

УДК 004.89

Дана робота присвячена дослідженню моделей архітектури серверних додатків корпоративних інформаційних систем. Розглянуто існуючі моделі архітектури, шляхом імітаційного моделювання виділені переваги та недоліки. На підставі результатів моделювання побудована модифікована модель архітектури, яка дозволить зменшити сукупну вартість володіння корпоративної інформаційної системи за рахунок оптимізації технічних показників системи

Ключові слова: модель архітектури, сукупна вартість володіння, комбінована модель, критерій ефективності

Данная работа посвящена исследованию моделей архитектуры серверных приложений корпоративных информационных систем. Рассмотрены существующие модели архитектуры, путем имитационного моделирования выделены преимущества и недостатки. На основании результатов моделирования построена модифицированная модель архитектуры, которая позволит уменьшить общую стоимость владения корпоративной информационной системы за счет оптимизации технических показателей системы

Ключевые слова: модель архитектуры, совокупная стоимость владения, модифицированная модель, критерии эффективности

ИССЛЕДОВАНИЕ МОДЕЛЕЙ АРХИТЕКТУР СЕРВЕРНЫХ ПРИЛОЖЕНИЙ КОРПОРАТИВНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Д. К. Михнов

Кандидат технических наук, доцент*

E-mail: mikhnov@kture.kharkov.ua

Е. А. Курилина*

E-mail: elkurilina@gmail.com

*Кафедра информационных управляющих систем
Харьковский национальный
университет радиоэлектроники
пр. Ленина, 14, г. Харьков, Украина, 61166

1. Введение

В настоящее время для повышения эффективности и минимизации издержек управления (временных, ресурсных и финансовых), широко применяются корпоративные информационные системы, помогающие осуществлять контроль бюджетных процессов, рабочего времени сотрудников, выполненных ими работ, хода реализации проектов, документооборота, и других управленческих функций.

Постоянно растущие требования пользователей к серверным приложениям корпоративных информационных систем влияют на архитектуру, развертывание, обслуживание и администрирование программного обеспечения. Совокупная стоимость владения программного обеспечения в настоящее время в основном состоит из затрат, возникающих после развертывания [1]. Приложение с хорошей моделью архитектуры обеспечит минимальную совокупную стоимость владения благодаря снижению затрат и времени, необходимых на развертывание приложения, обеспечение его работы, обновление.

Как правило, существующие модели архитектур ориентированы на выявление компонентов и связей между ними с точки зрения простоты поддержки решаемых задач. Они не учитывают возможность роста системы, то есть: увеличение базы данных и увеличение количества пользователей. На практике, рост данных показателей приводит к большим нагрузкам на комплекс технических средств. Как следствие,

большие нагрузки приводят к замедлению работы, снижаются такие показатели как время отклика системы, загрузка. Совокупная стоимость владения такими системами растёт пропорционально с ростом нагрузки на них.

Поэтому актуальна задача выбора рациональной модели архитектуры серверного приложения корпоративной информационной системы, которая позволит оптимизировать распределение нагрузки на комплекс технических средств и тем самым снизить совокупную стоимость владения системой.

2. Анализ литературных данных и постановка проблемы

Вопросам моделирования архитектур серверных приложений корпоративных информационных систем посвящены работы М. Фаулера [2] и П. Брауна [3]. Значительный вклад в развитие теории уровней моделей архитектуры внесли Д. Нильссон [4] и В. Маринеску [5]. Следует также отметить значительное количество публикаций на темы оптимизации архитектуры программных приложений и организации их взаимодействия с базами данных размещено на сайтах компаний, таких как Microsoft, Oracle [6].

На основании анализа данных источников, можно сказать, что на сегодняшний день существует 5 наиболее распространённых уровней моделей архитектур.

В модели по Фаулеру стоит различать три слоя: домен, источник данных и представление [2]. Она является самой простой и предназначена для решения общих задач.

Модель Брауна охватывает пять слоев: представление, контроллер/медиатор, домен (предметная область, бизнес-логика), отображение данных и источник данных.

Два дополнительных слоя, по существу, выполняют посреднические функции между базовыми слоями: контроллер/медиатор соединяет слои представления и домена, а слой отображения данных служит связующим звеном между предметной областью и источником данных [4].

Модель на платформе J2EE реализована в виде набора типовых решений Core J2EE. Здесь различаются следующие слои: клиент, представление, бизнес, интеграция и ресурсы. Слою бизнеса соответствует слой домена, а слою интеграции соответствует слой источника данных. Основное отличие модели связано с расщеплением слоя представления на две части между клиентом (слой клиента) и сервером (слой представления) [5].

