

*На правах рукописи*

Шилов Сергей Николаевич

**МОДЕЛИ И АЛГОРИТМЫ БАЛАНСИРОВКИ НАГРУЗКИ  
В КЛАСТЕРНОЙ СИСТЕМЕ С ПОДДЕРЖКОЙ  
МЕХАНИЗМА РЕПЛИКАЦИИ**

Специальность

05.13.17 – Теоретические основы информатики

Автореферат

диссертации на соискание ученой степени  
кандидата технических наук

Воронеж - 2015

Работа выполнена в Федеральном государственном бюджетном образовательном учреждении высшего профессионального образования «Воронежский государственный университет»

Научный руководитель доктор физико-математических наук, профессор Кургалин Сергей Дмитриевич

Официальные оппоненты Арзамасцев Александр Анатольевич, доктор технических наук, профессор, ФГБОУ ВПО «Тамбовский государственный университет имени Г.Р. Державина», заведующий кафедрой математического моделирования и информационных технологий

Зольников Владимир Константинович, доктор технических наук, профессор, ФГБОУ ВО «Воронежский государственный лесотехнический университет имени Г.Ф. Морозова», заведующий кафедрой вычислительной техники и информационных систем

Ведущая организация – Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Белгородский государственный национальный исследовательский университет»

Защита состоится 24 декабря 2015 года в 15:30 на заседании совета Д 212.038.24 при Федеральном государственном бюджетном образовательном учреждении высшего профессионального образования «Воронежский государственный университет» по адресу: 394006, г. Воронеж, Университетская площадь, д.1, ауд. 226.

С диссертацией можно ознакомиться в научной библиотеке Федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Воронежский государственный университет» и на сайте <http://www.science.vsu.ru>.

Автореферат разослан «        » ноября 2015 г.

Ученый секретарь  
диссертационного совета



Воронина Ирина Евгеньевна

## Общая характеристика работы

**Актуальность темы.** В настоящее время крайне актуальной задачей является разработка программных систем для новых информационных технологий, таких как программные комплексы балансировки нагрузки в компьютерных кластерных системах на основе распределенных хеш-таблиц.

Так как кластерная система состоит не менее чем из двух компьютеров, то возникает необходимость в программном комплексе балансировки нагрузки на узлы кластерной системы, необходимом для эффективного использования объединенных ресурсов. При наличии соответствующих механизмов программный комплекс, обеспечивающий функционирование кластерной системы, также позволяет организовать резервное копирование данных, сохраняя работоспособность системы при временных или безвозвратных выходах узлов.

Таким образом, актуальность темы диссертационных исследований обусловлена бурным ростом популярности кластерных систем, все больше демонстрирующих свою высокую эффективность, что позволяет говорить о кластеризации как о фундаментальной основе развития современных информационных технологий. Разработки в этой области ведутся крупнейшими университетами мира (Массачусетский технологический институт, Калифорнийский университет в Беркли, Принстонский университет и т.д.), а также ведущими мировыми производителями программного и аппаратного обеспечения, такими как Google, IBM, Microsoft, Sun, Amazon, Apache и т.д.

Представленная диссертационная работа посвящена разработке, реализации и исследованию моделей и алгоритмов, составляющих программный комплекс балансировки нагрузки и репликации в кластерных системах. Одной из основных областей применения комплекса являются кластеры, составленные из рекурсивных кэширующих DNS-серверов. В основе разработанных моделей и алгоритмов лежит инновационная технология распределенной хеш-таблицы. Один из наиболее успешных подходов к созданию распределенных хеш-таблиц, называемый консистентным хешированием, был предложен в работах D. Karger, E. Lehman, T. Leighton et al. (Кембридж, США), а в работе I. Stoica, R. Morris, D. Karger et al. (Кембридж, США) представлена соответствующая реализация распределенной хеш-таблицы.

В рамках данной диссертации предложен оригинальный подход к созданию распределенной хеш-таблицы, который позволил реализовать систему балансировки нагрузки, соответствующую предъявляемым к ней требованиям. Эффективность работы программного комплекса достигнута за счет специальным образом подготовленной инфраструктуры, которая позволяет определить узел, ответственный за элемент данных, с использованием нескольких арифметических операций. Также в рамках данной работы были разработаны и реализованы оригинальные алгоритмы репликации как часть программного комплекса балансировки нагрузки.

**Цель и задачи исследования.** Цель работы – разработка и анализ моделей и алгоритмов балансировки нагрузки в кластерной системе с поддержкой механизма репликации.

Для достижения цели в работе решались следующие задачи:

1. Разработка модели балансировки нагрузки на основе распределенной хеш-таблицы, соответствующей требованиям масштабируемости, быстродействия и равномерности распределения нагрузки.

2. Создание алгоритма поиска узла, ответственного за обработку элемента данных.

3. Создание алгоритмов репликации, повышающих стабильность, надежность и отказоустойчивость кластерной системы.

4. Разработка и анализ алгоритмов работы предложенных моделей, их реализация в форме комплекса компьютерных программ, проведение статистических исследований для верификации и тестирования программной реализации.

**Объект исследования** – кластерные системы.

**Предмет исследования** – модели и алгоритмы балансировки нагрузки и репликации в кластерной системе.

**Методы исследования.** При выполнении работы использовались: математические и статистические методы обработки данных, методы теории алгоритмов, теории вероятностей.

