

Моделирование работы web-сервера на основе системы массового обслуживания

Т.М. Попова, В.А. Слободчиков

Тихоокеанский государственный университет, Хабаровск

Аннотация: В работе рассмотрена имитационная модель работы веб-сервера Apache HTTP Server как системы массового обслуживания, проведено сравнение параметров соответствующей системы и Apache HTTP Server с использованием среды GPSS World. Для сравнения модели с реальным веб-сервером был развернут тестовый сервер. Генерация запросов и сбор статистики был осуществлен программой Apache JMeter. Было приведено сравнение обоих отчетов, указаны различия в характеристиках, и изложены предположения о причинах наличия расхождений. Модель может быть применена для установления требований к Apache HTTP Server с целью оптимизации его работы.

Ключевые слова: имитационное моделирование, система массового обслуживания, характеристики эффективности, тестовый сервер, поток заявок, каналы обслуживания, очередь.

В условиях стремительного роста объема интернет-трафика и увеличения числа веб-приложений, обеспечение высокой производительности и надежности веб-серверов становится критически важным, они выполняют ключевую роль в предоставлении доступа к содержимому веб-сайтов, обработке пользовательских запросов и взаимодействии с базами данных. Моделирование веб-серверов позволяет выявить слабые места, прогнозировать поведение системы под различными нагрузками и разрабатывать методы повышения их производительности. Таким образом, актуальность моделирования веб-серверов обусловлена потребностью в обеспечении стабильной и быстрой работы веб-приложений.

Веб-серверы могут быть рассмотрены как системы массового обслуживания, где каждый клиентский запрос представляет собой заявку на обслуживание.

Работа [1] посвящена измерению производительности моделей с очередями и множественными серверами с использованием методов теории

массового обслуживания, что позволяет оценить эффективность в условиях высокодинамичного трафика.

Так в работе [2] была рассмотрена модель доступа к радиоресурсам сети, которая включает ряд решений для повышения эффективности обслуживания за счёт настройки логической части сети. В этой модели применяются технология нарезки сети и эластичный трафик. В практической части работы было построено матрицы интенсивностей перехода для двух различных конфигураций.

В работе [3] представлена M/G/1/K*PS модель веб-сервера, получены аналитические выражения для метрик производительности веб-сервера, таких как среднее время отклика, пропускная способность и вероятность блокировки. Среднее время обслуживания и предел числа обрабатываемых запросов являются параметрами модели. Эти параметры оцениваются путем максимизации логарифмической функции правдоподобия на основе измеренного среднего времени отклика.

Моделирование потоков информации необходимо для проектирования работы серверов, анализа их чувствительности на изменение входящих потоков. В работе [4] авторы рассмотрели имитационную и аналитическую модели потоковой и сетевой компьютерных разведок, которая заключается в моделировании процессов идентификации информационных потоков защищаемого мобильного узла. Отличительной особенностью данного этапа является имитация поиска и идентификации формируемого мобильным узлом связи информационного потока с учетом динамического случайного переключения между информационными потоками, различного времени нахождения в данном потоке.

В статье [5] рассматриваются способы моделирования работы веб-серверов на языке GPSS, чаще всего применяемый для создания имитационных моделей систем массового обслуживания.

Моделирование работ различных технических устройств достаточно часто происходит методами систем массового обслуживания. В статье [6] рассматриваются вопросы моделирования взаимодействия человека и роботов в общей операционной среде в целях выбора оптимальных параметров такой системы. Система представлена в виде многоканальной системы массового обслуживания (в зоне зарядки).

