

На правах рукописи *Sign*

**Тощев Александр Сергеевич**

**Интеллектуальная система повышения  
эффективности IT службы предприятия**

Специальность 05.13.01 —  
«Системный анализ, управление и обработка информации  
(информационные технологии)»

**Автореферат**  
диссертации на соискание учёной степени  
Кандидат технических наук

Казань — 2015

Работа выполнена в организации Казанский (Приволжский) Федеральный Университет

Научный руководитель: доктор физико-математических наук, профессор  
**Елизаров А.М.**

Официальные оппоненты: **Соловьев Валерий Дмитриевич**,  
доктор физико-математических наук, профессор,  
Казанский (Приволжский) Федеральный Университет,  
Институт филологии и межкультурной коммуникации им. Льва Толстого,  
Ведущий научный сотрудник

**Оппонет2**,  
кандидат технических наук,  
**Оппонет2**,  
**Оппонет2**

Ведущая организация: Федеральное государственное бюджетное учреждение науки Институт проблем информатики Российской академии наук

Защита состоится **DD mmmmmmmm YYYY г. в XX часов** на заседании диссертационного совета **NN** на базе **Название учреждения** по адресу: **Адрес**.

С диссертацией можно ознакомиться в библиотеке **Название библиотеки**.

Автореферат разослан **DD mmmmmmmm YYYY** года.

Ученый секретарь  
диссертационного совета

**NN**, д-р физ.-мат. наук

*Sign*

Фамилия Имя Отчество

## **Общая характеристика работы**

**Целью** данной работы является разработка интеллектуальной системы повышения эффективности деятельности ИТ-службы предприятия.

**Область исследования** данной работы это разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации в ИТ-отрасли. **Предметом исследования** является процесс регистрации и устранения проблемных ситуаций, возникающих в ИТ-инфраструктуре предприятия.

Для достижения поставленной цели необходимо было решить следующие **задачи**:

1. Провести теоретико-множественный и теоретико-информационный анализ сложных систем в области поддержки информационной инфраструктуры
2. Создать модель целевой области
3. Исследовать модели мышления и выбрать наиболее подходящую
4. На основе выбранной модели мышления разработать модель проблемно-ориентированной системы управления, принятия решений и оптимизации технических объектов в области обслуживания информационной структуры предприятия
5. Создать архитектуру приложения на основе модели
6. Реализовать прототип на основе архитектуры
7. Провести апробацию прототипа на тестовых данных

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

1. Теоретико-множественный и теоретико-информационный анализ сложных систем в области поддержки информационной инфраструктуры
2. Модель проблемно-ориентированной системы управления, принятия решений и оптимизации технических объектов в области обслуживания ИТ
3. Прототип программной реализации модели проблемно-ориентированной системы управления, принятия решений и оптимизации технических объектов в области обслуживания ИТ
4. Апробация системы на контрольных примерах и ее результаты

### **Научная новизна:**

1. Была создана модель проблемно-ориентированной системы управления, принятия решений в области обслуживания информационной структуры предприятия на основе модели мышления
2. Была представлена новая модель данных для модели мышления и оригинальный способ ее хранения, эффективный по сравнению с другими базами данных
3. Было выполнено оригинальное исследование моделей мышления в области обслуживания информационной структуры предприятия
4. На основе модели была создана архитектура системы и ее прототип

**Практическая значимость** Система, разрабатываемая в рамках данной работы носит значимый практический характер. Идея работы зародилась из производственных проблем в IT отрасли, с которыми автор сталкивался каждый день. Только глубокое понимание области помогло выбрать правильное решение. **Достоверность** научных исследований и практических рекомендаций базируется на корректной постановке общих и частных, поставленных выше, задач, использовании известных фундаментальных теоретических положений технической кибернетики, достаточном объёме статистического моделирования и экспериментальном материале исходных данных для численных оценок достижимых качественных показателей.

В работе были проведены исследования согласно паспорту специальности 05.13.01, сопоставление приведено в Таблице 1.