**Новизна работы:**

1. Предложена модель балансировки нагрузки в кластерной системе, отличительными чертами которой являются оригинальная схема разбиения области значений базовой хеш-функции, отсутствие необходимости хранения метаданных на постоянном запоминающем устройстве для обеспечения функционирования алгоритмов и гарантированное определение ответственного узла за один цикл поиска.

2. Разработан алгоритм поиска узла, ответственного за обработку элемента данных, имеющий константную временную сложность и не ограничивающий масштабируемость системы.

3. Созданы алгоритмы репликации, обеспечивающие наличие заданного числа резервных копий в зависимости от конкретных требований к надежности и отказоустойчивости системы, имеющие константную временную сложность и не ограничивающие масштабируемость системы.

4. На основе разработанной модели распределения нагрузки созданы и исследованы новые алгоритмы, отличающиеся высоким быстродействием, проведено тестирование и верификация их программной реализации.

**Практическая значимость** результатов работы заключается в том, что предложенные модели, алгоритмы и программы можно использовать в различных кластерных системах, где требуется обеспечить равномерное распределение нагрузки на узлы, масштабируемость в режиме реального времени без остановки функционирования системы, а также производить резервное копирование данных. Также возможно их широкое использование в области распределенных файловых систем для равномерного размещения метаданных на выделенных для данной цели узлах. За счет высокого быстродействия предложенные алгоритмы позволяют существенно повысить эффективность работы кластерных систем, а наличие алгоритмов репликации обеспечивает их надежность и отказоустойчивость.

**Область исследования** – содержание диссертации соответствует паспорту специальности 05.13.17 – «Теоретические основы информатики» (технические науки), область исследований соответствует п. 1 «Исследование, в том числе с помощью средств вычислительной техники, информационных процессов, информационных потребностей коллективных и индивидуальных пользователей»; п. 2 «Исследование информационных структур, разработка и анализ моделей

информационных процессов и структур»; п. 14 «Разработка теоретических основ создания программных систем для новых информационных технологий».

**Реализация результатов исследования.** Результаты диссертации в форме комплексной системы распределения нагрузки с поддержкой механизма репликации ресурсных записей используются в проекте DNS-сервиса, реализуемом компанией «Релэкс» (г. Воронеж). В рамках данного проекта система успешно функционирует в составе ряда кластеров рекурсивных кэширующих DNS-серверов, расположенных в Европе и США.

#### **Основные результаты, выносимые на защиту:**

1. Модель балансировки нагрузки в кластерной системе, позволяющая избежать необходимости хранения метаданных на постоянном запоминающем устройстве для обеспечения функционирования алгоритмов, а также соответствующая требованиям масштабируемости, быстродействия и равномерности распределения нагрузки.

2. Алгоритм поиска узла, ответственного за обработку элемента данных, имеющий константную временную сложность и не ограничивающий масштабируемость системы.

3. Алгоритмы репликации, обеспечивающие наличие заданного числа резервных копий в зависимости от конкретных требований к надежности и отказоустойчивости системы, имеющие константную временную сложность и не ограничивающие масштабируемость системы.

4. Алгоритмы, построенные на основе разработанной модели распределения нагрузки, а также результаты их программной реализации.

**Апробация работы.** Основные положения диссертационной работы были представлены на XXI всероссийской научно-методической конференции «Телематика'2014», г. Санкт-Петербург, 2014 г.; XI, XII и XIV международных научно-методических конференциях «Информатика: проблемы, методология, технологии», г. Воронеж, 2011-2012, 2014 гг.; X международном семинаре «Физико-математическое моделирование систем», г. Воронеж, 2013 г. Результаты диссертационного исследования прикладного и теоретического характера нашли применение в проекте, реализуемом группой компаний «Релэкс» (г. Воронеж), внедрение результатов подтверждено соответствующим актом.

**Публикации.** Основные результаты диссертации опубликованы в 11 печатных изданиях [1-11], в том числе в четырех [1-4] – из списка изданий, рекомендованных ВАК РФ. Получены два свидетельства о государственной регистрации программ для ЭВМ [12, 13].

**Личный вклад автора.** Основные результаты по теме диссертации получены лично автором. Постановка задач диссертации предложена автором и научным руководителем. Выбор базисных принципов модели балансировки нагрузки, разработка схемы разбиения области значений хеш-функции на области ответственности узлов кластерной системы, алгоритмов репликации проводились лично автором. Реализация в виде комплекса алгоритмов, информационных структур и программ, исследование, проверка достоверности результатов, получение выводов и их интерпретация выполнены автором.

**Структура и объем работы.** Диссертация состоит из введения, четырех глав, заключения, списка использованных источников, включающего 60

наименований, и 4 приложений. Объем диссертации составляет 143 страницы, включая 126 страниц основного текста, содержащего 37 рисунков.

## СОДЕРЖАНИЕ РАБОТЫ

**Во введении** обоснована актуальность темы диссертационной работы, сформулированы цели и задачи исследования, определены научная новизна и практическая значимость. Обсуждается проблема построения кластерных систем. Ставится задача разработки модели и алгоритмов балансировки нагрузки с учетом особенностей сервиса DNS, алгоритмов репликации ресурсных записей, а также их реализации в виде комплекса программ.

**Первая глава** посвящена рассмотрению подходов к построению кластерных систем балансировки нагрузки, обзору и анализу существующих реализаций распределенных хеш-таблиц. В главе раскрыто понятие компьютерного кластера, рассмотрены основные виды кластеров, описаны их свойства и особенности. Также представлены понятие и структура распределенных хеш-таблиц, их назначение и области применения. Основной целевой областью применения разработанных моделей и алгоритмов является система доменных имен, имеющая свои свойства, особенности и ограничения, описанные в главе.