Модель Маринеску содержит пять слоев: представление, приложение, службы, домен и сохранение данных. Слой представления расщеплен на два слоя, отображающих структуру контроллера приложения. Домен также подвергся расщеплению: на основе модели предметной области сконструирован слой служб, что соответствует обычной практике деления бизнес-логики на две части [6].

Самая сложная модель архитектуры ПО предложена Нильссоном.

Она состоит из следующих слоев: потребитель, вспомогательный слой потребителя, приложение, домен, доступ к хранимым данным, общедоступные хранимые процедуры и приватные хранимые процедуры. Нильссон предусматривает активное использование хранимых процедур и поощряет размещение в них фрагментов бизнес-логики для повышения производительности приложения. Слои хранимых процедур Нильссона содержат как источники данных, так и бизнес-логику [7].

Однако рассмотренные модели архитектур не являются эффективными в случае их применения для корпоративных информационных систем. Это вызвано недостаточной разработанностью вопросов анализа поступающих запросов в систему, выделения частей модели архитектуры, организации взаимодействия между ними, оценке затрат совокупной стоимости владения.

3. Цель и задачи исследования

Основной целью исследования является разработка более гибкой модели архитектуры серверных приложений корпоративных информационных систем (ИС), которая позволит разрабатывать прикладное программное обеспечение (ПО) предприятий и организаций рассчитанное на рост нагрузки на комплекс технических средств.

Достижение цели обеспечивается решением следующих задач:

- анализ существующих уровневых моделей архитектур ПО, используемых при проектировании ИС;
- формирование модифицированной уровневой модели архитектуры ПО для ИС ориентированных на увеличение количества пользователей;
- формирование критерия эффективности моделей архитектур;
- выбор модели архитектуры ПО, рациональной для проектирования ИС рассчитанной на большие нагрузки.

4. Анализ существующих моделей архитектур и обработка результатов

Концепция уровней – одна из общепотребительных моделей, используемых разработчиками ПО для разделения сложных систем на более простые части.

В архитектурах компьютерных систем, например, различают слои кода на языке программирования, функций операционной системы, драйверов устройств, наборов инструкций центрального процессора и внутренней логики чипов [3].

Для проведения исследований моделей архитектур было использовано имитационное моделирование на основании теории массового обслуживания с помощью языка GPSS [8].

На этапе анализа было выявлено, что рассматриваемые модели архитектуры можно разделить на два типа по признаку разветвления процесса обработки запроса.

Можно выделить линейные модели архитектуры: модель Фаулера, модель J2EE, а нелинейные: модель Брауна, Маринеску, Нильссона.

На основании анализа моделей архитектур и проведения экспертной оценки работы ИС спроектированных с использованием рассматриваемых моделей архитектур, были определены входные параметры для проведения имитационного моделирования.

Входными данными являются:

- входящий поток запросов: поток Пуассона, минимальное время между запросами 0,5 с, максимальное – 1,5 с;
- система обслуживания, состоящая из накопителя и 2 узлов обслуживания;
- время работы узлов обслуживания сервера, характеризующиеся нормальным законом распределения, первый узел представляет собой обслуживание запроса сервером, занимает 0,6 с, второй узел представляет собой обслуживание запроса базой данных и занимает от 3,4 с до 10,2 с, заявка требует обслуживания базой данных с вероятностью 80 %;
- дисциплина ожидания: если обрабатывающий узел занят, то заявка становится в очередь.
- дисциплина обслуживания: заявки принимаются к обслуживанию в порядке очереди;
- дисциплина очереди: в системе будет использоваться одна очередь.

Было проведено моделирование поведения для выделенных типов моделей архитектур. Схема для моделирования поведения линейной модели архитектуры приведена на рис. 1.

