

На правах рукописи

Першин Алексей Владимирович

СИСТЕМЫ МОБИЛЬНЫХ АГЕНТОВ С НАСТРАИВАЕМОЙ  
АРХИТЕКТУРОЙ

Специальность: 05.13.11 – Математическое и программное обеспечение  
вычислительных машин, комплексов и компьютерных сетей

Автореферат

диссертации на соискание ученой степени  
кандидата технических наук

Санкт –Петербург –2011

Работа выполнена в Санкт-Петербургском государственном электротехническом университете «ЛЭТИ» им. В. И. Ульянова (Ленина).

Научный руководитель: доктор технических наук, профессор  
Куприянов Михаил Степанович

Официальные оппоненты: доктор технических наук, профессор  
Кутузов Олег Иванович

кандидат технических наук, ведущий инженер  
ЗАО «Моторола Солюшнз»  
Смирнов Александр Николаевич

Ведущая организация: ОАО «Информационные телекоммуникационные технологии» (г. Санкт-Петербург)

Защита состоится \_\_ октября в \_\_\_\_\_ часов на заседании совета по защите докторских и кандидатских диссертаций ДС 212.238.01 Санкт-Петербургского государственного электротехнического университета «ЛЭТИ» им. В. И. Ульянова (Ленина) по адресу: 197376, Санкт-Петербург, ул. Проф. Попова, д.5.

С диссертацией можно ознакомиться в библиотеке университета.

Автореферат разослан «\_\_» сентября 2011г.

Ученый секретарь совета  
по защите докторских  
и кандидатских диссертаций

Щеголева Н.Л.

# ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность темы.** Непрерывное развитие информационных компьютерных систем способствует всё большему их проникновению во все сферы деятельности. С увеличением количества пользователей требуется все бóльшая производительность, улучшение качества обслуживания и надёжности подобных систем. В работе рассматривается класс распределенных систем поиска, которые предназначены для поиска и бронирования товаров и услуг.

При разработке распределенных систем широко используется агентный подход, который объединяет в себе преимущества таких технологий, как объектно-ориентированные и распределённые программные системы, компонентная концепция разработки программного обеспечения, искусственный интеллект и экспертные системы.

Для эффективного поиска в распределенных хранилищах данных используется особый класс агентов — мобильные агенты, которые позволяют: 1) значительно повысить производительность при правильном распараллеливании алгоритма поиска; 2) снизить сетевой трафик благодаря локальному доступу к распределенным информационным ресурсам, тем самым снизив нагрузку на коммуникационные элементы сети. Но эффективность работы агентной системы сильно зависит от выбранной организации взаимодействия мобильных агентов, степени параллелизма.

Для обеспечения наиболее быстрого поиска мобильными агентами (минимизации времени отклика системы на запрос пользователя) в работе предлагается менять организацию взаимодействия агентов и одновременно варьировать степень параллелизма.

В существующих системах мобильных агентов агенты строятся как монолитные функциональные сущности. Для возможности изменения организации агентов предлагается настраиваемая архитектура агента, которая позволяет осуществлять сборку агента из программных компонентов. Благодаря настраиваемой архитектуре можно строить различные типы агентов, с нужным функциональным составом в процессе функционирования системы. Выбранная организация агентов определяет типы агентов и их функционально-модульный состав. Организация агентов, а также оптимальная степень параллелизма определяются благодаря предварительному моделированию.

**Цель диссертационной работы.** Разработка настраиваемой архитектуры системы мобильных агентов для поисковых систем.

Для достижения этой цели были решены следующие задачи:

1. Проведен анализ архитектур распределенных поисковых систем с мобильными агентами.
2. Разработана компонентная архитектура агента, позволяющая настраивать поисковых агентов и менять организацию взаимодействия агентов для

достижения наибольшей производительности при изменении условий функционирования среды.

3. Созданы модели агентных поисковых систем для исследования производительности агентных архитектур поиска.

4. Проведено моделирование агентных архитектур поиска для определения условий, в которых архитектура агентов достигает наибольшей производительности.

5. Разработана методика проектирования агентных поисковых систем.

6. Разработан прототип агентной системы с целью апробации предложенной архитектуры агента.

**Направление исследования.** Поисковые системы, построенные на базе распределённых архитектур, использующих мобильных агентов, математические и алгоритмические модели и методы построения таких систем на базе агент-ориентированного подхода.

**Методы исследования.** Для решения поставленных задач в работе используются методы математического моделирования, общей теории систем, теории систем массового обслуживания, теории автоматов, теории вероятностей, имитационного моделирования, анализа и проектирования объектно-ориентированных многозадачных программных систем.

**Достоверность и обоснованность результатов** подтверждается использованием современных научных методов исследования, глубоким и многосторонним рассмотрением объекта исследования, введением адекватных модельных ограничений, согласованностью данных модельного эксперимента и научных выводов, оценкой адекватности предложенных моделей, прототипированием архитектуры агентной системы, которое подтверждает работоспособность предложенного решения.

**Положения, выносимые на защиту:**

1. Модели агентных поисковых систем.
2. Настраиваемая компонентная архитектура агента.
3. Методика проектирования настраиваемых агентных поисковых систем.

**Научная новизна.** В работе получены следующие научные результаты:

1. Разработана компонентная архитектура агента, позволяющая строить агентные системы из настраиваемых программных компонентов.

2. Предложены модели агентных поисковых систем для определения условий, в которых достигается наибольшая производительность распределенной агентной системы поиска.

3. Предложена методика проектирования настраиваемых агентных поисковых систем.

**Практическая значимость.** В процессе выполнения данной работы были получены следующие практические результаты:

1. Реализован набор библиотечных модулей для построения имитационных моделей агентных поисковых систем в системе моделирования AnyLogic.

2. Построены имитационные и аналитические модели основных архитектур агентных систем.

3. Проведена серия модельных экспериментов, на основе полученных результатов проведено сравнение характеристик различных архитектур агентных поисковых систем на модельных данных.

4. Реализован прототип модульной агентной системы, подтверждающий возможность реализации предложенной компонентной настраиваемой архитектуры агентной системы.

