

УДК 004.4

UDC 004.4

**ОПТИМИЗАЦИЯ БАЛАНСИРОВКИ  
НАГРУЗКИ МЕЖДУ СЕРВЕРНЫМИ  
ПЛАТФОРМАМИ ВЕБ-ПРИЛОЖЕНИЙ**

**OPTIMIZATION OF LOAD BALANCING  
ACROSS SERVER PLATFORMS OF WEB  
APPLICATIONS**

Носков Владислав Юрьевич

Noskov Vladislav Yurievich

Гой Егор Александрович

Goy Egor Aleksandrovich

Пономарев Александр Владимирович

Ponomarev Aleksandr Vladimirovich

Костромин Алексей Вячеславович  
*Уральский федеральный университет,  
Екатеринбург, Россия*  
[vynoskov@gmail.com](mailto:vynoskov@gmail.com)

Kostromin Aleksey Vyacheslavovich  
*Ural Federal University, Yekaterinburg, Russia*  
[vynoskov@gmail.com](mailto:vynoskov@gmail.com)

В статье описывается разработанный алгоритм оптимального распределения нагрузки между разнородными по своей производительности системами при работе с одним распределенным веб-приложением

This article describes the developed algorithm for optimal load distribution between heterogeneous systems in terms of performance when working with a distributed web application

Ключевые слова: ВЕБ-ПРИЛОЖЕНИЕ,  
ПРОИЗВОДИТЕЛЬНОСТЬ,  
МАСШТАБИРОВАНИЕ, ОБЛАЧНЫЕ  
СИСТЕМЫ, БАЛАНСИРОВКА НАГРУЗКИ

Keywords: WEB APPLICATION, PERFORMANCE,  
SCALING, CLOUD SYSTEMS, LOAD  
BALANCING

Широкое использование сервисов, предоставляемых с использованием веб-ориентированных систем (ВОС), требует построения производительных решений в части построения сервисных платформ. Количество посетителей online-магазинов, SaaS платформ, развлекательных и образовательных порталов, растет быстрыми темпами (согласно исследованию, проведенному в сентябре 2011 года Фондом «Общественное мнение» и Microsoft [1] рост рынка интернет торговли составил около 26%, рост услуг SaaS, согласно материалам CNews Analytics, составляет более 100% в годовом исчислении [2]). Постоянный рост нагрузки на ВОС без изменения аппаратной платформы, на которой запущено приложение, приводит к постепенному увеличению времени отклика системы и, впоследствии, к частичной деградации предоставляемого сервиса ВОС. Типовыми решениями проблемы ограниченной производительности является создание многоуровневых ВОС (выделение различных аппаратных платформ для различных задач) а

также переход к облачным вычислительным системам (ОВС), предполагающим использование распределенного «облака» вычислительных платформ, на которых одновременно запущен некий фрагмент или полная копия веб приложения. Распределенная нагрузка по множеству аппаратных платформ позволяет достигать высоких показателей по обрабатываемой нагрузке, масштабированию, отказоустойчивости.

Важной задачей при построении ОВС становятся внедрение в составе ОВС механизмов эффективного распределения атомарной нагрузки между аппаратными платформами, входящими в состав ОВС, контроль их функционирования, возможности динамического перераспределения нагрузки между платформами.

Современные и перспективные высоконагруженные ОВС проектируются и строятся с размещением аппаратных платформ на разных географических площадках. В связи с этим, при решении задачи балансировки нагрузки в распределенных ОВС решаются две независимых задачи. Первая – распределение нагрузки между площадками, на которых размещены аппаратные ресурсы ОВС, и вторая – распределение поступающей нагрузки внутри площадки. Разработка алгоритма для оптимального горизонтального масштабирования ОВС является целью проведенной работы.