Таблица 1 — Сопоставление направлений исследования специальности 05.13.01 и исследований, проведенных в работе

Направление исследования	Результат работы
Разработка критериев и моделей описания и оценки эффективности решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации	В рамках работы была разработана модель системы принятия решения и обработки информации в области решения запросов пользователя на естественном языке.
Продолжение следует	

**Таблица 1 – продолжение**

Направление исследования	Результат работы
Разработка проблемно-ориентированных систем управления, принятия решений и оптимизации технических объектов	По модели, разработанной в предыдущем пункте был разработан прототип системы принятия решения Thinking-Understanding, который был испытан на модельных данных.
Методы получения, анализа и обработки экспертной информации	В рамках системы TU был разработан метод обработки экспертной информации - обучение при помощи модели мышления TU, основанной на принципах модели 6-ти Марвина Мински.
Разработка специального математического и алгоритмического обеспечения систем анализа, оптимизации, управления, принятия решений и обработки информации	В рамках разработки системы TU были созданы специальные алгоритмы для анализа запросов пользователя и принятия решений.
Теоретико-множественный и теоретико-информационный анализ сложных систем	В рамках работы был проведен комплексный анализ области поддержки программного обеспечения, с помощью которого была построена система данной области и выделены участки для оптимизации принятия решений.
Методы и алгоритмы интеллектуальной поддержки при принятии управленческих решений в технических системах	Система, разработанная в рамках данной работы включает в себя инновационные методы и алгоритмы поддержки принятия решений, использующих в своей основе модель мышления на базе модели мышления Человека, описанной в книге Марвина Мински.
Продолжение следует	

**Таблица 1 – продолжение**

<b>Направление исследования</b>	<b>Результат работы</b>
Визуализация, трансформация и анализ информации на основе компьютерных методов обработки информации	Представлена наглядная визуализация данных по системному анализу области удаленной поддержки инфраструктуры.

**Апробация работы.** Основные результаты работы докладывались на:

- Конференция Лобачевского - 2011
- WCIT-2012
- AINL-2013
- RCDL-2014
- AMSTA-2015

Апробация работы проводилась на выгрузка инцидентов из систем регистрации ОАО "АйСиЭл КПО ВС". Система показала хорошие результаты обработки данной информации. **Личный вклад.** Автор принимал активное участие в исследовании целевой области, разработке архитектуры приложения, реализации прототипа, проработки теории, тестировании прототипа.

**Публикации.** Основные результаты по теме диссертации изложены в 8 печатных изданиях [1], [2], [3], [4], [5], [6], [7], [8], 1 из которых изданы в журналах Scopus [7], 1 в журнале Web of Science [8], 1 в журнале РИНЦ [4], 4 в тезисах докладов [1], [2], [3], [4].

## **Содержание работы**

Во **введении** обосновывается актуальность исследования, проводимых в рамках данной диссертационной работы, дается общая характеристика работы. Проводится обзор области и производится постановка задачи. **Первая глава** посвящена обзору интеллектуальных систем регистрации и анализа проблемных ситуаций, возникающих в ИТ-инфраструктуре предприятия. Здесь представлен сравнительный анализ систем регистрации и устранения проблемных ситуаций. В главе определяются основные требования к интеллектуальным системам регистрации и анализа проблемных ситуаций в ИТ. Одним из важных элементов подобных систем является обработка естественного языка, поэтому здесь представлен сравнительный анализ методов и комплексов об-

работки текстов на естественном языке.

Во время анализа были использованы следующие обработчики естественного языка:

1. Open NLP
2. RelEx
3. StanfordParser

Результат работы подсчитывался при помощи метрик, представленных в Таблице 2.

Таблица 2 — Таблица метрик

Метрика	Описание	Формула
Precision	Точность	$P = \frac{tp}{tp + fp}$ <p>где P - precision, tp - успешно обработанные, fp - ложно успешные</p>
Recall	Чувствительность	$R = \frac{tp}{tp + fn}$ <p>где R-recall, tp - успешно обработанные, fn - ложно неуспешные</p>
F	F-measure (результативность)	$F = \frac{P * R}{P + R}$ <p>Где P - precision, R-recall.</p>

Результаты приведены на сводной диаграмме Рисунок ??.

Все рассмотренные системы не соответствуют полному комплексу необходимых требований. В Таблице 3 приведены сводные данные по системам. Ввиду развитости и доступности было решено использовать OpenCog RelEx.