**Реализация и внедрение результатов исследования.** Разработанная методика проектирования использовалась в учебном процессе при проведении практических занятий для студентов в рамках дисциплины «Распределенные программные системы и технологии» для профиля «Программное обеспечение и администрирование информационно-вычислительных систем и сетей» магистерской программы «Управление информационными системами и ресурсами».

**Апробация работы.** Основные положения и результаты работы докладывались на следующих научно-технических конференциях: на Международной конференции по мягким вычислениям и измерениям (SCM'2002), С.-Петербург, 2002 г.; Международной конференции по мягким вычислениям и измерениям (SCM'2004), С.-Петербург, 2004 г.; Всероссийской межвузовской научно-технической конференции студентов и аспирантов, С.-Петербург, 2007 г.; Международной конференции по мягким вычислениям и измерениям (SCM'2007), С.-Петербург, 2007 г.;

**Публикации.** По теме диссертации опубликовано 11 печатных работ, среди которых 3 публикации в ведущих рецензируемых изданиях, рекомендованных в действующем перечне ВАК, 3 статьи в других журналах, доклады доложены и получили одобрение на 5 международных, всероссийских и межвузовских научно-практических конференциях, перечисленных в конце автореферата.

**Структура и объем диссертации.** Диссертация состоит из введения, четырех глав, заключения, списка литературы, включающего 109 наименований. Содержание изложено на 188 страницах машинописного текста.

## ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обоснована актуальность работы, сформулированы основные цели, задачи, научная новизна, практическая ценность работы.

В **первой главе** проанализированы основные архитектуры поисковых систем: централизованная архитектура системы поиска с одним сервером, централизованная архитектура системы поиска с базой данных, частично центра-

лизованная архитектура с несколькими серверами, иерархическая архитектура с серверами групп и сегментов, децентрализованная архитектура без сервера.

В результате анализа выяснилось, что главной проблемой рассмотренных архитектур является наличие центрального сервера, который является самым медленным компонентом системы и ограничивает производительность и масштабируемость всей системы.

Системы с архитектурой из нескольких серверов позволяют распределить нагрузку между несколькими равноправными сегментами, серверы сегментов должны быть очень мощными компьютерами, поэтому затраты на развертывание и поддержку такой системы будут велики.

В иерархических системах с ростом глубины иерархии увеличивается степень параллелизма, нагрузка распределяется между серверами нижних уровней, соответственно затраты на серверы нижних уровней снижаются, но с ростом глубины иерархии возникают издержки при взаимодействии между уровнями.

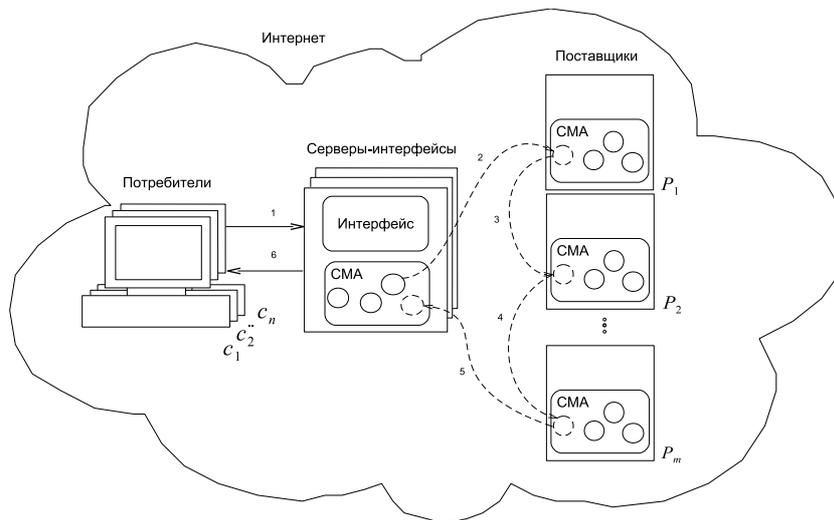
При построении систем без центрального сервера имеются технические ограничения, т.к. возникает необходимость в групповых запросах и широковещательных сообщениях, что в глобальной сети оказывается весьма проблематичным, поскольку не все маршрутизаторы поддерживают групповые и широковещательные сообщения. К тому же массовая рассылка сообщений перегружает сеть и узлы избыточной информацией, снижая общую производительность.

Устранение выявленных недостатков возлагается на применение агентного подхода, поэтому далее в первой главе рассматриваются архитектуры, использующие мобильных агентов для распределенного поиска.

Архитектура мобильного агента зависит от способа взаимодействия агентов при выполнении поиска и архитектуры узлов поисковой системы. Поэтому были рассмотрены следующие архитектуры поиска с мобильными агентами: 1) архитектура поиска с мобильными агентами, где агенты независимы и не обмениваются сообщениями между собой; 2) архитектура поиска с мобильными агентами-маршрутизаторами; 3) архитектура поиска с мобильными агентами-маршрутизаторами и агентами-наблюдателями.

Архитектура поиска с мобильными агентами, где агенты независимы и не обмениваются сообщениями между собой, показана на рисунке 1. Сплошными линиями показан обмен между потребителями и сервером-интерфейсом, пунктирными – перемещения поискового агента. На рисунке 2 приведена схема архитектуры системы мобильных агентов с использованием агента-маршрутизатора (на схеме он обозначается заштрихованным кружком). Поиск в таких системах включает следующие этапы:

1. Потребитель посылает запрос на сервер-интерфейс.
2. Сервер-интерфейс создает агента-маршрутизатора.



