МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕСИТЕТ»

На правах рукописи

Наместников Алексей Михайлович

ИНТЕЛЛЕКТУАЛЬНЫЕ РЕПОЗИТОРИИ ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ В ПРОЕКТИРОВАНИИ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

05.13.12 – Системы автоматизации проектирования (промышленность)

ДИССЕРТАЦИЯ
на соискание ученой степени
доктора технических наук

Научный консультант
д. т. н., профессор
Ярушкина Н. Г.

Ульяновск – 2017

Оглавление

Введение

Актуальность темы исследования.

Принятие проектных решений при создании сложных программно-аппаратных комплексов, к которым можно отнести современные автоматизированные системы (АС), сопряжено с необходимостью анализа большого объема разнородной информации. Системы автоматизированного проектирования (САПР) постоянно усложняются, и, как следствие, ужесточаются требования к их информационному обеспечению. Существующие подходы к формированию информационного обеспечения САПР позволяют решать задачи организации информационных баз с целью получения необходимых данных на всем протяжении жизненного цикла проектируемой системы. Однако все чаще начинает возникать проблема оперативной доступности информации, когда фактографические или документальные базы данных содержат необходимые данные для принятия проектных решений, но получить доступ к ним затруднительно по причине отсутствия дополнительных знаний о содержании информационных ресурсов. Жесткая система классификаторов, унифицированных форм технических документов, правил структурной организации массивов проектной информации в составе современных проектных репозиториев САПР уже не позволяют с прежней эффективностью осуществлять информационную поддержку процесса проектирования.

АС относятся к классу систем, которые интенсивно используют программное обеспечение. Автоматизация разработки программных систем предполагает использование различных шаблонов проектирования и фреймворков. Соответствующими артефактами проектирования в этом случае являются не только текстовые документы, но и исходные тексты программ и различные проектные диаграммы, разрабатываемые с использованием слабоформализованных нотаций (например, UML).

Отсутствие в современных электронных архивах проектных организаций

методов и средств выполнения контекстно-ориентированных запросов к слабоструктурированным гетерогенным информационным ресурсам, которые являются артефактами проектных процедур создания АС, не позволяет на начальных этапах проектирования эффективно использовать накопленный опыт формирования проектных решений с целью сокращения времени проектирования АС, что является **актуальной научно-технической проблемой**.

Решение данной проблемы может основываться на применении дополнительных знаний о предметной области проектной организации, которые способствуют повышению качества информационной поддержки процесса проектирования. Благодаря усилиям консорциума W3C разработаны и утверждены ряд стандартов в области Semantic Web, которые позволяют разрабатывать системы, основанные на знаниях, с использованием единого подхода к представлению, обмену и обработке информации не только на синтаксическом, но и на семантическом уровнях. К таким стандартам можно отнести расширяемый язык разметки XML, XMI – стандарт OMG для обмена метаданными с помощью языка XML, язык описания информационных ресурсов RDF, язык описания онтологий OWL и язык запросов к онтологическим хранилищам SPARQL.

Существующие семантические технологии ориентированы на формирование информационной среды, которая способна быть посредником между динамично изменяющейся внешней средой проектной организации и многочисленными гетерогенными источниками проектных данных. Такой подход к организации информационного обеспечения САПР позволяет повысить качество информационной поддержки процесса проектирования АС посредством включения в жизненный цикл проектируемых АС специализированных знаний предметной области и обеспечить возможность накопления индивидуального опыта специалистов в процессе выполнения проектных процедур. Значительный вклад в разработку методов представления предметных знаний на основе онтологии внесли такие исследователи, как Гаврилова Т.А., Загорулько Ю.А., Соловьев В.Д., Хорошевский В.Ф., Gruber T., Ushold M. В работах исследователей Но-

ренкова И.П., Малюх В.Н., Голенкова В.В., Смирнова С.В., Соснина П.И., Боргеста Н.М. подчеркивается актуальность применения онтологического анализа в процедурах проектирования сложных технических систем.

Очевидно, что проектировщику АС в своей деятельности приходится сталкиваться с задачами анализа не только структурированной информации в фактографических базах данных, но и со слабоструктурированной и неструктурированной проектной информацией. Содержимое документальных баз данных извлекается из технических документов, аннотаций программных модулей, всевозможных моделей и диаграмм, построенных с использованием различных нотаций (например, нотаций IDEF1X и UML). Для разработки единого подхода для интеллектуального анализа слабоструктурированных гетерогенных информационных ресурсов проектной организации требуется синтез методов, моделей и алгоритмов онтологического анализа в условиях неполной информации и неопределенности.

Принципиальная неполнота проектной информации, определяемая в работах Батыршина И.З., Берштейна Л.С., Борисова А.Н. накладывает ограничения на логико-лингвистические модели интеллектуального анализа содержимого проектных репозиториев автоматизированного проектирования. Синтез научного направления «мягкие вычисления (Soft Computing)», включающий в себя теорию нечетких множеств и генетические алгоритмы, с подходом представления экспертных знаний на основе дескриптивных логик (Description Logic) позволяет решать задачи информационной поддержки начальных стадий процесса проектирования сложных АС. Предметом данного исследования является именно этот класс задач.

В диссертации обобщены результаты теоретических и прикладных исследований в области моделирования процессов взаимодействия проектировщика АС с архивом технических документов на семантическом уровне.

Актуальность диссертационной работы обусловлена определенной выше проблемой и постоянно увеличивающимся количеством проектов, предполага-

ющих интенсивное взаимодействие проектных групп и, следовательно, формирование единого информационного пространства проектной организации.

Цели и задачи диссертационной работы.

Целью диссертационной работы является сокращение сроков выполнения начальных этапов проектирования АС за счет повышения точности и полноты выполнения профессиональных проектных запросов к электронным архивам проектных организаций на основе разработанных теоретических положений для реализации онтологического подхода к интеллектуальному анализу слабоструктурированных информационных ресурсов.

Для достижения указанной цели решены следующие задачи исследования:

1. Анализ современных подходов к реализации информационного обеспечения САПР АС на синтаксическом и семантическом уровне обработки информации.
2. Разработка теоретических основ нечетких онтологических систем информационной поддержки проектировщика АС.
3. Разработка методов и средств концептуального индексирования слабоструктурированных информационных ресурсов проектных репозиториев САПР.
4. Исследование и развитие комплекса моделей интеллектуального информационного взаимодействия субъекта проектирования с интеллектуальным проектным репозиторием.
5. Разработка онтологических программных средств информационной поддержки проектирования АС как интеллектуальной компоненты САПР АС.

Методы исследования.

При выполнении работы использованы основные положения и методы системного анализа, онтологического анализа, теории графов, искусственного интеллекта, теории нечетких множеств, приближенных множеств Павлака и дескриптивных логик.

Научная новизна.

В результате выполнения диссертационной работы были разработаны теоретические, методологические и практические основы онтологического подхода к анализу технической документации в проектировании автоматизированных систем, а именно:

1. Разработан онтологический подход, модели, методы и средства которого представляю собой теоретическую основу для анализа слабоструктурированных ресурсов проектной организации на начальных этапах проектирования сложных автоматизированных систем, нацеленных на сокращение времени проектных процедур и отличающийся от известных использованием нечетких логических формализмов при формировании контекстно-ориентированных профессиональных запросов к архивам технических документов.
2. Предложена интегрированная модель системы онтологий интеллектуального проектного репозитория для решения задачи информационной поддержки автоматизированного проектирования, отличающаяся новой структурой и позволяющая выполнять информационное взаимодействие с проектными репозиториями на семантическом уровне.
3. Разработан метод концептуального индексирования слабоструктурированных информационных ресурсов электронных архивов проектной организации, отличающийся единым подходом к интеллектуальному анализу проектной информации на основе описания предметной области в виде онтологии.
4. На основе введенного понятия концептуального индекса разработаны новые методы интеллектуального анализа текстовых документов при автоматизированном проектировании, позволяющие формировать навигационную структуру документов проектного репозитория в контексте жизненного цикла проектирования автоматизированных систем.
5. Разработан новый метод содержательной интерпретации кластеров техни-

ческих документов и технических временных рядов на основе лингвистических шкал и приближенных множеств Павлака, позволяющий реализовывать объяснительную компоненту интеллектуальной САПР на основе онтологии предметной области.

6. Разработаны и обоснованы нечеткая модель и методика оценки качества онтологии на основе свойств нечетких соответствий, позволяющие выполнять оперативный контроль процесса автоматизированного формирования онтологии.
7. Разработаны методологические основы построения интеллектуальных онтологических систем информационной поддержки процесса проектирования автоматизированных систем, основанные на интеграции нечетко-логического, графо-аналитического и вероятностного подходов к анализу слабоструктурированной информации с целью интенсификации процессов интеллектуализации проектных репозиториев.

Практическая значимость и результаты внедрения.

Разработана архитектура интеллектуального проектного репозитория. Разработан предметно-ориентированный редактор онтологий информационной поддержки процесса проектирования автоматизированных систем. Разработан комплекс программ, составляющий интеллектуальный проектный репозиторий и реализующий информационную поддержку проектировщика, который позволяет выполнять контекстно-ориентированные проектные запросы к электронным архивам технических документов и осуществлять структуризацию документов в соответствии с жизненным циклом проектируемых автоматизированных систем.

Результаты работы используются в ФНПЦ АО «НПО «Марс» (г. Ульяновск). Данное исследование было поддержано грантами РФФИ № 10-07-00064 в 2010, 2011 и в 2012 годах, РФФИ № 16-47-730742 и 16-47-732033 в 2016 и 2017 годах, а также выполнялось согласно тематическим планам научных исследований Федерального агентства по образованию в 2009-2010 годах. Результаты

диссертационной работы используются в учебном процессе кафедры «Информационные системы» при подготовке студентов направлений «Программная инженерия» и «Прикладная экономика». Под руководством автора защищены 2 кандидатские диссертации по тематике исследования.

Положения, выносимые на защиту:

1. Разработан подход к онтологическому анализу слабоструктурированных информационных ресурсов в проектных репозиториях, основанный на введенном понятии концептуального индекса проектного репозитория САПР. Данный подход позволяет выполнять анализ технических документов и проектных диаграмм на семантическом уровне, с учетом жизненного цикла проектируемых автоматизированных систем.
2. Свойство неполноты информационных ресурсов электронных архивов проектной организации является принципиальным и может быть формализовано в онтологии с использованием нечетко-логического подхода к представлению знаний предметной области.
3. Предлагается метод концептуального индексирования текстовых технических документов и проектных диаграмм, учитывающий особенности реализации проектной деятельности в виде применяемых стандартов и терминологических словарей и позволяющий выполнять контекстно-ориентированные профессиональные запросы к электронному архиву проектной организации.
4. Разработан метод нечетко-лингвистической интерпретации кластеров технических документов электронного архива, позволяющий формировать содержательную оценку навигационной структуры архива на базе системы понятий онтологии предметной области.
5. Разработан метод онтологической интерпретации технических временных рядов показателей проектируемых автоматизированных систем, позволяющий определять и интерпретировать фрагменты ряда в терминах предметной области объекта автоматизации.

6. Разработан способ доопределения понятийного аппарата онтологии предметной области системой терминов в виде концептуальной сети из внешних профессиональных структурированных wiki-ресурсов, нацеленный на сокращение трудоемкости построения онтологий проектных организаций за счет частичной автоматизации процесса формирования онтологических компонентов.
7. Разработана архитектура интеллектуального проектного репозитория, отличающаяся интеллектуальной компонентой, представление знаний в которой базируется на разработанной системе моделей онтологии информационной поддержки автоматизированного проектирования. Данное решение позволяет повысить точность и полноту проектных информационных запросов к электронному архиву и сократить время выполнения начальных этапов проектирования автоматизированных систем.

Степень достоверности и апробация результатов.

Достоверность научных положений и выводов, сформулированных в диссертации, подтверждается проведением вычислительных экспериментов, непротиворечивыми математическими моделями, результатами практического использования предложенных в диссертации методов и алгоритмов, подтвержденных актами об их применении.

Основные научные положения диссертации докладывались, обсуждались и получили одобрение на Всероссийской молодежно-практической конференции «Информационные и кибернетические системы управления и их элементы» (Уфа, 1997 г.); Научной сессии МИФИ-2001 (Москва, 2001 г.); Международном научно-практическом семинаре «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (Коломна, 2001 г.); Российской конференции с международным участием AIS'08 «Интеллектуальные системы» (Москва, 2008 г.); 11-ой национальной конференции по искусственному интеллекту с международным участием «КИИ-2008» (Дубна, 2008 г.); Всероссийской научной конференции «Нечеткие системы и мягкие вычисления (НСМВ-2008)» (Улья-

новск, 2008 г.); Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (Москва, 2009 г.); Международной конференции «Интеллектуальные системы (AIS'09)» (Геленджик, 2009 г.); Всероссийской конференции «Проведение научных исследований в области хранения, передачи и защиты информации» (Ульяновск, 2009 г.); 12-ой национальной конференции по искусственному интеллекту с международным участием «КИИ-2010» (Тверь, 2010 г.); 6-ой международной научно-технической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (Коломна, 2011 г.); 1-м международном симпозиуме «Гибридные и синергетические интеллектуальные системы: теория и практика» (Калининград, 2012 г.); 13-ой национальной конференции по искусственному интеллекту с международным участием «КИИ-2012» (Белгород, 2012 г.); 3-й международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2013)» (Минск, 2013 г.); 7-ой международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (Коломна, 2013 г.); 4-й международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2014)» (Минск, 2014 г.); 4-й Всероссийской научно-практической конференции «Нечеткие системы и мягкие вычисления» (Санкт-Петербург, 2014 г.); 2-м Международном симпозиуме «Гибридные и синергетические системы: теория и практика (ГИСИС'2014)» (Светлогорск, 2014 г.); 14-ой национальной конференции по искусственному интеллекту с международным участием «КИИ-2014» (Казань, 2014 г.); 5-й международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2015)» (Минск, 2015 г.); 8-ой международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (Коломна, 2015 г.); 6-й международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных

систем (OSTIS- 2016)» (Минск, 2016 г.); 15-ой национальной конференции по искусственному интеллекту с международным участием «КИИ-2016» (Смоленск, 2016 г.); 7-й международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2017)» (Минск, 2017 г.).

Публикации.

Материалы диссертации опубликованы в 86 печатных работах, из них 2 монографии, 22 статьи в журналах из перечня ВАК, 35 статей в сборниках трудов конференций, 3 свидетельства о государственной регистрации программ для ЭВМ.

Личный вклад автора.

Содержание диссертации и основные положения, выносимые на защиту, отражают персональный вклад автора в опубликованные работы. Подготовка к публикации полученных результатов проводилась совместно с соавторами, причем вклад диссертанта был определяющим. Все представленные в диссертации результаты получены лично автором в течение 1997–2017 годов на кафедре «Информационные системы» Ульяновского государственного технического университета.

Структура и объем диссертации.

Диссертация состоит из введения, шести глав, заключения и списка литературы. Общий объем диссертации – 304 страницы, из них 257 страниц текста, включая 78 рисунков. Библиография включает 202 наименования на 25 страницах.

Глава 1

Методы и средства формирования информационного обеспечения САПР автоматизированных систем

1.1. Информационное обеспечение проектирования современных автоматизированных систем

Современное производственное предприятие для реализации основной деятельности использует автоматизированные системы (АС). Можно выделить следующие стадии развития АС:

- решение задач обработки структурированных данных (50-е годы);
- реализация методов комплексной автоматизации и построение информационного обеспечения с использованием баз данных (БД) (60-е годы);
- построение вычислительных систем с распределенной терминальной сетью (70-е годы);
- использование персональных компьютеров и создание автоматизированных рабочих мест (АРМ) (80-е годы);
- активное использование при разработке АС телекоммуникационных средств, создание корпоративных (интегрированных) систем (90-е годы);
- создание автоматизированных систем, которые осуществляют взаимодействие на основе глобальной сети Internet (2000-е годы).

В ГОСТ 34.003-90 дается следующее определение: «*автоматизированная система* – это система, состоящая из персонала и комплекса средств автоматизации его деятельности, реализующая информационную технологию выполнения установленных функций» [22]. Такой тип систем стандарт IEEE 1471 определяет, используя термин «software intensive systems» – системы, которые интен-

сивно используют программное обеспечение. Все это указывает на «существенное влияние программного обеспечения, входящего в их состав» [103], [157].

Принцип, предполагающий единство информационной базы, является одним из важных принципов разработки АС [12], [87]. Реализация данного принципа предполагает построение «единой динамической информационной модели объекта проектирования, содержащей необходимый и достаточный перечень показателей по информационной поддержке всех этапов жизненного цикла разрабатываемой системы».

Для обеспечения возможности автоматизированного проектирования АС важным этапом является формирование информационного обеспечения систем автоматизированного проектирования (САПР), которое применяется для решения следующих задач [87]:

- экономичного и однозначного представления данных в системе с использованием кодирования объектов;
- предоставление возможности выполнения функций анализа информации и ее обработки, учитывающих характер связей между объектами на основе классификации;
- обеспечение диалога пользователей с системой с использованием экранных форм ввода-вывода данных;
- организация эффективного использования данных в процедурах управления деятельностью объекта автоматизации с применением унифицированной системы документации.

Одно из определений информационного обеспечения приведено в руководящем документе по стандартизации РД 50-680-88 «Методические указания. Автоматизированные системы. Основные положения»: «*Информационное обеспечение автоматизированной системы – совокупность системно-ориентированных данных, описывающих принятый в системе словарь базовых описаний (классификаторы, типовые модели, элементы автоматизации, форматы документации и т.д.), и актуализируемых данных о состоянии информацион-*

ной модели объекта автоматизации (объекта управления, объекта проектирования) на всех этапах его жизненного цикла» [92].

Информация, включаемая в информационное обеспечение САПР, является чрезвычайно важной для формирования эффективных проектных решений. Для того чтобы использовать в автоматизированном проектировании указанную информацию для поиска, вычислительной обработки и передачи по каналам связи, необходимо ее представить в цифровом виде. С этой целью информация сначала подвергается упорядочиванию (выполняются процедуры классификации), а затем происходит формализация (кодирование) с использованием классификатора. «*Классификатор – это документ, с помощью которого осуществляется формализованное описание информации, принимающей участие в процессе проектирования, содержащей наименования объектов, наименования классификационных группировок и их кодовые обозначения*» [87].

Вся проектная информация группируется в *информационные ресурсы*, управление и обработка которых осуществляется корпоративными автоматизированными системами. *Информационные ресурсы (ИР) – это совокупность данных, имеющих смысловую нагрузку, отражающих всю производственно-хозяйственную деятельность предприятия и документально зафиксированных на носителях различного вида.* По содержанию ИР представляют собой отображение реальных процессов проектирования, зафиксированных в проектно-конструкторской документации, плановых и отчетных документах, нормативных и инструктивных материалах и т. д.

В последнее время для специалистов в области проектирования АС начинает преобладать несколько иное представление об информационном обеспечении, которое представляется как *определенная совокупность элементов информации: реквизитов, составных единиц информации, показателей, классификаторов, языков записей данных, правил структурной организации массивов, документов, обеспечивающих структурную организацию информации в системе.* Здесь следует обратить внимание на новый акцент, связанный с

проблемами структуризации элементов информационного обеспечения САПР.

В соответствии с РД 50-34.698-90 внутримашинной информационной базой (ИБ) называют совокупность всех данных на машинных носителях, сгруппированных по определенному признаку [19]. В составе внутримашинной ИБ могут выделяться: фактографическая база данных, документальная база данных, база знаний. Алгоритмы обработки информации в указанных базах данных заметно различаются. Поэтому в зависимости от характера информационных ресурсов, которыми оперируют такие системы, принято различать два крупных класса информационных систем: фактографические и документальные. База знаний рассматривается как основа информационного наполнения особого класса информационных систем – *интеллектуальных информационных систем*, или *систем, основанных на знаниях (экспертных)* [107]. При формировании информационного обеспечения современной САПР применяются как фактографические, так и документальные информационные системы.