### **Моделирование поведения оборудования ЦОД**

Основной критерий, который показывает оптимальность распределения нагрузки внутри ЦОД — загруженность CPU на вычислительных платформах одного уровня. Фактически — величина загрузки CPU в рабочем диапазоне (до 70-80%) прямо пропорциональна числу пользовательских запросов, которые обрабатывает серверная

платформа. Таким образом, основная задача при моделировании, корректно учесть коэффициент пропорциональности, обусловленный разницей в частоте работы CPU и количестве ядер CPU. Оценку этого коэффициента можно предварительно дать по документу Microsoft [3]. В рамках данного документа приведены количество машинных тактов, которые используются при выполнении различных действий с HTTP протоколом и возникающая разница в количестве тактов между однопроцессорной и двухпроцессорной системой при выполнении этих действий. Количество машинных тактов, в расчете на одного пользователя ВОС, сильно зависят от используемого ПО ВОС, архитектуры аппаратных средств и т.п., однако это не влияет на характер самой зависимости.

Используя приведенные цифры, можно вывести формулу (1), описывающую загрузку CPU в рабочем диапазоне (до 80%), как функцию, зависящую от числа пользовательских запросов (N).

$$L(N) = K_{\text{ОС}} + K_{\text{доп}} + \frac{(C - N)}{(F \cdot M)} \quad (1)$$

Где,

$K_{\text{ОС}}$  - доля загрузки CPU, обусловленная постоянными процессами ОС (практически не зависит от числа обрабатываемых пользовательских запросов);

$K_{\text{доп}}$  - доля загрузки CPU, обусловленная дополнительными процессами на серверной платформе (не зависит от числа обрабатываемых пользовательских запросов, а зависит от запущенных доп. процессов);

$C$  - количество циклов машинного времени требуемого для обработки одной пользовательской сессии (зависит только от логики работы ПО ВОС и является неизменным при увеличении числа пользователей);

$F$  - тактовая частота работы процессора;

**$M$**  - коэффициент использования нескольких ядер в многоядерных/многопроцессорных системах

Коэффициент использования многоядерных систем зависит, прежде всего, от возможности задачи к распараллеливанию. Предполагаем, что использование кода и средств ПО достаточно хорошо оптимизировано для распараллеливания на уровне ядер/процессов (до 80% запросов может быть распределено между ядрами/процессорами) и при увеличении числа ядер коэффициент использования составляет значения, приведенные в таблице 1 [4].

Таблица 1 – Коэффициент использования ядер процессора

Количество ядер/процессоров	Коэффициент использования ядер
1	1
2	1,8
4	3,1

Для моделирования, мы используем значения коэффициентов  **$K_{ос}$**  и  **$K_{доп}$**  составляющие не более 1%. Коэффициенты  **$C = 2 \cdot 10^6$**  для серверов Front End и  **$C = 2 \cdot 10^7$**  для серверов Middle End.

Для проверки адекватности и точности разработанной модели проведем расчёт загруженности CPU при различных условиях функционирования и различных ресурсах аппаратных платформ.

Рассчитаем загрузку процессоров при равномерном распределении нагрузки между серверами для трёх серверов обладающих различными аппаратными ресурсами в ситуациях, когда эти сервера выполняют функцию Front или Middle End. Расчёт проведем для различного числа пользователей (1000 и 4000).

## Результаты вычисления тестовых задач поведения оборудования ЦОД

При данных расчетах используем следующие сумму коэффициентов  $K_{OC}, K_{доп}$  - 5%. Коэффициенты  $C = 2 \cdot 10^6$  для серверов Front End и  $C = 2 \cdot 10^7$  для серверов Middle End.

Результаты проведённых расчётов моделирования поведения оборудования ЦОД при различном числе пользователей и равномерном распределении нагрузки приведено в таблице 2.

Таблица 2 - Моделирование при равномерном распределении нагрузки

Аппаратная платформа, кол-во ядер/частота	Кол-во пользователей	Загрузка CPU FrontEnd, %	Загрузка CPU MiddleEnd, %
$1/1,2 \cdot 10^9$	1000	11,7	26,7
$1/2,4 \cdot 10^9$		10,8	18,3
$2/2,0 \cdot 10^9$		10,6	15,6
$1/1,2 \cdot 10^9$	4000	16,7	76,7
$1/2,4 \cdot 10^9$		13,3	43,3
$2/2,0 \cdot 10^9$		12,2	32,2

Проведённые расчёты показывают, что модель адекватно отрабатывает различие в загрузке CPU при различиях в аппаратных средствах серверных платформ и их месте в структуре веб приложения, а также позволяет выполнить моделирование с учетом изменения числа пользователей системы.