**Рисунок 1** — Архитектура системы поиска с мобильными агентами

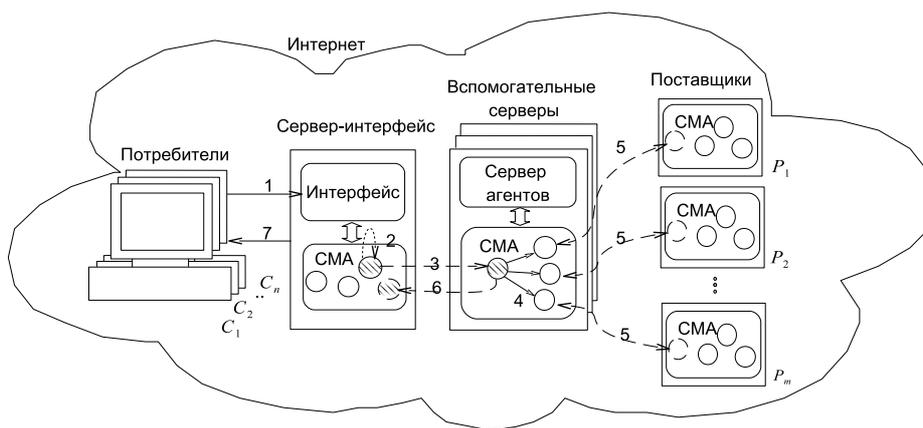
3. Агент-маршрутизатор перемещается на вспомогательный хост сети и согласно запросу пользователя определяет подмножество поставщиков для обхода.

4. Агент-маршрутизатор создает нескольких поисковых агентов для обхода нужных поставщиков, задает им маршруты и запускает их.

5. Поисковые агенты обходят заданных поставщиков и возвращаются с результатом.

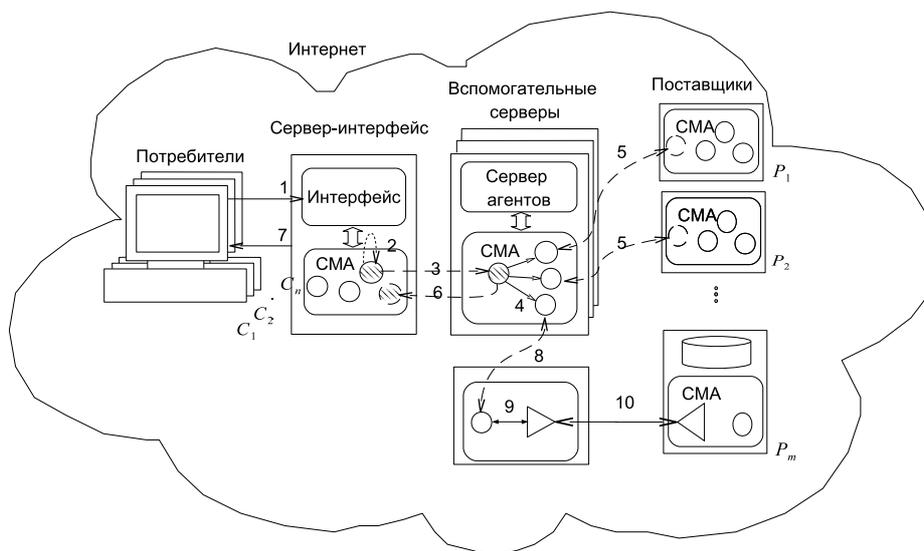
6. Агент-маршрутизатор возвращается с результатом на сервер-интерфейс.

7. Сервер-интерфейс формирует и передает результат потребителю.



**Рисунок 2** — Архитектура системы поиска с мобильными агентами-маршрутизаторами

В архитектуре с агентами-наблюдателями (рисунок 3) запросы поисковых агентов направляются не на сервер поставщика, а к агентам-наблюдателям, которые хранят копию данных сервера поставщика и поддерживают состояние этих данных в актуальном виде благодаря подписке на изменения в базе данных поставщика. Такая архитектура целесообразна при значительной нагрузке,



**Рисунок 3** — Архитектура системы поиска с мобильными агентами-маршрутизаторами и агентами-наблюдателями

когда сервер поставщика перестает справляться с обработкой поисковых запросов.

Выбор архитектуры определяется параметрами запроса (последовательный поиск, поиск первого подходящего решения и т. д.), рекомендациями и зависимостями, определенными в процессе имитационного моделирования, а также состоянием сети. Критерием выбора архитектуры, состава агентов и количества поисковых агентов является минимальное время поиска.

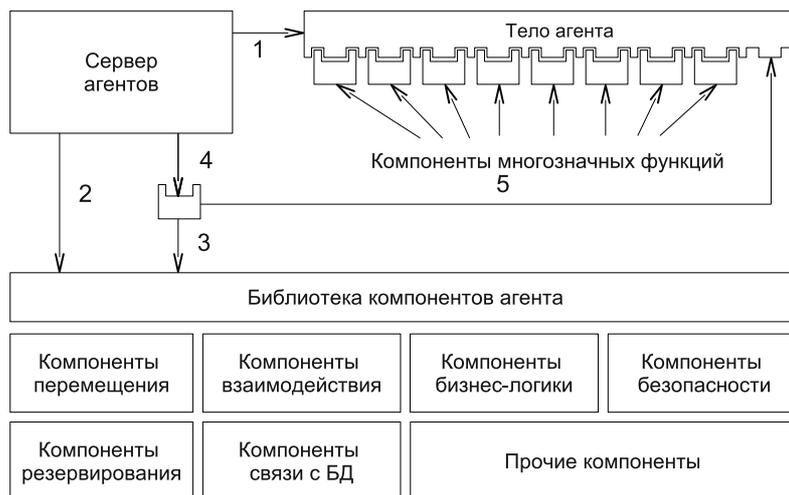
Во **второй главе** на основе анализа шаблонов (паттернов), используемых при проектировании агентов, предлагается «реконфигурируемый шаблон агента», позволяющий собирать специализированного агента из готовых компонентов, конфигурируемых в процессе создания. Поскольку критерием выбора архитектуры агента является текущая ситуация в сети, то архитектура агента должна позволять выбирать способы реализации требуемых функций на этапе инициализации агента.

Агент состоит из тела и компонентов. Тело агента будет управлять инициализацией и завершением, обеспечивать возможность динамической стыковки компонентов, их взаимодействие, возможность сохранения и восстановления состояния компонентов. Сборка агентов производится с помощью сервера агентов, последовательность сборки показана на рисунке 4:

1. Создание тела агента.
2. Запрос класса функционального элемента из библиотеки.
3. Создание компонента.
4. Инициализация и настройка компонента.
5. Интеграция компонента в тело агента.