В практике автоматизированного проектирования достаточно большой объем информации сосредоточен в технических документах, представленных в текстовом виде. Для работы с такими информационными ресурсами применяются документальные информационные системы (часто в литературе применяется термин – *информационно-поисковые системы (ИПС)*).

В ИПС документы могут быть представлены прямо либо косвенно. При прямом представлении документ хранится в памяти в обычной форме, а при косвенном представлении используются различные способы индексирования. По индексу можно получить адрес или идентификатор документа, который хранится в базе данных или в оперативной памяти. Как при прямом, так и при косвенном представлении документы можно хранить в виде полного текста или в усеченном виде. Из документов могут быть удалены так называемые «стоп-слова», а оставшиеся могут быть преобразованы к основной форме.

Простейшие запросы в документальных системах могут сводиться к совпадению с ключевыми словами, более сложные запросы состоят из логической

комбинации простых запросов и могут использовать относительное расстояние между ключевыми словами. В косвенном представлении и при кластеризации (классификации) документов используются ключевые слова или термины, выбираемые в соответствии с некоторой автоматической или полуавтоматической схемой индексирования. Ключевые слова и документы можно объединять в группы (кластеры). Группы ключевых слов используются для составления тезаурусов (словарей синонимов или ключевых слов), а группы документов – для разбиения очень больших документальных БД с целью их более эффективной обработки.

Документальные БД отличаются от фактографических систем тем, что первые разрабатываются нацеленные на частичное, приближенное представление данных, которые имеют достаточно сложную смысловую структуру (в основном, текст). В свою очередь, фактографические системы ориентированы на полное и точное представление информации, которая обладает достаточно простой смысловой структурой [87].

Несмотря на большое количество реализованных и применяемых на практике фактографических и документальных БД, имеются определенные проблемы их использования при формировании информационного обеспечения САПР. Классические модели данных и реляционные хранилища до сих пор интенсивно применяют разработчики инженерных программных систем CAD/CAM/CAE/PLM. Все попытки объединения разнородных, гетерогенных данных в рамках единой модели для производственных систем пока оказались неудачными. Причины этого кроются в фундаментальных ограничениях классической теории моделирования данных – наличие достаточно жестких границ между связями, атрибутами и сущностями. Имеет место разница в способах описания абстрактных понятий (пространство, время и т.д.) [88], а также традиционные представления о том, что мир можно разделить на классы, которые составляют единую иерархию [139].

При решении практических задач возникает множество проблем в том случае, когда необходимо сформулировать общий запрос сразу к нескольким базам данных (как к документальным, так и к фактографическим). В таких базах данных совершенно различным способом могут представляться объекты и атрибуты предметной области или существовать различные подходы учета изменения объектов, происходящие во времени. Такие условия и ограничения практически сводят на нет всеобщую унификацию.

Согласно данных национального института стандартизации и технологий США, решение проблем взаимодействия разнородных компьютерных систем поможет предприятиям, которые занимаются реализацией крупных капитальных проектов, экономить несколько миллиардов долларов только в США. Для обеспечения возможности эффективного управления проектированием и эксплуатацией сложного инженерного объекта необходимо иметь полный доступ ко всем данным объекта на протяжении всего жизненного цикла. В разных источниках указанная задача называется по-разному: формирование единого информационного пространства (ЕИП) жизненного цикла, построение цифровой модели и т. д.

Проблему интеграции корпоративных данных удобнее всего решать с использованием семантического подхода к моделированию данных. Данный подход основывается на том, что информация предоставляется в виде множества связанных отношениями элементов данных (графовая модель данных), а не в виде привычных таблиц. Сфера применения для инженерных и корпоративных данных в настоящее время, как правило, предполагает использование стандарта RDF (Resource Description Framework).

Семантические модели, основанные на онтологическом моделировании, разрабатываются для инженерных объектов на разных уровнях: от отраслевого уровня до международного уровня. На данный момент известен стандарт ISO 15926 «Industrial automation systems and integration. Integration of life-cycle data for process plants including oil and gas production facilities» (ГОСТ-Р ИСО 15926

«Промышленные автоматизированные системы и интеграция. Интеграция данных жизненного цикла для перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия») [18], в рамках которого разработана метамодель, которая является нейтральной по отношению к отдельным инженерным системам и вобрала в себя многие современные решения по моделированию данных.

Фактически, данный стандарт определяет универсальную инженерную онтологию – основные типы объектов и отношений, которые используются при представлении инженерной информации, позволяют выполнять упорядочивание терминологии, а также определяют принципы расширения стандартной терминологии через механизм федерированных библиотек справочных данных. В настоящее время многие крупные компании уже начинают применять этот стандарт на практике: компании-члены Norwegian Oil Industry Association, члены консорциума FIATECH, крупнейшие поставщики инжинирингового программного обеспечения, а также российские корпорации, такие как ГК «Росатом» и ОАО «Роснефть», изучают потенциальные возможности его использования.

Применение и реализация стандарта ISO 15926 в части представления, хранения и доступа к данным основана на использовании семантических стандартов консорциума W3C: RDF, OWL и SPARQL.

1.2. Организация проектных репозиториев

1.2.1. Формальные модели информационных ресурсов в документальных базах данных

В настоящее время во многих организациях, в которых осуществляется проектирование АС, документальные базы данных представлены в виде электронных архивов технической документации. К базовым функциям электронного архива относят [87]:

- управление информационными ресурсами и иерархической структурой

электронного архива;

- преобразование в цифровой вид, трансформация и представление исходных документов в различных форматах;
- ускорение записи большого множества типовых и гетерогенных информационных ресурсов в базу данных;
- управление Web-контентом;
- управление задачами и мониторинг статуса их выполнения;
- удобное и эффективное по времени выполнение поисковых запросов к документам.

Для реализации эффективных процедур поиска информации в электронных архивах применяются различные методы предварительной обработки (предобработки) и интеллектуального анализа текстовых документов [3]: стемминг (морфологический поиск), удаление стоп-слов, извлечение существенных для анализа понятий из текста, приведение регистра, N-граммы. Указанные методы используются для сокращения времени поиска информации и устранения незначащих слов [3]. Характеристики текстового документа, которые учитываются при его анализе и обработке, включаются в модель документа [27].

Самой простой моделью текстового документа в задачах информационной поддержки и информационного поиска, является булевская модель. Ограничением данной модели является то, что при ее использовании учитывается лишь факт наличия термина в документе. Подход, который является развитием булевской модели, предполагает, что каждому термину документа соответствует определенный «вес». Это дополнение превращает модель «множество слов (bag of words)» в модель «множество взвешенных слов (пары вес-слово)» [2], [3], [14], [30], [43], [93], [107], [178].

Булевская модель документа

Текстовый документ в булевской модели представляется в виде матрицы, где указывается связь между документами и терминами, содержащимися в данных документах [118], [45], [185], [196]. Словарем является множество

$T = \{t_1, \ldots, t_n\}$, где t_i – термины текстового документа, который является подмножеством словаря и представляется в виде $D \subset T$, где $D \in \{0, 1\}^n$.

Расширенная булевская модель документа

Расширенная булевская модель документа в отличие от простой булевской модели представляет термины не величинами $0, 1$, а весовыми коэффициентами с применением теории нечетких множеств [118], [45], [185], [196]. В этом случае, значение весового коэффициента определяется из интервала [0, 1], таким образом получаем, что $D \in [0, 1]^n$.

Векторная модель документа

Векторная модель формально представляет текстовые документы как матрицу терминов и документов [178]:

$$M = |F| \times |D|,$$

где $F = \{f_1, \ldots, f_k, \ldots, f_z\}$; $D = \{d_1, \ldots, d_i, \ldots, d_n\}$, d_i – вектор в z-мерном пространстве R^z.

Из терминов документа формируется множество F, исключая термины, у которых частота низкая и высокая. Конкретные пороговые значения определяются экспериментально [3].

В работах [3], [178] показано, что для каждого термина f_k в документе d_i вычисляется соответствующий вес $\omega_{k,i} \in [0, 1]$, который обозначает степень важности данного термина для конкретного документа электронного архива.

Матричная модель документа

Формально, матричная модель произвольного текстового документа содержит множества из n документов и m терминов, которые встречаются в одном или нескольких документах [118], [45], [185], [196]. Выделяют три основных типа матрицы сопряженности:

- «документ-документ» D. Отдельный элемент матрицы $d[i, j]$ определяет наличие общих терминов в i-м и j-м документах, или соответствует количеству терминов, которые являются общими для этих документов;

- «документ-термин» C. Отдельное значение $c[i, j]$ определяет наличие термина j в i-м документе или определяет значение веса данного термина в документе;
- «термин-термин» T. Отдельное значение $t[i, j]$ определяет наличие документов, содержащих одновременно i-й и j-й термины, или соответствует количеству таких документов.

Множество формальных моделей текстовых документов вида «множество слов» достаточно широко применяются на практике. Тем не менее, такое представление документа часто приводит к потере важной информации. Поэтому для решения данной проблемы применяются другие формальные модели документов, которые учитывают взаимное расположение слов в тексте [27], [119], [133], [136], [152], [160], [161], [168], [177], [179], [184], [185].

Методы формирование многословных терминов

В основе методов формирования многословных терминов лежит уточнение начального множества терминов за счет введения так называемых «псевдотерминов». Они состоят из нескольких отдельных терминов, которые устойчиво формируют все вместе одно целое понятие. Самым простым способом вычисления многословных терминов является извлечение из документов всех пар (или троек) слов, расположенных рядом друг с другом [27]. Ряд исследователей [119] предлагают такой подход, при котором индексируются только пары слов, которые наиболее часто встречаются в коллекции текстовых документов по определенной тематике.

Разбиение документа на фрагменты

В основе данного подхода лежит следующая идея: текстовый документ разбивается на отдельные фрагменты, которые рассматриваются обособленно относительно друг друга. В этом случае модель документа представляет собой не единственное терминологическое множество, а несколько множеств, которые связанных между собой.

В работе [177] подчеркивается, что разбиение текста на фрагменты основывается на гипотезе о неравномерности распределения терминов в тексте документа. Неравномерно распределены в документе именно значимые термины, т. к. их количество увеличивается в фрагментах, которые по смыслу связаны с данным термином, и сокращается в фрагментах, которые не связаны с рассматриваемым термином.

Исследователи в своей работе [160] дают описание модели документа, которая построена на основе принципа «скользящего окна», предполагающего использование информации о взаимном расположении слов. По мнению авторов, параметр CLC (Computing Lexical Cohesion), который определяется с помощью словаря, позволяет разбить документ на фрагменты, каждый из которых описывает только одну тему. Нахождение веса документа основывается на вычислении суммы весов фрагментов.

Модели текстовых документов, использующие синтаксический анализ

Под синтаксическим анализом понимается реализация процедур автоматического разбора текстовой информации и формирование синтаксических структур, входящих в его состав фраз, с применением лингвистических данных о терминах и их взаимном расположении [168]. После получения результатов анализа происходит формирование дополнительных маркеров, которые определяют синтаксическую роль терминов. Определенным недостатком данного подхода считают вероятностную природу распределения маркеров. Это связано с неопределенностью процессов выполнения синтаксического анализа текста на естественном языке.

Существует большое количество моделей, позволяющих представлять текстовые документы в задачах интеллектуального анализа. Достаточно простая модель «множество слов» часто заменяется более сложными моделями, среди которых наиболее распространенными являются следующие.

1. Модуль документа как представление множества весов терминов.

2. Модель документа как представление множества фрагментов.

Перечисленные разновидности моделей и их различные комбинации в последнее время получили достаточно широкое распространение в современных документальных базах данных.

1.2.2. Методы индексирования документальных баз данных

Для осуществления анализа информационных ресурсов документальных баз данных, входящих в информационное обеспечение САПР, необходимо провести предобработку их содержимого с помощью интеллектуальных методов анализа текстовой информации (Text Mining).

Данному процессу в работе [140] дано такое определение: «обнаружение знаний в тексте – это нетривиальный процесс обнаружения действительно новых, потенциально полезных и понятных шаблонов в неструктурированных текстовых данных». Под неструктурированными данными обычно понимают набор документов, которые представляют из себя логически связанный текст, исключая какие-либо ограничения, связанные с его структурой.

Процесс выполнения интеллектуального анализа текстовых документов состоит из последовательности шагов: [3], [45], [141], [186].

1. Поиск необходимой информации. Выбор документов для последующего анализа.
2. Предобработка текстовых документов. Сохранение документов в такой формат, с которым могут работать методы Text Mining.
3. Извлечение доступной информации. Определение в выбранных документах значимых понятий, которые впоследствии будут подвергаться анализу.
4. Использование методов Text Mining. Нахождение шаблонов и отношений, которые содержатся в тексте.
5. Содержательная интерпретация результатов. Данная интерпретация состоит или в представлении результатов на естественном языке, или в пред-

ставлении их в графическом виде.

Процедура предварительной обработки текста необходима для решения проблемы наличия в документе слов, которые не содержат полезной проектной информации, а также близких по смыслу слов. Количество терминов в результате данной процедуры в документе уменьшается благодаря удалению слов, не несущих важной информации, и приведению близких по смыслу слов к одному формату. Таким образом увеличивается скорость анализа текстовой информации.

Применяют следующие способы удаления неинформативных слов и повышения уровня строгости текстов [3], [45], [141], [186]:

- удаление стоп-слов. Стоп-словами называют такие слова, которые считаются вспомогательными и не несут много информации о содержании текстового документа. Наборы таких слов формируются заранее и при предобработке происходит их удаление из текста. В качестве примеров таких слов могут быть артикли, вспомогательные слова: «так как», «кроме того» и т. п.;
- морфологический разбор (операция стемминга). Данный разбор заключается в нормализации каждого слова. Нормальная форма предполагает исключение склонения слова, множественных форм, особенности устной речи и т. п. Например, такие слова, как «сжатие» и «сжатый» будут преобразованы в нормальную форму слова «сжимать». Все алгоритмы морфологического разбора учитывают особенности конкретного языка и поэтому являются алгоритмами, зависимыми от конкретного языка;
- построение N-грамм. Эта процедура является альтернативной относительно морфологического разбора и удаления стоп-слов. N-грамма есть часть строки, которая состоит из N символов. Например, слово «дата» может быть представлено 3-граммой «_да», «дат», «ата», «та_» или 4-граммой «_дат», «дата», «ата_». Здесь символ подчеркивания заменяет предшествующий или замыкающий слово пробел. N-граммы менее чувствитель-

ны к грамматическим и типографским ошибкам, если сравнивать их с операцией стемминга и стоп-словами. Важным является то, что N-граммы не требуют лингвистического представления слов. Это позволяет данной операции быть более независимой от языка. Тем не менее, N-граммы не способны полностью решить проблему уменьшения количества неинформативных слов;

- процедура приведения регистра. Данная процедура состоит в преобразовании всех символов к верхнему или нижнему регистру. Например, все слова «текст», «Текст», «ТЕКСТ» приводятся к нижнему регистру «текст».

На практике наиболее эффективным является совместное применение выше перечисленных методов.

Каждый метод индексирования текстового информационного ресурса предполагает использование понятия «вес термина», который, в первую очередь, зависит от частоты встречаемости указанного термина в текстовом документе [45], [118], [130], [135].

Достаточно простым способом определения численного значения веса термина является нахождение количества вхождений термина t в документ d. Данная схема вычисления веса называется *частотой термина* (term frequency) и обычно обозначается как tf_{td}, где t – термин документа, а d – анализируемый текстовый документ. Совокупность весов tf документа d можно назвать *представлением документа в числовом виде.*

Значительным недостатком предыдущего подхода определения весов является то, что он не учитывает важность терминов в документе. Например, термин «система» содержится практически в каждом документе коллекции, которая посвящена автоматизированному проектированию. С целью устранения указанного недостатка применяется механизм ослабления влияния термина, который встречается в коллекции очень часто и нет смысла учитывать его в процессе индексирования электронного архива. Поэтому вес термина tf уменьшается на некоторое значение, возрастающее по мере увеличения его встречаемости

в коллекции.

Еще одним вариантом является применение в процедурах анализа текстов показателя, который называется *документная частота* df_t (document frequency). Данный показатель представляет собой то количество документов коллекции, которые содержат термин t.

Показатель документной частоты используется при вычислении *обратной документной частоты* (inverse document frequency):

$$idf_t = \log\left(\frac{N}{df_t}\right),$$

где idf_t – обратная документная частота термина t в коллекции документов D, N – общее количество документов коллекции D, df_t – документная частота термина t в коллекции документов D.

Комбинируя данные показатели: частоту термина (term frequency – tf) и обратную документную частоту (inverse document frequency – idf)) можно рассчитать вес каждого термина в документе. Результирующий метод вычисления веса термина имеет следующий вид:

$$tfidf_{td} = tf_{td} \cdot idf_t$$

и обладает ниже приведенными свойствами.

1. Значение веса термина t достигает максимума тогда, когда термин t достаточно часто встречается в небольшом количестве текстовых документов.
2. Вес термина уменьшается, если термин присутствует в небольшом количестве документов с низкой частотой или встречается во многих документах.
3. Значение веса термина достигает минимума в том случае, если термин встречается во многих (практически во всех) документах.

Следующий шаг в развитии метода вычисления частоты термина (term frequency – tf) является метод, который использует в процессе определения

весового коэффициента для термина значение логарифма частоты:

$$wf_{td} = \begin{cases} 1 + \log(tf_{td}), tf_{td} > 0. \\ 0. \end{cases}$$

На практике часто указанный метод расчета веса термина комбинируется с другими альтернативными методами, например:

$$wfidf_{td} = wf_{td} \cdot idf_t.$$

Также распространенным методом расчета веса термина является нормировка весов tf терминов текстового документа d с помощью максимальной величины tf в этом документе. В этом случае, метод расчета нормированной частоты термина t в документе d будет иметь следующий вид:

$$ntf_{td} = a + (1 - a) \cdot \frac{tf_{td}}{\max(tf_{td})}, a = 0.4, a \in [0, 1],$$

где ntf_{td} – нормированная частота термина t в документе d, a – сглаживающий коэффициент, tf_{td} – частота термина t в документе d.

В процессе анализа документальных баз данных в составе электронных архивов технических документов следует рассматривать разные способы извлечения знаний из текстовых информационных ресурсов, например, поиск значимых понятий по частым наборам слов (например, из государственных стандартов, применяемых в проектной деятельности), идентификация фактов (представляемых в виде событий или отношений) и их характеристик и т. д. С целью идентификации фактов используют наборы шаблонов (паттернов).

Те факты, которые извлекаются из множества ключевых понятий из текста, применяются в процессе решения задач классификации, кластеризации и других. Огромное количество методов извлечения знаний, которые адаптированы для анализа текстовых документов, работают с отдельными понятиями, рассматривая их исключительно как атрибуты данных [3].

1.2.3. Формальные методы извлечения информации из документальных баз данных

Извлечение документов (или их фрагментов) из документальных баз данных электронных архивов выполняется на основе обработки поисковых запросов проектировщика. Такие запросы должны быть эффективно обработаны для того, чтобы проектировщик мог получить те документы, которые удовлетворяют его *информационную потребность*. Соответствующая область деятельности получила название Information Retrieval в западной литературе [45].

«*Информационный поиск (Information Retrieval)* – это комплексная деятельность по сбору, организации, поиску, извлечению и распространению информации при помощи компьютерных технологий» [45], [100].

Рассматривая задачу извлечения информационных ресурсов используются следующие базовые понятия.

