

На правах рукописи

**Тимченко Вадим Андреевич**

**ПРЕОБРАЗОВАНИЕ КЛАССОВ СЕМАНТИЧЕСКИХ СЕТЕЙ**

05.13.11 – математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

**Автореферат**

диссертации на соискание ученой степени  
кандидата технических наук



Владивосток

2010

Работа выполнена в лаборатории интеллектуальных систем Института автоматизации и процессов управления ДВО РАН.

Научный руководитель:

доктор технических наук, доцент  
Князева Маргарита Александровна

Официальные оппоненты:

доктор технических наук, доцент  
Шахгельдян Карина Иосифовна

кандидат физико-математических наук,  
старший научный сотрудник  
Голенков Евгений Александрович

Ведущая организация:

Московский энергетический институт  
(Технический университет), г. Москва

Защита состоится 11 февраля 2011 года в 14 часов на заседании диссертационного совета Д 005.007.01 в Институте автоматизации и процессов управления ДВО РАН по адресу: 690041, г. Владивосток, ул. Радио, 5.

С диссертацией можно ознакомиться в библиотеке Института автоматизации и процессов управления ДВО РАН.

Автореферат разослан 28 декабря 2010 г.

Ученый секретарь  
диссертационного совета Д 005.007.01,  
к.т.н.

  
\_\_\_\_\_ А.В. Лебедев

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность работы.** В настоящее время компьютерная обработка информации является одним из критически важных видов деятельности в большинстве теоретических исследований и прикладных областей. Этот вид деятельности включает задачи по хранению, инженерии и обмену различными видами разнородных данных и знаний между программными сущностями и их компонентами. При этом могут изменяться такие аспекты информации, как язык (формализм) представления, уровень детализации и т.д. В связи с этим на первый план выходят вопросы преобразования обрабатываемой программными системами информации (данных, знаний) из одной формы представления в другую.

Графовые структуры и семантические сети успешно используются в качестве наглядного и универсального средства представления информации в разных предметных областях. Разработано множество различных моделей и формализмов для представления семантических сетей. Следовательно, многие задачи преобразования информации, возникающие в различных областях профессиональной деятельности, технологических пространствах, а также на стыке разных дисциплин и технологических пространств, можно сформулировать в терминах преобразования семантических сетей.

Большой вклад в разработку и исследование методов решения таких задач внесли В.Н. Вагин, В.Ш. Кауфман, С.П. Крицкий, В.Н. Касьянов, В.А. Серебряков, В.Н. Скворцов, А.Н. Терехов, Н. Ehrig, G. Karsai, Н.-J. Kreowski, G. Rozenberg, A. Schürr, G. Taentzer и другие.

В ряде областей и технологических пространств (преобразование графов, преобразование программ) уже существуют свои эффективные, проверенные временем технологии и методы решения задач преобразования информации из одной формы в другую. Разработано множество языков и формализмов, поддерживающих эти технологии. В сравнительно молодых областях и технологических пространствах (преобразование моделей, инженерия онтологий) также достигнуты существенные результаты. При этом разрабатываются и исследуются новые подходы к преобразованию информации, накапливается опыт использования основанных на этих подходах моделей и методов, предпринимаются шаги по их стандартизации. Разработано множество программных и инструментальных средств поддержки существующих моделей, языков и методов, позволяющих эффективно определять и выполнять преобразования информации в различных задачах, областях профессиональной деятельности и технологических пространствах.

Однако существующие методы имеют весьма узкую область применения и рассчитаны на круг специалистов, обладающих необходимыми знаниями в этой области, либо вовсе ориентированы на конкретную задачу или класс задач. Разработанные системы рассчитаны, как правило, на конкретное технологическое пространство или на выполнение только одного вида преобразований – эндогенных или экзогенных. Помимо этого, большинство средств могут работать только в однопользовательском режиме, либо в локальной сети компании разработчиков, но не могут функционировать в сети Интернет. Это сокращает доступность и масштаб практического применения таких средств, а также препятствует накоплению опыта их использования.

Таким образом, разные задачи преобразования информации решаются различными методами, единый подход к преобразованию разных видов информации,

представленных семантическими сетями, отсутствует. При возникновении новой задачи приходится искать адекватный метод решения и программное средство его поддержки. Если задача возникает на стыке разных дисциплин и технологических пространств, и данный класс задач не охвачен ни одной программной системой, то необходимо разрабатывать новое специализированное средство, покрывающее этот случай.

Поэтому актуальной является разработка общего подхода к решению задач преобразования информации, представленной семантическими сетями, возникающих в разных технологических пространствах, а также на их пересечении, и реализация Интернет-средств, основанных на этом подходе.

**Целью диссертационной работы** является разработка моделей, методов и Интернет-средств преобразований информации, представленной семантическими сетями, включающих эндогенные и экзогенные преобразования, а также преобразования в разных технологических пространствах и между ними.

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

1. Разработать общую концепцию преобразования информации, представленной семантическими сетями, в терминах описания их классов.
2. Разработать модель для описания классов семантических сетей и формализм для описания спецификаций преобразования семантических сетей – структурных проекций в терминах описания их классов.
3. Разработать метод преобразования классов семантических сетей на основе описания структурной проекции.
4. Разработать методы реализации средств преобразования информации, представленной семантическими сетями, на основе описания структурных проекций.
5. Разработать технологию решения задач преобразования информации с помощью разработанного комплекса программных средств и показать ее практическую применимость при решении задач преобразования информации, включая эндогенные и экзогенные преобразования, а также преобразования в разных технологических пространствах и между ними.

**Методы исследования.** Для решения указанных задач использовались теория формальных языков, теория графов, элементы теории множеств, а также методы теории синтаксического перевода и компиляции, методы объектно-ориентированного анализа и проектирования, методы системного программирования.

**Научная новизна** работы состоит в следующем:

- предложена оригинальная концептуальная схема преобразования информации, представленной семантическими сетями, в терминах описания их классов, являющаяся инвариантной по отношению к технологическим пространствам. Данная схема представляет собой совокупность трех трехуровневых схем, различные комбинации которых позволяют определять как структурные преобразования, так и преобразования из текстового представления информации в структурное и наоборот;
- впервые разработана основанная на категориальном базисе *сущность – отношение* модель для описания структуры информации как класса семантических сетей, которая расширена формализмом для описания связи класса семантических сетей с элементами конкретного синтаксиса языка, определяющего текстовое представление информации. Данная связь при этом одновременно является спецификацией преобразований как из текстового представления информации в структурное, так и наоборот;

- разработан не имеющий аналогов формализм для описания спецификаций преобразования семантических сетей в терминах описания их классов, являющийся инвариантным по отношению к технологическим пространствам и позволяющий описывать как эндогенные, так и экзогенные преобразования; а также метод, определяющий, как по заданной спецификации преобразования на основе исходной семантической сети получить целевую сеть, удовлетворяющую этой спецификации.

**Практическая ценность** работы состоит в следующем:

- разработанная модель для описания классов семантических сетей позволяет создать и развивать базу описаний классов семантических сетей с помощью средства редактирования, управляемого данной моделью;
- разработанная модель для описания структурных проекций позволяет создать и развивать базу структурных проекций с помощью средства редактирования, управляемого данной моделью;
- разработан комплекс программных средств для преобразования информации, представленной семантическими сетями, на основе описания структурных проекций;
- применение платформы облачных вычислений *Многоцелевой банк знаний* позволило сделать комплекс программных средств доступным через сеть Интернет, при этом пользователям, имеющим соответствующие права, доступно не только выполнение преобразований, но и удаленное редактирование информационных баз, используемых при преобразовании.