Таблица 3 — Сравнительный анализ существующих решений

Сравнительный пункт	HP Open View	ServiceNOW	IBM Watson
Мониторинг	Да	Да	Да
Продолжение следует			

**Таблица 3 – продолжение**

<b>Сравнительный пункт</b>	<b>HP Open View</b>	<b>ServiceNOW</b>	<b>IBM Watson</b>
Регистрация инцидентов	Да	Да	Да
Управление системами	Да	Нет	Нет
Создание цепи обработки (Workflow) инцидента	Да	Да	Нет
Понимания и формализацию запросов на естественном языке	Нет	Нет	Да
Поиск решений	Нет	Нет	Да
Применение решений	Нет	Нет	Нет
Обучение решению инцидента	Нет	Нет	Да
Умение проводить логические рассуждения: генерализацию, специализацию, синонимичный поиск	Нет	Нет	Нет



**Вторая глава** посвящена построению модели интеллектуальной системы принятия решений для регистрации и анализа проблемных ситуаций в ИТ-инфраструктуре предприятия. Во время работы было рассмотрено и прототипировано четыре различных модели:

- Модель мышления на базе нейронных сетей
- Модель с использованием Деревьев Принятия Решений (Menta 0.1)
- Модель с использованием Генетических алгоритмов на базе модели мышления Питера Норвига (Menta 0.3)
- Модель мышления Марвина Мински (Модель 6-ти)

### **Модель мышления на базе нейронных сетей**

Модель на базе нейронных сетей (поддерживающая обучение) была отброшена на предварительной стадии оценки, так как имеет большие требования производительности несовместимые с Технико Экономическим Обоснованием.

### **Модель с использованием Деревьев Принятия Решений (Menta 0.1)**

Данная модель являлась одной из первых, которая была опробована. Модель была основана на деревьях принятия решений. В построение модели данной системы использовались следующие компоненты:

- Обработка запросов на естественном языке
- Поиск решения
- Применение решения

Система была ориентирована на выполнение простых команд, например, добавить поле на форму. Основная функция модели представлена следующим потоком:

1. Получение и формализация запроса
2. Поиск решения при помощи Деревьев Принятия Решений
3. Изменение модели приложения в формате OWL
4. Генерация и компиляция приложения

Основными проблемами данной модели являлось следующее:

1. Отсутствие устойчивости к ошибкам входной информации: грамматическим и содержательным. Например, входной файл не имел отношения к программной системе, модель которой была в базе знаний в формате OWL
2. Система поиска решения работала только в рамках модели одной программы

### 3. Отсутствовала функция обучения

**Модель с использованием Генетических алгоритмов на базе модели мышления Питера Норвига (Menta 0.2-0.3)** После работы над ошибками была предпринята попытка сделать поиск решения более универсальным. В рамках данной модели были сформированы основные компоненты системы:

- Критерии Приемки (Acceptance Criteria)
- Формат данных OWL
- Использование логических вычислений
- Универсальный поиск решения

Система содержала внутри себя модель приложения. При помощи генетического алгоритма модель строила из частей новую систему и проверяла ее при помощи логического движка NARS на соответствие входным критериям приемки, заданными пользователем. Основными недостатками подхода оказалось:

- Отсутствие обучения
- Отсутствие обработки естественного языка
- После апробации оказалось, что критерии приемки практически описывают необходимое решение (то которое должно быть найдено), что являлось недопустимым.

**Модель мышления Марвина Мински (Модель 6-ти)** Модель была построена с применением модели 6-ти Марвина Мински. Она содержит в себе основные концепции предыдущих моделей и показывает свою состоятельность на контрольных примерах. Реализация модели получила название TU.

- Обучение
- Поиск и применение решения
- Обработка естественного языка

Данная модель является более абстрактной и представляет собой верхнеуровневую архитектуру обработки запроса (мышления), где компонентами являются лучшие части предыдущих систем. Основным компонентом системы является Критик-Селектор-Образ мышления. На Рисунке 1 представлена схематичное изображение Критика-Селектора-Образа мышления.