## Разработка алгоритма оптимального распределения нагрузки

Задачу оптимального распределения вычислительной нагрузки на концептуальном уровне можно сравнить с задачей наполнения поилок — нескольких бочек с разными геометрическими характеристиками водой, которая поступает через трубу с вентилями. Один раз в секунду дно бочек

открывается, и вся вода из них попадает в поилки. Вода же, которая переливается через края бочек, уходит в землю бесцельно. Для исключения перелива необходимо выбрать такие регулирующие значения вентилей, чтобы вода в бочках держалась на одинаковом уровне. При увеличении потока воды уровень воды во всех бочках на момент сброса должен увеличиться на одинаковую величину, чтобы не допустить перелива воды ни в одной из бочек (необработанные запросы). Именно это позволит достигнуть максимального объема полезно употребленной воды (максимальное значение числа обработанных операций в многоузловых системах). Данный процесс можно представить на графике зависимости числа обработанных запросов от числа поступивших для системы, состоящей из двух узлов А и Б. Производительность системы Б в 2 раза выше, чем системы А. При оптимальном распределении нагрузки — 1/3 запросов на А и 2/3 запросов на Б (Рисунок 1, а) — величина предельного суммарного числа обработанных запросов, при котором не происходит сброс, вдвое выше, чем для варианта, представленного на рисунке 1, б (1/3 запросов приходится на Б и 2/3 запросов на А). Причина — быстрое достижение предела возможностей узла А и деградация производительности всей системы.

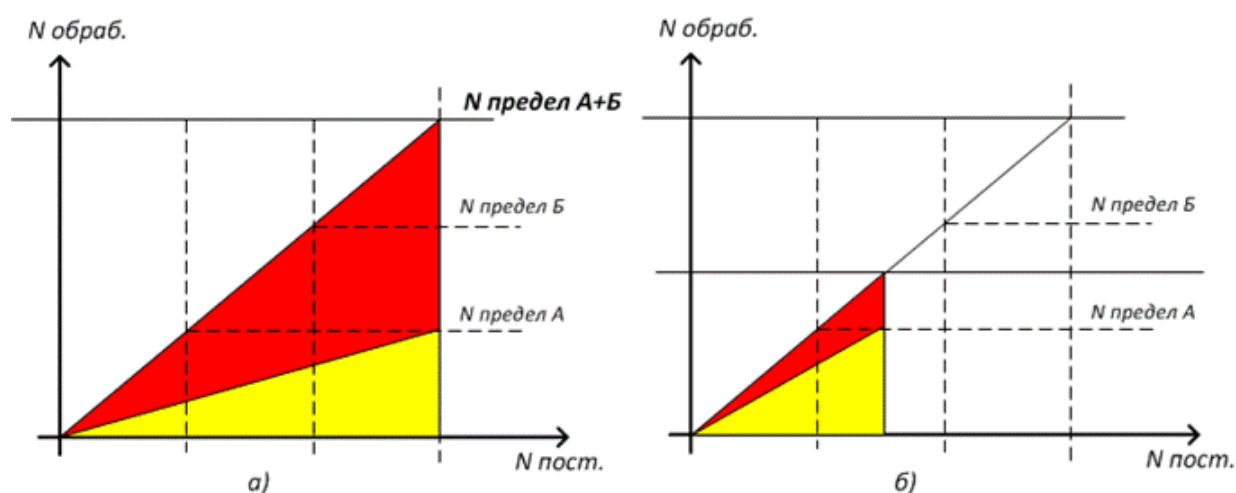


Рисунок 1. Предельная производительность системы

Таким образом, одна из важнейших задач балансировки — определение корректных и оптимальных пропорций распределения поступающих запросов и их последующая динамическая корректировка при изменении условий функционирования системы (изменение числа узлов, состава установленного и запущенных приложений, модернизация аппаратной платформы отдельных узлов и т. д.). Оптимальные пропорции в нашем случае — это такое соотношение числа обработанных запросов различными узлами, при котором уровни загрузки этих узлов приблизительно равны.