Рассмотрим моделирование работы веб-сервера Apache HTTP Server, так как он является одним из наиболее распространенных веб-серверов в мире, по последним статистическим данным [7], Apache HTTP Server занимает второе место среди самых используемых веб-серверов в мире, обслуживая 29,6% всех сайтов. Построение имитационной модели осуществляется с использованием системы моделирования GPSS World [8], которая не только значительно ускоряет процесс моделирования и исследования самых разнообразных систем массового обслуживания и непрерывных процессов, но и позволяет проводить оптимизационные эксперименты. Для моделирования работы веб-сервера Apache HTTP Server с использованием GPSS создана модель, которая включает следующие компоненты: источники запросов, очередь, каналы обслуживания и статистику, позволяет провести детальный анализ системы, исследовать различные методы оптимизации для повышения производительности и надежности сервера. Таким образом, рассматриваем веб-сервер, как систему массового обслуживания. Для моделирования Apache HTTP Server как системы массового обслуживания, важно сопоставить основные параметры СМО с соответствующими параметрами и конфигурациями Apache HTTP Server. Поток заявок на сервер Apache HTTP Server представляет собой HTTP-запросы, которые поступают на сервер в случайные моменты времени. Этот поток можно рассматривать как пуассоновский, поскольку запросы приходят независимо друг от друга и с постоянной средней интенсивностью.

В Apache HTTP Server параметры `MaxRequestWorkers` и `ThreadsPerChild` определяют максимальное количество процессов или потоков, которые могут одновременно обрабатывать запросы. Это аналогично количеству каналов в многоканальной СМО, где каждый канал в любой фиксированный момент времени обслуживает не более одного запроса. Время обработки запросов в Apache HTTP Server соответствует времени обслуживания в СМО. В предположении, что поток обработанных запросов пуассоновский, то время обслуживания есть показательно распределенная случайная величина с параметром обратным интенсивности потока высвобождения (обработанных запросов). Если запрос не может быть немедленно обработан и ожидает обработки, то можно рассматривать этот факт, как образование очереди с дисциплиной FIFO (first in, first out «первым пришёл — первым ушёл»). Параметр `ListenBackLog` в Apache HTTP Server определяет максимальное количество запросов, которые могут находиться в очереди на обработку без ограничения времени ожидания. Очередь имеет фиксированную максимальную длину, при заполнении всех мест новые запросы отклоняются. Исходя из вышеописанных характеристик, Apache HTTP Server можно рассматривать как многоканальную систему массового обслуживания с пуассоновским потоком заявок, экспоненциально распределенным временем обслуживания, ограниченной очередью и обслуживанием заявок в порядке их поступления в очередь (M/M/c/K/FIFO).

Для тестирования имитационной модели использовалось приложение Apache JMeter, которое можно использовать для имитации высокой нагрузки на сервер. Apache JMeter предоставляет множество возможностей для тестирования веб-серверов и других веб-приложений, простота его использования, гибкость и широкий набор функций делают Apache JMeter неотъемлемым компонентом процесса тестирования и оптимизации веб-серверов. В контексте моделирования работы веб-сервера как системы

массового обслуживания, Apache JMeter полезен тем, что он позволяет оценить достоверность построенной модели, сравнивая ее с реальной работой веб-сервера.

Значения параметров СМО, необходимые, для построения модели, были получены из файлов конфигурации Apache HTTP Server как значения по умолчанию, доступ к которым осуществлялся с помощью программы XAMPP. Она позволяет локально развернуть на компьютере тестовую среду, включающую в себя Apache HTTP Server, для того чтобы сравнить его работу с построенной моделью.

Параметры `ThreadsPerChild` и `ListenBackLog` в Apache, согласно документации [9], имеют значения по умолчанию 64 и 511 соответственно, однако в сборке XAMPP после установки, они могут быть перезаписаны. По этой причине всегда необходимо проверять файлы конфигурации, если планируется работа с первоначальными настройками. Значение параметра `MaxRequestWorkers`, отвечающего за количество каналов в модуле `Prefork`, который используется версией Apache для Unix-систем, по умолчанию равно 256, поэтому это же значение было установлено параметру `ThreadsPerChild`. На основе настроенной конфигурации веб-сервер Apache моделируется как система массового обслуживания с 256 каналами и 511 местами в очереди (M/M/256/511). Отчет с результатами моделирования изображен на рис. 1.