Критик (Critic) представляет собой определенный предикат: внешние обстоятельства, события или иное воздействие. Например, включился свет и зрачки сузились. Обожглись и одернули руку. Критик активируется только

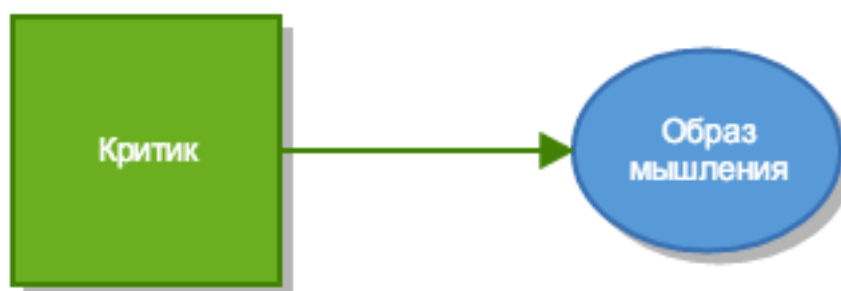


Рис. 1 — Критик-Селектор-Образ мышления

когда для этого достаточно входных стимулов. Одновременно могут активироваться несколько критиков. Например, человек решает сложную задачу. Идет активация множество критиков: считать, технические детали, кроме того параллельно может активироваться критик переработки, сообщающей о необходимости отдыха.

Селектор (Selector) занимается выбором определенных ресурсов, которыми также являются Образы мышления.

Образ мышления (WayToThink) это способ решения проблемы. Образ мышления также может активировать следующий критик.

На рисунке 2 представления расширенная модель работы триплета Критик-Селектор-Образ мышления. Критик активирует селектор, который активирует образ мышления (синий круг). образ мышления в свою очередь может активировать критик или же совершить определенные действия. Например, зажегся зеленый свет светофора, значит можно переходить дорогу.

Если активировалось много критиков, значит проблему нужно уточнить, так как степень неопределенности слишком высока. Если проблема очень похожа, то можно судить по аналогии. В Таблице 4 представлено описание уровней мышления.

Модель Menta 0.1 имеет следующие недостатки

1. Отсутствие устойчивости к ошибкам входной информации: грамматическим и содержательным. Например, входной файл не имел отношения к программной системе, модель которой была в базе знаний в формате OWL
2. Система поиска решения работала только в рамках модели одной программы

Таблица 4 — Описание уровней мышления модели 6-ти

Уровень	Описание
Инстинктивный уровень	На данном уровне происходят инстинктивные реакции (врожденные). Например, коленный рефлекс. Общую формулу для этого уровня можно выразить как "Если ..., то сделать так".
Уровень обученных реакций	На данном уровне происходит мышление обученных реакций, то есть тех реакций, которыми человек обучается в течение жизни. Например, переходить дорогу на зеленых свет. Общую формулу для этого уровня можно выразить как "Если ..., то сделать так".
Уровень рассуждений	а данной уровне происходит мышление с использованием рассуждений. Если я сделаю так, то будет ... Например, если перебежать дорогу на зеленый свет, то можно успеть вовремя. На данном уровне сравниваются последствия нескольких решений и выбирается оптимальное. Общую формулу для этого уровня можно выразить как "Если ..., то сделать так, тогда будет так".
Рефлексивный уровень	На данном уровне происходит рассуждение с учетом анализа прошлых событий. Например, в прошлый раз я побежал на моргающий зеленый и чуть не попал под машину.
Саморефлексивный уровень	На данном уровне происходит оценка себя. Строится определенная модель с помощью которой идет оценка своих поступков. Например, мое решение не пойти на это собрание было неверным, так как я упустил столько возможностей, я был легкомысленный.
Самосознательный уровень	На данном уровне идет оценка поступков человека с точки зрения высших идеалов и внешних оценок. Например, а что подумают мои друзья? А как бы поступил мой герой?