К телу агента могут подключаться следующие компоненты:

1. Компонент перемещения управляет перемещением агента, содержит



**Рисунок 4** — Сборка агента

маршрут, текущее положение агента в сети и т. д.

2. Компонент взаимодействия с другими агентами определяет протокол взаимодействия между агентами в группе, например, основанный на языке ACL.

3. Компонент взаимодействия с системой локального поиска (компонент связи с БД) определяет протокол взаимодействия с базой данных, т. е. является интерфейсом к базе данных.

4. Компонент бизнес-логики обеспечивает поведение агента, определяемое решаемой задачей, реализует функции поиска агента (бизнес-функции агента).

5. Компонент взаимодействия с сервером-интерфейсом определяет протокол взаимодействия между сервером-интерфейсом и агентом для обеспечения возможности передачи результатов работы и получения команд управления. Взаимодействие может быть как локальным, так и удаленным.

6. Компонент резервирования позволяет создать резервных агентов, обнаружить потерю резервируемого агента и заменить его резервным. Для обеспечения этих функций осуществляется информационный обмен между резервируемым и резервными агентами, в результате чего повышается надежность агента-маршрутизатора в ненадежной сетевой среде.

7. Компонент безопасности отслеживает изменение состояния агента, подлинность сертификатов, цифровых подписей от внутренних данных и кода агента, а также сообщений, принимаемых и отсылаемых агентом, взаимодействует с компонентами безопасности платформы системы мобильных агентов (СМА).

8. Компонент взаимодействия с платформой СМА обеспечивает возможность взаимодействия агента и платформы СМА. Внутреннее состояние агента и его компонентов сохраняются с помощью платформы СМА, платформа также обеспечивает передачу сообщений между агентами, управление инициализации

ей и завершением агента, поэтому следует выделить отдельный интерфейс для взаимодействия агента и платформы.

Тело агента должно обеспечивать возможность взаимодействия между компонентами произвольной функциональности. Для реализации такой архитектуры потребуется общий интерфейс для передачи межкомпонентных сообщений и обеспечения возможности сохранения и восстановления состояния компонента в процессе перемещения.

Для апробации предложенного «реконфигурируемого шаблона» и демонстрации идеи сборки агентов из типовых компонент был реализован прототип агентной системы. В качестве основы для реализации прототипа была взята агентная система JADE.

На диаграмме рисунке 5 показаны сущности системы JADE: **Agent** – базовый класс всех агентов в системе, **Behavior** – базовый абстрактный класс всех агентных поведений, **SimpleBehavior** – реализация поведения для типовых агентов.

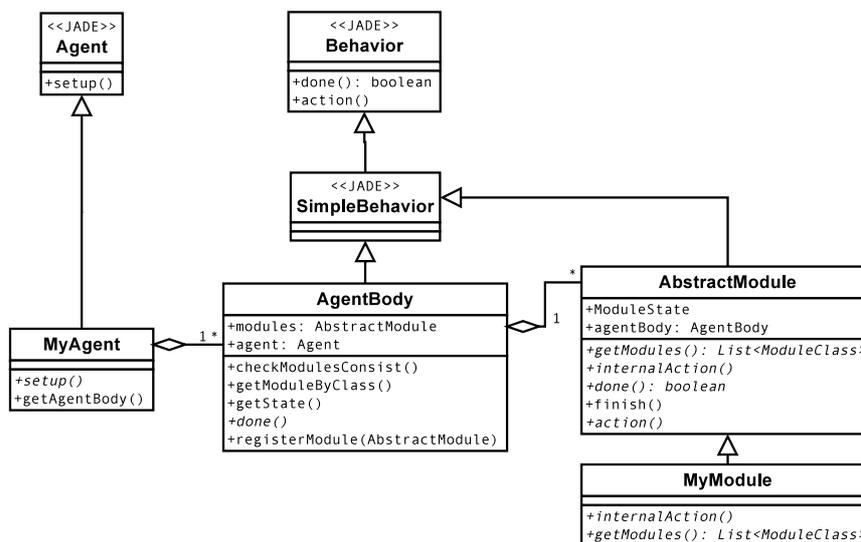


Рисунок 5 – Диаграмма классов прототипа компонентного агента в среде JADE

Элементы реализованного прототипа: **AgentBody** – базовый класс тела агента, которое содержит внутри себя компоненты агентов, унаследованных от базового класса **AbstractModule**.

Тело агента имеет состояние, которое можно получить вызовом **AgentBody.getState()**. Диаграмма состояний представлена на рисунке 6. **Третья глава** посвящена построению моделей систем мобильных агентов. В



Рисунок 6 – Диаграмма состояний тела прототипа агента

качестве сетевой среды рассматриваются соединённые сетью Интернет серверы поставщиков информационных услуг, которые являются сетевыми узлами. Между любыми узлами может быть установлен канал связи, параметрами которого является максимальная и средняя пропускная способность, задержка при передаче, зависящая от числа промежуточных узлов и их характеристик. Каждый сетевой узел характеризуется параметром производительности – временем выполнения поискового запроса. Данная величина зависит от архитектуры узла: типа ОС, типа используемой СУБД, эффективности реализации функционального ПО, а также от характеристик оборудования: процессоров, объема памяти, скорости работы и ёмкости устройств долговременного хранения информации и прочих факторов. При моделировании учитывается производная характеристика от данных параметров – среднее время выполнения базового запроса.

Для того, чтобы учесть наиболее существенные факторы (суммарную среднюю пропускную способность и задержку при передаче), осуществляется переход к модели каналов связи между узлами системы.

Для каналов связи основными характеристиками являются:  $\nu$  – пропускная способность,  $\tau$  – задержка времени возврата пакета. Узлы будут характеризоваться временем выполнения запроса  $t_{cs}$  на центральном сервере, временем выполнения запроса на сервере поставщика  $t_s$ .