- *Коллекция* – это совокупность документов, которые имеют некоторые общие свойства (например, коллекция документов по определенной тематике или коллекция документов, которые имеют единый формат представления).
- *Документ* – это наименьшая структурная единица информации с точки зрения хранения и извлечения из коллекции. Как правило, текстовый документ представляется последовательностью более мелких элементов: абзацы, предложения, слова, которые в некотором смысле тоже являются документами.

Вот некоторые примеры задач, которые можно отнести к области информационного поиска [100]:

- информационный поиск документов по запросу пользователя;
- автоматическая рубрикация документов по заранее заданному рубрикатору;
- автоматическая кластеризация документов – разбиение на кластеры близ-

ких по смыслу документов;

- разработка вопросно-ответных систем – поиск точного ответа на вопрос пользователя, а не целого документа;
- автоматическое составление аннотаций документа и другие.

Цель информационного поиска состоит в удовлетворении потребностей в информации пользователя [100]. Четко выразить, а тем более, формализовать собственную информационную потребность ни один пользователь не в состоянии. Самый простой способ – это использовать естественный язык, который, к сожалению, характеризуется многозначностью, избыточностью и значительно зависит от контекста.

Следующая проблема возникает из-за того, что пользователь способен оценить результат поиска сравнивая его исключительно со своей информационной потребностью, но не в соответствии с поисковым запросом. Однако, поисковая машина осуществляет поиск документов, которые релевантны введенному запросу.

Существуют различные типы релевантности: *тематическая релевантность* и *утилитарная релевантность*. Найденный документ может в полной мере удовлетворять индивидуальной информационной потребности по определенной теме (тематическая), но одновременно быть абсолютно бесполезным с точки зрения выполнения решаемой задачи (утилитарная).

Различные способы обработки запросов имеют различные недостатки.

1. Булев поиск. На запрос поисковая машина может возвратить слишком много документов. Поиск производится методом «проб и ошибок». Для выборки обозримого размера необходимо создавать сложную логическую формулу, что требует от пользователя хорошего знакомства с предметной областью. Релевантность всех документов в выборке одинакова (истина), все атомы логической формулы имеют одинаковый вес (важность).
2. Ранжированный поиск. Показывает более хорошие результаты по сравнению с булевым поиском за счет применения частот терминов. Но проблема

с релевантностью остается.

3. Вероятностная модель. Не учитывает, что релевантность одного документа может зависима от релевантности других документов. Показывает практические результаты на уровне ранжированного поиска.

Результативный поиск информации непосредственно связан с такими понятиями, как *задача пользователя* и *логическое представление документов* [118].

Пользователь системы информационного поиска должен представить свою информационную потребность в виде запроса на языке, который поддерживает система. Обычно предполагается, что такой запрос производится в виде набора слов, которые передают семантику информационной потребности пользователя.

Документы в репозитории исторически обычно представляются в виде множества проиндексированных термов или ключевых слов. Такие ключевые слова могут быть извлечены напрямую из текстов или определены специалистом конкретной предметной области. Вне зависимости от способа их получения (автоматически или вручную) такие ключевые слова образуют логическое представление документа.

Вычислительные способности современных компьютеров позволяют представлять документ посредством полного набора его слов. В этом случае говорят, что система информационного поиска работает с полнотекстовым логическим представлением документов. В случае достаточно большой коллекции даже современные компьютеры должны сокращать множество репрезентативных ключевых слов. Это может достигаться посредством ограничения количества *стоп-слов* (например, это могут быть различные соединительные слова), использованием *стемминга* (когда количество слов сокращается через приведение их к грамматической основе – корню), идентификацией групп имен существительных. Данные операции носят название *текстовых операций* (или трансформаций). Текстовые операции уменьшают сложность представления документа и позволяют перейти от полнотекстового представления к множеству проиндексированных термов. Некоторые промежуточные логические представления

документов показаны на рисунке 1.1.

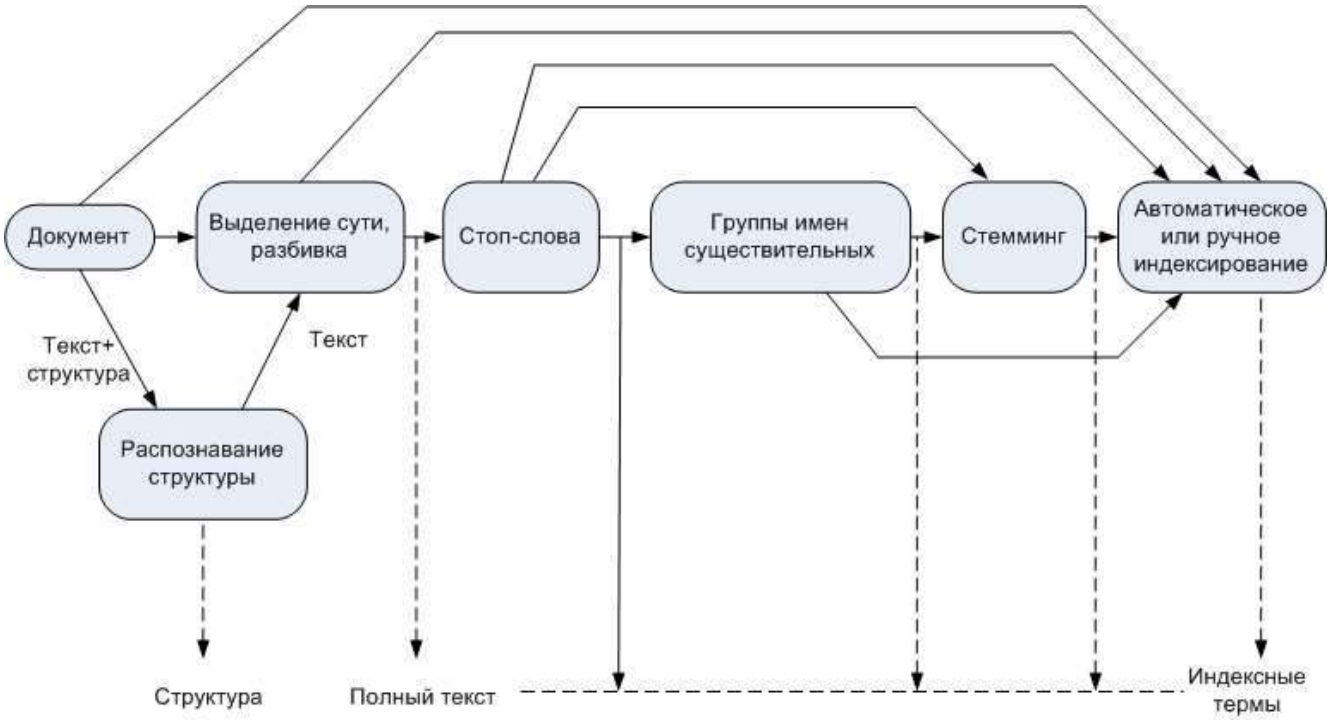


Рис. 1.1. Логическое представление документа: от полного текста к множеству индексных термов

Непосредственно процесс поиска показан на рисунке 1.2. Прежде всего, необходимо определить текстовую базу данных. Обычно это выполняет менеджер базы данных, который специфицирует следующее:

- коллекцию документов;
- операции, выполняемые над текстом;
- текстовую модель (структуру текста и какие элементы могут быть найдены).

Текстовые операции трансформируют исходные документы и генерируют их логическое представление.

Как только логическое представление документов определено, менеджер базы данных (используя модуль менеджера базы данных) создает индекс текста. Индекс является крайне необходимой структурой данных, так как позволяет осуществлять быстрый поиск в огромном массиве информации. Могут при-

меняться различные индексные структуры, но наиболее популярным является индекс типа *инвертированный файл*, как и показано на рисунке 1.2. Затраченные ресурсы (времени и объема памяти) на создание текстовой базы данных и построение индекса многократно окупаются в процессе запросов к поисковой системе.

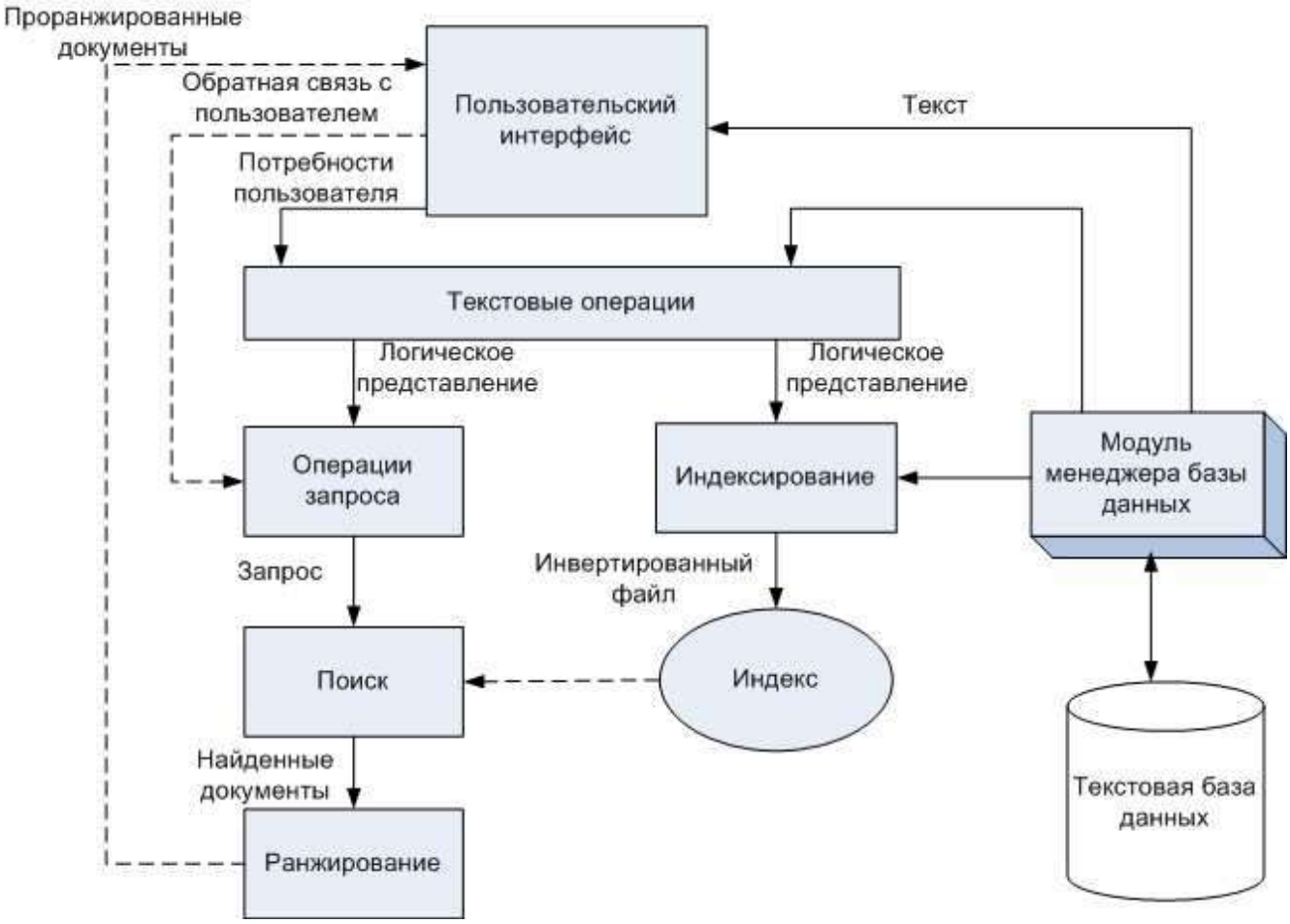


Рис. 1.2. Процесс поиска информации

Процесс поиска может быть инициирован, как только проиндексирована база данных документов. Сначала пользователь формирует свои информационные потребности, которые анализируются и трансформируются с применением текстовых операторов. Затем необходимо выполнить операции запроса над потребностями пользователя, выраженными в виде логического представления. Далее выполняется сам запрос, для ускорения которого используется ранее со-

зданный индекс.

Прежде чем найденные документы будут отправлены пользователю, они ранжируются по степени релевантности запросу. Пользователь затем исследует множество отсортированных найденных документов. С целью улучшения качества поиска пользователь может уточнить свой запрос на основе тех результатов информационного поиска, которые были получены ранее.

1.3. Применение онтологий в информационном обеспечении САПР

1.3.1. Понятие онтологии. Классификация онтологий

Прежде чем говорить о месте онтологии в информационном обеспечении САПР, необходимо дать формальное определение термину *онтология*. Согласно [147], [148], [149], [190], онтология – *формальная спецификация разделяемой концептуальной модели*. Онтология обычно состоит из различных классов сущностей, извлекаемых из предметной области, свойств данных классов, связей между этими классами и логических утверждений, которые построены на основе классов, их свойств и связей [149], [85], [166], [194].

Можно выделить два направления, в рамках которых развивались онтологические исследования. Первое направление предполагает представление онтологии как формальной системы, базирующейся на аксиомах, которые являются математически точными. Альтернативное направление развивалось в рамках когнитивных наук и компьютерной лингвистики. Здесь онтология представляется как совокупность понятий, которые существуют исключительно в сознании человека, и имеется возможность выразить данные понятия на естественном языке. Однако, в этом случае ничего нельзя сказать о точности и непротиворечивости такой системы. В настоящее время данные подходы тесно взаимодействуют.

Онтология предметной области может быть полезна для решения следующих задач [32].

1. Построение систем обучения. Для введения исследователей в новую предметную область полезно иметь в качестве «опорного сигнала» легко воспринимаемую структуру данной области. С использованием онтологии имеется возможность быстро находить разнообразные источники информации.
2. Разработка поисковых систем. В настоящее время имеет место переход от поиска информации по ключевым словам к определению семантически значимых текстовых фрагментов. Такой переход существенно упрощается, если используется онтология соответствующей предметной области.
3. Выполнение научных исследований. Огромное значение имеет унификация терминологии предметной области. Возможность использования онтологии позволяет автоматизировать процесс мониторинга полезных данных и знаний в потоке научно-технической информации.
4. Системный анализ предметной области. Онтологические ресурсы предоставляют частично формализованный и структурированный базис для выполнения системного анализа предметной области.
5. Выполнение интегрирования данных и знаний. В процессе объединения гетерогенных информационных баз онтология предметной области способна устанавливать семантическую эквивалентность тождественных фактов и понятий, которые могут быть сформулированы с использованием различных терминов.

Определим три основных способа классификации онтологий:

- классификация по степени формальности,
- классификация по цели создания,
- классификация по наполнению, содержимому.

Классификация онтологий по степени формальности

Онтологии могут быть использованы для того, чтобы представить кон-

кретную спецификацию имен терминов и значений терминов. В рамках такого понимания онтологии могут быть представлены совершенно по-разному в зависимости от деталей реализации: как каталоги на основе ID, как словари терминов, как тезаурус, как неформальные или формальные таксономии и т. д.

Классификация онтологий по цели создания

В рамках данной классификации выделяют четыре уровня: онтологии представления (концептуализация формального аппарата для представления знаний), онтологии верхнего уровня (предназначенные для повторного использования в различных предметных областях), онтологии предметных областей (предназначенные для повторного использования внутри конкретной предметной области) и прикладные онтологии (ориентация на конкретную задачу без возможности повторного использования).

Классификация онтологий по наполнению, содержимому

Рассматриваемая классификация очень похожа на предыдущую, однако здесь акцент делается на реальном содержимом онтологии. Выделяют три уровня: общие онтологии (включаются такие абстрактные понятия, как сущность, событие, пространство, время и другие), онтологии задач (применяется для конкретной задачи: классификация, составление расписания и т. д.) и предметные онтологии (акцент делается на предметах определенной области знания: вычислительная техника, учебные материалы и другие).

Теперь попробуем проанализировать рассмотренные классификации онтологий с точки зрения их применимости для решения задач формирования информационного обеспечения САПР. На рисунке 1.3 показаны вышеназванные уровни онтологий. Понятно, что не все виды онтологий будут интересны для управления проектной информацией на понятийном уровне.

Первая классификация – *по степени формальности*. На наш взгляд, здесь возможны любые варианты и нужная степень формальности будет определяться требованиями и ограничениями к информационному обеспечению.

Формальные таксономии предполагают точное определение отношения

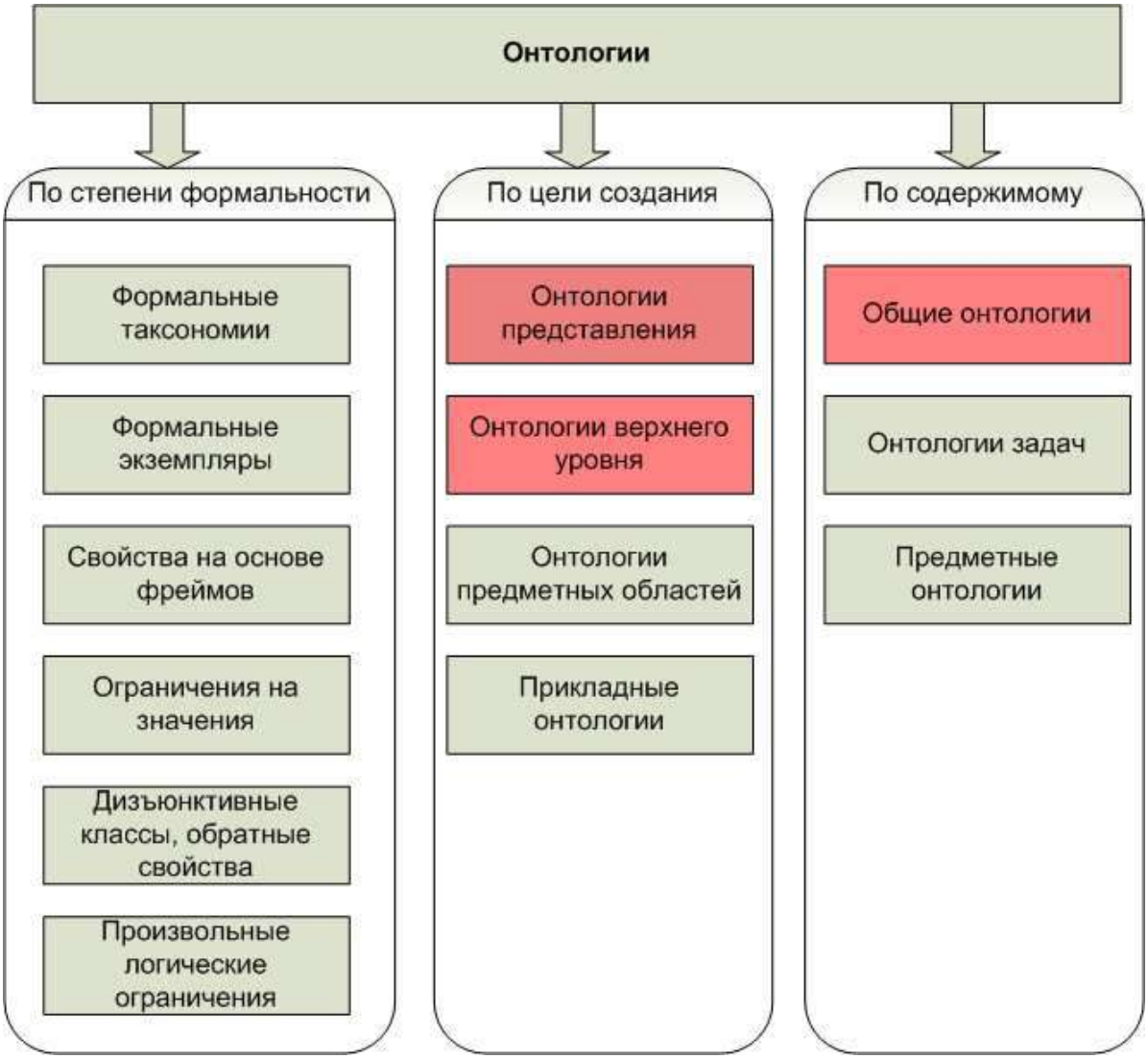


Рис. 1.3. Классификация онтологий

«is_A» (класс-подкласс) при строгом соблюдении транзитивности данного отношения. Организация проектной и технической информации часто удовлетворяет данному ограничению благодаря специфике самой предметной области.