**Реализация результатов работы.** Результаты диссертационной работы используются:

- в научных исследованиях, проводимых в рамках различных программ и проектов в лаборатории интеллектуальных систем Института автоматизации и процессов управления ДВО РАН, г. Владивосток;
- в учебном процессе на кафедре Программного обеспечения ЭВМ при разработке учебно-методических комплексов и чтении курсов лекций по дисциплинам “Теория вычислительных процессов и структур II. Теория и методы трансляции” и “Оптимизация программ” для студентов, обучающихся по специальности 010503.65 – “математическое обеспечение и администрирование информационных систем” в Дальневосточном федеральном университете, г. Владивосток.

Программное средство “Система, моделирующая процесс преобразования программ, управляемый знаниями” зарегистрировано в Реестре программ для ЭВМ 15 августа 2006 г. (Свидетельство об официальной регистрации программы для ЭВМ №2006612928).

Программное средство “Преобразователь семантических сетей” зарегистрировано в Реестре программ для ЭВМ 9 июля 2010 г. (Свидетельство о государственной регистрации программы для ЭВМ №2010614544).

**Обоснованность и достоверность полученных результатов.** Обоснованность и достоверность научных положений, выводов и прикладных результатов обеспечиваются используемыми методами исследования, а также практическим применением предложенных в диссертационной работе моделей, методов и алгоритмов.

**Апробация работы.** Основные положения диссертации докладывались и обсуждались на следующих международных и российских конференциях и семинарах: Дальневосточной математической школе-семинаре имени академика Е.В. Золото-

ва (г. Хабаровск, 2005, 2009, г. Владивосток, 2006, 2007, 2008, 2010), Международной конференции “Искусственный интеллект. Интеллектуальные и многопроцессорные системы” (г. Таганрог, 2005), Научно-практической конференции “Современные проблемы и пути их решения в науке, транспорте, производстве и образовании” (г. Одесса, 2005), Научной сессии МИФИ (г. Москва, 2006), Второй Международной конференции “Системный анализ и информационные технологии” (г. Обнинск, 2007), Международной конференции KDS-2008 (г. Варна, Болгария), Девятой международной научно-технической конференции “Искусственный интеллект. Интеллектуальные системы ИИ-2008” (пос. Кацивели, АР Крым, Украина, 2008), Одиннадцатой национальной конференции по искусственному интеллекту с международным участием “КИИ-2008” (г. Дубна, 2008), Рабочем семинаре “Наுகоемкое программное обеспечение” в рамках PSI’09 – Седьмой международной конференции памяти академика А. П. Ершова “Перспективы систем информатики” (г. Новосибирск, 2009), Международной научно-технической конференции “Искусственный интеллект. Интеллектуальные и многопроцессорные системы-2009” (г. Таганрог, Россия, 2009), Первой Российско-Тихоокеанской конференции по компьютерным технологиям и приложениям RPC 2010 (г. Владивосток, Россия, 2010), Открытом дальневосточном конкурсе программных средств студентов, аспирантов и молодых специалистов “Программист-2005”, “Программист-2006”, “Программист-2010” (г. Владивосток, 2005, 2006, 2010), Конкурсе научных работ молодых ученых и специалистов ИАПУ ДВО РАН (г. Владивосток, 2009), а также на совместных семинарах лаборатории интеллектуальных систем ИАПУ ДВО РАН и базовой кафедры программного обеспечения ЭВМ ДВГУ (2006 – 2009).

**Публикация результатов работы.** По материалам диссертации опубликовано 19 работ, из них 2 статьи в журналах, рекомендуемых ВАК РФ для опубликования научных результатов, получено 2 свидетельства о регистрации программ для ЭВМ.

**Структура и объем работы.** Диссертационная работа состоит из введения, пяти глав, заключения, списка литературы, включающего 156 наименований, и 20 приложений. Основная часть работы изложена на 130 страницах, содержит 50 рисунков и 6 таблиц.

## СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обоснована актуальность выполненных в диссертации исследований, сформулированы цель и задачи работы, рассмотрены научная новизна, практическая ценность результатов, приведены сведения об апробации и реализации основных положений диссертации, изложена структура диссертационной работы.

**Первая глава** содержит обзор литературы. В ней дан анализ современного состояния исследований в области преобразования семантических сетей. Рассмотрены наиболее распространенные модели и формализмы для представления семантических сетей, различные постановки задачи преобразования семантических сетей, а также задачи, к ней сводимые. Дана классификация преобразований. Представлены методы преобразования сетевых структур, языки и формализмы для спецификаций таких преобразований, а также рассмотрены инструментальные и программные средства их поддержки.

Во **второй главе** диссертации сформулированы принципиальные требования к средству преобразования информации, представленной семантическими сетями, пре-

дложена концептуальная схема преобразования информации.

Определение 1. Под *описанием класса семантических сетей информации* ( $S$ ) будем понимать описание устройства информации из некоторой предметной области – ее структуры, свойств и ограничений на возможное содержание в виде семантической сети.

Определение 2. Под *представлением информации в виде семантической сети* или просто *семантической сетью* ( $S_{inst}$ ) будем понимать семантическую сеть, являющуюся экземпляром (информацией более низкого уровня общности) по отношению к  $S$ , представляющей собой информацию более высокого уровня общности – метайнформацию.

Это означает, что вершины  $S_{inst}$ , представляющие понятия, входящие в объемы понятий, представленных вершинами из  $S$ , являются экземплярами этих вершин из семантической сети  $S$ , тогда как сами вершины семантической сети  $S$  являются прототипами для вершин из семантической сети  $S_{inst}$ . Таким образом,  $S$  определяет, в сущности, язык для описания множества  $S_{inst}$ .

Определение 3. Под *структурной проекцией* ( $\mu$ ) (спецификацией преобразования) описания класса семантических сетей исходной информации на описание класса семантических сетей целевой информации будем понимать множество правил (соответствий)  $\mu = \{Rule_i\}_{i=1}^{rulescount}$ . Каждое правило  $Rule_i$  определяет описание устройства (структуру и содержание) подсети семантической сети целевой информации, являющейся экземпляром некоторой подсети из описания класса семантических сетей целевой информации, сопоставляемое понятию из описания класса семантических сетей исходной информации.

Определение 4. Под *преобразованием информации (семантических сетей)* понимается генерация целевой семантической сети ( $TS_{inst}$ ), являющейся экземпляром описания класса семантических сетей целевой информации ( $TS_{nc}$ ), на основе исходной семантической сети ( $SS_{inst}$ ), являющейся экземпляром описания класса семантических сетей исходной информации ( $SS_{nc}$ ), по описанию структурной проекции описания класса семантических сетей исходной информации на описание класса семантических сетей целевой информации (рис. 1).