В разработанных моделях использовались следующие компоненты:

1. **Сеть.** Состоит из узлов сети и коммуникационных связей между узлами. Узлы сети могут быть каналами данных, переключателями (маршрутизаторами), хостами (компьютеры пользователей, серверы).

Канал данных – элемент модели, связывающий узлы сети, обеспечивает возможность взаимодействия между узлами сети, характеризуется задержкой передачи данных, которая может меняться в зависимости от текущей загрузки соответствующего канала.

Переключатель (маршрутизатор) – узел сети, который выполняет перенаправление пакетов в требуемый канал данных согласно маршруту пакета. В маршрутизаторе может сходиться много каналов данных.

Хост – узел сети, который предоставляет некоторый ресурс пользователю. Обычно соединяется одним каналом данных с пользователем или переключателем. Хост является модельным отображением сервера.

2. **Сервер.** Является моделью хоста сети. Состоит из следующих объектов: приёмника запросов, очереди запросов, обработчика запросов, отправителя результатов. Ёмкость очереди и время обработки запроса могут быть различными для разных серверов.

3. **Пользователь.** Посылает запросы серверам и получает от них ответы. Является генератором запросов и получателем ответов. Интенсивность посылки запросов у пользователей может быть различной.

При моделировании используются следующие входные параметры:

1. Конфигурация сети. Совокупность узлов и связей между ними.
2. Серверы. Характеризуются распределением времени обработки запроса и длиной очереди.
3. Каналы данных. Характеризуются временем передачи запросов в зависимости от их пропускной способности, задержки и загрузки.
4. Переключатели. Характеризуются задержкой при передаче запросов.
5. Пользователи. Характеризуются интенсивностью посылки запросов и её распределением.

Выходные параметры моделирования:

1. Трафик в сети. Количество сообщений в единицу времени. Должна быть возможность просмотра трафика каждого канала данных в отдельности для выявления «узких» мест.
2. Среднее время обслуживания заявки.
3. Средняя длина очереди заявок.
4. Количество отказов.

**Ограничения модели.** Поток событий в модели ординарный, так как одновременное появление двух событий маловероятно (к тому же в реальной жизни из-за ограничения времени распространения сигналов понятие одновременности относительно), поэтому минимальный интервал между появлением событий будет равен машинной погрешности. Для моделирования работы информационных систем было использовано событийное моделирование с дискретным временем. Время обработки запроса может задаваться произвольным распределением, но в работе при моделировании использовалось экспоненциальное распределение.

В данной работе для построения моделей была использована система Anylogic. Компоненты моделей строились на основе библиотечных элементов Anylogic, таких как очередь, задержка, коммутатор, блок управления.

На рисунке 7 приведена схема модели сервера, она имеет входную очередь запросов, обработчик, блок управления, генератор ответов и агентов, выходную очередь. Обработка запроса потребителя заключается в создании группы агентов, их запуске, обработке результатов поиска после возвращения всех агентов. Затем происходит формирование и отправка ответа компоненту-пользователю.

Алгоритм работы блока управления сервера задается картой состояний, показанной на рисунке 8. Карта состояний сильно зависит от алгоритма поиска информации с помощью мобильных агентов. Данные алгоритмы подробно описаны в работе.

Модель сервера является получателем запросов и генератором ответов. Интенсивность и распределение посылки запросов у разных пользователей может быть различной.

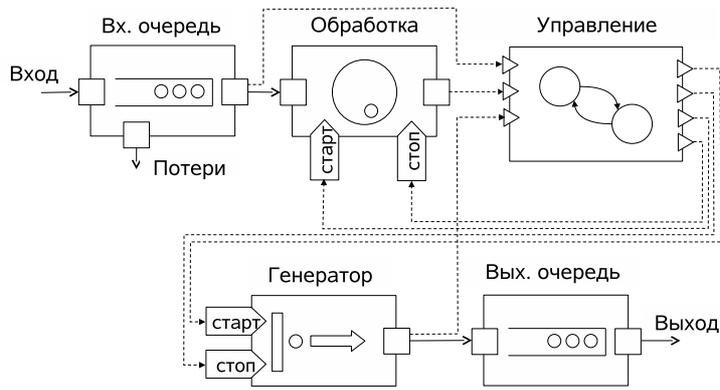


Рисунок 7 — Модель сервера

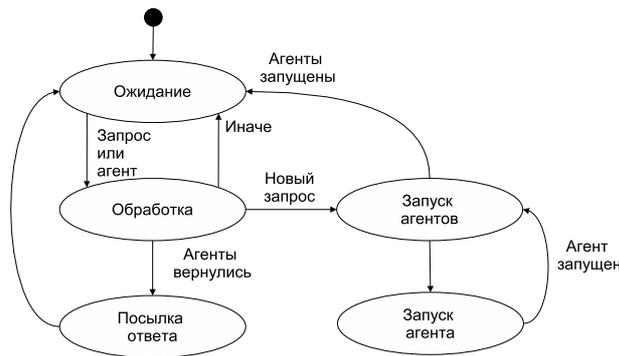


Рисунок 8 — Модель состояний блока «управление» сервера

Входными параметрами сервера являются: 1) Интенсивность обработки запросов и функция распределения. 2) Длина входной очереди. 3) Количество посылаемых агентов. 4) Количество серверов поставщиков для обхода. 5) Размер пакета. 6) Размер системной части пакета. 7) Размер пакета подтверждения. 8) Размер агента.

Выходными параметрами сервера являются: 1) Загрузка. 2) Средняя длина очереди, количество запросов. 3) Количество обработанных запросов. 4) Количество отказов из-за переполнения очереди.

Для моделирования различных архитектур требуется собрать из перечисленных компонентов модель сети и задать алгоритмы обработки запросов и агентов. Для определения значений параметров модели поисковой системы была проанализирована задача поиска и бронирования авиабилетов.

В четвертой главе приведены модели архитектур и результаты моделирования. Целью моделирования архитектур агентных поисковых систем является выбор оптимальной архитектуры.