Распределенная хеш-таблица (Distributed Hash Table, DHT) – это класс децентрализованных распределенных систем, которые обеспечивают модель поиска, похожую по принципу работы на классическую хеш-таблицу, при этом каждый участвующий в кластерной системе узел может рациональным образом осуществлять поиск значения, ассоциированного с данным ключом. Свойство балансировки нагрузки вытекает из использования базовой хеш-функции как части распределенной хеш-таблицы, где данная функция генерирует псевдослучайные значения для входящих последовательностей.

Первые четыре реализации распределенных хеш-таблиц – CAN, Chord, Pastry и Tapestry – были выполнены в 2001-2004 гг. В работе S. Ratnasamy, P. Francis, M. Handley et al. (2001 г.) изложены основные принципы построения CAN, которая обеспечивает функциональность хеш-таблицы в масштабах Интернета. Работа I. Stoica, R. Morris, D. Karger et al. (2001 г.) описывает систему Chord, где идентификаторы и ключи получаются в результате так называемого консистентного хеширования (кольцевидного представления пространства ключей). Инновационная идея принципа консистентного хеширования, позволяющего избежать полного перераспределения ключей при изменении состава участвующих узлов, впервые было изложена в работе D. Karger, E. Lehman, T. Leighton et al. в 1997 г. Отличительной чертой систем Pastry (A. Rowstron, P. Druschel, 2001 г.) и Tapestry (B. Y. Zhao, L. Huang, J. Striblin et al., 2004 г.) является построение локальных оптимальных таблиц маршрутизации для более эффективного обмена сообщениями, при этом Pastry реализует работу с распределенной хеш-таблицей аналогично Chord.

Что касается применения систем, основанных на технологии распределенной хеш-таблицы, в сфере системы доменных имен, то существует работа R. Cox, A. Muthitacharoen, R. T. Morris (2002 г.), описывающая реализацию сервиса поиска ресурсных записей на основе Chord.

Особое влияние на идеи, изложенные в данной диссертации, оказала работа G. DeCandia, D. Hastorun, M. Jampani et al. (2007 г.), в которой приведено подробное описание распределенного хранилища высокой доступности Dynamo от Amazon.

Итогом рассмотрения и анализа существующих методов построения систем распределения нагрузки, а также изучения свойств и особенностей системы доменных имен являются следующие выводы: наиболее эффективный подход состоит в использовании распределенной хеш-таблицы как базисной технологии комплекса алгоритмов балансировки нагрузки. Помимо непосредственно распределения нагрузки данный подход позволяет в полной мере обеспечить реализацию важнейших преимуществ кластерных систем: масштабируемость и высокую доступность предоставляемого сервиса. Также на базе данной технологии возможна реализация механизмов репликации данных, позволяющих повысить стабильность, надежность и отказоустойчивость кластерных систем.

**Вторая глава** посвящена разработке модели балансировки нагрузки в кластерной системе с учетом особенностей системы доменных имен. Разработаны элементы модели балансировки, включающие в себя схему разбиения области значений базовой хеш-функции на области ответственности узлов кластера с учетом требований масштабируемости системы и равномерности распределения нагрузки на узлы, а также аспект хранения схемы разбиения, определяющий способ ее хранения в виде таблиц вариантов распределения. Создан алгоритм поиска ответственного за домен узла, проанализированы ситуации входа и выхода узлов из состава кластера.

В данной диссертации модель балансировки основывается на факте существования в многоканальной системе массового обслуживания устройства, которое для конкретной заявки является предпочтительным (выделенным) за счет существенного сокращения времени обслуживания. Критерием определения подобного устройства служит некоторый признак заявки. В работе было показано, что отношение среднего времени обслуживания заявки ( $\bar{t}$ ) при случайном выборе обрабатывающего устройства и времени обслуживания на выделенном устройстве ( $t_1$ ) определяется следующим соотношением:

$$\frac{\bar{t}}{t_1} = \frac{n\mu_1 - \mu_1 + \mu_2}{n\mu_2}, \quad (1)$$

где  $n$  – количество обрабатывающих устройств,  $\mu_1$  – интенсивность потока обслуживания на выделенном для заявки устройстве,  $\mu_2$  – интенсивность потока обслуживания на устройстве, которое не является выделенным для данной заявки.

Компоненты разработанной модели балансировки нагрузки:

- схема разбиения области значений базовой хеш-функции на области ответственности узлов кластерной системы;
- способ хранения схемы разбиения в виде таблиц вариантов распределения.

Схема разбиения области значений хеш-функции базируется на механизме консистентного хеширования, который при переконфигурировании кластера сохраняет связь между ключами и узлами. Основная идея данного подхода состоит в следующем: вся область значений хеш-функции (например, в случае 64-х битной хеш-функции область ее значений будет составлять отрезок  $[0, 2^{64}-1]$ ) представляется в виде кольца, в котором пространство «склеивается» в своем максимальном и минимальном значениях (0 и  $2^{64}-1$ ). Каждый узел получает некоторую позицию на кольце, при этом два соседних узла образуют область ответственности, за которую отвечает один из них (с большим или меньшим идентификатором). Каждый элемент

данных идентифицируется ключом (путем вычисления хеш-функции) и также получает некоторую позицию на кольце, попадая в область ответственности какого-либо узла. Так элемент данных связывается с определенным узлом системы (рис. 1).