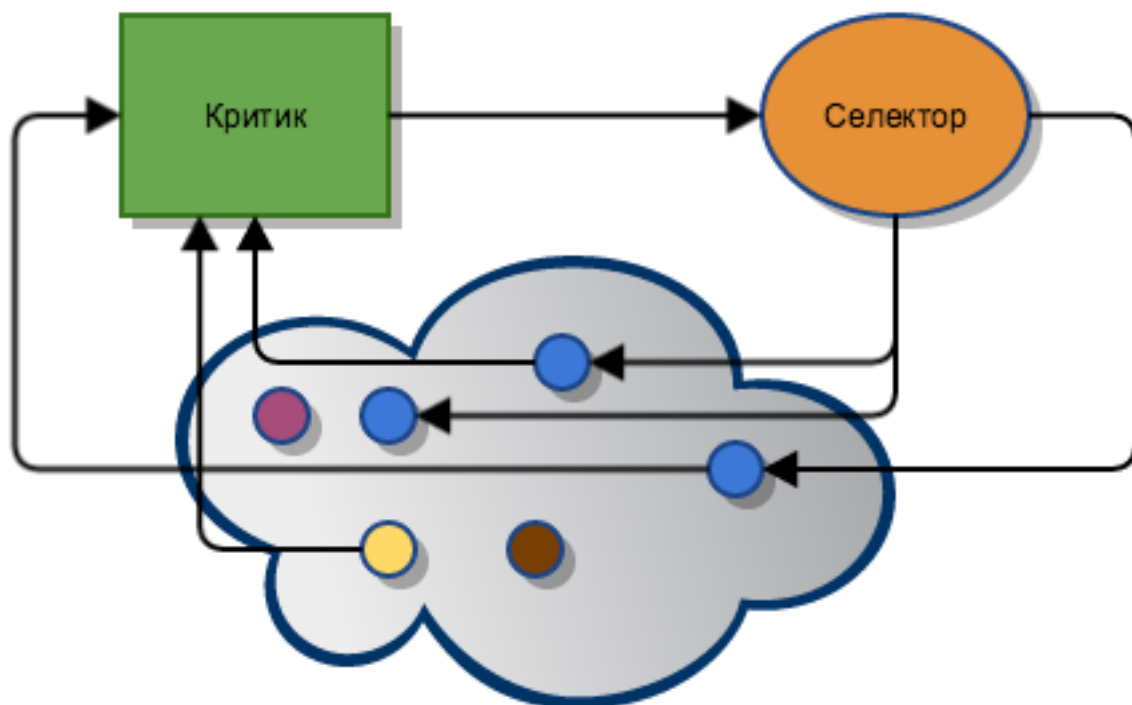


Рис. 2 — Критик-Селектор-Образ мышления в разрезе ресурсов

### 3. Отсутствовала функция обучения

Модель Menta 0.3 имеет следующие недостатки:

- Отсутствие обучения
- Отсутствие обработки естественного языка
- HyperGraphDB оказалась непригодной для промышленного использования
- NARS в виду своих особенностей оказался непригодным для промышленного применения на значительном объеме фактов ( $>20$ ). Так как содержал в себе комбинаторный взрыв. Например при 10 фактов количество сочетаний будет равно 45 на первом уровне, далее будут сравнивать результаты этих сочетаний.
- Апробации оказалось, что критерии приемки практически описывают необходимое решение, что являлось недопустимым. Данный подход был описан в статье [?].

Для программной экспертной системы очень важно обладать способностью мыслить и рассуждать. Например, очень важно для системы уметь действовать

по аналогии. Так как множество запросов типичны и отличаются лишь параметрами. Например, пожалуйста, установить Office, Antivirus и т.д.

Также для экспертной системы важно уметь абстрагировать специализированные рецепты решения. К примеру, система научилась решать инцидент "Please install Firefox". Абстрагировав данный инцидент до степени "Please install browser" система сможет теми же способами попробовать решить новый инцидент.

После рассмотрения нескольких моделей была выбрана модель мышления Марвина Мински, так как данная модель наиболее точно ложится на целевую область решения инцидентов в области IT. На основе подхода Мински была построена модель системы, которая поддерживает основные функции: обучение, понимание инцидента, поиск решения, применение решения. Более подробно с результатами апробации моделей можно ознакомиться на сайте <http://tu-project.com>

В третьей главе приведено описание архитектуры и реализация системы, основанной на модели TU Архитектура системы представляет собой модульную систему. Основными компоненты системы описаны в Таблице 5. Система может функционировать в режиме обучения и в режиме решения запросов.