В данной главе определялась производительность архитектур агентов при варьировании нагрузки на систему. Критерием оптимальности является выполнение поиска за минимальное время на базе имеющихся сетевых вычислительных ресурсов. Пусть  $\lambda$  – нагрузка на систему,  $i$  – организация агентов,  $I$  – множество организаций агентов,  $i \in I$ ,  $\Lambda : \lambda \in [0; l], l \in \mathbb{R}, \lambda \in \mathbb{R}$  – интервал

исследуемой нагрузки,  $t(i, \lambda)$  – целевая функция, поскольку  $T \in \mathbb{R}$ , то целевая функция отображает  $t : I \times \Lambda \rightarrow \mathbb{R}$ , тогда задача оптимизации заключается в нахождении такого  $i$ , когда  $t(i, \lambda) \rightarrow \min_i$ .

Дополнительной задачей оптимизации является определение оптимального количества агентов (степени параллелизма) при использовании выбранной организации агентов. Если степень параллелизма  $k \in [1, \infty)$ ,  $k \in \mathbb{N}$ , то целевая функция  $t : I \times \mathbb{N} \times \Lambda \rightarrow \mathbb{R}$ ,  $t(i, k, \lambda) \rightarrow \min_{i,k}$ .

Конечным результатом моделирования является формирование данных, пригодных для автоматического выбора организации агентов в меняющихся условиях работы. Для каждой из моделируемых архитектур представлен алгоритм поиска и алгоритмы маршрутизации агентов на главном сервере, серверах потребителей, дополнительных серверах.

Имитационное моделирование было проведено для следующих типов архитектур:

1. Независимой группы агентов со статической балансировкой нагрузки.
2. Группы агентов с маршрутизатором со статической балансировкой нагрузки.
3. Независимой группа агентов при различной производительности поставщиков.
4. Группы агентов с маршрутизатором при различной производительности поставщиков. Также выполнена аналитическая оценка оптимальной степени параллелизма для простейшего случая: независимая группа агентов со статической балансировкой и с одинаковой производительностью хостов в установившемся режиме.

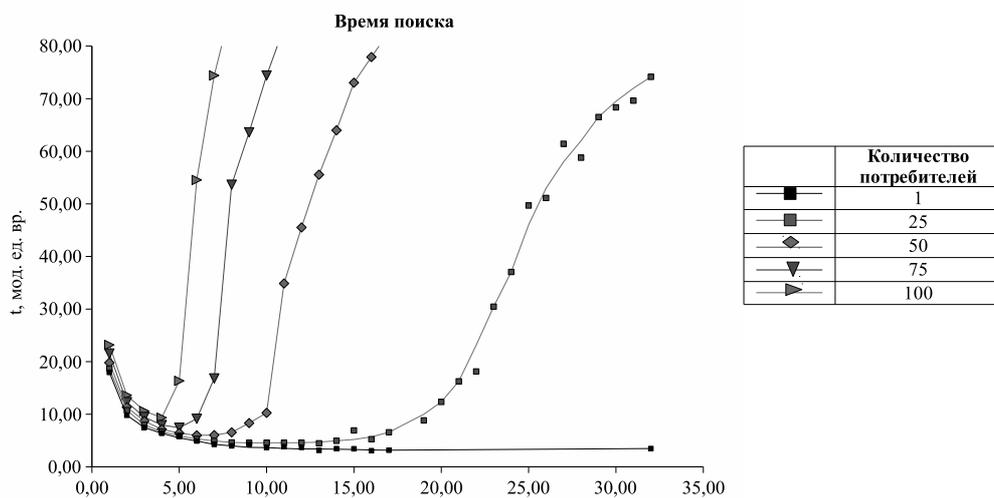
В процессе моделирования были получены результаты, которые представлены в виде семейства графиков рисунке 9, каждый график соответствует определённой нагрузке на систему, создаваемой определенным количеством работающих потребителей (1,25,50,75,100) в системе.

Видно, что у каждого графика есть свой характерный минимум в соответствующих точках на оси абсцисс: 32, (7-14), 6, 5, 4<sup>1</sup>; значения которых соответствуют количеству агентов, при котором поиск выполняется наиболее быстро ( $t \rightarrow \min$ ). Значения оси абсцисс в точках минимума являются оптимальным количеством агентов для заданной нагрузки. При увеличении нагрузки точка оптимального количества агентов становится более выраженной и смещается влево, то есть оптимальное количество поисковых агентов уменьшается при увеличении нагрузки системы.

Данные семейства графиков были получены для всех четырех типов архитектур. Результатом явилась методика проектирования агентных систем поиска, состоящая из следующих основных этапов:

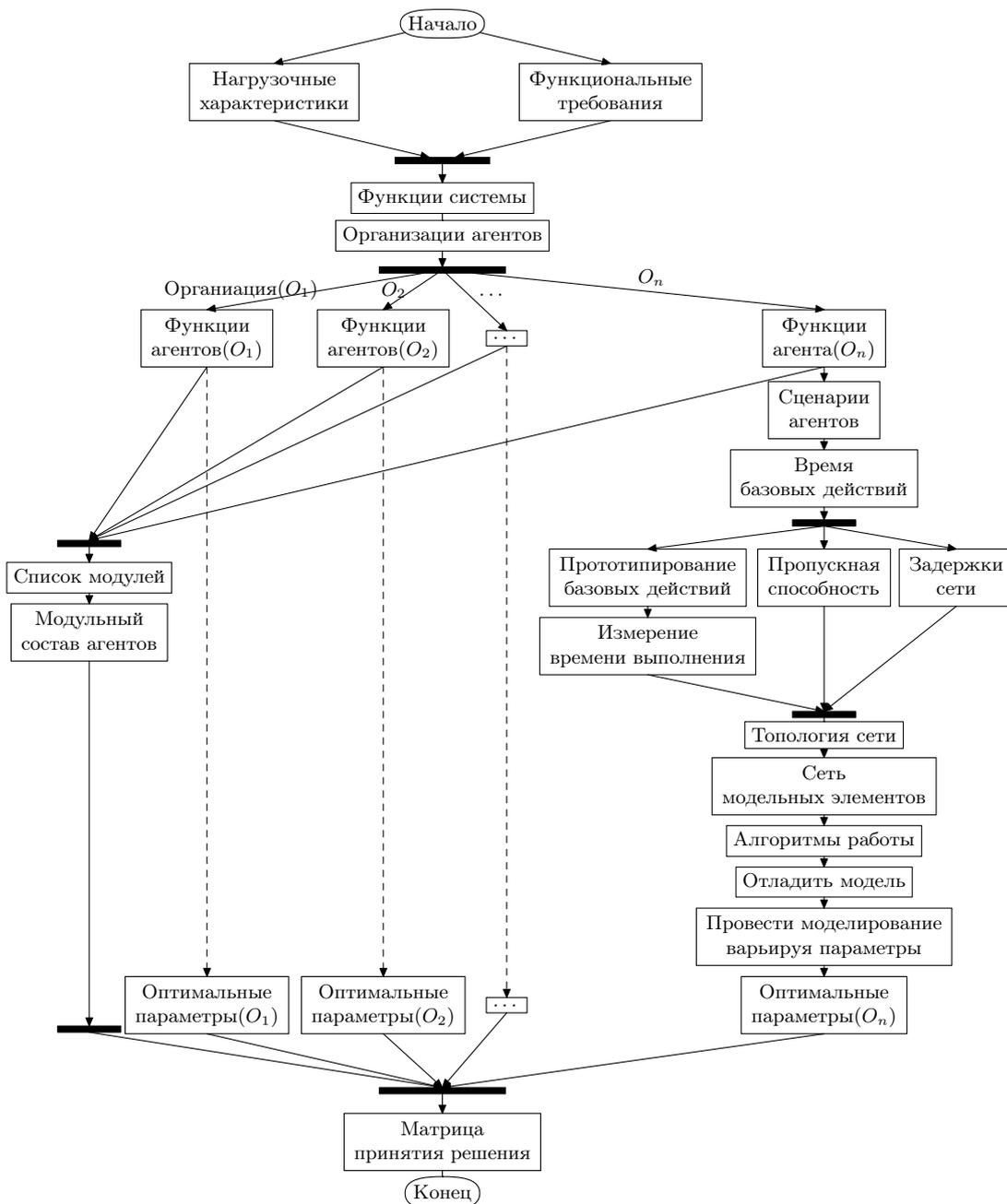
---

<sup>1</sup>Для кривой (25) минимум соответствует интервалу значений в интервале 7-14



**Рисунок 9** —Время поиска группой независимых агентов

1. Определение функциональных требований к системе.
2. Определение требований к нагрузочным характеристикам системы.
3. Задание функций системы, которые будут определять действия агентов.
4. Определение возможных организаций агентов для выполнения поиска.
5. Для каждой из организаций агентов определение:
  - 5.1. Набора функций для каждого типа агентов.
  - 5.2. Выявление базовых действия агентов и задание сценариев их работы.
  - 5.3. Определение времени выполнения базовых действий агента. Для этого необходимо выполнить следующие действия:
    - 5.3.1. Провести прототипирование и измерение времени выполнения базовых операций.
    - 5.3.2. Определить пропускную способность и задержки системы.
    - 5.3.3. Перевести полученные измерения в модельные единицы.
  6. Выявить топологию сети, в которой будут работать агенты.
    - 6.1. Представить сеть в виде функциональных модельных элементов, связанных сетевыми модельными элементами.
    - 6.2. Задать алгоритмы работы модельных элементов, отражающие сценарии работы агентов.
    - 6.3. Откомпилировать и отладить модель.
    - 6.4. Провести серию модельных экспериментов. Для этого необходимо:
      7. Определение параметров, при которых та или иная схема взаимодействия достигает наибольшей эффективности.
      8. Представление результатов в виде матрицы для принятия решения.



**Рисунок 10** – Методика проектирования

Перечисленные этапы методики проектирования агентной системы с перестраиваемой архитектурой представлены в виде диаграммы, рисунке 10.

В случае изменения интенсивности запросов более оптимальной может стать другая организация мобильных агентов, эта информация следует из анализа матрицы принятия решения, полученной в результате моделирования.

Матрица принятия решения в зависимости от количества запросов пользователей в единицу времени будет выглядеть следующим образом, см. таблицу 1, где  $n$  – количество архитектур,  $\lambda$  – нагрузка, и  $\lambda_1 < \lambda_2 \dots < \lambda_{n-1} < \lambda_n$ . Данная матрица определяет выбор архитектуры для одного из варьируемых параметров, для  $n$  параметров это будет  $n$ -мерная таблица.

После выбора архитектуры необходимо определить степень параллелиз-

**Таблица 1** — Матрица выбора архитектуры

Нагрузка, запросов в ед.вр.	Арх.1	Арх. 2	...	Арх. n-1	Арх. n
$(0; \lambda_1]$	+		...		
$(0; \lambda_2]$		+	...		
...	...	...	...	...	...
$(0; \lambda_{n-1}]$			...	+	
$(0; \lambda_n]$			...		+

ма, т.е. количество агентов, параллельно выполняющих поиск. Степень параллелизма также является результатом моделирования и представлена в виде таблицы 2, где  $a, b \in \mathbb{N}, a < b, a > 0$ . В результате моделирования были получены U-образные кривые, что позволяет говорить не об оптимальном значении количества агентов, а об оптимальном интервале степеней параллелизма  $[a_i, b_i]$ . После

**Таблица 2** — Выбор степени параллелизма

Нагрузка, запросов в ед.вр.	Количество агентов
$(0; \lambda_1]$	$[a_1, b_1]$
$(0; \lambda_2]$	$[a_2, b_2]$
...	...
$(0; \lambda_{m-1}]$	$[a_{m-1}, b_{m-1}]$
$(0; \lambda_m]$	$[a_m, b_m]$

определения оптимальной организации агентов производится сборка агентов из функциональных модулей, каждый тип агентов будет иметь свой модульный состав. Сборку агентов производит сервис агентов, который принимает запрос от интерфейса пользователя, инициирует старт агентов, принимает результаты поиска и возвращает их пользователю системы.