Решить подобную задачу можно путём введения относительного рейтинга аппаратных платформ, на которых выполняются приложения. Относительный рейтинг, может быть рассчитан исходя из оценки доли запросов ( $N_i/\sum N$ ), которые были отправлены на аппаратную платформу и загрузки CPU ( $L_i$ ), которая возникает в результате обработки данной доли запросов. По своему смыслу, относительный рейтинг указывает на долю запросов, которые обрабатываются аппаратной платформой в расчёте на один процент загрузки CPU. Формула (2) позволяет вычислить относительный рейтинг.

$$R_i = \frac{N_i}{L_i * \sum_1^m N_i} \quad (2)$$

где,

$R_i$  - относительный рейтинг аппаратной платформы;

$N_i$  - количество пользователей обслуживаемых сервером;

$L_i$  - значение загрузки CPU на аппаратной платформе

$m$  - общее количество аппаратных платформ с запущенным приложением.

Обладая сведениями об относительном рейтинге, возможно рассчитать такие значения доли запросов, отправляемых на каждую платформу, при которых значения загрузки CPU различных аппаратных платформ будут приблизительно равны. Т.е. для значений загрузки CPU будет выполняться равенство  $L_1 = L_2 = \dots = L_m$ .

Это достигается при соблюдении равенства приведённого в формуле (3):

$$\frac{N_1}{R_1 * \sum_1^m N_i} = \frac{N_2}{R_2 * \sum_1^m N_i} = \dots = \frac{N_m}{R_m * \sum_1^m N_i} \quad (3)$$

где,

$R_i$  - относительный рейтинг аппаратной платформы;

$N_i$  - количество пользователей обслуживаемых сервером;

$m$  - общее количество аппаратных платформ с запущенным приложением.

Таким образом, при предварительно найденных значениях рейтинга ( $R_i$ ), становится возможным рассчитать необходимые пропорции по оптимальному распределению нагрузки. Для выполнения этого действия необходимо нормализовать полученные значения относительных рейтингов (нормализовать относительно суммарного значения 100% (4) или единицы, в зависимости от методов задания конфигурации балансирующего устройства).

$$R_i^{\text{норм}} = \frac{100 * R_i}{\sum R_i} \quad (4)$$



Где

$R_i^{норм}$  - доля/процент от запросов, которые требуется отправить на аппаратную платформу;

$R_i$  - относительный рейтинг аппаратной платформы, рассчитанный на предыдущем шаге.

Для проверки адекватности и точности алгоритма балансировки проведем моделирование функционирования распределения нагрузки внутри ЦОД для следующей архитектуры (Рисунок 2).