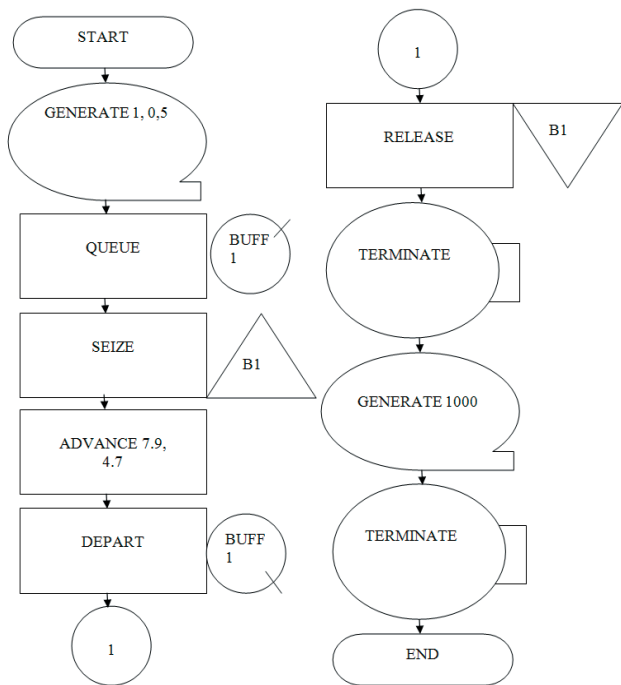


Рис. 1. Схема линейной модели архитектуры

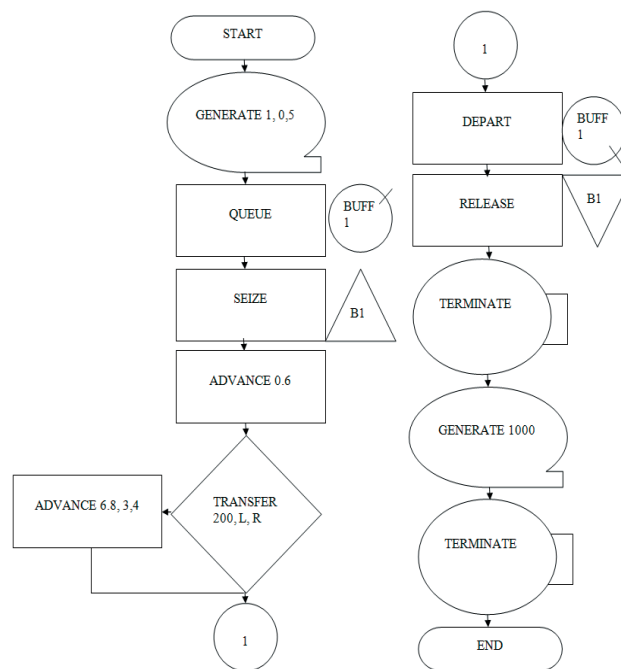


Рис. 2. Схема нелинейной модели архитектуры

Так же было выявлено, что во всех моделях работа с базой данных занимает наибольшее время. База данных должна хранить непротиворечивые данные, поэтому в один момент времени с базой данных должен работать один пользователь. За счёт этого формируется очередь запросов. Но в условиях больших нагрузок на систему, когда пользователи посылают запросы каждую секунду, можно заметить, что большинство пользователей получают устаревшую информацию.

Часто случается, что пользователь получает информацию, которая через незначительный момент времени становится устаревшей, так как множество пользователей с ней работают одновременно и изменяют её.

На основании этого, была проведена модификация нелинейной модели архитектуры. Если пользователи, в основном, получают устаревшие данные, то имеет смысл выделить отдельную базу данных, которая будет хранить простые структуры данных, предназначенные только для чтения. Доступ к такой базе не потребует синхронизации и, следовательно, формирования очереди.

Схема для моделирования поведения нелинейной модели архитектуры приведена на рис. 2. Схема модифицированной нелинейной модели архитектуры приведена на рис. 3.

Для построения комплексного критерия эффективности модели архитектуры был проведён анализ взаимодействия её элементов.

Все элементы были разбиты на три типа: домен, представление, источник данных.

Были использованы следующие частные критерии, предложенные Фаулером: время отклика (response), пропускная способность (throughput), загрузка (load), способность к вертикальному и горизонтальному масштабированию (scalabilityV, scalabilityG) [9].