Представленная методика проектирования агентных систем поиска позволяет: 1) получить характеристики различных организаций агентов для поиска в сети, 2) на основе данных характеристик, произвести выбор наиболее эффективной архитектуры в данных параметрах окружения 3) для выбранной архитектуры определить оптимальную степень параллелизма агентов, 4) применить результаты моделирования для модульных агентных систем поиска.

## ЗАКЛЮЧЕНИЕ

В процессе проведенных в диссертационной работе исследований получены следующие основные результаты:

1. В работе проведен анализ основных архитектур распределенных поисковых систем, в результате которого выявлены их основные недостатки: плохая масштабируемость, высокая стоимость, низкая производительность. Для устранения этих недостатков предложено использовать агентный подход. Благодаря

использованию мобильных агентов происходит более оптимальное распределение нагрузки между компонентами системы и, тем самым, оптимизируется время поиска системы. Агенты позволяют строить более гибкие системы, имеющие меньшую стоимость за счет возможности развертывания на существующей сетевой инфраструктуре. Но производительность агентной системы поиска сильно зависит от способа организации агентов и текущей загрузки сети, поэтому было решено разработать архитектуру, которая позволяет менять организацию агентов в зависимости от текущего состояния сети. На основании этого сформулированы задачи: разработка настраиваемой агентной архитектуры, разработка методики проектирования агентных распределенных систем поиска.

2. Систематизированы основные архитектуры мобильных агентов для задачи поиска в сети, определены условия и границы их применимости. Определены основные типы агентов, описаны взаимодействия и связи между ними.

3. Разработана компонентная настраиваемая архитектура агента, позволяющая настраивать поисковых агентов и менять организацию взаимодействия агентов для достижения наибольшей производительности при изменении условий функционирования среды.

4. Создан прототип агентной системы с целью апробации компонентной архитектуры агента.

5. Разработаны аналитические и имитационные модели агентных систем поиска с целью определения оптимальной архитектуры системы при изменении параметров функционирования сетевой среды, определения оптимальной степени параллелизма агентной системы, при которой достигается наибольшая производительность всей системы.

6. Реализован набор библиотечных модулей для построения имитационных моделей агентных поисковых систем в системе моделирования AnyLogic.

7. В процессе моделирования были получены зависимости времени выполнения поискового запроса от нагрузки на систему, количества агентов, параллельно выполняющих поиск, и организации агентов в системе. Определены точки экстремума, значения которых соответствуют количеству агентов при котором поиск выполняется с наибольшей производительностью, данное количество агентов является оптимальной степенью параллелизма. Полученные данные позволяют сравнивать различные организации агентов и выбирать наиболее оптимальную в заданных условиях.

8. Разработаны основы методики, позволяющей проектировать эффективные настраиваемые агентные системы поиска благодаря возможности изменения структуры агентной системы, применению мобильных агентов с разным компонентным составом, а также определению оптимальных значений степени параллелизма для выбранной структуры агентов, полученных при моделировании.

**Апробация.** Основные результаты работы докладывались на междуна-

родных конференциях по мягким вычислениям и измерениям (SCM) в 2002–2008г., Всероссийской межвузовской научно-технической конференции студентов и аспирантов в 2007 г.

**Публикации.** По теме диссертации опубликовано 11 печатных работ, включающих 6 статей и тезисы к 4 докладам на международных и всероссийской научно-технических конференциях.

*Публикации в изданиях, рекомендованных ВАК России:*

1. Фомичев В.С., Першин А.В. Архитектуры сетевых поисковых систем. //Известия СПбГЭТУ «ЛЭТИ». 2006. № 3. С. 3-8.

2. Першин А.В. Архитектура настраиваемого агента. //Информационные технологии. 2008. № 6

3. Куприянов М.С.,Першин А.В. Методика моделирования агентных поисковых систем. //Известия СПбГЭТУ «ЛЭТИ». 2010. № 9. С. 61-66.

*Другие статьи и материалы конференций:*

4. Фомичев В.С., Холод И.И., Першин А.В., Степанов А.Г. Поток сообщений в системах мобильных агентов. // Материалы Междунар. конф. по мягким вычислениям и измерениям (SCM2002).СПб., 25-27 июня 2002. СПб: Изд-во СПбГЭТУ «ЛЭТИ».

5. Першин А.В. Классификация шаблонов проектирования систем мобильных агентов. // Сб. докл. Междунар. конф. по мягким вычислениям и измерениям, СПб., 17-19 июня 2004. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2004. Т.1. С. 238-243.

6. Першин А.В. Построение моделей мобильных агентов в системе Anylogic. Материалы Всероссийской межвузовской науч.-техн. конф. студентов и аспирантов. Ч.V. СПб: Изд-во. СПбГПУ. Политехнического университета, 2007. с.181-182.

7. Фомичев В.С., Першин А.В. Система мобильных агентов для поиска в сети. //Известия СПбГЭТУ «ЛЭТИ». 2007. № 1.

8. Першин А.В. Моделирование процесса поиска с мобильными агентами посредниками. // Сб. докл. Междунар. конф. по мягким вычислениям и измерениям, СПб., 25-27 июня 2007. СПб.: Изд-во СПбГЭТУ «ЛЭТИ». 2007. Т.1.

9. Першин А.В. Модель агентной системы для поиска в сети. //Известия СПбГЭТУ «ЛЭТИ». 2008. № 7. С. 19-24.

10. Першин А.В. Аналитическая модель агентной поисковой системы //Известия СПбГЭТУ «ЛЭТИ». 2008. № 8. С. 21-26.

11. Першин А.В. Динамическая балансировка нагрузки при поиске в сети мобильными агентами. // Сб. докл. Междунар. конф. по мягким вычислениям и измерениям, СПб., 23-25 июня 2008. СПб.: Изд-во СПбГЭТУ «ЛЭТИ». 2008. Т.1.