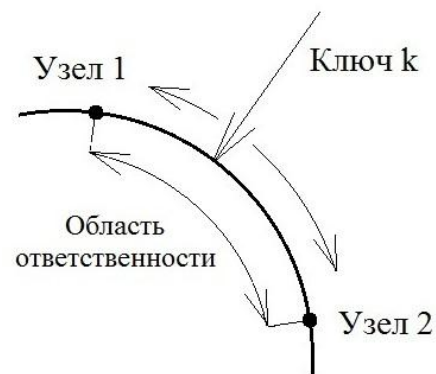


Рис. 1. Отображение ключа в узел кластерной системы

Основной проблемой при реализации распределенной хеш-таблицы является схема разбиения пространства ключей между участниками кластера.

После проведенного анализа была реализована следующая схема разбиения области значений хеш-функции. В данной работе была использована хеш-функция, генерирующая 64-битные ключи, следовательно, пространство ключей имеет размер  $2^{64}$ . Все это пространство разбивается на  $2^{32}$  одинаковых частей, называемых блоками. В свою очередь, каждый блок в равных частях делится на количество узлов кластера. Получившиеся отрезки назовем областями ответственности узлов.

Для того чтобы при выходе узла из состава кластера участки его областей ответственности не переходили одним и тем же соседним узлам после деления блоков на количество узлов нового состава, порядок следования областей ответственности узлов должен быть неодинаковым для равномерности перераспределения запросов. Поэтому было решено ввести 512 вариантов распределения узлов, повторяющихся циклично на всем пространстве области значений хеш-функции, и хранить их в двумерном массиве, называемом таблицей вариантов распределения (концепция хранения схемы разбиения).

При построении варианта распределения для каждого узла рассчитывается хеш-значение на основе комбинации его IP-адреса и номера варианта, после чего полученные значения сортируются в порядке возрастания. В случае возникновения коллизий (совпадения хеш-значений) происходит сравнение значений IP-адресов узлов в численном представлении. Полученный порядок следования хеш-значений определяет позиции узлов в каждом варианте распределения. Отображение таблицы вариантов распределения на область значений хеш-функции представлено на рис. 2.

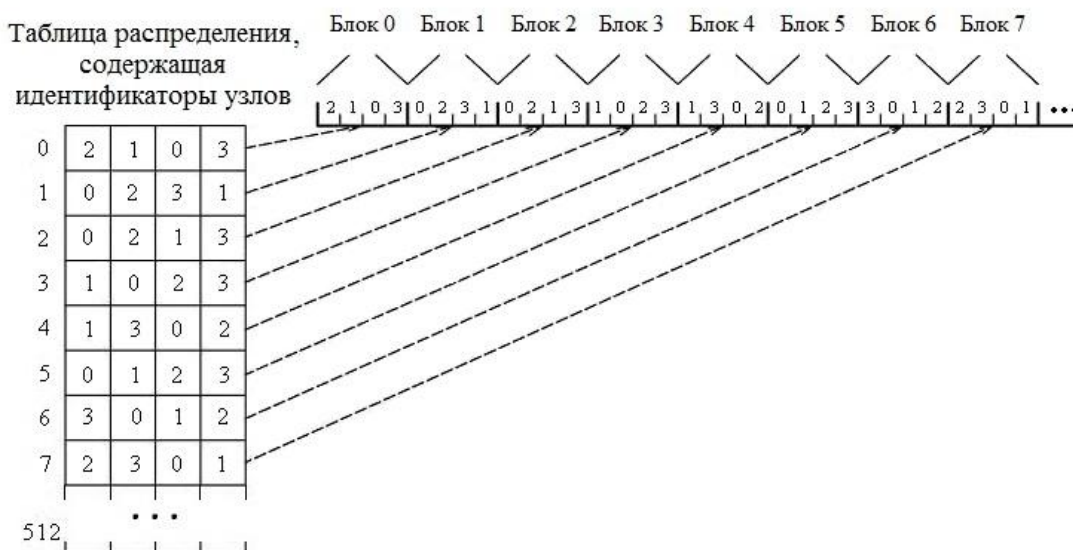


Рис. 2. Отображение таблицы вариантов распределения на область значений хеш-функции



**Утверждение 1.** Пусть  $L_1, L_2, \dots, L_n$  – суммарные размеры областей ответственности узлов с идентификаторами  $1, 2, \dots, n$  соответственно,  $n$  – количество узлов в кластерной системе. Тогда при отсутствии неработоспособных узлов каждый из узлов системы суммарно отвечает за равную часть области значений хеш-функции:

$$L_1 = L_2 = \dots = L_n. \quad (2)$$

В этой главе также представлены две модели организации таблиц вариантов распределения: одноуровневая и двухуровневая. В одноуровневой модели строится одна таблица распределения, при этом в описанном выше алгоритме построения учитываются только «живые» (т.е. функционирующие в нормальном режиме) узлы кластера. Данная модель имеет высокие показатели равномерности распределения запросов, а также легко масштабируется. Недостаток одноуровневой модели состоит в наличии эффекта смещения пространств ответственности узлов при выходе какого-либо узла из состава кластера, что приводит к миграции некоторого количества ключей, т.к. пространство ключей всегда делится только между «живыми» узлами.