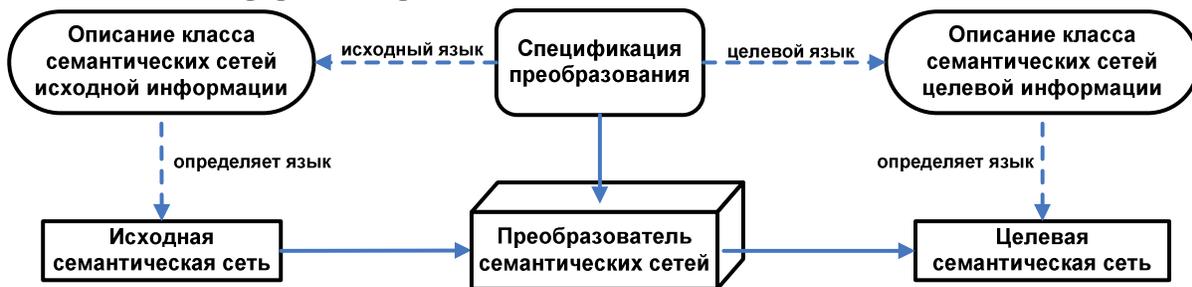


Рис. 1. Общая схема преобразования информации, представленной семантическими сетями

К средству преобразования информации, представленной семантическими сетями, предъявляются следующие *принципиальные* требования.

Требование 1. Средство преобразования должно быть спроектировано с соблюдением принципа разделения программных компонентов и обрабатываемой

ими информации, которая должна храниться отдельно и редактироваться независимо от средств ее обработки.

Требование 2. Должна быть обеспечена возможность выполнять как эндогенные, так и экзогенные преобразования, а также преобразования внутри различных технологических пространств и между технологическими пространствами.

Требование 3. Должна обеспечиваться возможность выполнять преобразования вида “текст – семантическая сеть” и “семантическая сеть – текст”.

Требование 4. Должна обеспечиваться возможность работы с программным средством через локальную вычислительную сеть и Интернет.

Схему преобразования информации, представленной семантическими сетями, можно представить как совокупность трех концептуальных схем, каждая из которых соответствует одному из трех базовых типов преобразования: “семантическая сеть – семантическая сеть”, “текст – семантическая сеть” и “семантическая сеть – текст” и их комбинаций.

В **третьей главе** представлена модель описания классов семантических сетей, модель и язык для описания структурных проекций, а также метод преобразования классов семантических сетей на основе описания структурной проекции.

Модель описания классов семантических сетей есть четверка:  $\langle Concepts\_Network, Id\_Concept, Const\_Concept, Syntax\_Restrictions \rangle$ .

1.  $Concepts\_Network = \langle Concepts, Relations, Attributes, Initial\_Concept \rangle$  – описание класса сетей понятий ( $S_{nc}$ ).

1.1.  $Concepts = \{Concept_i\}_{i=1}^{conceptscount}$  – конечное непустое множество понятий. Каждое понятие  $Concept_i$  описывается следующим образом:  $Concept_i = \langle Concept\_Value, Concept\_Kind, Concept\_TerminalType \rangle$ .

1.1.1.  $Concept\_Value = \langle Concept\_Value\_Type, Concept\_Value\_Value \rangle$  – значение понятия описывается своим типом  $Concept\_Value\_Type$  и значением  $Concept\_Value\_Value$ . Значение понятия есть непустая последовательность символов, которая идентифицирует определенный класс объектов описываемой предметной области, множество значений (сорт) или представляет в ней конкретное (константное) значение.  $Concept\_Value\_Type \in \{“Строковое”, “Целое”, “Вещественное”, “Логическое”\}$ .  $Concept\_Value\_Value \in String \cup Integer \cup Real \cup Boolean$ , где  $String$  – множество строк,  $Integer$  – множество целых чисел,  $Real$  – множество вещественных чисел,  $Boolean$  – множество {“true”, “false”}. Значением понятия может быть последовательность символов, интерпретируемых как строковая константа, целое число из множества целых чисел, вещественное число из множества вещественных чисел, элемент множества {“true”, “false”}.

1.1.2.  $Concept\_Kind \in \{“Терминальное”, “Нетерминальное”\}$  – тип понятия. Понятие может быть терминальным или нетерминальным.

1.1.3.  $Concept\_TerminalType \in \{“Идентификатор”, “Константа”, “Сорт”, “Не определено”\}$  – тип терминального понятия. Терминальное понятие может быть идентификатором, или константой, или обозначать некоторый сорт. Если понятие  $Concept_i$  является нетерминальным, то  $Concept\_TerminalType = “Не определено”$ .

1.2.  $Relations = \{Relation_i\}_{i=0}^{relationscount}$  – конечное, возможно пустое, множество отношений. Каждое отношение  $Relation_i$  является направленным бинарным отношением (дугой), имеющим, возможно, спецификатор множественности и

связывающим два понятия. Отношение описывается следующим образом:  $Relation_i = \langle Relation\_EndSp, Begin\_Concept, End\_Concept \rangle$ .

1.2.1.  $Relation\_EndSp \in \{“Конкретность”, “Единственность”, “Множество”\}$  – спецификатор множественности (кардинальность) – характеристика отношения, определяющая способ порождения понятия-экземпляра (понятия из  $S_{inst}$ ) вместе с отношением к нему по понятию-концу данного отношения ( $End\_Concept$  – понятие из  $S_{nc}$ ).  $Relation\_EndSp = “Конкретность”$  (или пустой спецификатор) означает, что может быть порождено только одно понятие-экземпляр и его значение (имя) должно совпадать со значением (именем) его понятия-прототипа ( $End\_Concept$ ).  $Relation\_EndSp = “Единственность”$  означает, что по  $End\_Concept$  может быть порождено только одно понятие-экземпляр, но его значение (имя) обязательно должно быть задано при порождении. Порожденное понятие при этом является сущностью из класса объектов или элементом из множества значений, идентифицируемого понятием-прототипом ( $End\_Concept$ ) (строка, целое значение, вещественное значение, логическое значение).  $Relation\_EndSp = “Множество”$  означает, что по  $End\_Concept$  может быть порождено любое количество (но, по крайней мере, один) понятий-экземпляров и их значения (имена) обязательно должны быть заданы при порождении. Каждое порожденное понятие при этом является сущностью из класса объектов или элементом из множества значений, идентифицируемого понятием-прототипом ( $End\_Concept$ ).

1.2.2.  $Begin\_Concept$  – понятие, из которого дуга исходит – понятие-начало отношения.  $Begin\_Concept \in Concepts \setminus Terminal\_Concepts$ , где  $Terminal\_Concepts$  – множество терминальных понятий,  $Terminal\_Concepts \subset Concepts$ . Понятием-началом отношения может быть любое нетерминальное понятие.

1.2.3.  $End\_Concept$  – понятие, в которое дуга входит – понятие-конец отношения.  $End\_Concept \in Concepts \setminus \{Initial\_Concept\}$ . Понятием-концом отношения может быть любое понятие за исключением *начального понятия*.

1.3.  $Attributes = \{Attribute_i\}_{i=0}^{attributescount}$  – конечное, возможно пустое, множество атрибутов. Каждый атрибут  $Attribute_i$  представляет некоторое свойство понятия и описывается следующим образом:  $Attribute_i = \langle Attribute\_Name, Attribute\_Argument, Attribute\_Value \rangle$ .

1.3.1.  $Attribute\_Name$  – имя атрибута, представляющее собой непустую последовательность символов.

1.3.2.  $Attribute\_Argument \in Concepts$ . Аргументом атрибута может быть понятие из множества описанных понятий.