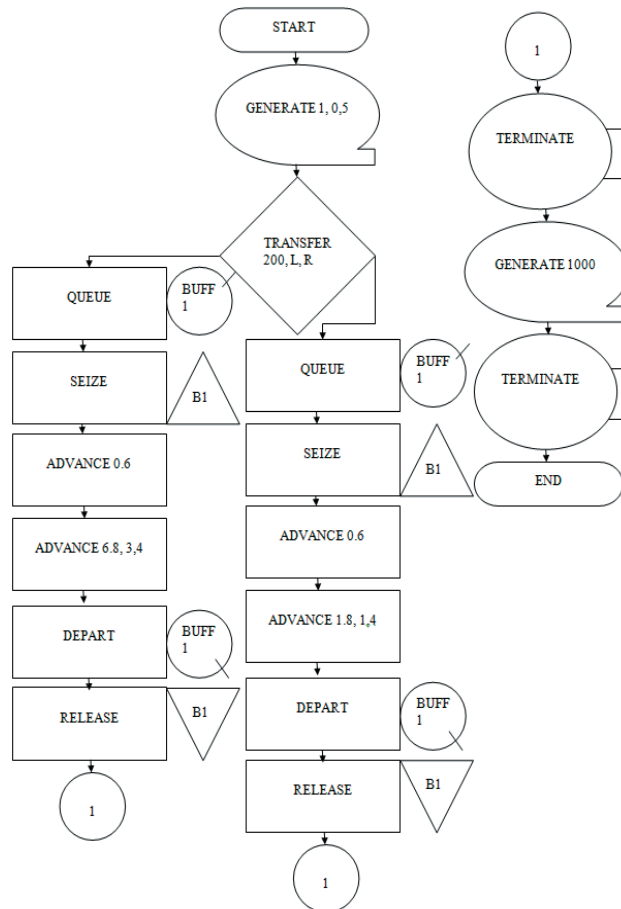


Рис. 3. Схема модифицированной нелинейной модели архитектуры

Способность к масштабированию – свойство, характеризующее поведение системы при добавлении

Таблица 1

Значения интегрального критерия

Название модели	Параметры эффективности	Значение интегрального критерия
Нелинейная модель	Load=19973 Throughput=2711 Response=1,2 ScalabilityV=19 ScalabilityG=78	668,7*108
Линейная модель	Load=20121 Throughput=2367 Response=1,5 ScalabilityV=36 ScalabilityG=49	560*108
Модифицированная модель	Load=21590 Throughput=3695 Response=1,2 ScalabilityV=19 ScalabilityG=60	757,6*108

аппаратных ресурсов. Масштабируемой принято считать систему, производительность которой возрастает пропорционально объему приобщенных ресурсов (например, вдвое при удвоении количества серверов). Вертикальное масштабирование это увеличение мощности отдельного сервера (например, за счет увеличения объема оперативной памяти). Горизонтальное масштабирование – это наращивание потенциала системы путем добавления новых серверов.

Данные параметры были выбраны, так как они дают возможность оценить совокупную стоимость владения корпоративной информационной системой. Известно, что совокупную стоимость владения включает в себя косвенные и прямые затраты.

Из косвенных затрат моделировались затраты от простоя пользователей с помощью параметров: время отклика и загрузка, рост которых, прямопропорционален росту затрат от простоев пользователей.

Из прямых затрат моделировались затраты на приобретение комплекса технических средств с помощью таких параметров как вертикальное и горизонтальное масштабирование. Рост данных параметров обратно пропорционален росту затрат на оборудование.

Исходя из физического смысла выбранных частных критериев, недопустимым является компенсирование недостаточной величины одного частного критерия избыточной величиной другого и сглаживание уровней частных критериев за счет неравнозначных первоначальных значений частных критериев. Исходя из этого, для сведения параметров эффективности к единому показателю был использован интегральный критерий эффективности:

$$K(x) = \max \left(\frac{4 \cdot \text{load} \cdot \text{throughput} \cdot \text{scalabilityV} \cdot \text{scalabilityG}}{\text{response}} \right) \quad (1)$$

Данный критерий является безразмерным и не требует нормирования частных критериев [10]. Весовые коэффициенты критерия были определены с помощью экспертной оценки.

В результате проведения имитационного моделирования было получено множество значений параметров эффективности и проведен усреднение с помощью правила среднего арифметического [11]. Далее были рассчитаны значения интегрального критерия для каждой модели (табл. 1).

Из анализа полученных результатов имитационного моделирования можно увидеть, что минимизируются затраты на поддержку комплекса технических средств и уменьшаются потери от простоя пользователей. Таким образом, модифицированная модель позволит снизить совокупную стоимость владения корпоративной информационной системой за счёт снижения косвенных и прямых затрат на систему [12].

Практическая значимость исследования заключается в возможности применения предложений и разработок, являющихся результатом работы, для оптимизации распределения нагрузок на комплекс технических средств серверных приложений во время проектирования, что будет способствовать сокращению времени и расходов на поддержку, сокращению времени ожидания пользователей.

5. Выводы

(1) Были проанализированы 5 наиболее распространённых моделей архитектур с точки зрения корпоративных информационных систем.

На основании анализа моделей архитектур и проведения имитационного моделирования с использованием интегрально критерия эффективности была выбрана наиболее рациональная модель – нелинейная.

После проведения анализа работы корпоративных систем и проведения экспертной оценки была предложена модификация нелинейной модели архитектуры, основанная на выделении из очереди запросов к системе, тех, которые требуют работы с базой данных на запись. Данная модификация позволила оптимизировать такие параметры эффективности как: время обработки запросов, пропускная способность, горизонтальная масштабируемость.