Формальные экземпляры в основе своей имеют формальное отношение «класс-экземпляр». Такие онтологии включают в себя не только иерархию классов, но и содержат на нижнем уровне экземпляры (индивиды). В онтологии информационного обеспечения это может быть выражено следующим образом: помимо таких классов, как «нормативные документы»-«стандарты»-«стандарты на технические задания», добавляется экземпляр «ГОСТ 34.602-89».

Свойства на основе фреймов принимают во внимание тот факт, что клас-

сы (фреймы) могут иметь информацию о свойствах (слотах). Полезность таких свойств заключается в том, что они могут наследоваться от классов верхних уровней нижестоящим классам. В таких областях, как проектирование сложных технических и программных систем, наследование информации играет важную роль и способно значительно упростить процесс моделирования рассматриваемой предметной области. Например, согласно документообороту проектной организации, у каждого документа должен быть автор, дата и номер. В этом случае указанные свойства могут наследоваться от класса «технический документ» всем нижестоящим классам.

Онтологии, которые включают *ограничения на область значения свойств*, обладают большей выразительностью. В данном случае значения свойств берутся из заранее определенного множества (целые числа, символы алфавита) или из подмножества понятий (концептов) онтологии (множество экземпляров некоторого класса, множество классов). Например, для свойства «ИмеетАвтора» класса «ТехническийДокумент» значения можно получать как экземпляры класса «Проектировщик».

Следующий вид онтологии, показанный на рисунке 1.3 как *Дизъюнктивные классы, обратные свойства*, позволяет объявить два или более класса непересекающимися (дизъюнктивными). Это означает, что у таких классов не существует общих экземпляров. Обратные свойства дают возможность осуществлять вывод одного отношения между классами через обратное и наоборот. Например, для свойства «ИмеетАвтора» класса «ТехническийДокумент» обратным свойством будет «Разработал».

Произвольные логические ограничения позволяют определять произвольные логические утверждения о концептах – аксиомы.

Следующая классификация – *по цели создания*. Здесь *онтологии представления* и *онтологии верхнего уровня* для задач формирования информационного обеспечения САПР АС разрабатывать нецелесообразно. Цель *онтологий представления* состоит в описании области представления знаний, в создании

языка спецификаций для других онтологий более низких уровней. Примером может служить определение понятий языка OWL, в основе которых лежит модель RDF/RDFS. *Онтологии верхнего уровня* описывают абстрактные междисциплинарные понятия и их отношения. Такие онтологии могут быть полезными для проектных репозиториев только тогда, когда требуется при описании предметной области выйти за рамки проектных задач. Например, это может быть интересно при высокоуровневой интеграции различных документальных и фактографических информационных баз, функционирующих в различных предметных областях.

Последняя классификация – *по содержимому*. Здесь наблюдается похожая ситуация с предыдущей классификацией. *Общие онтологии* применять для рассматриваемой нами предметной области не представляется возможным и целесообразным в силу ее излишней абстрактности.

Онтология с позиций Семантического Web

Автором концепции Semantic Web (Семантический Веб) является Тим Бернерс-Ли. Он же является одним из создателей всемирной паутины (World Wide Web) и председателем WWW-консорциума (W3C). Данный проект нацелен на организацию такого представления информации в глобальной сети, чтобы внимание разработчиков акцентировалось не только на визуализацию данных (применяя формат HTML), но и на эффективную автоматическую обработку распределенных данных различными программами. Таким образом предлагается трансформация традиционного Web в систему семантического уровня. С точки зрения создателей Semantic Web должен реализовать «понимание» информации программными системами, определение наиболее подходящих по некоторым критериям данных, и только потом – предоставление информации конечным пользователям.

В настоящее время уже существуют поддерживаемые консорциумом W3C стандарты, которые легли в основу данной концепции. Регулярно издается научный журнал, полностью посвященный вопросам теоретических исследований

и практическим результатам Semantic Web (Web Semantics: Science, Services and Agents on the World Wide Web).

В работе [191] исследователи A. Hotho, G. Stumme и B. Berendt подводят итоги многолетних исследований в области Semantic Web и определяют основные направления исследований, которые предполагают быть перспективными. В данной работе центральным понятием является Semantic Web Mining. Оно является пересечением двух связанных понятий: Semantic Web и Web Mining (аналог понятия Data Mining в системах интеллектуальной обработки данных), которое часто переводится на русский язык как интеллектуальный анализ Web-контента.

В данном диссертационном исследования полезно рассмотреть основное содержание Semantic Web [51], [52]. Понятие Web Mining будет рассмотрено кратко с перечислением связанных с ним научных направлений.

Определение и формальное представление смысла ресурсов, которые располагаются во всемирной паутине можно считать основной целью Семантического Веба [187]. Для того, чтобы достичь указанную цель, на практике применяют несколько слоев структур, которые все вместе представляют исследуемое понятие (рисунок 1.4).

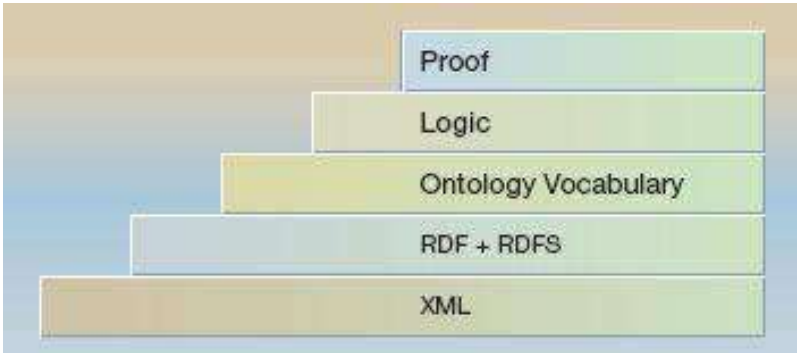


Рис. 1.4. Основные элементы архитектуры Semantic Web

Указанные слои имеют следующие предназначения.

- Структуры данных представляет слой eXtensible Markup Language (XML).
- Смысловое описание данных содержит слой Resource Definition Framework

(RDF).

- Разделяемое соглашение о смысле элементов структур данных закрепляется за слоем онтологий (Ontology).
- Интеллектуальный вывод позволяет реализовывать логический слой (Logic).
- Для обеспечения реализации взаимодействия между программными агентами на нужном доверительном уровне в архитектуру Semantic Web включается слой доказательств (Proof). При этом обеспечивается понимание того, как получается необходимая информация.

Исследователями в работе [187] особо подчеркивается, что эффективность использования технологии Semantic Web значительно возрастает при увеличении объема машинно-интерпретируемого Интернет-контента и количества соответствующих программных систем, которые способны обрабатывать данный контент без участия человека.

Рассмотрим структуру Semantic Web [191] более подробно. Технология XML предоставляет синтаксис для структурированных документов. Никакие семантические ограничения на содержание данных документов не налагаются. Существуют XML-схемы (XML-schema) для формирования структуры документов XML, а также для дополнения XML конкретными типами данных. Фактически, RDF является первым слоем, где информация становится машинно-интерпретируемой [121], [128], [153]. Согласно рекомендациям консорциума W3C, именно RDF является основой для создания метаданных и обеспечивает свойство интероперабельности между программными системами в Web.

Документы RDF включают в себя сущности трех видов: ресурсы, свойства и утверждения. Ресурсами могут выступать Web-страницы, фрагменты или совокупности Web-страниц, а также любые объекты физического мира. Адресация ресурсов в RDF всегда происходит с помощью унифицированного идентификатора ресурсов (URI – Uniform Resource Identifier). В качестве свойств выступают атрибуты, характеристики или отношения, которые описывают ресурсы. Конкретный ресурс вместе со определенным свойством и соответствующим

значением свойства позволяют образовать утверждение в виде RDF-графа. У свойства значением может быть ресурс, литерал или некоторое утверждение. В документах RDF утверждения могут записываться как триплеты, имеющие структуру: объект-атрибут-значение.

Модель данных, лежащая в основе RDF, обычно представляется в виде ориентированного графа. Для описания классов, отношений между классами, отношений между свойствами и ограничений по доменам и диапазонам значений для свойств служит специальный язык RDF-схема.

Слой следующего уровня представляет собой словарь онтологий. Согласно [191], онтология есть «явная формализация разделяемого понимания концептуализации». Различные исследователи по-разному дают определения онтологии, но большинство из них согласны с тем, что онтология должна включать в себя множество концептов (понятий), которые организованы в иерархию, и отношения между концептами. Дополнительно к этому ряд исследователей включают в данное понятие набор логических аксиом.

Создание языка OWL для представления структурированных онтологий стало в последнее время очень важным этапом работ по развитию направления Semantic Web консорциумf W3C. Для этих целей в составе W3C была создана специальная рабочая группа – Web Ontology Working Group. В итоге, 10 февраля 2004 года W3C-консорциум определил для онтологического языка OWL статус «рекомендованной к реализации технологии» [125]. Учитывая особенности OWL, дано следующее определение: «онтология – это совокупность утверждений, задающих отношения между понятиями и определяющих логические правила для рассуждений о них». Программные системы способны «понимать» информации, размещенной на на веб-страницах, просто переходя по гиперссылкам, которые ведут на специализированные онтологические ресурсы. В этом случае, онтология должна включать описания классов из некоторой предметной области, свойств и индивиды указанных классов [143], [169], [181].

Логические выводы имеется возможность получать, используя формаль-

ную семантику OWL на основе онтологий. Происходит извлечение фактов, которые не представлены явно, а следуют из семантики онтологии. Данные выводы могут основываться на анализе большого количества документов, размещенных в глобальной сети. Распределенное представление семантических ресурсов обеспечивается возможностью онтологий быть связанными. Допустимым является непосредственный импорт данных из других онтологических ресурсов. Для определения онтологии, которая может однозначно интерпретироваться и использоваться программными системами, применяются синтаксис и формальная семантика языка OWL.

Следующий — логический — слой позволяет выводить новое знание из тех данных, которые заданы в явном виде, основываясь на наборе логических аксиом в онтологии.

Проверку степени достоверности утверждений, выведенных в Semantic Web, обеспечивает слой доказательств. В последнее время в данном направлении исследования только начинаются.

1.3.2. Роль и место онтологии в информационном обеспечении САПР

Онтология позволяет определить общий словарь для исследователей, у которых имеется необходимость совместно использовать информацию в общей предметной области. Такая онтология включает машинно-интерпретируемые определения основных понятий исследуемой предметной области и систему отношений между ними [120], [172], [170].

Каковы причины возникновения потребности в создании онтологии? Основными причинами являются следующие.

- Имеется необходимость в совместном использовании людьми или программными системами разделяемого понимания структуры информации.
- Необходимо осуществлять повторное использование ранее определенных знаний в некоторой предметной области.

- Необходимо определить явные допущения в некоторой предметной области.
- Необходимо четко разделить знаний в предметной области от оперативных знаний.
- Необходимо выполнять процедуры анализа предметных знаний в ограниченной области.

Необходимость в совместном использовании людьми или программными системами разделяемого понимания структуры информации является наиболее общей причиной для разработки онтологий [170], [134], [192]. Например, различные веб-сайты могут содержат информацию, относящуюся к проектированию сложных технических или программных систем. В процесс проектирования могут быть вовлечены сразу несколько проектных организаций, и у каждой организации имеется свой сайт. В том случае, если для данных веб-сайтов разработана и опубликована единая базовая онтология терминов, которые используются на сайтах, то программные системы имеют возможность извлекать информацию из этих различных сайтов и сохранять ее для дальнейшего анализа. Могут быть построены программные системы, которые способны в автоматическом режиме применять сохраненную информацию для последующих ответов на специализированные запросы пользователей или как входной набор данных для сторонних систем.

Необходимость осуществлять повторное использование ранее определенных знаний в некоторой предметной области является одной из самых важных причин в изучении онтологий. Существует большое количество понятий, которые должны включаться в модели многих различных предметных областей. Например, имеется необходимость сформулировать понятие времени. Данное представление должно включать понятие временных интервалов, моментов времени, относительных мер времени и т. д. Если кто-то способен детально разработать подобную онтологию, то другие исследователи имеют возможность повторно использовать ее в собственных предметных областях. В том случае, если

нужно построить крупную онтологию, то имеется возможность интегрировать несколько существующих онтологий, представляющие отдельные фрагменты большой предметной области. Кроме того, имеется возможность повторно использовать базовую онтологию и расширить ее для описания дополнительной предметной области.

Необходимость определения явных допущений в некоторой предметной области, дает возможность оперативно корректировать такие допущения в том случае, если наши знаний о предметной области изменяются. Включение в код программы предположений об окружающем мире на языке реализации программ приводит к тому, что данные предположения сложно обнаружить и интерпретировать. Кроме того, явное представление знаний в ограниченной предметной области весьма полезны для новых пользователей, которые должны ознакомиться с существующей терминологией предметной области.

Необходимость четкого разделения знаний в предметной области от оперативных знаний — еще один способ универсального использования онтологий. Предположим, что имеется возможность описать задачу конфигурирования продукта из его компонентов в соответствии с требуемой спецификацией. Далее необходимо разработать программу, которая позволяет этой конфигурации быть независимой от продукта и самих компонентов. Для этого можно разработать онтологию, например, компонентов и характеристик вычислительных систем, и применить данный алгоритм для конфигурирования нетиповых вычислительных систем. Также есть возможность использовать тот же алгоритм для конфигурирования другой системы, если разработана онтология, описывающая компоненты данной системы.

Необходимость выполнять процедуры анализа знаний в предметной области возникает тогда, когда имеется декларативная спецификация терминов предметной области. Формальный анализ терминов имеет высокую ценность при попытке повторного использования ранее созданных онтологий и при их расширении на другие предметные области.

Часто целью является не онтология предметной области сама по себе. Построение онтологии напоминает процедуры определения наборов данных и их структур (моделей) для применения другими программными системами. Методы решения задач, контекстно-независимые приложения и программные системы используют в качестве входных данных онтологии и базы знаний интеллектуальных систем, построенные на основе этих онтологий.

Теперь выполним анализ возможностей применения различных видов онтологий для реализации функций организации информационных ресурсов (ИР), включаемых в состав информационного обеспечения САПР [131], [144], [165]. Выделим следующие функции (таблицы 1.1–1.3):

- кластеризация ИР документальных и фактографических баз данных с целью составления дерева категорий,
- обработка поступающих технических документов (выполнение классификации и публикация их в соответствующих категориях),
- обеспечение возможности реализации поисковых запросов по текстам технических документов,
- обеспечение возможности поиска по дереву категорий,
- слежение за тем, чтобы дерево категорий всегда покрывало поступающие ресурсы (чтобы не было ИР, не относящихся ни к одной категории или, наоборот, относящихся ко всем категориям).

Таблица 1.1 содержит оценки применения видов онтологий по степени формальности. Относительно функций кластеризации наиболее подходящими видами онтологий можно назвать формальные таксономии, свойства на основе фреймов и дизъюнктивные классы. Предположительно для функций классификации было бы полезно использовать формальные экземпляры, формальные таксономии и ограничения на значения. Функции реализации поисковых запросов требуют привлечения логико-лингвистических знаний предметной области [89], [90]. Для них предполагается эффективным применение формальных таксономий, дизъюнктивных классов и логических ограничений. Для функций

Таблица 1.1. Возможность применения онтологий в информационном обеспечении САПР АС (классификация онтологий – по степени формальности)

Функция/ Вид онтологии	**Кластеризация**	**Классификация**	**Поисковые запросы**	**Поиск категорий**	**Анализ покрытия**
Формальные таксономии	+	+	+	+	+
Формальные экземпляры		+			
Свойства на основе фреймов	+				
Ограничения на значения		+			
Дизъюнктивные классы	+		+		+
Логические ограничения			+	+	+

поиска категорий наиболее применимыми являются формальные таксономии и логические ограничения. Эффективность функции анализа покрытия может быть повышена путем применения формальных таксономий, дизъюнктивных классов и логических ограничений.

Таблица 1.2 содержит оценки применения видов онтологий по цели создания. Онтологии представления напрямую не способствуют решению анализируемых функций. Однако они предоставляют метаинформацию для построения любой онтологии в принципе. Онтологии верхнего уровня слишком абстрактны для большинства задач информационного обеспечения, за исключением, по-

Таблица 1.2. Возможность применения онтологий в информационном обеспечении САПР АС (классификация онтологий – по цели создания)

Функция/ Вид онтологии	**Кластеризация**	**Классификация**	**Поисковые запросы**	**Поиск категорий**	**Анализ покрытия**
Онтологии представления					
Онтологии верхнего уровня			+		
Онтологии предметных областей	+	+	+	+	+
Прикладные онтологии	+	+	+	+	+

жалуй, формирования и анализа поисковых запросов к электронным архивам технических документов. Знания, выраженные в виде онтологий предметных областей и прикладных онтологий, полезны для решения каждой в отдельности функции.

В таблице 1.3 представлены оценки применения видов онтологий по содержимому. Общие онтологии эффективно могут использоваться лишь для поисковых запросов. Онтологии задач и предметные онтологии могут в одинаковой степени способствовать решению задач организации ИР информационного обеспечения САПР.

Теперь необходимо ответить на вопрос: каким образом потребность в разработке онтологии соотносится с функциями организации ИР?

В таблице 1.4 представлены оценки соответствия потребности в разработке

Таблица 1.3. Возможность применения онтологий в информационном обеспечении САПР АС (классификация онтологий – по содержимому)

Функция/ Вид онтологии	**Кластеризация**	**Классификация**	**Поисковые запросы**	**Поиск категорий**	**Анализ покрытия**
Общие онтологии			+		
Онтологии задач	+	+	+	+	+
Предметные онтологии	+	+	+	+	+

онтологии тем функциям организации ИР, которые были ранее нами определены.

Необходимость общего понимания структуры информации в большей степени существует тогда, когда решается задача интеграции множества источников информации в семантически единый репозиторий. В нашем случае данная потребность хорошо согласуется с функцией формирования информационных запросов в силу того, что релевантный ответ будет получен только в том случае, когда имеет место семантическая согласованность запроса и информации, хранящейся в репозитории.

Фактически, неоднократное использование одних и тех же знаний, представленных в виде онтологии предметной области, как и выявление явных допущений, есть основная потребность для формирования семантического уровня информационного обеспечения САПР АС. Поэтому все функции имеют соответствие для данных потребностей (таблица 1.4).

Явное разделение оперативных знаний от знаний предметной области необходимо для выделения шаблонов или паттернов в предметной области, которые

Таблица 1.4. Соответствие потребности в разработке онтологии функциями организации ИР

Функция/ Потребность в онтологии	**Кластеризация**	**Классификация**	**Поисковые запросы**	**Поиск категорий**	**Анализ покрытия**
Совместное использование			+		
Повторное использование знаний	+	+	+	+	+
Выявление явных допущений	+	+	+	+	+
Разделение знаний	+	+			
Анализ знаний	+				+

можно использовать повторно для различных оперативных задач [4]. Решение проблемы анализа знаний часто связывается с формальным анализом терминов, который, в свою очередь, может быть использован в задачах кластеризации [170]. Кроме того, потребность в анализе знаний может быть интересна для реализации функции анализа покрытия.

1.3.3. Формальные модели онтологий

Основная проблема построения формальной структуры онтологии заключается в том, что в настоящее время не существует единственно правильного решения этой задачи. Одним из таких решений является проект KAON (Karlsruhe Ontology framework) [191].