Решением данной проблемы является двухуровневая модель, где присутствует две таблицы распределения. При этом в таблицу распределения 1-го уровня записываются все узлы (включая «мертвые», т.е. не функционирующие в нормальном режиме), а в таблицу 2-го уровня – только «живые» (аналогично одноуровневой схеме).

В главе детально рассмотрены алгоритмы поиска ответственного узла в одноуровневой и двухуровневой моделях. Для двухуровневой модели блок-схема алгоритма представлена на рис. 3.

При поступлении домена на обработку сначала для него рассчитывается соответствующее хеш-значение посредством хеш-функции. Далее на первом шаге алгоритм выполняет попытку определения ответственного узла, используя таблицу 1-го уровня, которая может содержать «мертвые» узлы. При попадании хеш-значения в область ответственности «живого» узла данный узел выбирается ответственным, и на этом алгоритм прекращает работу без использования таблицы 2-го уровня. Однако если хеш-значение попало в область ответственности «мертвого» в данный момент узла, алгоритм на втором шаге принимает решение, используя таблицу 2-го уровня, где ключ гарантированно попадает в об-

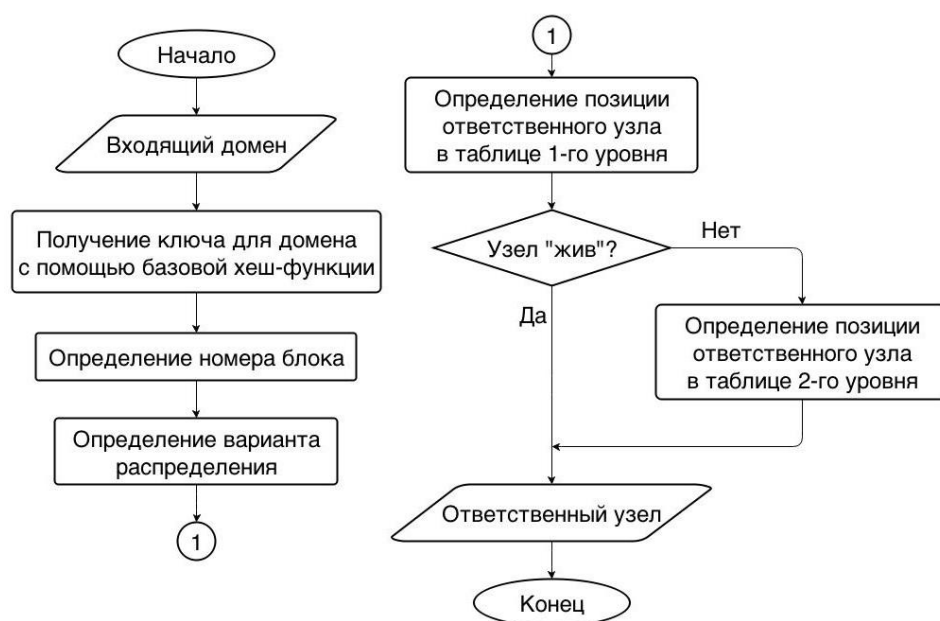


Рис. 3. Блок-схема алгоритма поиска ответственного узла с применением таблиц 1-го и 2-го уровней

ласть ответственности «живого» узла.

**Утверждение 2.** Для любого элемента данных, идентифицируемого базовой хеш-функцией, существует ответственный узел в рамках текущего непустого состава кластерной системы:

$$\forall d \in D \exists s \in S, S \neq \emptyset, \quad (3)$$

где  $D$  – множество элементов данных, которые могут быть идентифицированы базовой хеш-функцией,  $S$  – непустое множество узлов кластерной системы.

Непосредственно алгоритм поиска ответственного узла является компактным и содержит малое количество операций, что определяет его высокое быстродействие.

Временная сложность алгоритма поиска ответственного узла как в случае одноуровневой, так и в случае двухуровневой модели не зависит от количества узлов кластера и определяется как  $O(C)$ . Таким образом, алгоритм будет одинаково быстро работать при десятках и тысячах участвующих узлов. Этот факт позволяет заключить, что данный алгоритм никаким образом не ограничивает масштабируемость кластерной системы.

Временная сложность алгоритма построения таблиц вариантов распределения определяется как  $O(N \log(N))$ , где  $N$  – количество узлов кластера.

Так как в распределенной хеш-таблице используется хеш-функция, генерирующая ключи для элементов данных, то неизбежны ситуации, при которых возникают коллизии, т.е. получение одинаковых ключей для различных входных последовательностей. Однако в случае распределенных хеш-таблиц не требуется наличие специального механизма разрешения коллизий. Полученный с помощью хеш-функции ключ участвует только в операции определения узла, ответственного за обработку запроса. Далее вся работа происходит непосредственно с элементом данных без участия ассоциированного с ним ключа. Таким образом, получение одинаковых ключей для различных элементов данных означает лишь их принадлежность одному и тому же узлу кластера.

**Третья глава** посвящена разработке и анализу алгоритмов репликации ресурсных записей на основе схемы разбиения пространства ключей на области ответственности узлов кластера, описанной во второй главе. Представлены и проанализированы два оригинальных алгоритма, первый из которых основан на взаимном перекрытии областей ответственности узлов, второй базируется на использовании некоторого количества ближайших к ответственному узлу областей ответственности.

В результате анализа проблемы, проектирования и разработки алгоритмов было реализовано два алгоритма репликации.