1.3.3.  $Attribute\_Value = \langle Attribute\_Value\_Type, Attribute\_Value\_Value \rangle$  – значение атрибута описывается своим типом  $Attribute\_Value\_Type$  и значением  $Attribute\_Value\_Value$ .  $Attribute\_Value\_Type \in \{“Понятие”, “Строковое”, “Целое”, “Вещественное”, “Логическое”, “Файл”\}$ .  $Attribute\_Value\_Value \in Concepts \cup String \cup Integer \cup Real \cup Boolean$ . Значением атрибута может быть ссылка на понятие, последовательность символов, интерпретируемых как строковая константа, целое число из множества целых чисел, вещественное число из множества вещественных чисел, элемент множества  $\{“true”, “false”\}$  или содержимое файла в бинарном виде ( $Attribute\_Value\_Value$  в этом случае интерпретируется как путь к файлу).

1.4. *Initial\_Concept* – начальное понятие в описании сети понятий информации, оно единственно, и через него не может быть выражено ни одно другое понятие из описываемого класса сетей понятий.  $Initial\_Concept \in Concepts$ .

2. *Id\_Concept* – понятие, представляющее идентификатор в описании класса сетей понятий информации, оно единственно и не может быть выражено через другие понятия, т.е. является терминальным понятием типа “Сорт”.  $Id\_Concept \in Concepts$ .

3. *Const\_Concept* – понятие, представляющее константу в описании класса сетей понятий информации, оно единственно и не может быть выражено через другие понятия, т.е. является терминальным понятием типа “Сорт”.  $Const\_Concept \in Concepts$ .

4. *Syntax\_Restrictions* =  $\langle Lexica, Syntax \rangle$  – описание синтаксических ограничений. Оно может отсутствовать, если текстовое представление информации не требуется, или же информация вовсе не имеет текстового представления.

$$4.1. Lexica = \{ \langle Lexem\_Type_i, Definition_i \rangle \}_{i=1}^5$$

$$4.2. Syntax = \{ \langle Concept_i, Definition_i \rangle \}_{i=1}^{conceptscount-2}$$

$Lexem\_Type_i \in \{ \text{“Идентификатор”, “Целое число”, “Вещественное число”, “Строковая константа”, “Ограничитель строковой константы”} \}$ .

$Concept_i \in Concepts \setminus \{ Id\_Concept, Const\_Concept \}$ .

$Definition_i$  – строка символов, включающая метасимволы и представляющая конкретное лексическое или синтаксическое ограничение для конкретного понятия или вида лексемы.

Синтаксическое определение представляет собой определение объемлющего понятия из описания класса сетей понятий через объемлемые понятия и элементы конкретного синтаксиса языка текстового представления. В определении фиксируется также порядок следования объемлемых понятий и элементов конкретного синтаксиса. Лексический словарь языка определяется фиксированным набором вершин, соответствующих базовым типам данных – целое, вещественное и строковое и, кроме этого, определением вида идентификатора, или имени.

Все ограничения на вид лексем контекстно-свободного языка задаются с помощью регулярных выражений с использованием нотации perl5. Метасимволы не распространяются на лексические ограничения.

Описание структурной проекции  $\mu$  представляет собой множество правил  $\mu = \{ Rule_i \}_{i=1}^{rulescount}$ . В общем виде каждое правило  $Rule_i$  можно представить следующим образом:  $\alpha \rightarrow \beta$ , где  $\alpha \in Source\_Concepts$ , где  $Source\_Concepts$  – множество понятий из  $SS_{nc}$ ;

$$\beta = \{ Target\_Concept\_Descr_i \}_{i=0}^{conceptscount} \cup \{ Target\_Relation\_Descr_j \}_{j=0}^{relationscount} \cup \{ Target\_Attribute\_Descr_k \}_{k=0}^{attributescount}, \text{ где}$$

$\{ Target\_Concept\_Descr_i \}_{i=0}^{conceptscount}$  – возможно пустое множество описаний порождаемых понятий в семантической сети  $TS_{inst}$ .  $Target\_Concept\_Descr_i$  есть описание порождения понятия в  $TS_{inst}$ .

$\{ Target\_Relation\_Descr_j \}_{j=0}^{relationscount}$  – возможно пустое множество описаний создаваемых отношений в семантической сети  $TS_{inst}$ .  $Target\_Relation\_Descr_j$  есть описание порождения отношения в  $TS_{inst}$ .

$\{Target\_Attribute\_Descr_k\}_{k=0}^{attributescount}$  – возможно пустое множество описаний создаваемых атрибутов в семантической сети  $TS_{inst}$ .  $Target\_Attribute\_Descr_k$  есть описание порождения атрибута в  $TS_{inst}$ . При этом  $\beta \neq \emptyset$  и для любых  $\alpha_1, \alpha_2 \in Source\_Concepts$ , если правила  $\alpha_1 \rightarrow \beta_1, \alpha_2 \rightarrow \beta_2$  входят в  $\mu$ , то  $\alpha_1 \neq \alpha_2$ .

Описание создания понятия в  $TS_{inst}$  ( $Target\_Concept\_Descr$ ) представляет собой шестерку:  $Target\_Concept\_Descr = \langle Proto\_Concept\_Name, SConceptValue, isTProtoConceptName, Storable, Loadable, Concept\_Value \rangle$ .

1.  $Proto\_Concept\_Name$  – имя понятия-прототипа из описания класса сетей понятий целевой информации  $TS_{nc}$ , которое представляет собой непустую последовательность символов.

2.  $SConceptValue = \langle isSConceptValue, isTerminal, isAttributeValue, attributeName \rangle$ . Описывает выбор значения для создаваемого в  $TS_{inst}$  понятия на основе имени (значения) понятия из  $SS_{inst}$  или на основе значения атрибута понятия из  $SS_{inst}$ .

2.1.  $isSConceptValue \in \{“true”, “false”\}$ . Определяет, нужно ли создаваемому в  $TS_{inst}$  понятию присваивать в качестве значения имя (значение) понятия из  $SS_{inst}$  или значение атрибута понятия из  $SS_{inst}$ .

2.2.  $isTerminal \in \{“true”, “false”\}$ . Определяет, значение какого понятия (или его атрибута) из  $SS_{inst}$  нужно присвоить: экземпляра понятия, стоящего в левой части правила (“false”), или терминального понятия-потомка экземпляра понятия, стоящего в левой части правила (“true”).

2.3.  $isAttributeValue \in \{“true”, “false”\}$ . Определяет, присваивать создаваемому в  $TS_{inst}$  понятию в качестве значения имя (значение) понятия из  $SS_{inst}$  (“false”) или значение атрибута этого понятия (“true”).

2.4.  $attributeName$  – строка символов, возможно пустая, представляющая имя атрибута, значение которого нужно присвоить создаваемому в  $TS_{inst}$  понятию в качестве значения.

3.  $isTProtoConceptName \in \{“true”, “false”\}$ . Определяет, нужно ли создаваемому в  $TS_{inst}$  понятию присваивать в качестве значения имя (значение) его понятия-прототипа из  $TS_{nc}$ .

4.  $Storable = \langle isStorable, Alias, storeConcept \rangle$ . Описывает сохранение значений в хеш-таблицу.

4.1.  $isStorable \in \{“true”, “false”\}$ . Определяет, нужно ли значение, которое будет присвоено создаваемому в  $TS_{inst}$  понятию, поместить (сохранить) в хеш-таблицу.

4.2.  $Alias$  – строка символов, возможно пустая, представляющая псевдоним, под которым нужно сохранить значение понятия в хеш-таблицу.