Основу онтологии составляет ядро, представляющее следующую структу-

ру:

$$O := (C, \leq_C, R, \sigma, \leq_R, A).$$

Она состоит из:

- множеств C и R, которые не пересекаются и включают *идентификаторы понятий* и *идентификаторы отношений*;
- частичного порядка $\leq_C$, определенного на множестве C – *иерархия понятий* или *таксономия*;
- функции $\sigma : R \to C^+$, которую называют *сигнатурой*,
- частичного порядка $\leq_R$, определенного на R – иерархия отношений;
- множества A логических аксиом.

Согласно работам [32] и [13], онтология (O) представляет собой формальное представление структуры исследуемой предметной области, которая состоит из терминов (T), обозначающих индивиды и основные понятия предметной области, отношения (R) между терминами и определения (D) этих понятий и отношений:

$$O = \langle T, R, D \rangle.$$

В работе [7] формализм онтологии обеспечивает гибкое представление понятий предметной области и разнообразных семантических связей между ними. Есть возможность упорядочивания понятий предметной области в иерархию «общее-частное» с поддержкой механизма наследования свойств понятий по данной иерархии. Кроме того, есть возможность задания ограничений на значение свойств объектов предметной области и описания семантики отношений в виде аксиом.

Формально онтология записывается следующим образом:

$$O = \{C, R, T, D, A, F, A_x\},$$

где $C = \{C_1, \ldots, C_n\}$ – конечное непустое множество классов, которое представляет понятия ограниченной предметной области;

$R = \{R_1, \ldots, R_m\}, R_i \subseteq C \times C, R = \{R_T, R_P\} \cup R_A$ – конечное непустое множество отношений, которые задаются на понятиях (классах):

- R_T – бинарное, транзитивное, антисимметричное, нерефлексивное отношение наследования, задающее частичный порядок на множестве понятий C,
- R_P – бинарное, транзитивное отношение включения («часть-целое»),
- R_A – конечное множество отношений, определяющих ассоциацию.

$T = \{t_1, \ldots, t_n\}$ – множество стандартных типов (таких, как «Строка», «Дата», «Число»);

$D = \{d_1, \ldots, d_k\}$ – конечное множество доменов $d_i = \{s_1, \ldots, s_r\}$, где s_i – значение стандартного типа t_j;

$TD = T \cup D$ – обобщенный тип данных, включающий множество стандартных типов и множество доменов;

$A = A_C \cup A_R = \{a_1, \ldots, a_w\}$ – множество атрибутов, которые представляют свойства понятий C и отношений R_A в онтологии;

F – множество, задающее возможные ограничения на значения атрибутов понятий и отношений. Другими словами, указанное множество есть набор предикатов вида $pi(e_1, \ldots, e_m)$, где e_k – есть или имя атрибута ($e_k \in A$), или константа ($e_k \in td_j$, где $td_j \in TD$);

A_x – аксиомы, которые определяют семантику отношений онтологии; в качестве аксиом используются свойства транзитивности и наследования отношений R_T и R_P.

Отдельно рассматриваются так называемые *лингвистические онтологии.* Их главной характеристикой является то, что эти онтологии связаны со значениями языковых выражений (слов, именных групп и т. п.). В работе [96] лингвистическая онтология определяется пятеркой вида $\langle V, W, T, F, D \rangle$,
где V – словарь, включающий минимальные единицы текста – лексемы и лексические конструкции,

W – словарь словосочетаний,

T – тезаурус, который устанавливает классические тезаурусные отношения между элементами словарей V и W,

F – множество упорядоченных наборов схем фактов (порядок отражает последовательность применения схем фактов во время анализа),

D – множество моделей документов, для каждой из которых может быть определен собственный набор схем фактов.

1.3.4. Онтологические модели интеллектуального анализа документальных баз данных

Модели, ориентированные на онтологию

Традиционные модели в системах информационного поиска основываются на представлении документа в виде набора проиндексированных термов. При этом частота встречаемости термов в документе учитывается при вычислении степени релевантности документа пользовательскому запросу.

Наряду с использованием онтологий представляется целесообразным использовать для моделирования знаний пользователя о предметной области поиска частный случай онтологии – тезаурус, построение которого относительно проще. До недавнего времени термины «онтология» и «тезаурус» использовались как синонимы, однако теперь тезаурус чаще применяют для описания лексики в проекции на семантику, а онтологию – для формализованного представления семантики и прагматики, учитывая язык представления онтологии.

В работе [180] онтология применяется не напрямую в процессе информационного поиска, а выступает в качестве основы нового подхода к представлению документов в репозитории. Отмечается, что большинство пользовательских запросов являются *лексически направленными* без какого-либо семантического содержания.

Идея применения онтологии для представления документов основывается на применении отношений между понятиями предметной области. Лингвистических отношений между понятиями известно большое количество, но исполь-

зование их всех в автоматическом режиме – это очень сложная задача. Поэтому авторами работы [180] предлагается их аппроксимация до уровня двух видов: семантические и физические отношения.

Идея анализа семантических отношений заключается в использовании понятия «дистанция» между словами. В документе отношение между двумя словами, расположенными в границах одного предложения, должно отличаться от отношения между словами в рамках двух различных абзацев. Дополнительно, если основная идея повторяется сразу в нескольких абзацах, то она должна интерпретироваться как более важная, чем если бы она была идентифицирована только в одном абзаце. Значение семантического коэффициента отношения между двумя словами-термами определяется следующим образом:

$$S = \frac{\sum_{occur(t_i,t_j)} \frac{1}{exp(sentence \cdot (parag+1))}}{num(occur(t_i,t_j))} \cdot \frac{num(parag - coocur(t_i,t_j))}{num(totalparag)}, \qquad (1.1)$$

где t_i, t_j – i-й и j-й термы соответственно; $sentence$ – расстояние, выраженное в количестве предложений между термами; $parag$ – расстояние, выраженное в количестве абзацев между термами; $num(occur(t_i,t_j)$ – количество совпадений t_i и t_j; $num(parag - coocur(t_i,t_j))$ – количество абзацев, где существует совместная встречаемость термов t_i и t_j; $num(totalparag)$ – число абзацев в документе.

Физические отношения представляют понятия, которые тесно связаны с термами, являющимися частью онтологии. Возможны следующие отношения: наследование, агрегация, синонимия и другие. Можно определить два основных источника для получения таких отношений: тезаурус или синтаксический анализ проиндексированных документов. Значение, соответствующее весу физического отношения, всегда равно единице.

В качестве самостоятельного направления развиваются подходы к интеллектуальным методам формирования поисковых запросов в рамках концепции Semantic Web [132], [195]. Характерной чертой данных подходов является активное использование стандартов, имеющих отношение к Semantic Web. Так,

в работе [195] предлагается модель применения базы знаний, основанной на онтологии с целью улучшения качества поисковых запросов в репозиториях с большим количеством документов. Модель поиска основывается на адаптации классической модели векторного пространства. Семантический поиск применяется вместе с поиском на основе ключевых слов для допущения неполноты базы знаний.

Согласно взгляду авторов [195] на процесс семантического информационного поиска, база знаний создается и соединяется с информационными ресурсами (базой документов) с использованием одной или нескольких предметных онтологий, которые описывают понятия, появляющиеся в текстовых документах.

Непосредственно процесс формирования поисковых запросов представлен на рисунке 1.5.

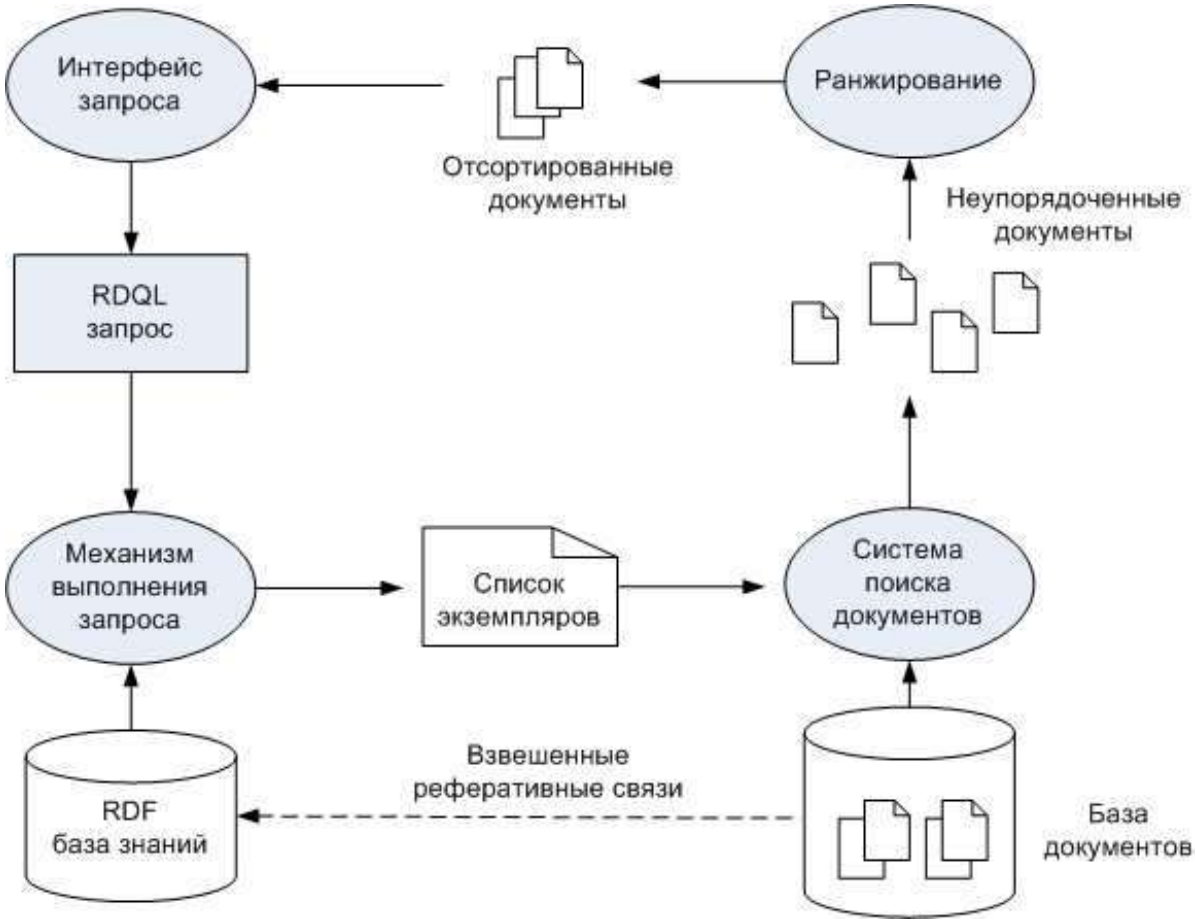


Рис. 1.5. Представление поиска документов, основанного на онтологии

Система в качестве входа принимает формальный запрос к тройкам RDF

(RDQL). Данный запрос выполняется к базе знаний и возвращает список кортежей, удовлетворяющих запросу. Наконец, документы, снабженные примечаниями в виде соответствующих кортежей, находятся, сортируются и предоставляются пользователю.

Запрос RDQL может содержать условия, значимые для экземпляров предметной онтологии и свойств документов (таких как автор, дата, издатель и другие). Задача системы поиска документов состоит в нахождении всех документов, которые соответствуют экземплярам кортежей из базы знаний. Если кортежи соответствуют только понятиям предметной области, то система поиска будет анализировать все примечания экземпляров по ссылкам. Если кортежи содержат экземпляры классов документов (по причине того, что в запрос были включены прямые условия на документы), выполняется похожая процедура, но ограничивается документами в результирующем множестве (вместо полного репозитория).

Как только список документов сформирован, поисковая система вычисляет значение семантической схожести между запросом и каждым документом следующим образом. Пусть O представляет множество всех классов и экземпляров онтологии, D представляет множество всех документов в пространстве поиска. Пусть q – RDQL-запрос, а V_q – множество переменных в выражении SELECT q. Пусть $T_q \subset O^{|V_q|}$ – список кортежей в результирующем множестве запроса, где для каждого кортежа $t \in T_q$ и $v \in V_q$ $t_v \in O$.

Каждый документ в пространстве поиска представляется как вектор документа $d \in D$, где d_x – вес понятия x в документе для каждого $x \in O$, если ссылка на понятие существует, иначе – 0. Определяется расширенный вектор запроса q следующим образом: $q_x = |\{v \in V_q | \exists t \in T_q, t_v = x\}|$, т. е. координата вектора запроса, соответствующая x есть число переменных в RDQL-запросе, для которого существует кортеж t, где переменная означивается x. Если x не появляется ни в одном кортеже, $q_x = 0$. Степень схожести между документом

d и запросом q вычисляется по следующей формуле:

$$sim(d, q) = \frac{d \cdot q}{|d| \cdot |q|}.$$

Модели поиска документов в условиях неполноты

Неполнота проектной информации является принципиальной, если речь идет о проектировании сложных систем [1]. В данном разделе приводятся подходы к решению задач, относящихся к формированияю поисковых запросов в условиях неполных данных [150].

Авторами [15] предлагается формальная модель для представления поиска, основанного на применении нечеткой логики. В работе приводятся основные свойства модели, к которым относят: критерий допустимости информационного графа и критерий полноты базового множества. Дается описание перехода от четкой постановки задачи поиска к нечеткой. Приводятся условия, при которых решение четкой задачи информационного поиска будет соответствовать решению нечеткой задачи.

Задача информационного поиска в [15] определяется следующим образом. Пусть X – определяет допустимое множество запросов; Y – есть допустимое множество записей (объектов поиска); ρ – определяет бинарное отношение на $X \times Y$, которое называется отношением поиска; тройка $S = \langle X, Y, \rho \rangle$ определяет тип; в свою очередь, тройка $I = \langle X, V, \rho \rangle$, где V – некоторое конечное подмножество множества Y, есть задача информационного поиска (ЗИП) типа S и будем полагать, что ЗИП $I = \langle X, V, \rho \rangle$ содержательно состоит в перечислении для произвольного запроса $x \in X$ всех записей $y \in V$, таких что выполняется: $x \rho y$.

По аналогии вводится постановка задачи нечеткого поиска [15]. Предположим, что аналогичным образом X – есть допустимое множество запросов, Y – определяет допустимое множество записей. Пусть задано отображение $\eta(x, y) : X \times Y \to [0, 1]$, которое называется отношение нечеткого поиска. Тройка $S = \langle X, Y, \eta \rangle$ определяет тип нечеткого поиска; тройку $I = \langle X, V, \eta \rangle$, где V – неко-

торое конечное подмножество множества Y, будем называть задачей нечеткого поиска (ЗНП) типа S и будем считать, что ЗНП $I = \langle X, V, \eta \rangle$ содержательно состоит в том, чтобы для произвольного числа $c \in [0, 1]$ и произвольного запроса $x \in X$ перечислить все те и только те записи $y \in V$, такие что $\eta(x, y) \geq c$.

В работах [154] и [171] предлагается строить модели информационного поиска на основе знаний. Более конкретно, в работе [154] представлена модель, центральным компонентом которой является нечеткая концептуальная сеть. Работа [171] посвящена разработке и исследованию модели нечеткого информационного поиска, в основе которой лежит понятие нечеткого тезауруса. Рассмотрим суть данных моделей.

В модели Хорнга (Horng) [154] база знаний представляет собой нечеткую концептуальную сеть, которая определяет отношения и соответствующие степени релевантности между концептами (понятиями). Данная сеть описывает четыре типа нечетких отношений: нечеткое обобщение (G), нечеткая специализация (S), нечеткая положительная ассоциация (P), нечеткая отрицательная ассоциация (N). Нечеткая концептуальная сеть представляется матрицей U_r следующего вида:

$$U_r = \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1y} \\ u_{21} & u_{22} & \dots & u_{2y} \\ \vdots & & & \\ u_{y1} & u_{y2} & \dots & u_{yy} \end{pmatrix},$$

где U_r – матрица относительной релевантности, u_{ij} – степень релевантности между понятиями c_i и c_j, основанная на отношении $r : R \in \{P, N, G, S\}, u_{ij} \in [0, 1]$. Такая модель позволяет находить документы, которые не напрямую связаны с пользовательским запросом. Кроме того, в работе [154] приводится описание *алгоритма расширения вектора запроса*, который добавляет новые понятия нечеткой концептуальной сети к пользовательскому запросу. В качестве результата отмечается большее количество найденных релевантных документов.

В модели, предлагаемой исследователем Ogawa [171], предполагается нечеткая система информационного поиска документов, использующая матрицу связей ключевых слов для представления степени схожести между ключевыми словами. Элементы матрицы задаются следующим образом:

$$W_{ij} = \begin{cases} \frac{N_{ij}}{N_i + N_j - N_{ij}}, & i \neq j \\ 1, & i = j \end{cases}$$

где W_{ij} – значение отношения между j-м и i-м ключевыми словами; N_{ij} – количество документов, содержащих как i-е, так и j-е ключевое слово; N_i – количество документов, которые содержат i-е ключевое слово и N_j – количество документов, которые содержат j-е ключевое слово. После того, как матрица связей ключевых слов создана, генерируются нечеткие индексы для каждого терма из коллекции документов. Значения нечетких индексов определяют нечеткие отношения между ключевыми словами и документами. Для определения релевантных запросу документов должны быть выполнены следующие три шага.

1. Генерация нечетких индексов.
2. Вычисление степени релевантности для каждого подзапроса.
3. Вычисление обобщенной степени релевантности.

Найденные документы предоставляются пользователю в порядке уменьшения степени релевантности.

Авторы работы [176] развивают идеи, предложенные в [154] и [171], и отмечают, что типичными примерами успешного использования теории нечетких множеств в системах информационного поиска являются *механизмы нечеткого индексирования*, *методы нечеткой кластеризации*, *нечеткие системы интеллектуального анализа данных* и *нечеткие распределенные системы информационного поиска*.

Подход к нечеткому моделированию систем информационного поиска, представленный в работе [176], основывается на нечеткой реляционной онтологической модели. Суть ее заключается в том, что онтология предметной области

представляет собой двухслойную структуру. Первый слой содержит имена понятий, тогда как второй слой содержит ключевые слова, связанные с именами понятий предметной области первого слоя. Имена понятий и ключевые слова выбираются из содержимого документов, представленных в коллекции. Каждое отдельное понятие c_i связано с ключевым словом k_j степенью нечеткой ассоциации $r_{ij} \in [0, 1]$.

Матрица релевантности R определяет нечеткую реляционную онтологию следующим образом:

$$R = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \vdots & & & \\ r_{n1} & r_{n2} & \dots & r_{nm} \end{pmatrix},$$

где $1 \leq i \leq n$, n – количество ключевых слов во втором слое; $1 \leq j \leq m$, m – количество понятий в первом слое и $r_{ij} \in [0, 1]$ – значение степени релевантности между c_i и k_j.

Особенность такой модели заключается в том, что база знаний представлена нечеткой реляционной онтологией, а нечеткое отношение определяется в пространстве слов и понятий.

1.3.5. Интеграция данных на основе онтологий. Понятие связанных данных (Linked Data)

В качестве формальной основы фактографических информационных баз часто в настоящее время применяют реляционные модели. Несмотря на то, что реляционные базы данных разрабатываются с использованием общепринятых стандартов, их интеграция не является тривиальной задачей из-за возможной сложности их структуры. Так, две базы данных, содержащие записи о сущностях двух схожих предметных областей, могут быть спроектированы на разных уровнях абстракции, таблицы могут иметь разную степень нормализации, домены и ограничения целостности могут иметь различные формулировки. Помимо

этого возникает проблема семантической гетерогенности, связанная с наличием разных понятий предметной области и их интерпретаций [129].