Таблица 5 — Основные компоненты системы ThinkingUnderstanding

Компонент	Описание
TU Webservice	Основной компонент взаимодействия со внешней системой, включая пользователя.
CoreService	Ядро системы, содержит основные классы.
DataService	Компонент работы с данными.
Reasoner	Компонент вероятностной логики.
ClientAgent	Компонент выполнения скриптов на целевой машине.
MessageBus	Шина данных для системы.

В главе приводится основной рабочий поток работы приложения.

1. Поступает запрос от пользователя  
User had received wrong application. User has ordered Wordfinder Business Economical. However she received wrong version, she received Wordfinder Tehcnical instead of Business Economical. Please assist.
2. GoalManger устанавливает цель системы HelpUser
3. Активируется набор Critic, привязанный к данной цели
4. PreliminaryAnnorator разбирает фразу
5. KnowledgeBaseAnnotator создает семантическую сеть и ссылки на нее
6. Critic, привязанный к цели HelpUser на Рефлексивном уровне запускает WayToThink ProblemSolving с целью: ResolveIncident
7. Critic на Рефлексивном уровне выбирает WayToThink KnowingHow
  - (a) Запускаются параллельно все Critic, которые привязаны к IncidentClassification Critic, который привязан к ResolveIncident цели, в данном слу-

чае это DirectInstruction, ProblemWithDesiredState, ProblemWithoutDesiredState

- (b) Selector выбирает наиболее вероятный результат работы среди всех результатов компонентов. В данном случае будет результат работы Problem Description with desired state.
- (c) KnowingHow сохраняет варианты выбора Selector.
- (d) Simulation WayToThink с параметрами "Создать модель текущий ситуации" создает: CurrentSituation, User, Software
- (e) Reformulation WayToThink, используя результаты предыдущего шага синтезирует артефакты, которых не хватает, чтобы получить из CurrentState DesiredState, так как он не указан явно. WayToThink запускает Critic размышления, чтобы найти корень проблемы. Он находит CurrentState- Wordfinder Tehcnical, DesiredState- Wordfinder Business Economical
- (f) Рефлексивные Critic оценивают состояние системы - на каком шаге она находится, и если цель не достигнута, то запускают другой WayToThink, например, DirectInstruction.
- (g) Critic генерации решения запускает KnowingHow WayToThink, ExtensiveSearch.
- (h) Selector выбирает наиболее вероятный образ мышления. В данном случае ExtensiveSearch, который будет находить решения, позволяющие привести систему в необходимое состояние (DesiredState). Если он не сможет, то он инициирует коммуникацию с пользователем.

8. Рефлексивный Critic проверяет состояние системы. Если Цель достигнута, то пользователю посылается ответ.

9. Само Сознательные Critic активируется на данном шаге и сохраняют информацию о затратах на решение.

В главе приведено описание созданной специально для системы модели данных TUKnowledge 3.