4.3.  $storeConcept \in \{“true”, “false”\}$ . Определяет, нужно ли поместить (сохранить) в хеш-таблицу создаваемое в  $TS_{inst}$  понятие целиком или же только его значение (имя).

5.  $Loadable = \langle isLoadable, Alias \rangle$ . Описывает извлечение значений из хеш-таблицы.

5.1.  $isLoadable \in \{“true”, “false”\}$ . Определяет, будет ли значение, которое необходимо присвоить создаваемому в  $TS_{inst}$  понятию, извлекаться из хеш-таблицы.

5.2.  $Alias$  – строка символов, возможно пустая, представляющая псевдоним, по которому нужно получить значение для понятия из хеш-таблицы.

6.  $Concept\_Value = \langle Concept\_Value\_Type, Concept\_Value\_Value \rangle$ . Описывает значение (имя), присваиваемое создаваемому в  $TS_{inst}$  понятию.

6.1. *Concept\_Value\_Type*  $\in$  {"Строковое", "Целое", "Вещественное", "Логическое", "Вычислимое целое", "Вычислимое вещественное", "Вычислимое строковое"} – тип значения.

6.2. *Concept\_Value\_Value* – собственно значение понятия – строка символов, интерпретируемая в зависимости от значения типа понятия. Значением понятия может быть строковая константа, целое число из множества целых чисел, вещественное число из множества вещественных чисел, элемент множества {"true", "false"}. Если тип значения понятия "Вычислимое целое" (*Concept\_Value\_Type* = "Вычислимое целое"), то значение понятия интерпретируется как выражение, значение которого требуется вычислить, и результат вычисления будет интерпретироваться как целое число. Если тип значения понятия "Вычислимое вещественное" (*Concept\_Value\_Type* = "Вычислимое вещественное"), то результат вычисления будет интерпретироваться как вещественное число. Если тип значения понятия "Вычислимое строковое" (*Concept\_Value\_Type* = "Вычислимое строковое"), то вычисленное значение будет строковой константой. Под выражением здесь понимаются арифметико-логическое выражение, операции над строками, множество вложенных условных операторов.

Описание создания отношения в *TS<sub>inst</sub>* (*Target\_Relation\_Descr*) представляет собой тройку: *Target\_Relation\_Descr* =  $\langle$ *Relation\_Begin\_Concept*, *Relation\_End\_Concept*, *Number* $\rangle$ .

1. *Relation\_Begin\_Concept* =  $\langle$ *Begin\_Concept\_Name*, *isProtoConceptName*, *isLoadable*, *Alias* $\rangle$ . Описывает понятие-начало отношения.

1.1. *Begin\_Concept\_Name* – имя понятия-начала отношения, представляющее собой непустую последовательность символов.

1.2. *isProtoConceptName*  $\in$  {"true", "false"}. Определяет, является ли это имя именем понятия из *TS<sub>nc</sub>*.

1.3. *isLoadable*  $\in$  {"true", "false"}. Определяет, будет ли понятие-начало отношения, извлекаться из хеш-таблицы.

1.4. *Alias* – строка символов, возможно пустая, представляющая псевдоним, по которому нужно получить понятие из хеш-таблицы.

2. *Relation\_End\_Concept* =  $\langle$ *End\_Concept\_Name*, *isProtoConceptName*, *isLoadable*, *Alias* $\rangle$ . Описывает понятие-конец отношения. Аналогично описанию *Relation\_Begin\_Concept*, только *End\_Concept\_Name* есть имя понятия-конца отношения, которое также представляет собой непустую последовательность символов, а *isLoadable* определяет, будет ли понятие-конец отношения, извлекаться из хеш-таблицы.

3. *Number* – целое положительное число – порядковый номер отношения. Определяет номер, под которым создаваемое отношение будет добавлено к отношениям, в которых уже участвует понятие-начало.

Описание создания атрибута в *TS<sub>inst</sub>* (*Target\_Attribute\_Descr*) представляет собой шестерку: *Target\_Attribute\_Descr* =  $\langle$ *Attribute\_Name*, *SConceptValue*, *Attr\_Storable*, *Attr\_Loadable*, *Attribute\_Argument*, *Attribute\_Value* $\rangle$ .

1. *Attribute\_Name* – имя атрибута, представляющее собой непустую последовательность символов.

2. *SConceptValue* =  $\langle$ *isSConceptValue*, *isTerminal*, *isAttributeValue*, *attributeName* $\rangle$ .

Описывает выбор значения для создаваемого в  $TS_{inst}$  атрибута на основе имени (значения) понятия из  $SS_{inst}$ . или на основе значения атрибута понятия из  $SS_{inst}$ . Аналогично описанию выбора значения для создаваемого понятия.

3.  $Attr\_Storable = \langle isStorable, Alias \rangle$ . Описывает сохранение значения атрибута в хеш-таблицу. Описание компонентов аналогично описанию сохранения понятия в хеш-таблицу.

4.  $Attr\_Loadable = \langle isLoadable, Alias \rangle$ . Описывает извлечение значений из хеш-таблицы. Описание компонентов аналогично описанию извлечения значений для понятий из хеш-таблицы.

5.  $Attribute\_Argument = \langle Attribute\_Argument\_Value, isProtoConceptName \rangle$ . Описывает аргумент атрибута.

5.1.  $Attribute\_Argument\_Value$  – собственно аргумент атрибута – строка символов, интерпретируемая как имя (значение) понятия из  $TS_{inst}$  или  $TS_{nc}$ .

5.2.  $isProtoConceptName \in \{“true”, “false”\}$ . Определяет, является ли имя понятия-аргумента именем понятия из  $TS_{nc}$  (“true”) или  $TS_{inst}$  (“false”).

6.  $Attribute\_Value = \langle Attribute\_Value\_Type, Attribute\_Value\_Value \rangle$ .

6.1.  $Attribute\_Value\_Type \in \{“Понятие”, “Строковое”, “Целое”, “Вещественное”, “Логическое”, “Файл”, “Вычисляемое целое”, “Вычисляемое вещественное”, “Вычисляемое строковое”\}$ .

6.2.  $Attribute\_Value\_Value$  – собственно значение атрибута – строка символов, интерпретируемая в зависимости от типа. Если тип значения атрибута “Понятие”, то значением атрибута является ссылка на понятие. Если тип значения атрибута “Файл” то  $Attribute\_Value\_Value$  интерпретируется как путь к файлу, а значением атрибута является содержимое этого файла в бинарном виде.

Для формирования пользователем множества правил в терминах приведенной выше модели разработан язык для описания структурных проекций.

Метод преобразования классов семантических сетей на основе описания структурной проекции можно представить как совокупность трех базовых методов преобразования: “текст – семантическая сеть”, “семантическая сеть – текст” и “семантическая сеть – семантическая сеть”.

Метод преобразования “текст – семантическая сеть” включает анализ текстового представления, построение последовательности номеров правил вывода грамматики и формирования семантической сети на основе полученной последовательности. Для анализа текста и построения вывода, используется алгоритм Эрли. Для формирования семантической сети предназначен алгоритм, основная идея которого состоит в том, чтобы, двигаясь от вершины сети (начального понятия), достраивать сеть понятий в соответствии со сформированной по описанию синтаксических ограничений КС-грамматикой и сгенерированной последовательностью списков ситуаций Эрли.

Метод преобразования “семантическая сеть – текст” состоит в рекурсивном обходе заданной семантической сети в глубину с параллельным просмотром правил сформированной по описанию синтаксических ограничений КС-грамматики для синтеза фрагментов результирующего текста.