В главе предложен первый вариант алгоритма репликации, основанный на принципе пересечения областей ответственности узлов. Основная идея состоит во взаимном перекрытии отрезков ответственности, в соответствии с которой определяется не только узел, отвечающий за данный участок пространства ключей, но и узел, чей отрезок имеет с ним зону перекрытия. При попадании ключа в область перекрытия соответствующий узел будет выбран для репликации данных. Размер области перекрытия является настраиваемым параметром.

Блок-схема такого алгоритма репликации представлена на рис. 4.

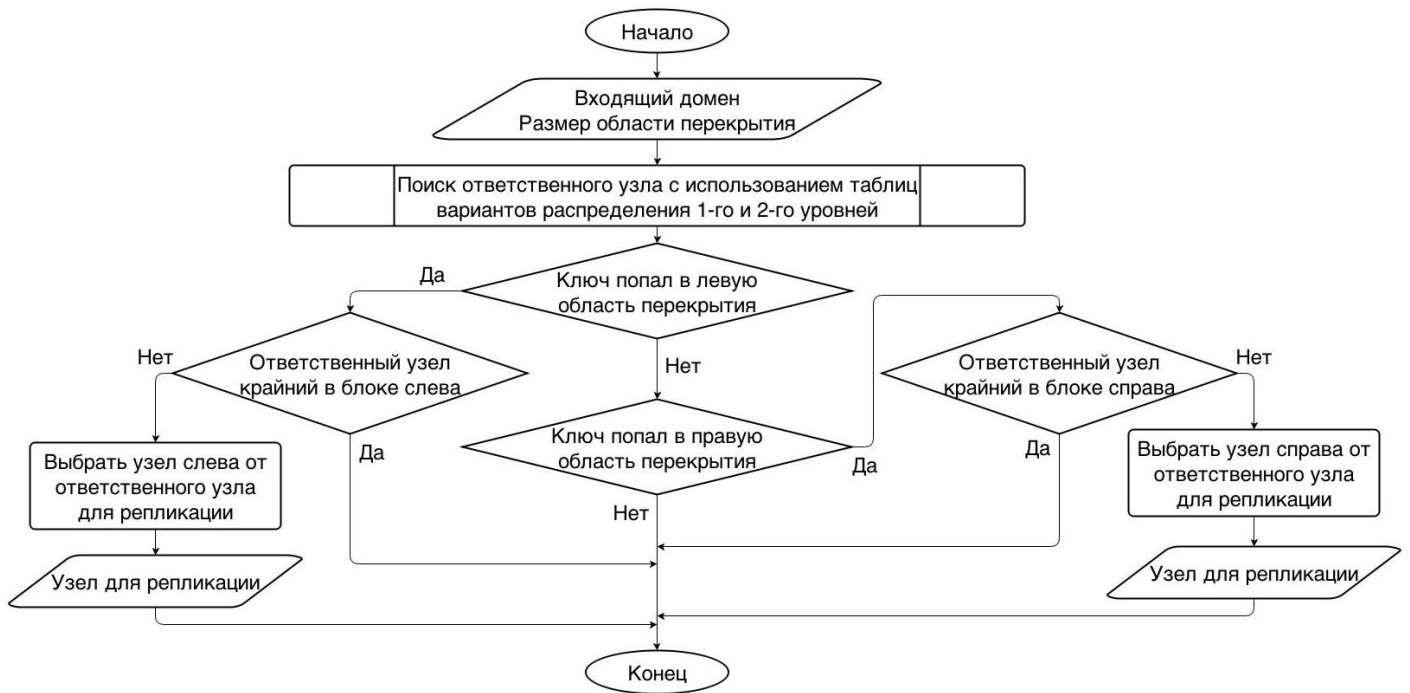


Рис. 4. Блок-схема алгоритма репликации на основе взаимного перекрытия областей ответственности узлов

В главе также предложен другой алгоритм репликации (рис. 5). Основная его идея состоит в том, чтобы производить репликацию только на четное количество узлов, которые потенциально могут стать ответственными за данный домен. Алгоритм позволяет задавать уровень репликации, т.е. число копий: 2, 4, 6 и т.д.



Рис. 5. Блок-схема алгоритма репликации на основе использования ближайших к ответственному узлу областей ответственности

Анализ алгоритмов репликации показывает, что второй из рассмотренных вариантов с репликацией на четное количество узлов выглядит предпочтительнее первого, где репликация производится на основе перекрытия участков ответственности, прежде всего за счет возможности иметь множество копий ресурсных записей, вплоть до резервного копирования на все узлы кластера.

Однако первый вариант алгоритма репликации также может иметь преимущества в определенных ситуациях, например, при ограниченных ресурсах памяти, где репликация на четное количество узлов может стать слишком ресурсоемкой. В таком случае подобная «частичная» репликация поможет повысить стабильность работы кластера и его надежность при небольших затратах памяти на хранение части реплицированных ресурсных записей.

Временная сложность алгоритма репликации, основанного на взаимном перекрытии областей ответственности узлов, не зависит от количества узлов кластера и является константной, т.е. определяется как  $O(C)$ . Временная сложность алгоритма репликации, основанного на использовании некоторого количества ближайших областей ответственности узлов, не зависит от количества узлов кластера и является константной, т.е. определяется как  $O(C)$ .

Таким образом, оба варианта алгоритма репликации не ограничивают масштабируемость кластерной системы.

В конечном итоге структурная схема взаимодействия основных модулей выглядит так, как представлено на рис. 6.