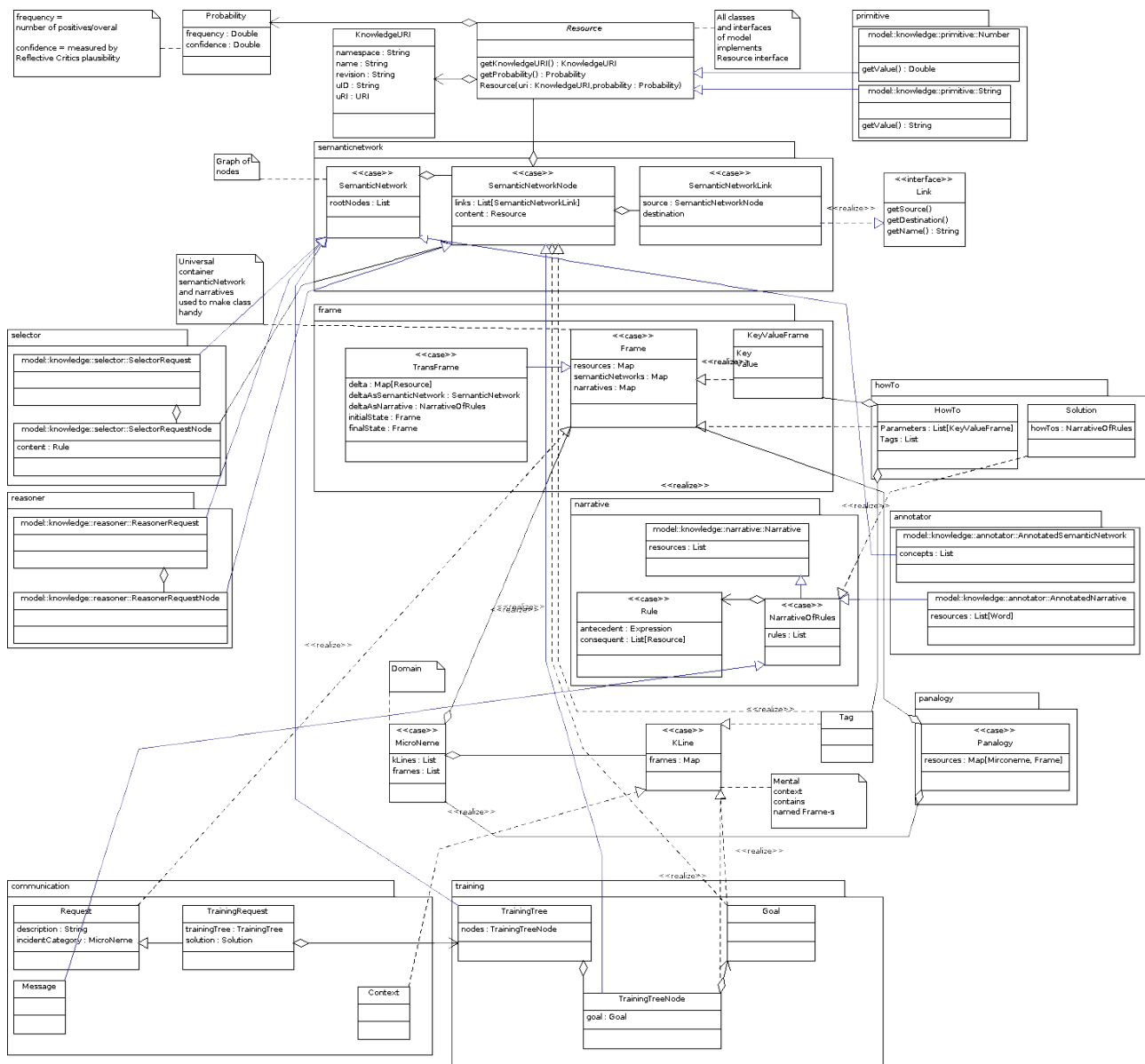


Рис. 3 — Схема данных TU Knowledge в формате UML

В **Главе 4** приведены Экспериментальные исследования эффективности работы модели TU. В данной главе были представлены основные результаты работы:

- Теоретико-множественный и теоретико-информационный анализ сложных систем в области поддержки информационной инфраструктуры
- Проблемно-ориентированная система управления, принятия решений и оптимизации технических объектов в области обслуживания IT
- Архитектура системы, ее реализация и испытания на модельных данных

Система показала свою жизнеспособность на модельных данных. Были проведены тесты в сравнении с работой человеческого специалиста. Был выбран контрольный список инцидентов. Сравнивался поиск решения для инцидентов. Основное время при опросе специалиста тратилось на коммуникацию. В Таблице приведены результаты сравнения **6**. Тесты были выполнены на машине Intel Core i7 1700 MHz, 8GB RAM, 256 GB SSD, FreeBSD.

Таблица 6 — Результаты сравнения с работой человеческого специалиста

Инцидент	TSS1 (.мс)	TU (.мс)
Tense is kind of concept.	15000	385
Please install Firefox.	9000	859
Browser is an object.	20000	400
Firefox is a browser.	5000	659
Install is an action.	8000	486
User miss Internet Explorer 8.	10000	10589
User needs document portal update.	15000	16543
Add new alias Host name on host that alias is wanted to: hrportal.lalala.biz IP adress on host that alias is wanted to: 322.223.333.22 Wanted Alias: webadviser.lalala.net	10000	18432
Outlook Web Access (CCC) - 403 - Forbidden: Access is denied	15000	10342
Продолжение следует		

**Таблица 6 – продолжение**

<b>Инцидент</b>	<b>TSS1 (.мс)</b>	<b>TU (.мс)</b>
PP2C - Cisco IP communicator. Please see if you can fix the problem with the ip phone, it's stuck on configuring ip + sometimes Server error rejected: Security etc.	13000	12343

В **заключении** приведены основные выводы по работе.