Метод преобразования “семантическая сеть – семантическая сеть” основан на операционной семантике языка описания структурных проекций. Его суть состоит в

обходе исходной семантической сети в глубину и параллельном достраивании подсети в целевой семантической сети в соответствии с описанием правой части выбранного правила. После применения очередного правила формируемая целевая сеть всегда соответствует своей метainформации.

**Четвёртая глава** диссертационной работы содержит описание архитектуры и методов реализации средства преобразования классов семантических сетей на основе описания структурной проекции.

На рис. 2 изображена архитектура средства преобразования классов семантических сетей как совокупность программных и информационных компонентов. На данной схеме отражены потоки данных между программными компонентами, а преобразователь классов семантических сетей представлен в виде совокупности трех подсистем – подсистема анализа и синтеза текстовых представлений информации (АИСТ), генератор семантических сетей и подсистема управления.

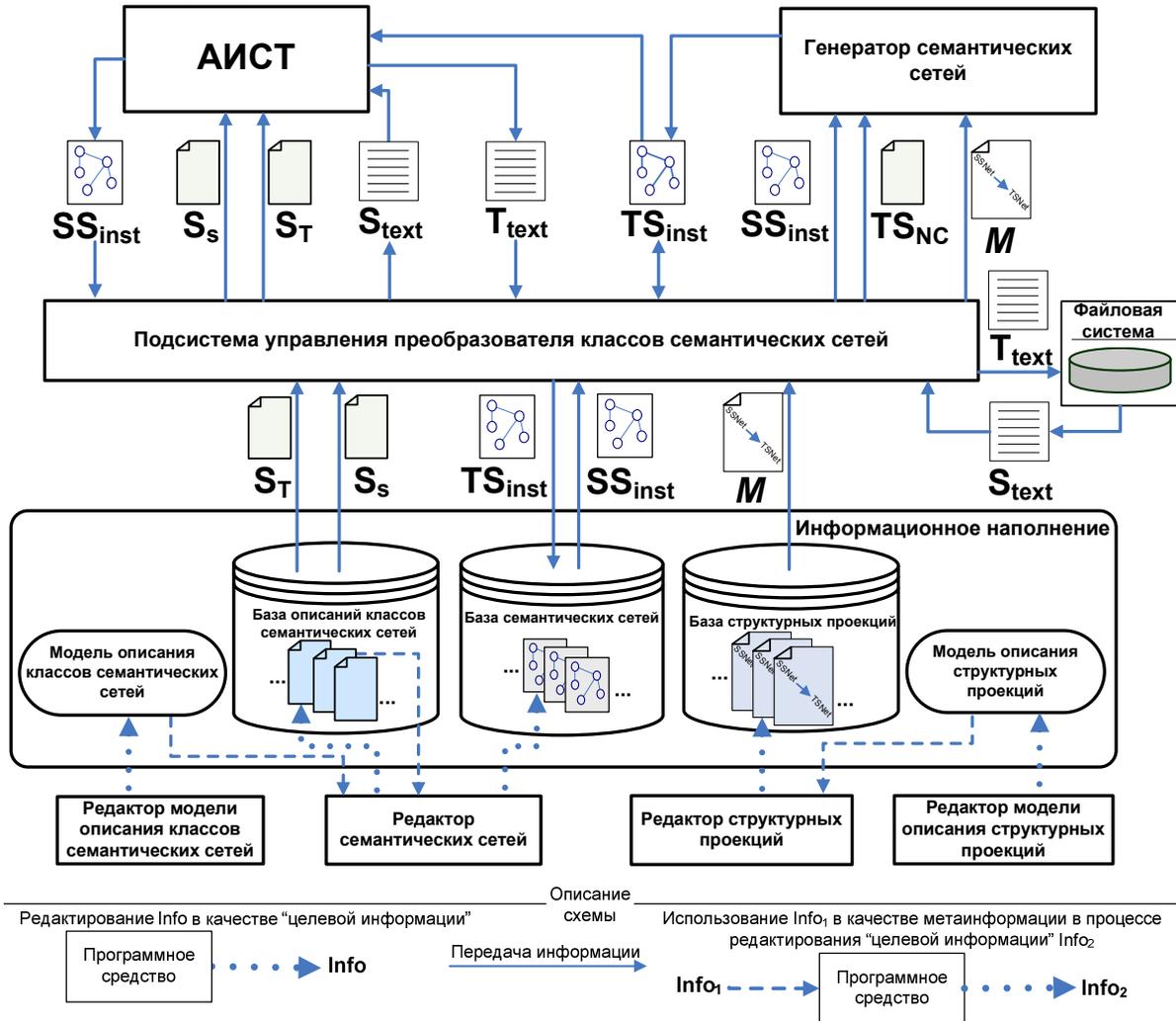


Рис. 2. Архитектура средства преобразования классов семантических сетей на основе описания структурной проекции

Для обеспечения возможности работы с программным средством через Интернет/Интранет оно реализовано как клиент-серверное приложение с использованием платформы облачных вычислений *Многоцелевой банк знаний (МБкЗ)*.

Все информационные компоненты программного средства расположены на **сервере**, на котором развернут МБкЗ, в *информационном наполнении МБкЗ*. Для

каждого информационного компонента средства преобразования классов семантических сетей через административную систему зарегистрирован и создан соответствующий ему *информационный ресурс* в МБкЗ. На **клиентском компьютере** пользователя находятся средства редактирования и преобразователь классов семантических сетей.

Каждое средство редактирования реализуется с использованием *Редактора информации различного уровня общности (ИРУО)* как инструментального средства для создания специализированных средств редактирования информационных ресурсов МБкЗ.

Оставшиеся три компонента программной части средства реализованы в виде отдельного приложения (преобразователя классов семантических сетей), состоящего из трех подсистем – подсистемы управления, подсистемы АИСТ и генератора семантических сетей. Для реализации программного средства выбран язык программирования Java фирмы Sun Microsystems, поскольку он позволяет разрабатывать кроссплатформенные приложения, работающие под управлением виртуальной Java-машины. Для удаленного доступа к информационному наполнению МБкЗ используется специальный программный интерфейс (API), предоставляемый платформой МБкЗ.

В **пятой главе** описана технология решения практических задач преобразования информации, представленной семантическими сетями, с помощью разработанных программных средств. Продемонстрированы примеры описаний проекций семантических сетей в различных технологических пространствах, а также приведены их количественные характеристики.

Пользователей комплекса программных средств можно разделить на две группы: *разработчики спецификации преобразования* и *прикладные пользователи*. В задачи разработчиков спецификации преобразования входит формирование Базы описаний классов семантических сетей и формирование Базы структурных проекций. В задачи прикладных пользователей входит: формирование Базы семантических сетей – формирование представлений информации в виде семантических сетей и занесение их в базу, формирование файлов с текстовым представлением информации и выполнение процесса преобразования семантических сетей, представляющих информацию.

На рис. 3 изображена общая схема использования программного комплекса при решении задач преобразования информации, представленной семантическими сетями. Цифры на стрелках обозначают очередность, в которой прикладной пользователь задает информацию, необходимую для процесса преобразования. Надпись на стрелке вида “[i]” означает, что действие, указанное в ромбе, в который входит стрелка с цифрой i, не является обязательным для запуска процесса преобразования. Надпись на стрелке вида “[i]”j” означает, что действие, указанное в ромбе, в который входит стрелка с такой надписью, может быть выполнено либо i-м, либо j-м по порядку.