В отчете можно наблюдать, что система после генерации 1000 заявок продолжила генерировать их дальше, пока не была обслужена последняя заявка. Таким образом, было создано еще 293 заявки, 256 из которых дошли до каналов обслуживания, а 37 ожидают в очереди. Однако, лишние заявки не повлияли на результат моделирования в данном случае, поскольку симуляция остановилась ровно в момент завершения обслуживания тысячной заявки. С момента запуска симуляции до конца ее работы прошло 12,75 временных единиц, за такую единицу была взята 1 секунда.

Следовательно, от Apache HTTP Server будем ожидать обработки 1000 заявок за 12,75 секунд. Зная количество заявок и время, за которое они были обслужены, можно вычислить пропускную способность модели, которая составила 78,43 заявки в секунду. Ни одна заявка не получила отказ, значит можно говорить, что система не переходила в состояние, когда заняты все места в очереди.

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	12.750	9	0	2

NAME	VALUE
QUEUE_IS_FULL	9.000
REQUEST_WORKER	10001.000
WAITING_CLIENTS	10000.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
	1	GENERATE	1293	0	0
	2	GATE	1293	0	0
	3	ENTER	1293	37	0
	4	ENTER	1256	1	0
	5	LEAVE	1255	0	0
	6	ADVANCE	1255	255	0
	7	LEAVE	1000	0	0
	8	TERMINATE	1000	0	0
QUEUE_IS_FULL	9	TERMINATE	0	0	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES AVL.	AVE.C.	UTIL.	RETRY	DELAY
WAITING_CLIENTS	511	473	0	40	1293	1	5.752	0.011	0 0
REQUEST_WORKER	256	0	0	256	1256	1	208.507	0.814	0 37

Рис. 1 – Отчет с результатами моделирования работы веб-сервера в GPSS

Стоит отметить, что имитационное моделирование веб-сервера не только позволяет предсказать его эффективность, но и помогает определить требования к его работе. В случае невыполнения таких требований со стороны веб-сервера администратором может быть принято решение провести работы по его оптимизации.

Для проверки созданной модели, необходимо сравнить ее с реальным веб-сервером Apache. С помощью программы XAMPP локально была

развернута среда тестирования и запущен Apache HTTP Server. С помощью файлов конфигурации были настроены параметры `ThreadsPerChild` и `ListenBackLog`, им были указаны значения 256 и 511 соответственно. Значение порта было изменено с 80, стандартного для протокола HTTP, на 8081, так как порт 80 на устройстве уже использовался другим процессом.

На тестовом сервере расположен единственный файл – `index.php` с кодом на языке PHP, который осуществляет генерацию псевдослучайного числа с экспоненциальным распределением методом обратной функции [10, с.53] и задержку на время, равное этому числу, в секундах.

Порядок событий на тестовом сервере можно описать следующим образом:

1. Пользователь отправляет HTTP запрос на адрес `http://localhost:8081/`.
2. Apache HTTP Server слушает данный запрос, после чего подключает свободный рабочий процесс, если такой имеется или запрос помещается в очередь. Если очередь заполнена, то сервер отправляет ответ с HTTP-статусом 503 Service Unavailable.
3. В конфигурации Apache уже предусмотрено перенаправление с адреса `localhost/` на адрес `localhost/index.php`, по которому находится файл с PHP-кодом.
4. Далее идет выполнение кода PHP.
5. Apache возвращает ответ пользователю в виде получившегося числа.
6. Браузер получает ответ от сервера и выводит число на странице

В результате тестирования 1000 пользовательских запросов были обработаны приблизительно за 13 секунд, также был предоставлен два отчета от каждого из слушателей. На рис. 2 продемонстрирован фрагмент отчета View Results Tree, который выводит список всех ответов от сервера и позволяет рассмотреть подробно каждый из них.

Значения параметров, предоставленные отчетом Summary Report, можно посмотреть в таблице 1.

Стоит заметить, что стандартный таймер Apache JMeter показывает только целое количество секунд, однако благодаря собранной статистике можно посчитать точное время работы симуляции. Для этого надо разделить количество сгенерированных пользователей на пропускную способность. Получаем, что сервер обработал 1000 заявок за 12,96 секунд.