В работе были выполнены следующие задачи и достигнуты следующие результаты.

1. Была создана модель проблемно-ориентированной системы управления, принятия решений в области обслуживания информационной структуры предприятия на основе модели мышления
2. Была представлена новая модель данных для модели мышления и оригинальный способ ее хранения, эффективный по сравнению с другими базами данных
3. Было выполнено оригинальное исследование моделей мышления в области обслуживания информационной структуры предприятия
4. На основе модели была создана архитектура системы и ее прототип
5. Были созданы специальные алгоритмы для анализа запросов пользователя и принятия решений.
6. Система, разработанная в рамках данной работы, включает в себя инновационные методы и алгоритмы поддержки принятия решений, использующих в своей основе модель мышления на базе модели мышления Человека, описанной в книге Марвина Мински.
7. Была представлена наглядная визуализация структуры области удаленной поддержки инфраструктуры

Представленная в данной работе модель мышления, ее архитектура и реализация является уникальной в своем роде. На момент написания это была единственная реализация модели мышления Марвина Мински.

Разработанная в рамках работы системы не является узкоспециализированной. Она также подходит для других областей, где требуется поддержка принятия решений. Например, при постановке медицинского диагноза, чтобы отбросить ложные диагнозы.

Например, систему можно обучить органам человека и их взаимосвязи. Далее

можно обучить каким заболеваниям подвержен тот или иной орган. Далее к каждому заболеванию добавить симптом. После этого можно делать запрос с симптомами и система выдаст список вероятных заболеваний со способами их лечения.

В области диагностики проблем в машиностроении. Обучить систему узлам автомобиля, проблемам с ними связанными, признаками этих проблем и способами их устранения.

## **Публикации автора по теме диссертации**

1. *Тоцев А. С.* К новой концепции автоматизации программного обеспечения // *Труды Математического центра имени Н.И. Лобачевского. Материалы Десятой молодежной научной школы-конференции 'Лобачевские чтения - 2011. - Казань, 31 октября - 4 ноября 2011'.* — 2011. — Vol. 44. — 2 pp.
2. *Toshchev A. Talanov M. Krehov A. Khasianov A.* Thinking-Understanding approach in IT maintenance domain automation // *Global Journal on Technology, Vol 3 (2013): 3rd World Conference on Information Technology (WCIT-2012).* — 2013. — Т. 3. — Режим доступа: <http://www.world-education-center.org/index.php/P-ITCS/issue/view/96>.
3. *Toshchev A. Talanov M.* Thinking model and machine understanding of English primitive texts and it's application in Infrastructure as Service domain // *Proceedings of AINL-2013.* — 2013. — Режим доступа: <http://ainlconf.ru/material201303>.
4. *Toshchev A. Talanov M.* ARCHITECTURE AND REALIZATION OF INTELLECTUAL AGENT FOR AUTOMATIC INCIDENT PROCESSING USING THE ARTIFICIAL INTELLIGENCE AND SEMANTIC NETWORKS // *Ученые записки ИСГЗ 2078-6980.* — 2014. — Vol. 2.
5. *Toshchev A. Talanov M.* Computational emotional thinking and virtual neurotransmitters // *International Journal of Synthetic Emotions (IJSE).* — 2014. — Т. 05/2014.

6. *Toshchev A. Talanov M. Appraisal, Coping and High Level Emotions Aspects of Computational Emotional Thinking // International Journal of Synthetic Emotions (IJSE). — 2015. — T. 06/2015.*
7. *Toshchev A. Thinking model and machine understanding in automated user request processing // CEUR Workshop Proceedings. — 2014. — T. 1297.*
8. *Toshchev A. Talanov M. Thinking Lifecycle as an Implementation of Machine Understanding in Software Maintenance Automation Domain // Agent and Multi-Agent Systems: Technologies and Applications: 9th KES International Conference, KES-AMSTA 2015 Sorrento, Italy, June 2015, Proceedings (Smart Innovation, Systems and Technologies). — 2015.*