В сентябре 2012 года консорциум W3C опубликовал документы, в которых представлены официальные рекомендации для спецификаций языков отображения DM (Direct Mapping) и R2RML (RDB to RDF Mapping Language). Согласно этим спецификациям, язык DM представляет собой, как можно судить из названия, автоматическое прямое отображение реляционных таблиц. Структура результирующего RDF-графа при прямом отображении базы данных напрямую зависит от структуры базы (например, названия таблиц преобразуются в имена классов, а названия столбцов – в названия свойств). При этом ни структура, ни целевой словарь не могут быть изменены [124]. Само по себе прямое отображение приемлемо только в самых простых случаях.

Фактографические базы данных в составе информационного обеспечения современных САПР могут иметь очень сложную структуру, так как она напрямую зависит от уровня компетентности ее разработчика и от предметной области. Так, например, в базах может содержаться не одна сотня таблиц, в которых хранятся сущности реальной предметной области, в то время как другие таблицы носят лишь вспомогательный характер. Для построения отображения для таких случаев была разработана спецификация R2RML.

В отличие от прямого отображения отображение с использованием спецификации R2RML является полностью настраиваемым. Помимо этого, оно позволяет работать с SQL-представлениями таблиц, посредством встраивания SQL-запросов прямо в файл с отображением [137].

Языки DM и R2RML не исключают, а дополняют друг друга. Сначала пользователь с помощью DM может увидеть, что будет представлять собой RDF-граф и получить предварительную версию файла отображения. Затем, используя R2RML, пользователь может произвести тщательную настройку отображения для дальнейшей работы.

В работе исследователей из Цюрихского университета [164] предпринята

попытка сравнения между собой существующих языков отображения реляционных данных в RDF-множества, посредством анализа индивидуальных функций и особенностей, которые присутствуют в этих языках. В результате исследуемые языки были разделены на четыре группы: языки прямого отображения, языки общего назначения (чтение), языки общего назначения (чтение и запись), языки специального назначения.

В работе [36] описывается подход доступа к реляционным данным, основанного на применении онтологии при помощи использования дескрипционной логики. В данном случае происходит преобразование конъюнктивного запроса в терминах онтологии в запрос над реляционной базой данных.

Учитывая современный уровень развития документальных и фактографических информационных баз, данные по-прежнему фактически «заперты» в корпоративных информационных системах. Для их представления используются различные, обычно не согласованные между собой словари и схемы. В результате имеет место фрагментированная информационная среда, в которой задачи обнаружения, повторного использования и осмысления данных из различных источников становятся трудновыполнимыми.

Современная концепция *связанных данных* направлена на решение указанных задач [201]. Связанные данные (Linked Data) – это набор основных принципов проектирования средств совместного использования машиночитаемых данных в Web органами госуправления, коммерческими структурами и отдельными гражданами.

Тим Бернерс-Ли сформулировал четыре принципа проектирования, применительно к связанным данным [201].

1. Применение унифицированных идентификаторов ресурсов (URI) для однозначной идентификации объектов (элементов данных).
2. Использование стандартных HTTP URL-адресов данных URI для обеспечения возможности осуществления информационного поиска.
3. Необходимость добавления метаданных в соответствии с такими откры-

тыми стандартами, как RDF.

4. Формирование ссылок на соответствующие URI для обеспечения возможности обнаружения информационных объектов.

В настоящее время фактографические информационные базы формируются на основе реляционного подхода, а информационный обмен возможен только при четко определенной структуре, обычно с использованием схем XML. Совместное использование данных в соответствии с некоторой схемой (XML) служило технической парадигмой в последние десятилетия. По мере развития таких схем необходимо соответствующим образом адаптировать использующие их информационные системы (выступающие в роли источников или потребителей данных). В долговременной перспективе поддержка таких схем требует значительных усилий. Особенно это касается информационного обеспечения САПР современных АС, поскольку требования к таким системам и внешняя среда, в которой они функционируют, изменяются достаточно быстро.

В этом заключается главная причина появления новой парадигмы обмена данными, основанной на схеме описания ресурсов RDF. Согласно консорциуму W3C, *«схема RDF обладает свойствами, облегчающими слияние данных, даже если лежащие в основе схемы различны; в частности, она поддерживает эволюцию схем с течением времени, не требуя модификации у всех потребителей данных»* [202].

Архитектура Связанных Данных для решения задачи интеграции данных представлена на рисунке 1.6.

Связанные данные используются для решения задачи интеграции данных, которые могут быть размещены в гетерогенных источниках, имеющих как разные форматы, так и неструктурированных (основные концепции архитектуры изложены в работе [200]).

На первом этапе происходит определение постоянных URI, которые назначаются данным и дается описание этих данных в виде RDF-графа с использованием общепринятых метаданных. Для структурированных данных (реляцион-

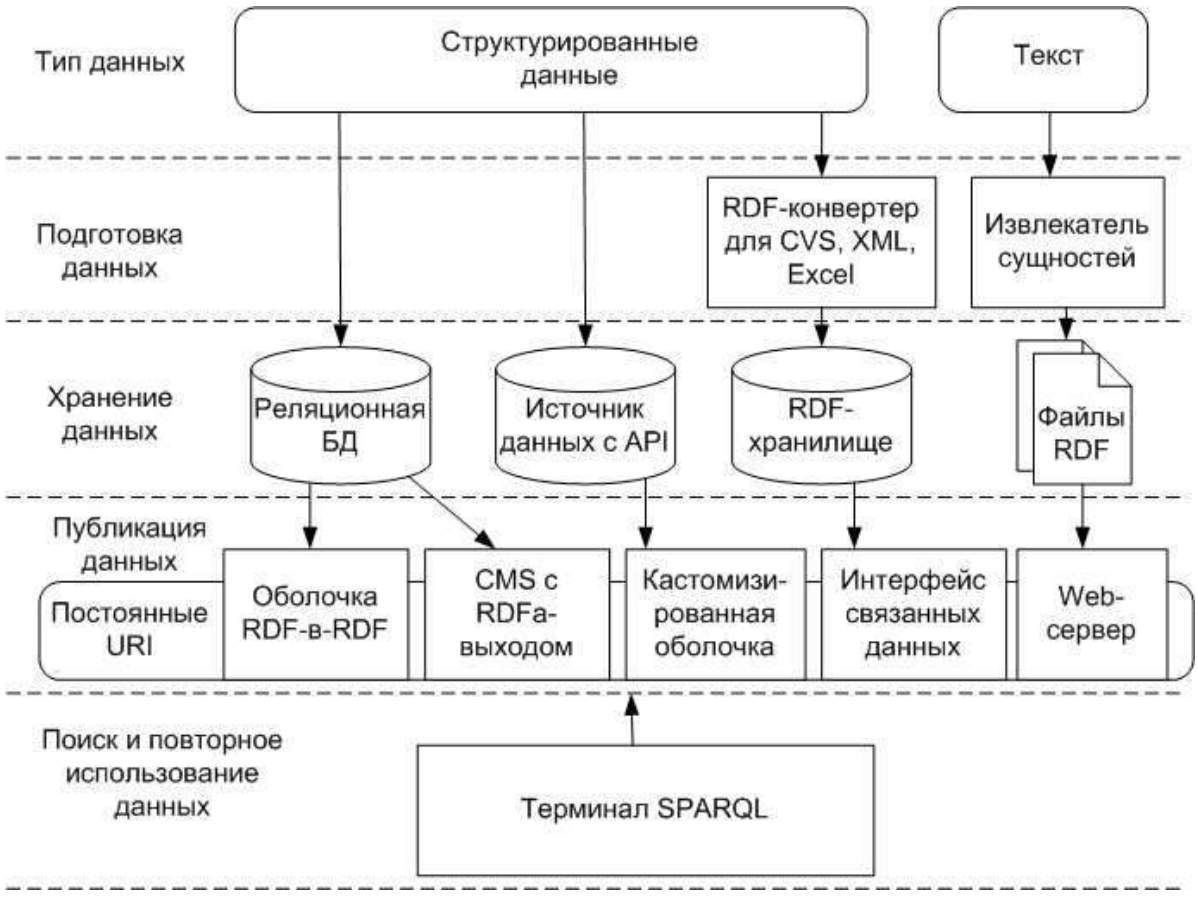


Рис. 1.6. Архитектура связанных данных

ные базы данных, файлы CSV, Excel или XML) происходит адаптация RDF-оболочек, модулей экспорта и API для последующего преобразования этих данных в RDF-графы с последующей публикацией в сети или в любых других источниках, которые допускают обмен этим типом данных. Для слабоструктурированных данных (информационные ресурсы документальных баз данных) в случае их публикации в виде связанных данных необходимо использовать средства извлечения сущностей и углубленного анализа текстов с целью обнаружения информационных объектов.

В том случае, когда данные представлены в виде RDF-графов, их необходимо связать с другими данными, которые поступают из надежных источников. Это необходимо выполнить для того, чтобы сформировать их контекстное окружение и расширить смысловое содержание. После того, как связанные данные опубликованы, любая система, которая способна обрабатывать семантические

запросы (например, при помощи языка запросов SPARQL), будет иметь возможность извлекать и повторно использовать релевантную информацию.

1.4. Формализация неполноты проектной информации

1.4.1. Виды неполноты информации

Известны различные аспекты неполноты информации, имеющей место в проектной деятельности. Часто различают три основных аспекта: неточность, нечеткость и неопределенность [1], [35], [40], [106], [115], [138]. *Неточные данные* определяются в интервальной форме $D+E$, т. е. в виде интервала $[D-E, D+E]$. Термин *неопределенность* имеет многозначную интерпретацию в российской научной литературе. В искусственном интеллекте данный термин употребляется часто в контексте вычисления степени истинности утверждения. Понятие *нечеткость данных* связано с определением функции принадлежности элементов множества. В этом случае, семантика функции принадлежности задается как распределение возможностей [29].

В своей работе Д. И. Шапиро [113] выделяет такие категории неопределенности, как:

- незнание;
- субъективная вероятность;
- неточность (ошибка наблюдения);
- неполнота;
- неопределенность (недостаточность информации);
- расплывчатость.

При проведении сравнительного анализа подходов к формализации неполноты информации имеет смысл исследовать их особенности, опираясь на ряд признаков, к которым относятся: способ описания исходных данных, понятийная основа, человеческий фактор, количество объектов, способы формирования операций, совокупность прикладных задач, аксиомы [115], [193], [197]. В рабо-

тах [44], [198] представлены результаты исследования по применению нечеткой математики при анализе сложных систем.

1.4.2. Современный подход Заде к формализации неполноты

Основная идея работ Л. Заде в настоящее время заключается в развитии гранулярных вычислений [199]. В них отмечается необходимость определения такого уровня точности описания объектов предметной области, который был бы согласован с требованиями решаемой задачи. В развитие этого положения, Л. Заде разработал *Theory of Precisiation of Meaning*, теорию уточнения значений. Базовые положения данной теории представляются следующим образом.

1. Концепция точности (неточности) способов выражения и содержания понятий. Каждое понятие включает содержание (*value*), которое может определяться точно или не точно (*v-precise, v-imprecise*). Каждое понятие имеет свою форму значения (meaning), которая также может выражаться как точно, так и не точно (*m-precise, m-imprecise*). В своих работах Л. Заде атрибут *m-precise* использует в качестве аналога термина «математически определенный». Так, например, если задана следующая пропозиция:

   $p:\ x\ is\ X$,

   где X – случайная переменная, имеющая гауссово распределение, математическое ожидание m и дисперсию σ , m и σ – действительные числа, в этом случае говорят, что p наделен атрибутами *v-imprecise* и *m-precise*. Указанную концепцию можно выразить достаточно кратко: *теория нечетких систем – это точная наука о неточности.*
2. Грануляция есть ожидаемое следствие *v-imprecise*. Для представления неточного значения вместо синглетона (единичного значения) необходимо применять или интервал или распределение некоторой функции множества, другими словами, гранулу сложной структуры. Вообще говоря, можно утверждать об экстенсиональном и интенсиональном (*attribute-based*) представлении значений. Поэтому возможность реализовывать операции

над гранулами приводит к понятию *гранулярных вычислений.*

3. При определении гранулы применяется *принцип обобщенных ограничений* (generalized constraint). Обобщенное ограничение может быть задано, как $X\ isr\ R$, где X – ограниченная переменная, r – тип модальности, R – ограничивающее (нечеткое) отношение.

   Перечислим возможные типы ограниченных переменных:

   - X – n-арная переменная, $X = (X_1, \ldots, X_n)$,
   - X – обусловлена другой переменной X/Y,
   - X – функция другой переменной: $X = f(Y)$,
   - X – пропозиция,
   - X имеет структуру, например, X= *Location(Residence(Carol))*
   - X обобщенное ограничение $X:\ Y\ isr\ R$.

   X групповая переменная $G[A]:\ (Name_1, \ldots, Name_n)$, с каждым элементом группы $Name_i$, $i = 1, \ldots, n$, ассоциируется атрибут A_i.

   Перечислим возможные типы обобщенных ограничений $X\ isr\ R$:

   $r :=$ ограничение эквивалентности: $X = R$ аббревиатура $X\ is\ R$,

   $r :\leq$ ограничения неэквивалентности: $X \leq R$,

   $r :\subset$ ограничения вложенности: $X \subset R$,

   $r : blank$ возможностное ограничение; $X\ is\ R$; R распределение возможности на X,

   $r : v$ истинностное ограничение; $X\ isv\ R$; R – распределение истины на X,

   $r : p$ вероятностное ограничение; $X\ isp\ R$; R – распределение вероятностей на X,

   $r : bm$ бимодальное ограничение; X – случайная переменная; R – бимодальное распределение,

   $r : rs$ ограничение случайных множеств; $X\ isrs\ R$; R – множество-значное распределение вероятностей на X,

   $r : fg$ ограничения нечеткого графа; $X\ isfg\ R$; X – функция и R – ее

нечеткий граф,

$r : u$ ограничения «традиции» (привычной практики usually); $X\ isu\ R$ «обычно означает» ($X\ is\ R$),

$r : g$ групповое ограничение; $X\ isg\ R$ означает, что R ограничивает все значения атрибутов гранулы.

4. С целью обеспечения возможности символической записи гранулярных пропозиций был предложен язык гранулярных вычислений *Generalized Constraint Language (GCL)*. Операцию дедукции следует рассматривать как распространение ограничений (*deduction = generalized constraint propagation*). Символическая запись дедуктивных рассуждений ведется с помощью языка протоформ *ProtoForm Language (PFL)*. $PF(p)$: – абстрактная форма, глубинная структура p. На заданном уровне абстракции, объекты p и q PF-эквивалентны, если $PF(p) = PF(q)$. Например,

   p : Большинство шведов высокие $Count(A)\ is\ Q$,

   q : Некоторые профессора богаты $Count(A)\ is\ Q$.

5. Конструктивный характер *Theory of Precisiation of Meaning* (TPM) придает развитый символьный аппарат дедукции протоформ и вычислительный аппарат гранулярных вычислений (таблица 1.5).

Л. Заде предлагает по-новому взглянуть на понятие неопределенности с точки зрения теории обобщенных ограничений:

- Свойство неопределенности является основным атрибут информации. Оглядываясь, в первую очередь, на работы К. Шеннона изучают статистическую природу неопределенности. Однако, теория обобщенных ограничений неопределенности (*Generalized Theory of Uncertainty, GTU*) отличается по своей сути.
- Тезис о статистической природе неопределенности заменяется в GTU тезисом о том, что информация есть ни что иное, как обобщенные ограничения. В свою очередь, статистическое представление информации представляет собой всего лишь частный случай.

Таблица 1.5. Иллюстрация конструктивного характера TPM

Правило дедукции	**Символьное правило (на языке прото-форм)**	**Вычислительное гранулярное правило**
Композиционное правило вывода	$X\ is\ A$ $(X, Y)\ is\ B$ $Y\ is\ A \circ B$	$\mu_B(v) = sup(\mu_A(u) \wedge \mu_B(u, v))$
Принцип расширения Заде	$X\ is\ A$ $Y = f(X)$ $Y = f(A)$	$\mu_y(v) = sup_u(\mu_A(u))$ $v = f(u)$
Правило дедукции на основе принципа расширения Заде	$X\ is\ A$ $f(X)\ is\ B$	$\mu_B(v) = sup_u(\mu_A(u))$ $v = f(u)$
Обобщение на основе принципа расширения	$f(X)\ is\ A$ $g(X)\ is\ B$	$\mu_B(v) = sup_u(\mu_A(f(u)))$ $v = g(u)$

- Бивалентность наличия свойства (в том числе истинности) заменяется степенью проявления свойства.
- Главная цель GTU заключается в способности представления информации, используя естественный язык (NL-capability). Применяя символическую форму, можно записать следующий образом: $I(X) = GC(X)$, где X – переменная, определенная на U, $I(X)$ – информация о X, GC – обобщенные ограничения.

Способность представления информации на естественном языке в любой теории неопределенности Л. Заде считает критерием ее приемлемости при по-

строении гуманистических систем, которые основаны на знаниях.

1.4.3. Формальное представление приближенных множеств Павлака

Теория приближенных множеств (Rough Sets Theory) [175] была представлена З. Павлаком в качестве нового математического аппарата, который предназначался для работы с неопределенностью и неточностью.

Основная идея приближенных множеств базируется на предположении, что с каждым суждением мы ассоциируем некоторую информацию (данные, знания). Объекты, характеризуемые одной информацией, являются неразличимыми с точки зрения имеющейся о них информации. Полученные таким образом неразличимые отношения определяют математическую основу теории приближенных множеств.

Тот факт, что всякий объект может «проявляться» только через ту информацию, которая о нем доступна, приводит к представлению, что все знания имеют гранулированную структуру. По причине гранулирования знаний ряд объектов, которые могут быть интересны, не могут быть различимы и выступают как совершенно одинаковые. С другой стороны, неопределенное понятие (в противоположность определенному) не может быть охарактеризовано в терминах элементов, которые его образуют. Поэтому в предлагаемом подходе происходит замена каждого неопределенного понятия двумя определенными: нижней и верхней аппроксимацией неопределенного понятия. Нижняя аппроксимация включает все объекты, которые *точно* соответствуют понятию, а верхняя – включает все объекты, которые *возможно* соответствуют понятию. Приближенным множеством (Rough Set) называют множество, которое задано через нижнюю и верхнюю аппроксимации. Очевидно, что разница между верхней и нижней аппроксимациями определяет границу области неопределенного понятия.

Отношение неразличимости может быть определено математически, однако для большей наглядности оно рассматривается на примере табличного пред-

ставления набора данных – *информационной системы* [175]. Информационная система в контексте теории приближенных множеств – это таблица, в которой строки обозначены элементами универсума, а столбцы – атрибутами. На пересечении строк и столбцов записываются значения атрибутов для элементов, входящих в универсум.

Отношение неразличимости, сгенерированное любым подмножеством атрибутов, есть отношение эквивалентности. Так, каждое подмножество атрибутов определяет разбиение универсального множества на гранулы, содержащие элементы, имеющие схожее описание с точки зрения значений атрибутов. Каждая такая гранула может представляться как основной строительный блок наших знаний об универсуме.

Предположим, что имеется два ограниченных, непустых множества U и A, где U – *универсум*, а A – множество *атрибутов*. С каждым атрибутом $a \in A$ ассоциируется множество V_a его *значений*, которое называется *доменом* a. Любое подмножество $B \subset A$ определяет бинарное отношение $I(B)$ на U, которое называется *отношением неразличимости* и определяется следующим образом:

$$xI(B)y \text{ тогда и только тогда, когда } a(x) = a(y) \text{ для каждого } a \in B,$$

где $a(x)$ означает значение атрибута a для элемента x.

Очевидно, что $I(B)$ есть отношение эквивалентности. Семейство всех классов эквивалентности $I(B)$, т. е. разбиений, определяемых B, обозначается $U/I(B)$, или проще U/B; класс эквивалентности $I(B)$, т. е. блок разбиения U/B, содержащий x, обозначается $B(x)$. Если (x, y) принадлежат $I(B)$, то из этого следует, что x и y являются *неразличимыми* относительно B.