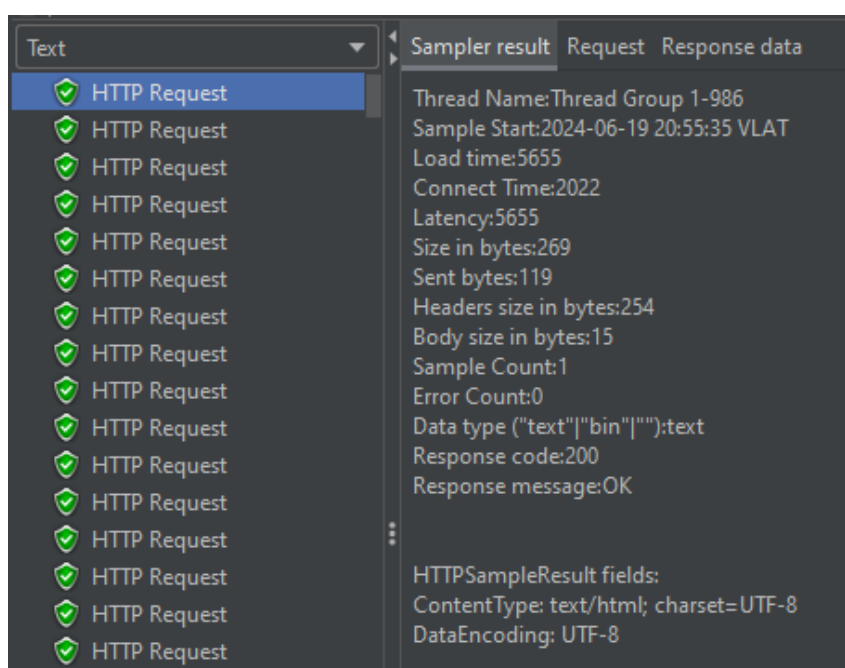


Рис. 2 – Список ответов от сервера и пример ответа.

Доля отклоненных заявок составила 0%. Это означает, что аналогично имитационной модели, не возникало такого состояния, при котором все места в очереди были бы заняты.

При сравнении отчета, полученного в результате моделирования работы веб-сервера в GPSS World и представленного на рисунке 1, с отчетом, который был предоставлен программой Apache JMeter по завершении тестирования сервера и оформлен в виде таблицы 1, можно заметить несущественную разницу в сроках выполнения симуляции. Имитационная

модель обработала 1000 заявок за 12,75 секунд, в то время как настоящий веб-сервер справился с тем же объемом запросов за 12,96 секунд

Таблица № 1

Отчет слушателя Summary Report

№ п/п	Название параметра	Описание параметра	Значение параметра
1	Samples	Количество выполненных запросов	1000
2	Average	Среднее время отклика	6,431 с
3	Min	Минимальное время отклика	0,04 с
	Max	Максимальное время отклика	12,95 с
	Std. Dev.	Стандартное отклонение	3,778 с
	Error %	Процент запросов, завершившихся с ошибками	0,00%
	Throughput	Пропускная способность	73,0 с ⁻¹
	Received KB/sec	Скорость получения данных от сервера	19,18 КБ/с
	Sent KB/sec	Скорость отправки данных на сервер	8,48 КБ/с
	Avg. Bytes	Средний размер данных, переданных за один запрос	269,0 Б

Имитационная модель адекватно описала работу веб-сервера, несмотря на незначительное различие некоторых характеристик, при этом можно изменять конфигурации настроек Apache, так как у него есть более 700 параметров, и рассмотреть их влияние на функциональность веб-сервера. Для повышения пропускной способности веб-сервера Apache можно подобрать более эффективную конфигурацию, используя методы оптимизации.