Таким образом, была получена более гибкая модель архитектуры серверных приложений корпоративных информационных систем, которая позволит снизить совокупную стоимость разработанного программного обеспечения за счёт более гибкого распределения нагрузки на комплекс технических средств.

Литература

1. MacCormack, A. The True Costs of Software [Electronic resource] / A. MacCormack. – Available at: http://www.computerworld.com/s/article/81590/The_True_Costs_of_Software?pageNumber=1/. – 23.09.2013.
2. Фаулер, М. Архитектура корпоративных программных приложений [Текст] / М. Фаулер. – К. : Издательский дом “Вильямс”, 2006. – 560 с.
3. Brown, K. Enterprise Java Programming with IBM WebSphere [Text] / K. Brown. – М : IBM Press, 2003. – 960 с.

4. Deepak, A. Core J2EE Patterns: Best Practices and Design Strategies. [Text] / A. Deepak — Sun Microsystems Press, 2001. — 393 p.
5. Parsons, D. Designing Component-Based Applications [Text] / D. Parsons. — K : Sun Microsystems Press, 2004. — 449 p.
6. Marinescu, F. EJB Design Patterns. [Text] / F. Marinescu. — K : Wiley Computer Publisher, 2002. — 259 p.
7. Sztrik, J. Basic Queueing Theory. [Текст] / J. Sztrik. — K. : University of Debrecen, Faculty of Informatics, 2012. — 193 p.
8. Мониторинг аппаратных серверов [Электронный ресурс] : интеракт. учеб. / Режим доступа : <http://ashirobokov.blogspot.com/>. — 30.05.2013. — Загл. с экрана.
9. Имитационное моделирование систем массового обслуживания [Электронный ресурс] / Московский Государственный Технический Университет им. Н. Э. Баумана. — Режим доступа : <http://gps.h11.ru/norenkov.php/>. — 20.09.2013. — Загл. с экрана.
10. Астахов, А. В. Краткий словарь важнейших правовых, экономических, экологических терминов и понятий. [Текст] / А. В. Астахов, В. К. Зайденварг. — К.: Издательский дом "Вильямс", 2009. — 383 с.
11. Губарь, Ю. Р. Введение в математическое моделирование, Лекция 5: Компьютерное имитационное моделирование. Статистическое имитационное моделирование. [Текст] / Ю. Р. Губарь. — К. : Интуит.ру, 2012. — 223 с.
12. Совокупная стоимость владения и сервисы ИТ [Электронный ресурс] : интеракт. учеб. / Режим доступа : <http://userdocs.ru/informatika/754/index.html?page=4/>. — 20.09.2013. — Загл. с экрана.

У статті вирішено задачу порівняльної експериментальної оцінки ефективності застосування метрик мультимножин в задачах інформаційного пошуку. В результаті імітаційного моделювання визначено «період напіврозпаду корисності» як основний показник ефективності. Отримані значення показника зіставлені з показниками альтернативних методів. Показано, що застосування метрик мультимножин в задачах інформаційного пошуку сприяє підвищенню якості інформаційного пошуку

Ключові слова: WEB-орієнтовані системи, адаптивні алгоритми, інформаційний пошук, критерії ефективності, мультимножество

В статье решена задача сравнительной экспериментальной оценки эффективности применения метрик мультимножеств в задачах информационного поиска. В результате имитационного моделирования определен «период полураспада полезности» как основной показатель эффективности. Полученные значения показателя сопоставлены с показателями альтернативных методов. Показано, что применение метрик мультимножеств в задачах информационного поиска способствует повышению качества информационного поиска

Ключевые слова: WEB-ориентированные системы, адаптивные алгоритмы, информационный поиск, критерии эффективности, мультимножество

УДК 004.413:629.78.064.018

ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ МЕТРИК МУЛЬТИМНОЖЕСТВ В ЗАДАЧАХ ИНФОРМАЦИОННОГО ПОИСКА

Д. С. Негурица

Аспирант

Кафедра программной инженерии

Харьковский национальный

университет радиоэлектроники

пр. Ленина, 14, г. Харьков, Украина, 61166

E-mail: neguritsa@gmail.com

1. Введение

Существует пять видов WEB-ориентированных систем, которые используются в настоящее время как прикладные области в большинстве научных работ по оптимизации методов поиска информации.

1. Системы дистанционного обучения имеют относительно небольшое гиперпространство представле-

ния отдельного курса или раздела учебного материала по конкретной теме [1, 2].

2. Сетевые информационные системы (On-Line Information Systems) – обеспечивают справочный доступ к информации для пользователей с различным уровнем знания темы, которые нуждаются в различной информации о понятиях и в различных уровнях детализации [3].