Практическая применимость разработанной технологии рассматривалась на следующих задачах: преобразование базы знаний о заболеваниях; преобразование диаграмм классов на языке UML в их представление в формате XML; разработка

конвертера языка программирования MILAN на язык программирования ASPLE;  
задача построения модели структурных программ.

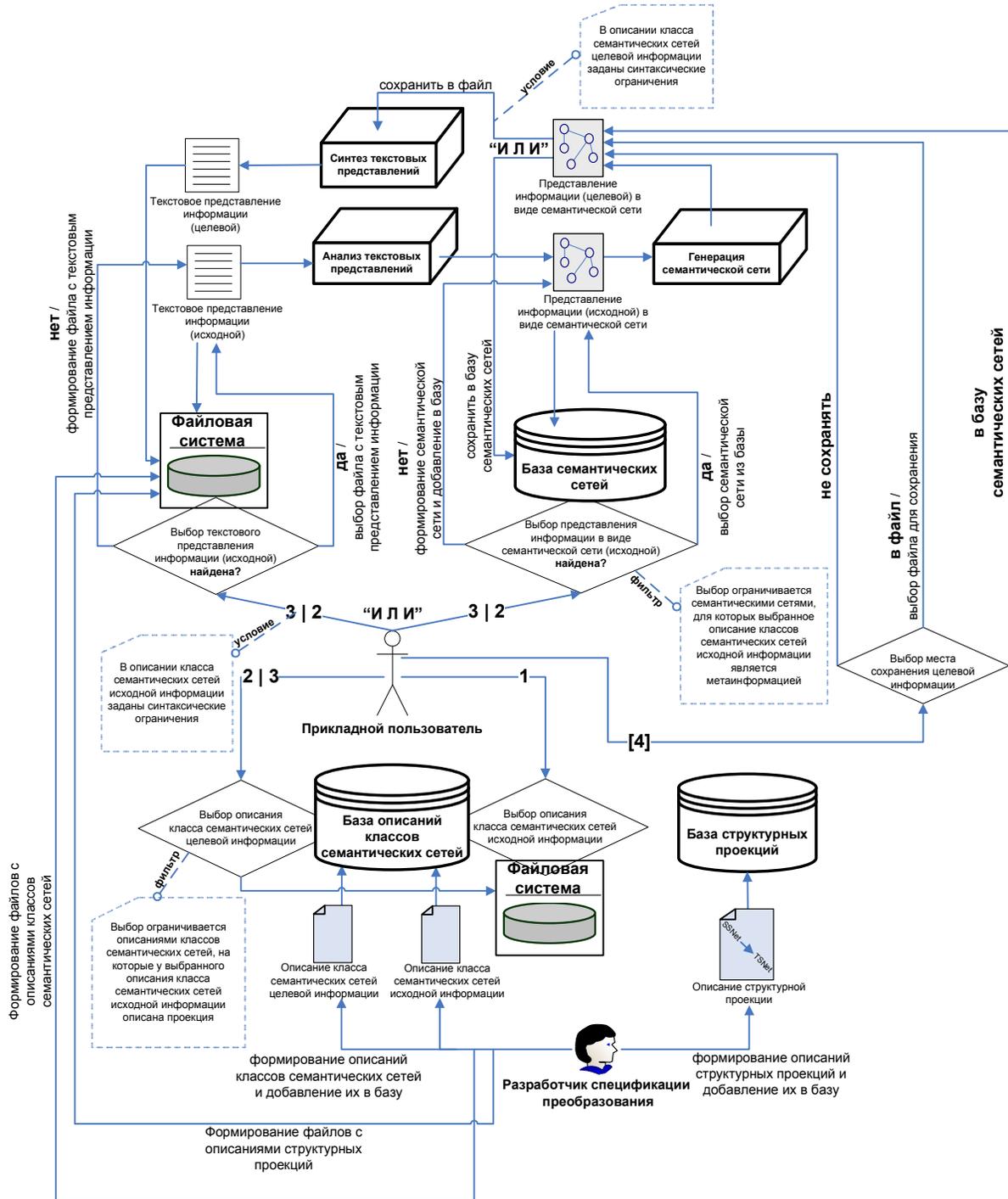


Рис. 3. Общая схема использования программного комплекса при решении задач преобразования информации, представленной семантическими сетями

Описанная совокупность примеров преобразования информации охватывает весь спектр видов преобразований: эндогенные и экзогенные, вертикальные и горизонтальные, а также преобразования внутри разных технологических пространств и между технологическими пространствами. При этом применяются все базовые типы преобразований: “семантическая сеть – семантическая сеть”, “текст – семантическая сеть” и “семантическая сеть – текст”. Помимо этого, задействуются

все три комбинации базовых типов преобразований: “семантическая сеть – семантическая сеть – текст”, “текст – семантическая сеть – семантическая сеть” и “текст – семантическая сеть – семантическая сеть – текст”.

Количественные характеристики классов семантических сетей, представляющих описание использованных в примерах видов информации, а также количество правил в описании структурной проекции приведены в сводной таблице 1.

Таблица 1 – Количественные характеристики классов семантических сетей, представляющих описание перечисленных видов информации и количество правил в описании проекции

Характеристика Вид информации	Количество понятий	Количество отношений	Количество синтаксических ограничений	Количество правил в описании проекции	
Описание базы знаний о заболеваниях	8	7	—	4	
Описание преобразованной базы знаний о заболеваниях	4	3	—		
Описание диаграммы классов языка UML	7	11	—	7	
Описание языка XML	12	16	11		
Описание языка MILAN	41	60	39	35	
Описание языка ASPLE	46	65	44		
Ограниченное подмножество языка Pascal	105	171	103	59	—
Ограниченное подмножество языка C	144	229	142		80
Язык модели структурных программ	16	15	—		

## ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. На основе анализа основных задач преобразования различных видов информации, решаемых в научных исследованиях и профессиональной деятельности, а также моделей, методов и средств, существующих для решения таких задач, сформулирован и обоснован набор принципиальных требований к средству преобразования информации, представленной семантическими сетями. На основе требований предложена трехкомпонентная трехуровневая концептуальная схема преобразования информации, представленной семантическими сетями. Данная схема ориентирована на возможность описания преобразований “семантическая сеть – семантическая сеть”, “семантическая сеть – текст” и “текст – семантическая сеть” (трехкомпонентность), описания эндогенных и экзогенных преобразований, а также преобразований внутри и между разными технологическими пространствами (трехуровневость).

2. Разработана модель для описания классов семантических сетей. Данная модель базируется на формализме представления семантических сетей, основанном на максимально абстрактных категориях “сущность” и “отношение”. Расширение модели позволяет специфицировать преобразования вида “текст – семантическая сеть” и “семантическая сеть – текст” одинаковым образом: используется один и тот же формализм, нотация, вид правил и т.д.

3. Разработаны модель и язык для описания структурных проекций. Язык предназначен для описания преобразований вида “семантическая сеть – семантическая сеть”. Описан абстрактный и конкретный синтаксис языка. Данный язык не ориентирован на конкретное технологическое пространство и позволяет специфицировать преобразования в самых общих категориях – “сущность”, “свойство”, “отношение”.