Литература

1. Win L. Application of Queuing Theory for Internet Server // Annual University Journal on Innovative Research and Products. 2019. Vol. 2 Issue. 1. pp.508-515.
 2. Гончаренко С. Н. Радимов И.Р. Оптимизация разделения радиоресурсов и интенсивностей перехода состояний в двухсервисной модели сети 5G с эластичным типом трафика // Инженерный вестник Дона, 2024. № 5. URL: ivdon.ru/ru/magazine/archive/n5y2024/9197/.
 3. Cao J, Andersson M., Nyberg C., Kihl M. Web Server Performance Modeling using an M/G/1/K*PS Queue // 10th International Conference on Telecommunications, ICT 2003, vol. 2, no. 2, 2003, pp. 1501–1506.
 4. Добрышин М. М., Закалкин П. В., Жук С. И. Модель узла компьютерной сети как объекта компьютерной разведки учитывающая динамику использования арендуемых информационных потоков // Инженерный вестник Дона, 2019. № 4. URL: ivdon.ru/magazine/archive/n4y2019/5887/.
 5. Хваталин М. М. Моделирование работы web-сервера на языке GPSS // APRIORI. Серия: Естественные и технические науки. 2015. №6. URL: cyberleninka.ru/article/n/modelirovanie-raboty-web-servera-na-yazyke-gpss (дата обращения: 19.07.2024).
 6. Филяев Г. А., Вилисов В. Я. Алгоритм концептуального проектирование системы мониторинга объекта коллаборативной мультиагентной робототехнической системой // Инженерный вестник Дона, 2021. № 5. URL: ivdon.ru/ru/magazine/archive/n5y2021/6965/.
 7. Usage of Apache broken down by operating systems // W3Techs - World Wide Web Technology Surveys. URL: w3techs.com/technologies/breakdown/ws-apache/operating_system (дата обращения: 17.07.2024).
-



8. GPSS World Student Version// URL: gps-world-student-version.software.informer.com/ (дата обращения 17.07.2024).
9. Apache HTTP Server Documentation // The Apache HTTP Server Project URL: httpd.apache.org/docs/ (дата обращения: 17.06.2024).
10. Петров Ю. В., Аникин С. Н., Южно С. А. Моделирование случайных величин: учебное пособие. СПб.: Балт. гос. техн. ун-т., 2020. 90 с.

References

1. Win L. Annual University Journal on Innovative Research and Products. 2019. Vol. 2 Issue. 1. pp.508-515.
2. Goncharenko S. N. Radimov I. R. Inzhenernyj vestnik Dona, 2024. № 5. URL: ivdon.ru/ru/magazine/archive/n5y2024/9197/.
3. Cao J., Andersson M., Nyberg C., Kihl M. 10th International Conference on Telecommunications, ICT 2003, vol. 2, № 2, pp. 1501–1506.
4. Dobryshin M. M., Zakalkin P. V., Zhuk S. I. Inzhenernyj vestnik Dona, 2019. № 4. URL: ivdon.ru/magazine/archive/n4y2019/5887/.
5. Khvatalin, M. M. APRIORI. Ceriia: Estestvennye i tekhnicheskie nauki. 2015. № 6. URL: cyberleninka.ru/article/n/modelirovanie-raboty-web-servera-na-yazyke-gpss (date assessed 19.07.2024)
6. Filiaev G. A. Vilisov V. Ia. Inzhenernyi vestnik Dona, 2021. № 5. URL: ivdon.ru/ru/magazine/archive/n5y2021/6965/.
7. Usage statistics and market shares of web servers. URL: w3techs.com/technologies/overview/web_server (date assessed: 17.06.2024).
8. GPSS World Student Version. URL: gps-world-student-version.software.informer.com/ (date assessed 17.07.2024)
9. Apache HTTP Server Documentation. The Apache HTTP Server Project URL: httpd.apache.org/docs/ (date assessed: 17.06.2024).



10. Petrov Iu.V., Anikin S.N., Iukhno S.A. Modelirovanie sluchainykh velichin: uchebnoe posobie [Modelling of random variables] SPb.: Balt. gos. tekhn. un-t., 2020. 90 p.

Дата поступления: 25.06.2024

Дата публикации: 3.08.2024