Так как к системе предъявлялись особые требования по быстродействию и уровню производительности, то для реализации программного комплекса в качестве языка программирования был выбран язык С. Программный комплекс способен функционировать на семействах операционных систем Linux и Windows.

**Четвертая глава** посвящена исследованию алгоритмов балансировки нагрузки. В данной главе приведены различные статистические показатели, найденные при использовании тестового набора доменов, проанализирована их динамика с ростом числа принятых системой запросов. Также для тестового набора доменов и доменов, взятых из DNS-трафика от реальных пользователей, произведена проверка качества алгоритмов в сфере равномерности распределения нагрузки.

Для получения статистических данных было собрано  $7 \times 10^6$  уникальных доменов. Чтобы статистические оценки были наиболее близкими к реальности, размер тестового кластера был выбран равным пяти узлам. Это количество соответствует среднему размеру DNS-кластеров, используемых в рамках упомянутого выше проекта компании «Релэкс», где были внедрены результаты диссертации, для обслуживания клиентов. Данной тестовой системе посы-



Рис. 6. Структурная схема взаимодействия основных модулей

лалось различное количество уникальных запросов из собранного множества доменов: 50, 100, 500, 1000 и т.д. При этом фиксировалось число попавших на каждый узел запросов и на основе этих данных рассчитывались абсолютные и относительные статистические показатели.

Особый интерес в полученных оценках вызывают относительные статистические показатели, такие как коэффициент вариации. Получено, что коэффициент вариации убывает с ростом числа входящих уникальных запросов. Т.е. показатели равномерности распределения улучшаются с течением времени работы системы при наличии входящего потока уникальных доменов. Наглядно данный факт продемонстрирован на рис. 7, где показана зависимость значения коэффициента вариации от количества поступивших в систему запросов.

На графике, изображенном на рис. 7, присутствуют отсечки на 50000 и 100000 запросов. На данном интервале статистические показатели системы изменяются крайне незначительно (в частности, наблюдается изменение коэффициента вариации в пределах 0.05%). Данный факт помогает выявить достаточный объем выборки для

применения критерия согласия Пирсона. Данный критерий использовался для проверки соответствия числа запросов, получаемых каждым из узлов кластера, равномерному закону распределения.

Для получения статистических данных, необходимых для применения критерия согласия Пирсона, тестовой системе, включающей 5 узлов, было по-

слано  $10^5$  уникальных запросов. Промежуточные результаты алгоритма критерия и итоговое фактическое значение  $\chi^2$  представлены в табл. 1.

Таблица 1

Расчет фактического значения  $\chi^2$  для  $10^5$  уникальных запросов

№ узла (i)	Факт. кол-во запросов ( $n_i$ )	Теор. кол-во запросов ( $np_i$ )	$n_i - np_i$	$\frac{(n_i - np_i)^2}{np_i}$
1	19937	20000	-63	0.19845
2	20128	20000	128	0.8192
3	19953	20000	-47	0.11045
4	19897	20000	-103	0.53045
5	20085	20000	85	0.36125
				$\chi^2 = 2.0198$

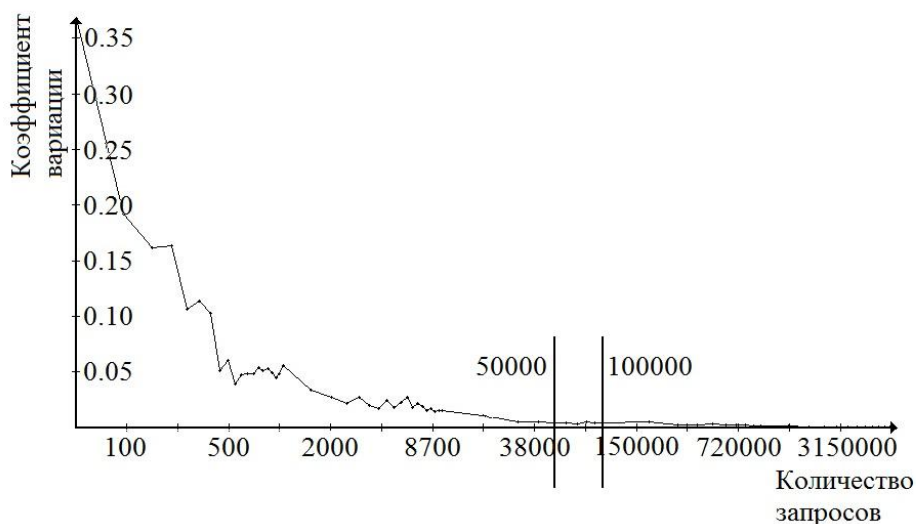


Рис. 7. Зависимость коэффициента вариации от количества входящих запросов

При выборе уровня значимости равным 0.05 (число степеней свободы – 3) критическое значение  $\chi^2$  составляет 7.81. Полученное фактическое значение  $\chi^2$  равно 2.0198, что значительно ниже табличного критического значения, поэтому можно сделать вывод, что теоретическое равномерное распределение удовлетворительно описывает эмпирические данные.

Также была получена оценка для DNS-трафика реальных пользователей, для этого было также собрано  $10^5$  доменов (табл. 2).