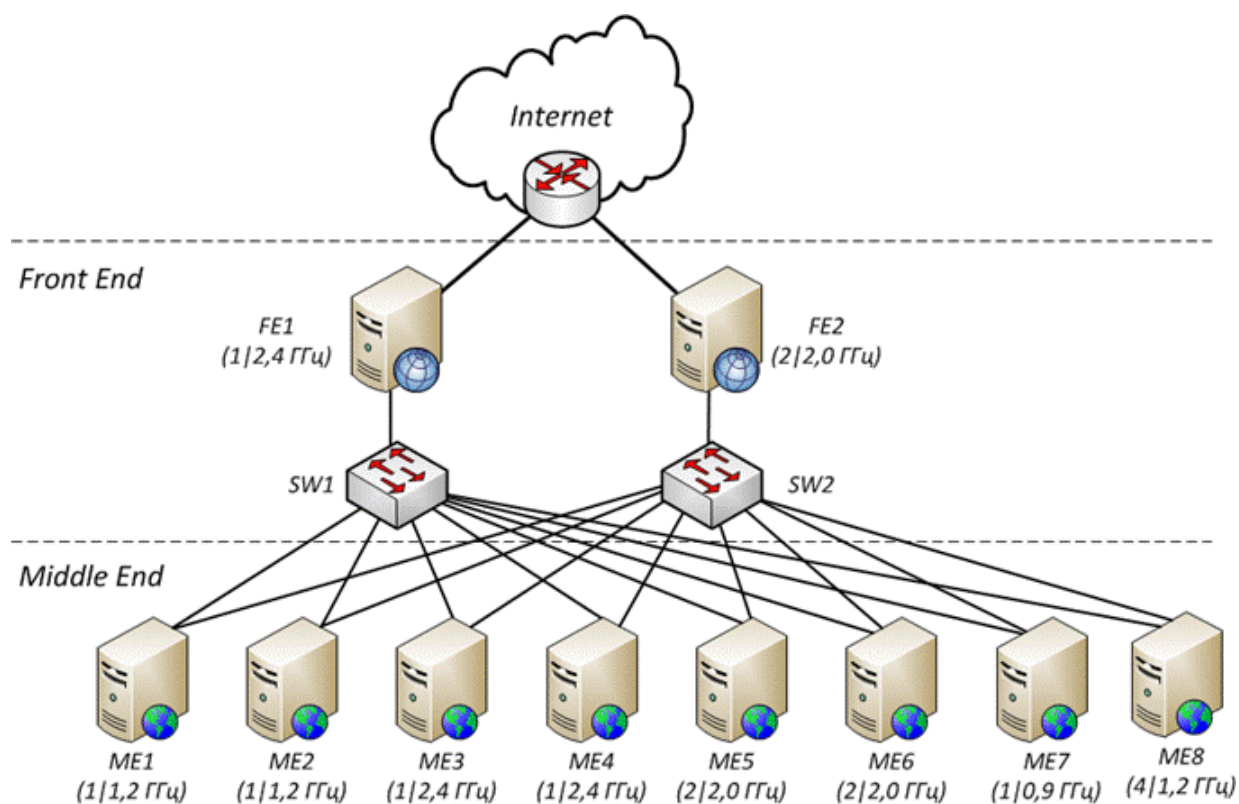


Рисунок 2. Схема ЦОД для моделирования

В таблице 3 представлены сводные характеристики аппаратных средств ЦОД.

Таблица 3 - Сводные характеристики аппаратных ресурсов ЦОД

Сервер	Кол-во ядер/частота	Уровень функционирования
ME1	$1/1,2 \cdot 10^9$	Middle End
ME2	$1/1,2 \cdot 10^9$	
ME3	$1/2,4 \cdot 10^9$	
ME4	$1/2,4 \cdot 10^9$	
ME5	$2/2,0 \cdot 10^9$	
ME6	$2/2,0 \cdot 10^9$	
ME7	$1/0,9 \cdot 10^9$	
ME8	$4/1,2 \cdot 10^9$	

Для проверки всех требуемых свойств алгоритма проведем моделирование и проведем сравнение следующих типовых задач:

1) Определим максимальное количество пользователей, которые будут обслуживаться без потерь пакетов в ЦОД приведенной архитектуры и используемых аппаратных средств, в условиях равномерного распределения нагрузки и оптимизированного в соответствии с разработанным алгоритмом;

2) Проведем расчет вычисленных пропорций перераспределения нагрузки и определим максимальное количество пользователей, которые могут быть обслужены без потерь пакетов в условиях выхода из строя одной из аппаратных платформ ОВС;

3) Определим загрузку аппаратных ресурсов ВОС в условиях равномерного распределения нагрузки и оптимизированного при одинаковой пользовательской нагрузке.

Проведем расчет максимального количества пользователей, при котором загрузка CPU любой платформы не превышает 80% (находится в

рабочем диапазоне) в условиях равномерного распределения нагрузки. Используем сумму коэффициентов  $K_{ос}$  и  $K_{доп}$  - 1%.

Достигнутый при моделировании при подобных начальных условиях и равномерном распределении запросов между платформами результат — около 28451 активных пользователей. При этом расчетная результирующая загрузка аппаратных средств ВОС представлена в таблице 4.