Каждому подмножеству X универсума U можно поставить в соответствие два множества $B_*(X)$ и $B^*(X)$:

$$B_*(X) = \{x \in U : B(x) \subseteq X\},$$

$$B^*(X) = \{x \in U : B(x) \cap X \neq \emptyset\},$$

называемые *B-нижняя* и *B-верхняя* аппроксимации множества X соответственно. Множество

$$BN_B(X) = B^*(X) - B_*(X)$$

определяется как *B-граничная область* множества X.

Если граничная область X есть пустое множество, т. е. $BN_B(X) = \emptyset$, тогда множество X будет называться *четким (точным)* множеством относительно B. В противоположном случае, т. е. если $BN_B(X) \neq \emptyset$, множество X понимается как *приближенное (неточное)* относительно B.

Граничная область множества состоит из всех элементов универсума, о которых нельзя однозначно сказать, принадлежат они этому множеству или не принадлежат. Данная идея представляется в числовом виде как коэффициент, называемый *точностью аппроксимации* и определяемый следующим образом:

$$\alpha_B(X) = \frac{|B_*(X)|}{|B^*(X)|},$$

где $|X|$ обозначает мощность множества X.

Очевидно, что $0 \leq \alpha_B(X) \leq 1$. Если $\alpha_B(X) = 1$, тогда множество X является четким относительно B. Напротив, если $\alpha_B(X) < 1$, тогда X есть приближенное множество относительно B.

Метод представления приближенного множества, описанный выше, называется топологическим. Кроме него, применяется другой метод описания приближенных множеств, который называется вероятностным и также представлен в работе [175].

Суть вероятностного метода представления приближенного множества заключается в том, что наряду с понятием аппроксимации можно использовать понятие функции принадлежности так, как это часто делается в отношении множеств вообще. *Приближенная функция принадлежности* определяется следующим образом:

$$\mu_X^B(x) = \frac{|X \cap B(x)|}{|B(x)|}.$$

Очевидно, что

$$\mu_X^B(x) \in [0,1].$$

Может показаться, что это нечеткая функция принадлежности. То, что это не так, ясно видно из следующих свойств:

1. $\mu_{U-X}^B(x) = 1 - \mu_X^B(x), \forall x \in U$
2. $\mu_{X\cup Y}^B(x) \geq \max(\mu_X^B(x), \mu_Y^B(x)), \forall x \in U$
3. $\mu_{X\cap Y}^B(x) \leq \min(\mu_X^B(x), \mu_Y^B(x)), \forall x \in U$

Можно заметить, что приближенная принадлежность, в отличие от нечеткой принадлежности, имеет очевидную вероятностную особенность и может быть интерпретирована как разновидность условной вероятности [175].

Приближенная функция принадлежности может быть также использована для определения аппроксимаций и граничных областей множества:

$$B_*(X) = \{x \in U : \mu_X^B(x) = 1\}$$

$$B^*(X) = \{x \in U : \mu_X^B(x) > 0\}$$

$$BN_B(X) = \{x \in U : 0 < \mu_X^B(x) < 1\}.$$

Оба описанные представления приближенных множеств не являются эквивалентными. Первое определение связано с понятием множества и выражает невозможность точного описания множества. В то время как второе определение связано с элементами множества и выражает нашу невозможность классификации элементов в определенные понятия (концепты). В работе [175] делается вывод о том, что два описанных способа представления приближенных множеств (через точность аппроксимации и приближенную функцию принадлежности) соотносятся с различными аспектами неполноты знаний: первый способ – с нечеткостью понятий (концептов), а второй – с неопределенностью элементов.

1.5. Понятие единого информационного пространства проектной организации

Прежде чем говорить о роли информационного обеспечения в формировании единого информационного пространства проектной организации, необходимо дать формальное определение термину *проектная информация* и какое отношение она имеет к проектной документации. В работе [47] дано следующее определение: «проектная информация — это более частный вариант информационного объекта, который, понимается как описание некоторой сущности (реального объекта, явления, процесса, события) в виде совокупности логически связанных реквизитов (информационных элементов)».

В данной работе будем считать, что проектная информация — это информация, которая необходима проектировщику для принятия решений в процессе разработки технической системы. Виды представления проектной информации могут быть весьма различными: таблицы реляционных баз данных, текстовые документы, табличные документы, графики, модели и чертежи в своих форматах и другие. Вне зависимости от представления, проектная информация есть содержимое проектных документов, а проектные документы определяют форму представления проектной информации.

В работе [94] отмечается, что проектная документация служит для демонстрации продвижения проекта и повышения личной ответственности его участников за результат — на многих стадиях проекта при отсутствии документально подтвержденных результатов работ планирование развития проекта становится чрезвычайно трудной задачей.

В настоящее время практически вся проектная документация является доступной в электронном виде. Поэтому каждый отдельный проектный документ может считаться принадлежащим к классу электронных документов.

В работе [95] указывается, что электронный документ является не только одной из форм представления информации. Главное предназначение электрон-

ного документа — это *передача информации и знаний (человеку или машине)*. По этой причине можно утверждать, что набор проектной документации определяет информационное пространство проектируемой системы, что очень хорошо согласуется с концепцией CALS.

Концепция CALS (Continuous acquisition and life-cycle support), означающая непрерывную компьютерную поддержку всего ЖЦ изделия, лежит в основе создания ЕИП предприятий (в аглоязычной литературе часто применяется термин – Shared data environment – среда совместно используемых данных). В данной концепции определены три взаимосвязанных аспекта [114].

1. Рост числа задач, которые решаются с использованием автоматизированных систем. Данное утверждение связано с увеличением числа прикладных программных продуктов, относящихся к различным предметным областям и создаваемых независимыми друг от друга производителями.
2. Интеграция различных программных систем и реализация их интероперабельности. Основным направлением, в данном случае, является свойство интероперабельности относительно данных, то есть совместимость данных, создаваемых и принимаемых приложениями. Самой трудоемкой задачей, в этом случае, является достижение совместимости данных не только на физическом и логическом уровнях, но и на концептуальном уровне.
3. Использование средств, позволяющих выполнять интеграцию данных с целью повышения эффективности бизнес-процессов: внедрение PDM-технологий, реализации стратегий всеобщего управления качеством и реинжиниринга.

ЕИП должно обладать следующими свойствами:

- вся информация представлена в электронном виде;
- ЕИП охватывает всю информацию, созданную об изделии;
- ЕИП является единственным источником данных об изделии (прямой обмен данными между участниками ЖЦ исключен);

- ЕИП строится только на основе международных, государственных и отраслевых информационных стандартов;
- для создания ЕИП используются программно-аппаратные средства, уже имеющиеся у участников ЖЦ;
- ЕИП постоянно развивается.

В работе [39] отмечается, что технологии интеграции данных, с одной стороны, выступают в качестве **хранилища всех данных об изделии** и взаимодействуют с прикладными программами, создающими или использующими данные об изделии. Данные, созданные любой прикладной программой, передаются на хранение в PDM-систему и становятся доступными любому участнику ЖЦ изделия, имеющему соответствующие права доступа.

С другой стороны, PDM-системы должны решать задачу **повышения эффективности работы** отдельного пользователя. В этом случае они должны выступать в качестве рабочей среды пользователя, предоставляя ему нужные данные в нужное время и в нужной форме.

Авторами статьи [11] отмечается, что создание ЕИП наиболее актуально в опытно-конструкторских и проектных организациях, где остро стоит вопрос об информационной поддержке изделия на всех стадиях его ЖЦ.

Понятие жизненного цикла неразрывно связано с процессами проектирования сложных систем. В работе [87] отмечаются некоторые источники проблем, приводящие к необходимости иметь описание жизненного цикла разрабатываемой системы: разнородность составных элементов системы (оборудование, люди, программное обеспечение), комплексное использование компьютерных технологий, недостаточная интеграция применяемых дисциплин.

Определение стандарта ISO/IES 15288 (Системная и программная инженерия - Практики жизненного цикла системы) гласит: *жизненный цикл* (ЖЦ) – это эволюция системы, продукции, услуги, проекта или иного рукотворного объекта от замысла до прекращения использования [87]. Каждая система, вне зависимости от ее вида и масштаба, проходит весь свой ЖЦ согласно некото-

рому описанию. Продвижение системы по частям этого описания и есть ЖЦ системы.

1.6. Основные выводы и направление исследования

Анализ современного состояния информационного обеспечения САПР и текущего уровня сложности автоматизированных систем позволяет сделать вывод о необходимости включения в состав информационного обеспечения предметных знаний и опыта проектировщиков. Информационные ресурсы проектной организации должны быть структурированы с использованием моделей представления знаний, которые поддерживаются международными профессиональными сообществами. Подробнее представим выводы следующим образом.

1. Сложность современных АС приводит к необходимости кооперации в процессе проектирования большого количества групп разработчиков (проектных команд). Сложившийся принцип построения информационного обеспечения таких АС, предполагающий использование субъективно структурированных данных (реляционные таблицы, XML-файлы, атрибутивное представление документов), не способен обеспечить эффективную информационную поддержку распределенного процесса проектирования АС.
2. Интеграцию информационных ресурсов как в фактографических, так и в документальных информационных базах следует выполнять на семантическом уровне с использованием международных стандартов и рекомендаций, например, разработанные консорциумом W3C. Классификаторы, словари и унифицированные формы документов могут составлять основу для формирования семантического метауровня информационного обеспечения проектирования современных АС, основанного на знаниях.
3. В основе модели представления знаний интеллектуального проектного репозитория (ИПР) должна применяться *интегрированная система онтологий*, позволяющая представлять знания не только предметной области

проектной организации, но и знания, описывающие особенности поведения, состояния и структурные аспекты объекта проектирования.

В соответствии с указанными выводами основыми направлениями теоретических и прикладных исследований будем считать следующие.

1. Разработка формальных моделей прикладных онтологий для решения задач информационной поддержки процесса проектирования сложных автоматизированных систем. Структурно-аналитическое представление прикладных онтологий должно быть нацелено на решение задач структурирования, агрегации, содержательной интерпретации и формирование поисковых запросов к слабоструктурированным информационным ресурсам электронных архивов крупных проектных организаций.
2. Составление методики формирования прикладных онтологий информационной поддержки проектирования автоматизированных систем, включающей в себя набор процедур лексического описания понятий онтологии и способ оценивания качества онтологии на основе анализа ее фрагментов.
3. Формальное представление процедуры концептуального индексирования документальных информационных ресурсов и аналитическое описание концептуального индекса электронного архива проектной организации как отображения системы онтологий ИПР автоматизированных систем на информационную базу САПР.
4. Разработка формальных процедур концептуальной структуризации электронных архивов и алгоритма содержательной интерпретации кластеров электронных архивов.
5. Разработка способа содержательной интерпретации тенденций технических временных рядов, позволяющего в терминах предметной области формировать заключения о динамике технических показателей на различных промежутках времени.
6. Формализация информационной потребности проектировщика на концептуальном уровне, учитывающей отличия решаемых проектировщиком за-

дач на различных стадиях жизненного цикла разрабатываемой автоматизированной системы.

Глава 2

Структурно-логическая модель онтологии интеллектуального проектного репозитория

2.1. Семантический базис проектного репозитория

Проектирование современных АС является примером крайне сложной человеческой деятельности, требующей привлечения высококвалифицированных специалистов из различных предметных областей. Процесс проектирования АС можно назвать специфической интеллектуальной деятельностью человека (или группы лиц), которая состоит в принятии взаимосвязанных проектных решений на основе знаний по отношению к объекту проектирования.

В современном высококонкурентном мире на первое место выходят такие показатели эффективности процесса разработки новых изделий, как время от замысла до изготовления серийных образцов, уровень технологичности, степень унификации составных подсистем и т. д. Достижение приемлемых значений указанных показателей возможно только при условиях использования методологии коллективной разработки АС, формирования ЕИП процесса проектирования и максимального использования накопленного опыта и знаний высококвалифицированных специалистов. Указанные условия накладывают ряд ограничений на информационное обеспечение (ИО) САПР АС [81] и, в частности на проектные репозитории.

ИО проектирования АС организуется в соответствии с основными этапами проектной деятельности [156]. Совокупность знаний, которые требуются субъекту проектирования в процессе разработки нового изделия, можно разделить на две категории: *системные знания* и *ситуативные знания*. Для системных знаний характерно наличие устойчивых связей и свойств предметов и явлений объективного мира. Источником таких знаний часто является научно-техниче-

ская литература. В свою очередь, ситуативные знания часто отражают соотнесенные с определенной ситуацией связи и отношения. Информация для ситуативных знаний содержится в методологической, нормативно-технической, производственной и справочной литературе.

Жизненный цикл сложной системы (рисунок 2.1) согласно ГОСТ 22487-77 «Проектирование автоматизированное. Термины и определения» включает следующие стадии (фазы) [20].

1. Формирование требований к системе и разработка ТЗ (внешнее проектирование или макропроектирование).
2. Проектирование (внутреннее проектирование или микропроектирование).
3. Изготовление, испытание и доводка опытных образцов.
4. Серийное производство.
5. Эксплуатация и целевое применение.
6. Консервация и хранение.
7. Модернизация и эксплуатация.

Стадии жизненного цикла, входящие в НИОКР, являются наиболее важными с точки зрения вероятности появления ошибок в проекте и увеличения времени проектирования нового изделия. Указанные стадии предполагают использование и системных и ситуативных знаний. В рамках каждой стадии или отдельно взятого этапа проектирования АС решаются вполне конкретные проектные задачи, на выходе которых имеет место набор артефактов проектирования, включая техническую документацию. Можно утверждать, что ситуативные знания соотносятся с отдельными стадиями и (или) этапами проектирования АС.

Знания, используемые в проектировании АС будем рассматривать как многоаспектную систему, включающую в себя (рисунок 2.2):

- аспект по степени объективности, различающий знания *объективные* и *субъективные*;
- аспект по степени обобщения, позволяющий различать *системные* и *си-*

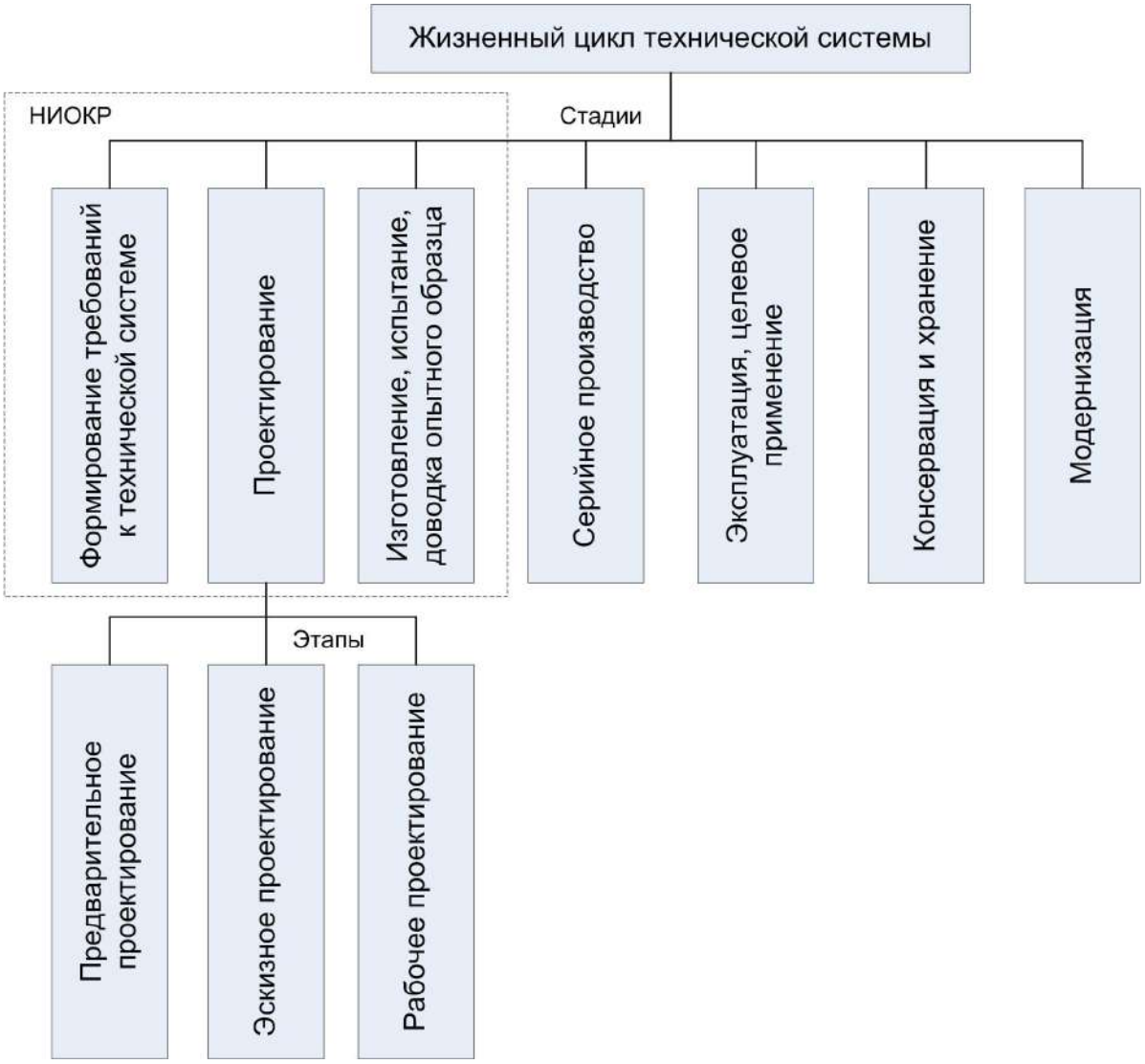


Рис. 2.1. Структура жизненного цикла сложной системы

туативные знания.

Использование системы знаний в проектных процедурах САПР АС нацелено на решение таких задач, эффективное решение которых требует семантической обработки гетерогенных информационных ресурсов и учета принципиальной неполноты информации, связанной с применением естественного языка (текстовые ресурсы и полуформальные графические нотации, применяемые в проектной деятельности (например, UML, IDEF0, IDEF3, DFD и IDEF1X)). Среди них в данной диссертационной работе будем выделять следующие (рис. 2.2):

- интегрирование информационных ресурсов проектной организации;
- семантическая структуризация информационных ресурсов;

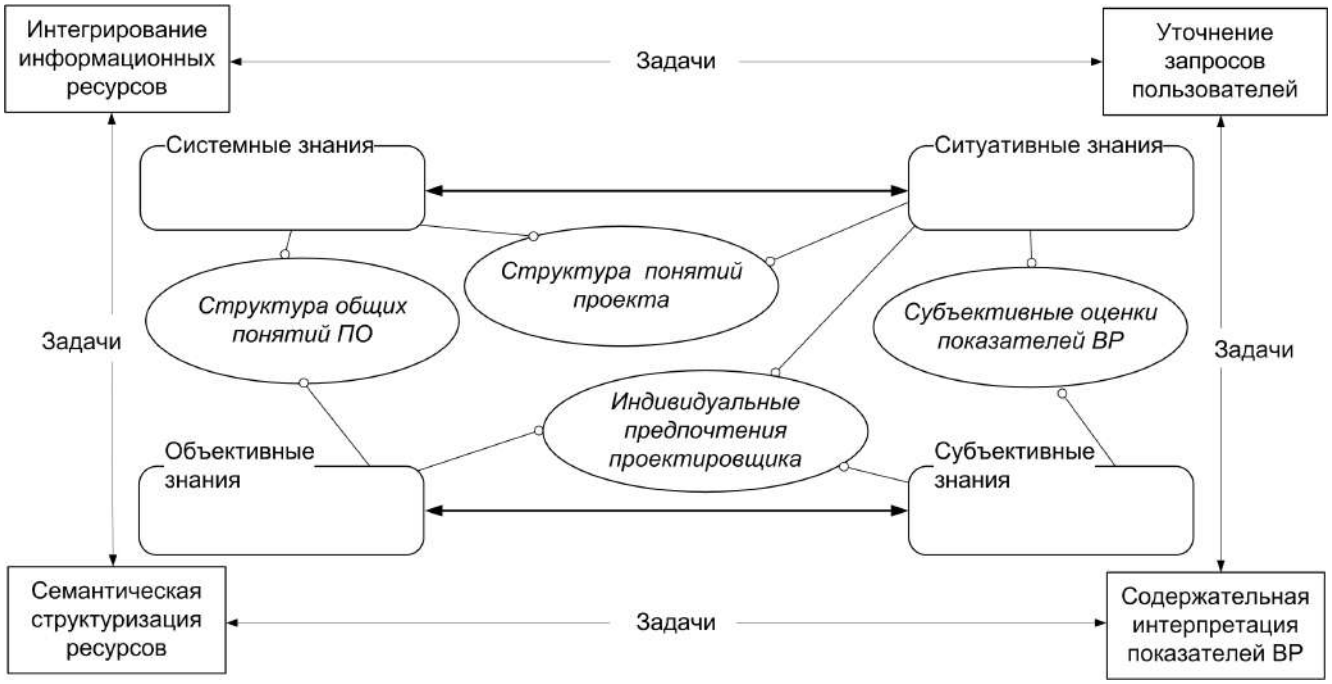


Рис. 2.2. Структура знаний проектной организации

- уточнение проектных запросов пользователей (субъектов проектирования);
- содержательная интерпретация изменений показателей технический временных рядов (ВР).