4. Разработаны методы преобразований “текст – семантическая сеть”, “семантическая сеть – текст” и “семантическая сеть – семантическая сеть”, определяющие, как по соответствующей спецификации преобразования на основе входной информации получить выходную, удовлетворяющую этой спецификации. Используемый метод преобразования “текст – семантическая сеть” позволяет анализировать текст на языке, задаваемом КС-грамматикой без ограничений.

5. Разработаны методы реализации программного средства преобразования информации, представленной семантическими сетями, на основе описания структурных проекций. Для обеспечения процесса отдельного, независимого (от программных компонентов) сопровождения информации соответствующими специалистами все информационные компоненты программного средства выделены в отдельные информационные базы. Для каждой базы разработан свой специализированный редактор, позволяющий удаленно формировать и поддерживать в актуальном состоянии соответствующую информационную базу. Преобразователь классов семантических сетей обеспечивает возможность работы как в “сетевом” (через Интернет/Интранет), так и в “локальном” режиме, а также позволяет формировать и сохранять графические представления семантических сетей.

6. Предложена технология решения различных задач преобразования информации, представленной семантическими сетями, с помощью разработанного комплекса программных средств. Показана практическая применимость разработанной технологии и комплекса программных средств при решении задач преобразования информации, возникающих в разных областях профессиональной деятельности и технологических пространствах. Для этого была подготовлена выборка примеров задач преобразования информации, которая покрывала бы весь спектр видов преобразований. Результаты практического применения показали, что разработанное средство удовлетворяет предъявляемым к нему требованиям и позволяет успешно решать указанные задачи.

### ОСНОВНЫЕ ПУБЛИКАЦИИ

1. Князева М.А., Тимченко В.А. Структурные редакторы программ на языках программирования высокого уровня и генератор моделей структурных программ в Банке знаний о преобразованиях программ // Искусственный интеллект. – 2005. – Т.4. – С. 200 – 208.
2. Князева М.А., Тимченко В.А. Средства структурного редактирования информационного наполнения в Банке знаний о преобразованиях программ // **Вестник компьютерных и информационных технологий**. – Москва. – 2006. – №6. – С. 43 – 49.
3. Клещев А.С., Князева М.А., Тимченко В.А. Интеллектуальная система генерации единого внутреннего представления в системе преобразований программ // Вторая Международная конференция “Системный анализ и информационные технологии”: Труды конференции. В 2 т. – Москва. – 2007. – Т.1. – С. 130 – 133.
4. Князева М. А., Тимченко В.А. Подсистема генерации единого внутреннего представления в системе преобразований программ // **Программные продукты и системы**. – 2008. – №1. – С. 58 – 62.
5. Knyazeva M.A., Timchenko V.A. Ontology-based model of representation of knowledge about language mappings // International Book Series “INFORMATION SCIENCE & COMPUTING”, Number 5. Supplement to the International Journal “INFORMATION TECHNOLOGIES & KNOWLEDGE” – Bulgaria, Sofia: FOI ITHEA. – 2008. – Vol. 2. – Pp. 111 – 118. – ISSN: 1313-0455 (printed), 1313-048X (online), 1313-0501 (CD/DVD).
6. Князева М.А., Тимченко В.А. Генерация программ в целевом представлении на основе описаний проекций языков исходного представления на языки целевого представления // Искусственный интеллект. Интеллектуальные системы: Материалы IX Международной научно-технической конференции. – Донецк. – 2008. – Т.2. – С. 343 – 347.

7. Князева М.А., Тимченко В.А. Преобразование программ из исходного представления в целевое представление на основе описаний проекций языка исходного представления на язык целевого представления // Искусственный интеллект. – 2008. – Т.3. – С. 681 – 689.
8. Князева М.А., Тимченко В.А. Модель онтологии проекций языков программирования высокого уровня на единое представление программ // Одиннадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-08. Труды конференции. – Москва. – 2008. – Т.1. – С. 124 – 132.
9. Князева М.А., Тимченко В.А. Система преобразования графовых структур на основе проекций // Седьмая международная конференция памяти академика А. П. Ершова “Перспективы систем информатики” Рабочий семинар “Научное программное обеспечение”. Информационный бюллетень. – Новосибирск. – 2009. – С. 168 – 172.
10. Князева М.А., Тимченко В.А. Преобразование графовых структур представления информации // Искусственный интеллект. – 2009. – Т.4. – С. 425 – 436.
11. Тимченко В.А. Преобразователь семантических сетей // Открытый Дальневосточный конкурс программных средств студентов, аспирантов и молодых специалистов “Программист-2010”. Сборник докладов. Владивосток. – 2010. – С. 17 – 20.
12. Тимченко В.А. Концепция преобразования информации на основе проекций классов семантических сетей // XXXV Дальневосточная математическая школа-семинар имени академика Е.В. Золотова: сб. докл. [Электронный ресурс]. – Владивосток. – 2010. – С. 886 – 891.
13. Knyazeva M.A., Timchenko V.A. The concept of information transformation tool based on mapping of graph structures // Proceedings of the First Russia and Pacific Conference on Computer Technology and Applications (RPC 2010). [Electronic resource]. – Vladivostok, Russia. – 2010. – Pp. 145 – 150.
14. Князева М.А., Бердник А.Н., Волков Д.А., Жеравин М.В., Зотов И.Ю., Маевский М.С., Плохих С.А., Тимченко В.А. Система, моделирующая процесс преобразования программ, управляемый знаниями. Зарегистрировано в Реестре программ для ЭВМ 15 августа 2006 г. Свидетельство Федеральной службы по интеллектуальной собственности, патентам и товарным знакам об официальной регистрации программы для ЭВМ № 2006612928.
15. Князева М.А., Тимченко В.А. Преобразователь семантических сетей. Зарегистрировано в Реестре программ для ЭВМ 9 июля 2010 г. Свидетельство Федеральной службы по интеллектуальной собственности, патентам и товарным знакам о государственной регистрации программы для ЭВМ № 2010614544.

**Личный вклад автора.** Все результаты, составляющие основное содержание диссертации, получены автором самостоятельно. В работах [1, 2] автором описаны виды информации, с которой работает программное средство для преобразования информации на основе проекций классов семантических сетей, а также средства редактирования этой информации. В работах [3, 4, 7, 9, 10, 13] автору принадлежит определение концепции системы для преобразования информации на основе проекций классов семантических сетей и методов ее реализации. В работах [5, 8, 10, 13] автором описаны модель и язык для описания структурных проекций. В работе [6] автором дано описание алгоритма преобразования семантических сетей на основе описания структурной проекции. В работе [10] автор приводит описание использования программного средства в рамках системы преобразований программ. В работе [14] разработанное программное средство “Преобразователь классов семантических сетей” является одной из подсистем в системе, моделирующей процесс преобразования программ, управляемый знаниями. В работе [15] автором разработаны подсистема генерации семантических сетей, подсистема анализа и синтеза текстов и подсистема управления.

Тимченко Вадим Андреевич

ПРЕОБРАЗОВАНИЕ КЛАССОВ СЕМАНТИЧЕСКИХ СЕТЕЙ

Автореферат

Подписано к печати 21.12.2010  
Формат 60x84/16

Усл. печ. л. 1,0  
Тираж 100

Уч.-изд. л. 0,8  
Заказ 42

---

Издано ИАПУ ДВО РАН. Владивосток, ул. Радио, 5  
Отпечатано участком оперативной печати ИАПУ ДВО РАН  
Владивосток, ул. Радио, 5