Таблица 2

Расчет фактического значения  $\chi^2$  для  $10^5$  запросов из DNS-трафика пользователей

№ узла (i)	Факт. кол-во запросов ( $n_i$ )	Теор. кол-во запросов ( $np_i$ )	$n_i - np_i$	$\frac{(n_i - np_i)^2}{np_i}$
1	19932	20000	-68	0.2312
2	20321	20000	321	5.15205
3	19883	20000	-117	0.68445
4	19990	20000	-10	0.005
5	19874	20000	-126	0.7938
				$\chi^2 = 6.8665$

Для данной выборки фактическое значение  $\chi^2$  равно 6.8665. Это значение превышает величину  $\chi^2$ , полученную при исследовании статистических свойств системы с использованием уникальных доменов, где она была равна 2.0198. Однако это значение также ниже критического 7.81 (как и в первом случае, уровень значимости – 0.05). Таким образом, опытные данные, полученные при участии DNS-трафика реальных пользователей, также не противоречат равномерному закону распределения.

На основании проведенных статистических исследований можно сделать вывод, что рассматриваемый комплекс программ полностью соответствует предъявляемым к ним требованиям о равномерности распределения входящих DNS-запросов (а, следовательно, и суммарной нагрузки) среди узлов кластера.

**В заключении сформулированы основные результаты работы:**

1. Разработана модель балансировки нагрузки в кластерной системе, позволяющая избежать необходимости хранения метаданных на постоянном запоминающем устройстве для обеспечения функционирования алгоритмов, а также соответствующая требованиям масштабируемости, быстродействия и равномерности распределения нагрузки.

2. Создан алгоритм поиска узла, ответственного за обработку элемента данных, имеющий константную временную сложность и не ограничивающий масштабируемость системы.

3. Созданы алгоритмы репликации, обеспечивающие наличие заданного числа резервных копий в зависимости от конкретных требований к надежности и отказоустойчивости системы, имеющие константную временную сложность и не ограничивающие масштабируемость системы.

4. На основе разработанной модели распределения нагрузки созданы и исследованы новые алгоритмы, отличающиеся высоким быстродействием, проведено тестирование и верификация их программной реализации.

**Основные публикации по теме диссертации:**

1. Шилов С.Н. Анализ и реализация механизма репликации ресурсных записей в DNS кластере / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Информационные технологии. – 2014. – № 6. – С. 38-43.

2. Шилов С.Н. Реализация инфраструктуры распределенной хеш-таблицы в рамках кластерной системы DNS / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Вестник ВГУ. Серия: Системный анализ и информационные технологии. – 2012. – № 2. – С. 74-79.

3. Шилов С.Н. Двухуровневая схема организации таблиц распределения запросов в кластерной системе DNS / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Вестник ВГУ. Серия: Системный анализ и информационные технологии. – 2014. – № 1. – С. 90-96.

4. Шилов С.Н. Статистическое исследование системы распределения нагрузки в DNS кластере / С.Н. Шилов. Системы управления и информационные технологии. – 2014. – № 3(57). – С. 96-99.

5. Шилов С.Н. Вариант реализации распределения ключей и разбиения пространства ключей в рамках распределенной хеш-таблицы (keys distribution and partition) / С.Н. Шилов, А.А. Крыловецкий // Информатика: проблемы, методология, технологии : материалы XI международ. научно-методической конф. – Воронеж, 2011. – Т. 2. – С. 448-454.

6. Шилов С.Н. Выявление идентичных файлов и последовательностей данных с помощью технологии Context Triggered Piecewise Hashing (Fuzzy Hash) / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Информатика: проблемы, методология, технологии : материалы XII международ. научно-методической конф. – Воронеж, 2012. – Т. 1. – С. 450-460.

7. Шилов С.Н. Моделирование распределения нагрузки в кластерной системе серверов DNS на основе использования распределенной хеш-таблицы / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Физико-математическое моделирование систем: материалы X Международ. семинара. – Воронеж, 2013. – С. 32-38.

8. Шилов С.Н. Усовершенствованный вариант реализации системы распределения нагрузки в DNS кластере / С.Н. Шилов. Информатика: проблемы, методология, технологии : материалы XIV международ. научно-методической конф. – Воронеж, 2014. – Т. 3. – С. 214-218.

9. Шилов С.Н. Реализация системы распределения нагрузки в DNS-кластере на основе распределенной хеш-таблицы / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Телематика'2014 : материалы XXI Всерос. научно-методической конф. – СПб., 2014. – С. 121-122.

10. Шилов С.Н. Реализация механизма репликации ресурсных записей в составе системы распределения нагрузки в DNS-кластере / С.Н. Шилов, С.Д. Кургалин, А.А. Крыловецкий // Телематика'2014 : материалы XXI Всерос. научно-методической конф. – СПб., 2014. – С. 122-123.

11. Шилов С.Н. Анализ статистических показателей системы распределения нагрузки в DNS кластере / С.Н. Шилов // Информационные технологии моделирования и управления. – 2014. – № 4(88). – С. 341-347.
12. Свидетельство о государственной регистрации программы для ЭВМ «Система локальной эмуляции окружения DNS кластера и расчета статистических показателей системы распределения нагрузки» / С.Н. Шилов. – № 2014619700, зарегистрировано в Реестре программ для ЭВМ 19 сентября 2014 г.
13. Свидетельство о государственной регистрации программы для ЭВМ «Система распределения нагрузки в DNS кластере с поддержкой механизма репликации» / С.Н. Шилов. – № 2014618308, зарегистрировано в Реестре программ для ЭВМ 14 августа 2014 г.

**Работы № 1–4 опубликованы в изданиях, соответствующих перечню ВАК РФ.**