Таблица 4 - Моделирование при равномерном распределении нагрузки

Сервер	кол-во ядер/частота	Доля от общего числа запросов	Количество пользователей	CPU, %
ME1	$1/1,2 \cdot 10^9$	$1/8$	3556	60,25
ME2	$1/1,2 \cdot 10^9$	$1/8$	3556	60,25
ME3	$1/2,4 \cdot 10^9$	$1/8$	3556	30,63
ME4	$1/2,4 \cdot 10^9$	$1/8$	3556	30,63
ME5	$2/2,0 \cdot 10^9$	$1/8$	3556	20,75
ME6	$2/2,0 \cdot 10^9$	$1/8$	3556	20,75
ME7	$1/0,9 \cdot 10^9$	$1/8$	3556	80
ME7	$4/1,2 \cdot 10^9$	$1/8$	3556	20,11
Итого			28451	

Используя разработанный алгоритм, вычисляем оптимальные пропорции перераспределения нагрузки между серверами Middle-End. После этого, производим предельный расчет максимального числа пользователей системы, при котором загрузка CPU любой платформы не превышает 80%.

Достигнутый при оптимальном распределении запросов между платформами результат — около 73 141 активных пользователей на тех же аппаратных ресурсах. Вычисленная результирующая загрузка аппаратных средств ВОС представлена в таблице 5.

Таблица 5 - Моделирование при оптимизированном распределении

Сервер	кол-во ядер/частота	Доля от общего числа запросов	Количество пользователей	CPU, %
ME1	$1/1,2 \cdot 10^9$	0,065	4722	79,68
ME2	$1/1,2 \cdot 10^9$	0,065	4722	79,68
ME3	$1/2,4 \cdot 10^9$	0,127	9289	78,4
ME4	$1/2,4 \cdot 10^9$	0,127	9289	78,4
ME5	$2/2,0 \cdot 10^9$	0,187	13710	77,15
ME6	$2/2,0 \cdot 10^9$	0,187	13710	77,15
ME7	$1/0,9 \cdot 10^9$	0,049	3556	80,01
ME8	$4/1,2 \cdot 10^9$	0,19	14144	77,03
Итого			73141	

Используя разработанный алгоритм, проведем моделирование ситуации и вычисления пропорции перераспределения нагрузки между серверами Middle-End в условиях отказа одного из серверов (ME4). После этого, производим расчет максимального числа пользователей системы, при котором загрузка CPU любой платформы не превышает 80%.

Достигнутый при оптимальном распределении запросов между платформами результат — около 63 850 активных пользователей на тех же аппаратных ресурсах. Вычисленная результирующая загрузка аппаратных средств ВОС представлена в таблице 6.

Таблица 6 - Моделирование при аварии в одном из ЦОД

Сервер	кол-во ядер/частота	Доля от общего числа запросов	Количество пользователей	CPU, %
ME1	$1/1,2 \cdot 10^9$	0,074	4721	79,68
ME2	$1/1,2 \cdot 10^9$	0,074	4721	79,68
ME3	$1/2,4 \cdot 10^9$	0,145	9289	78,39
ME4	$1/2,4 \cdot 10^9$	0	0	0
ME5	$2/2,0 \cdot 10^9$	0,215	13709	77,15
ME6	$2/2,0 \cdot 10^9$	0,215	13709	77,15
ME7	$1/0,9 \cdot 10^9$	0,056	3556	80,01
ME8	$4/1,2 \cdot 10^9$	0,22	14144	77,03
Итого			63850	

Анализ полученных результатов показывает, что разработанный алгоритм перераспределил пропорции таким образом, чтобы сохранить равномерную загрузку аппаратных ресурсов в изменившихся условиях функционирования. При этом обеспечено сохранение равномерной загрузки между аппаратными средствами.

Теперь проведем моделирование загрузки CPU аппаратных средств при равной пользовательской нагрузке (20 тыс. пользователей) в условиях равномерного и оптимизированного распределения нагрузки. В таблице 7 представлены результаты вычислений.

Таблица 7 - Моделирование для 20 тыс. пользователей

Имя	кол-во ядер/частота	Вариант равномерного распределения		Вариант оптимизированного распределения	
		Сессий	CPU, %	Сессий	CPU, %
ME1	$1/1,2 \cdot 10^9$	2500	42,6	1303	22,7
ME2	$1/1,2 \cdot 10^9$	2500	42,6	1303	22,7
ME3	$1/2,4 \cdot 10^9$	2500	21,8	2546	22,2
ME4	$1/2,4 \cdot 10^9$	2500	21,8	2546	22,2
ME5	$2/2,0 \cdot 10^9$	2500	14,9	3734	21,7
ME6	$2/2,0 \cdot 10^9$	2500	14,9	3734	21,7
ME7	$1/0,9 \cdot 10^9$	2500	56,5	983	22,8
ME8	$4/1,2 \cdot 10^9$	2500	14,4	3850	21,7
Средняя загрузка, %			28,7		22,2

Проведенное моделирование показывает, что разработанный алгоритм обеспечивает решение задачи балансировки нагрузки в пределах ЦОД со следующими сводными характеристиками:

1) загрузка процессоров всех аппаратных платформ находится на максимально низком уровне при заданной величине нагрузки на ВОС для всех аппаратных платформ образующих ВОС (28,7 при равномерном распределении и 22,2 при использовании алгоритма оптимальной балансировки);

2) разброс прогнозируемой загрузки процессоров не превысил 4% в пределах одного архитектурного уровня одного ЦОД (77% - 80%);

3) для аппаратных платформ находящихся в аварийном состоянии величина направляемой на них нагрузки составляет 0% от общей нагрузки (доля запросов на ME4 выставлена в 0%).

Таким образом, использование разработанного алгоритма балансировки позволяет обеспечить эффективную балансировку и

горизонтальное масштабирование аппаратных средств серверных платформ в условиях разнообразия используемых аппаратных ресурсов внутри одной площадки размещения аппаратных средств. При этом, алгоритм обладает необходимыми средствами адаптации к изменяющимся условиям в функционировании аппаратных средств ОВС.

Работа поддерживается Министерством образования и науки Российской Федерации, ГК 14.514.11.4043.

### Список литературы:

1 Рынок e-commerce в России [Электронный ресурс] //Microsoft: [web-сайт] . - USA, 2011. - Режим доступа: <http://www.microsoft.com/ru-ru/cpe/ecommerce.aspx>. - 24.10.2012.

2 Мария Попова, Любовь Байдалина Рынок SaaS в России разгоняется и нагревается [Электронный ресурс] //CNewsAnalytics: [web-сайт] . - М, 2011. - Режим доступа: <http://www.cnews.ru/reviews/free/saas/articles/articles1.shtml>. - 24.10.2012

3 Best Practices for Performance in ISA Server 2004 [Electronic resource] //Microsoft: [web-сайт] . - USA, 2012. - Mode Access: <http://technet.microsoft.com/library/cc302518.aspx#PlanningISAServerCapacity>. - 26.03.2013.

4 Сергей Пахомов Эра многоядерных энергоэффективных процессоров [Электронный ресурс] //КомпьютерПресс: [web-сайт] . - М, 2006. - Режим доступа: <http://www.compress.ru/article.aspx?id=16962&iid=786>. - 26.03.2013.

### Referenses:

1 Rynok e-commerce v Rossii [Electronic resource] //Microsoft: [website]. - USA, 2011. - Mode Access: <http://www.microsoft.com/ru-ru/cpe/ecommerce.aspx>. - 24.10.2012

2 Marija Popova, Ljubov' Bajdalina Rynok SaaS v Rossii razgonjaetsja i nagrevaetsja [Electronic resource] //CNewsAnalytics: [website] . - M, 2011. - Mode Access: <http://www.cnews.ru/reviews/free/saas/articles/articles1.shtml>. - 24.10.2012

3 Best Practices for Performance in ISA Server 2004 [Electronic resource] //Microsoft: [website] . - USA, 2012. - Mode Access: <http://technet.microsoft.com/library/cc302518.aspx#PlanningISAServerCapacity>. - 26.03.2013

4 Sergej Pahomov Jera mnogojadernyh jenergojeffektivnyh processorov [Electronic resource] //Komp'juterPress: [website] . - M, 2006. - Mode Access: <http://www.compress.ru/article.aspx?id=16962&iid=786>. - 26.03.2013