Определим фрагменты знаний в проектной организации, имеющие непосредственное значение для вышеуказанных задач информационной поддержки проектирования АС.

Структура общих понятий ПО специфицирует набор понятий [102] и отношения между ними той части ПО проектной организации, которая инвариантна относительно реализуемых проектов. Источником знаний здесь часто являются системные знания и форма представления знаний, в основном, декларативная. В онтологии интеллектуального проектного репозитория структуру общих понятий будем представлять в виде таксономии понятий, дополненную универсальными семантическими отношениями (например, «является частью», «включается в» и другими).

В отличие от структуры общих понятий *структура понятий проекта* об-

разует систему понятий, которые являются специфичными для отдельно взятых проектов АС. Данную структуру определяют в большей степени ситуативные знания. Форма представления знаний является также декларативной. Множество понятий проекта состоит из понятий, наследуемых от общих понятий ПО.

Индивидуальные предпочтения проектировщика отражают опыт его взаимодействия с информационными ресурсами в процессе проектирования АС. Формально такие предпочтения могут быть представлены в виде графа, вершинами которого являются подмножество объединения множеств общих понятий ПО и понятий проектов, в реализации которых проектировщик принимал участие (выполнял проектные запросы к электронному архиву технической документации).

Субъективные оценки показателей временных рядов позволяют выполнять содержательную интерпретацию основные трендов технических временных рядов (ВР). Данные оценки формируются на основе ситуативных знаний (набор анализируемых показателей определяется структурой проектируемой АС) с использованием субъективных знаний о лингвистических оценках.

Основанный на знаниях подход к формированию информационного обеспечения САПР АС приводит к необходимости переформулирования требований к проектным репозиториям САПР. Одним из перспективных направлений развития современных проектных репозиториев является их интеллектуализация. В данном исследовании реализацию интеллектуальных проектных репозиториев предлагается рассматривать как частный случай построения интеллектуальной САПР.

2.2. Требования к онтологии проектного репозитория. Структура интегрированной онтологии

При построении модели предметной области САПР в виде онтологии интеллектуального проектного репозитория (ИПР) для решения задач анализа информационных ресурсов информационных баз САПР необходимо сформулировать основные требования к такой онтологии. Указанные требования должны опираться на особенности предметной области, решаемые проектные задачи. Кроме того, должны учитываться особенности информационных ресурсов, к которым выполняются проектные запросы.

Основной целью использования онтологий является формирование базы знаний об информационном окружении проектируемых систем для сокращения времени нахождения необходимых информационных ресурсов. Фактически, для каждого слабоструктурированного информационного ресурса (технического документа, проектной диаграммы) онтология задает новую систему координат, в которой задачи структурирования, поиска и содержательной интерпретации могут решаться на принципиально новом семантическом уровне [69], [71].

На основании вышесказанного сформулируем основные требования к онтологии интеллектуального проектного репозитория.

- Онтология ИПР должна включать в себя характерные особенности применяемых в организации методологий разработки и моделей ЖЦ проектируемых АС.
- Структура и содержание онтологии должны основываться на применяемых в проектной организации нормативно-справочной информации, стандартов, руководящих документов, с учетом которых определяется фактическая понятийная и терминологическая структура.
- Понятийная структура онтологии должно включать в себя те концепты, которые извлекаются из реализованных проектов, сохраненных в электронных архивах проектной организации (фактически, речь идет об ис-

пользовании накопленного опыта в проектировании АС).

- Основу онтологии ИПР должна определять таксономия понятий соответствующей предметной области. Дополнительные классификации понятийного аппарата определяются с применением выбранного набора семантических отношений (например, «часть-целое», «ассоциация» и другие).
- На уровне отдельных информационных ресурсов информационного обеспечения САПР АС атомарными информационными единицами являются термины, извлекаемые из документов (технические документы), наименование классов, атрибутов и методов (объектно-ориентированные нотации моделей программных систем), наименование сущностей и атрибутов (модели данных). Соответствующая онтология должна включать в себя функции интерпретации (семантического отображения) атомарных уровней на понятийный уровень.

Автоматизированная система как объект проектирования является сложной системой. Проектные процедуры предполагают всесторонний анализ возможных способов реализации подсистем объекта проектирования и их последующий синтез. Все выполняемые проектные процедуры сопровождаются разработкой технической документации и других артефактов проектирования. Онтология, как модель представления знаний о различных аспектах информационного обеспечения автоматизированного проектирования АС, должна включать в себя многие закономерности как проектных процедур, так и знания об объекте проектирования, субъективных предпочтениях проектировщиков и т.д. Переходя от одной стадии проектирования к другой стадии, у объекта проектирования изменяется контекст, в котором реализуются проектные процедуры. Предлагаемая система контекстов автоматизированного проектирования АС в самом общем виде представлена в следующей главе.

В данном исследовании предлагается использовать для реализации базы знаний ИПР не одну онтологию, а интегрированную систему онтологий (рисунок 2.3). Компонентами интегрированной системы онтологий являются: онтоло-

гия предметной области, концептуальные сети проектов, онтология проектных диаграмм, онтология жизненного цикла, тезаурус проектной организации и онтология анализа технических временных рядов.

Рис. 2.3. Интегрированная система онтологий ИПР

Исходными данными для онтологии предметной области являются различного уровня стандарты (ГОСТы, ОСТы, стандарты предприятия), техническая документация и научно-техническая литературы. Данная онтология является основой формальной концептуализации предметной области проектирования АС в организации. Для каждого вновь создаваемого проекта формируется отдельная концептуальная сеть - набор специализированных понятий, извлекаемых из проектной документации на начальных этапах проектирования АС. Дополнительно для расширения концептуальных сетей привлекаются внешние профессиональные wiki-ресурсы. Онтология проектных диаграмм включает в себя концептуализацию основных нотаций, применяемых при моделировании программных подсистем проектируемых АС. Кроме того, в состав данной онто-

логии входит набор основных шаблонов проектирования. Онтология жизненного цикла состоит из множества стадий и этапов проектирования АС, связанных между собой в соответствии с принятой методологией проектирования, которая, в свою очередь, определяет применяемую модель жизненного цикла АС.

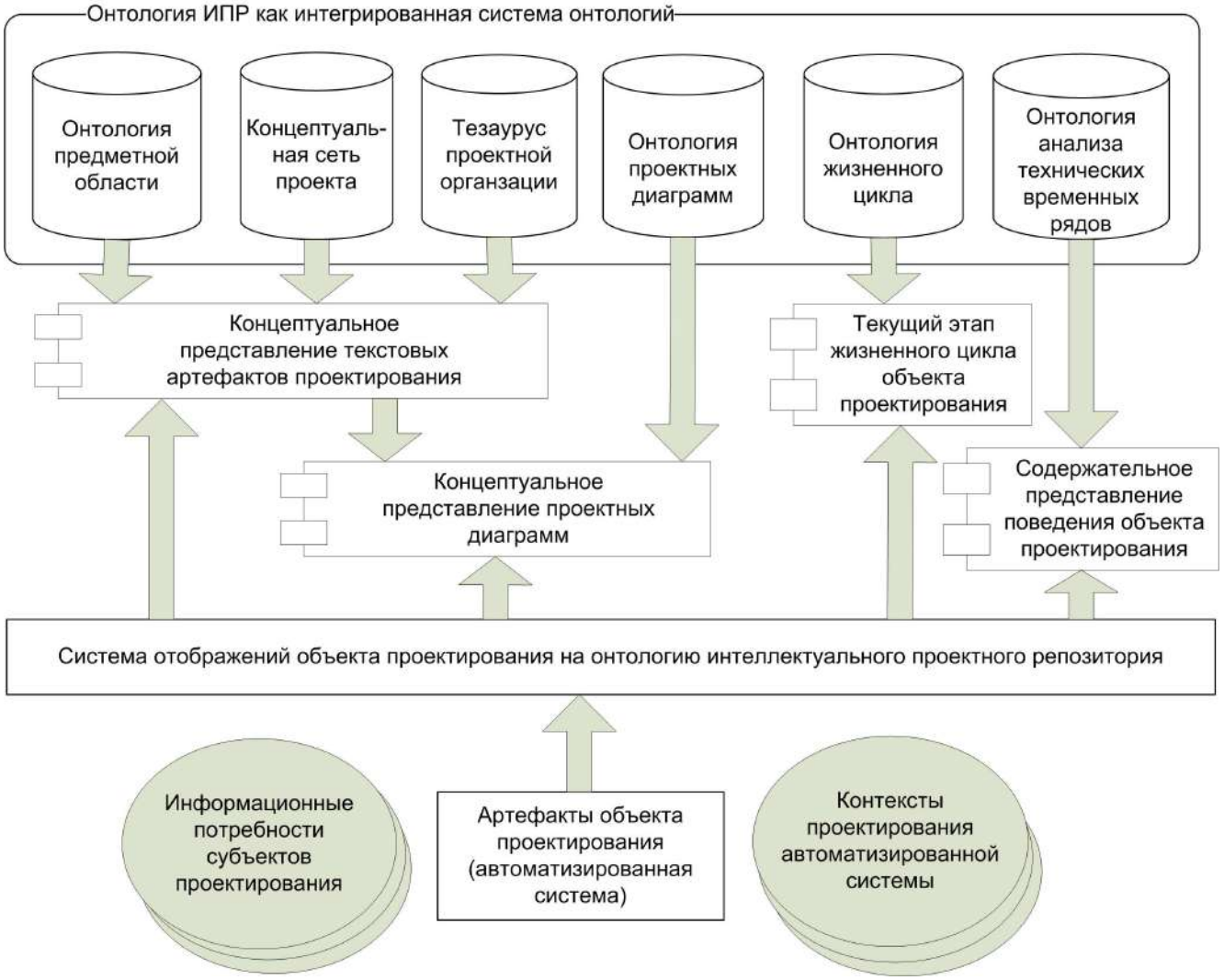


Рис. 2.4. Отображение объекта проектирования на онтологию ИПР

Тезаурус проектной организации непосредственно связан с онтологией предметной области и фактически определяет набор терминов, семантически связанных с понятиями онтологии (текстовые входы понятий). Это позволяет осуществлять концептуальное индексирование текстовых документов электронного архива. Онтология анализа технических временных рядов включает в себя множество показателей компонентов проектируемых АС для обеспечения воз-

можности автоматического выполнения процедур содержательной интерпретации значений указанных показателей, изменяющихся во времени. Интерпретация динамики изменения показателей осуществляется на основе экспертных оценок и шаблонов лингвистических меток, зафиксированных в онтологии.

Каждая отдельно взятая онтология является средством отображения множества артефактов объекта проектирования (технические документы, модели, диаграммы) на концептуальную плоскость, формируя *концептуальный индекс* электронного архива проектной организации (рисунок 2.4).

2.3. Теоретико-множественная модель онтологии интеллектуального репозитория

2.3.1. Модель онтологии предметной области

Проектирования сложных АС накладывает определенные требования к структуре онтологии предметной области. Особенность структуры и содержания информационных ресурсов электронных архивов и проектной деятельности такова, что онтология предметной области разделена на метауровни, как представлено на рисунке 2.5.

Формально набор компонентов онтологии предметной области запишем как кортеж вида [49]:

$$O^{dom} = \langle PL, DL, CL, R^{dom}, F^{dom} \rangle,$$

где PL – метауровень проектов, содержащий информацию о реализуемых проектах: таксономия классов проектов и экземпляры проектов, содержащих технические документы; DL – метауровень документов, включающий таксономию классов документов и экземпляры документов; CL – метауровень понятий, основу которого составляет таксономия понятий (предметной области проектной организации и реализуемых проектов), дополнительно используются такие отношения, как «имеет часть», «связан с» и другие; R^{dom} – множество отношений

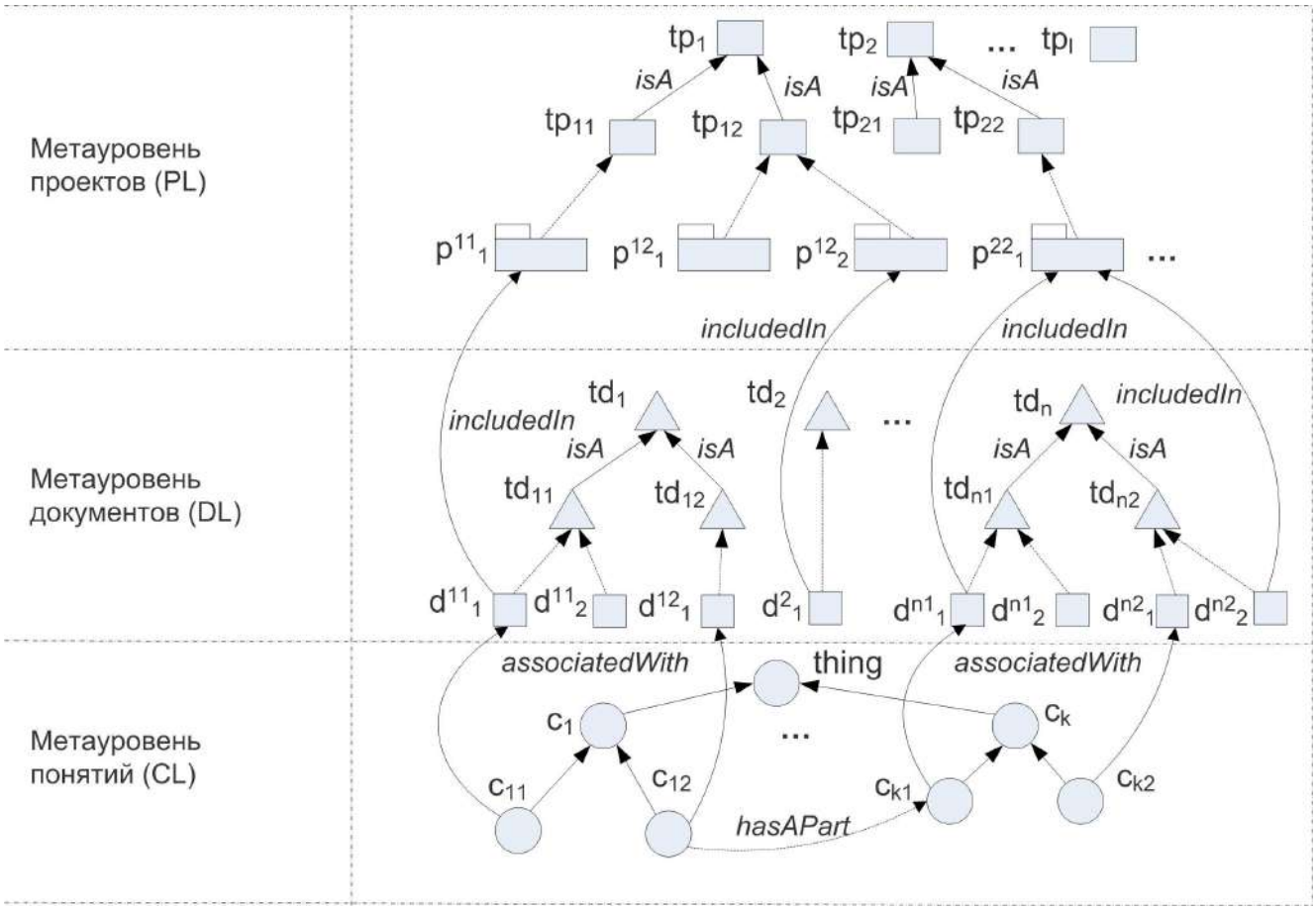


Рис. 2.5. Структура онтологии предметной области

между концептами и/или экземплярами, относящимися к различным метауровням онтологии; F^{dom} – множество функций интерпретации на уровне описания предметной области проектной организации.

Рассмотрим каждый метауровень более детально. Метауровень проектов представим в виде множества:

$$PL = \{TP, P, R^{PL}\}, \tag{2.1}$$

где TP – концепт, представляющий множество типов проектов; P – множество проектов в электронном архиве проектной организации; $R^{PL} = \{R_G, R_R\}$ – множество, включающее отношение обобщения R_G и отношение реализации R_R.

Концепт онтологии $P = \{p_1, p_2, \ldots p_s\}$ включает в себя в качестве индивидов (экземпляров) все ранее реализованные проекты. Таким образом каждый

p_i представляет собой отдельный проект организации.

Отношение обобщения R_G (на рисунке 2.5 – «isA») является бинарным, транзитивным и связывает между собой пары типов проектов из множества TP. Допустимым является запись вида: $tp_i R_G tp_j, tp_i, tp_j \in TP$, которая означает, что tp_i является подтипом tp_j. В свою очередь, бинарное нетранзитивное отношение реализации R_R будем определять между проектом и типом проекта, которое может быть записано в следующем виде: $p_i R_R tp_j, p_i \in P, tp_j \in TP$, что означает принадлежность проекта p_i к типу tp_j.

Рассмотрим структуру метауровня документов, представляемого следующим образом:

$$DL = \{TD, D, R^{DL}\}, \tag{2.2}$$

где TD –множество типов документов (на самом верхнем уровне иерархии имеет место всего три типа: текстовый технический документ, диаграмма классов и модель данных «сущность-связь»); D –множество документов:

$$D = \{d_1^1, \dots, d_i^j, \dots\},$$

где d_i^j – i-й документ j-го типа; $R^{DL} = \{R_G, R_R\}$ – множество, включающее отношение обобщения R_G и отношение реализации R_R.

По аналогии с предыдущим метауровнем отношение обобщения R_G связывает между собой пары типов документов из множества TD. Запись вида: $td_i R_G td_j, td_i, td_j \in TD$ означает, что td_i является подтипом td_j. В свою очередь, отношение реализации R_R для данного метауровня будем определять между документом и типом документа, которое может быть записано в следующем виде: $d_i R_R td_j, d_i \in D, td_j \in TD$, что означает принадлежность документа d_i к типу td_j.

Формально метауровень понятий представим в следующем виде:

$$CL = \{thing, C, R^{CL}\}, \tag{2.3}$$

где $thing$ – корневой концепт, включающий все понятия; C – множество понятий (концептов) предметной области проектной организации; $R^{CL} = \{R_G, R_P^{CL}\}$ – отношения, включающее множество отношений обобщения R_G и множество бинарных транзитивных отношений «часть-целое (hasAPart)» R_P^{CL}.

Множество понятий C состоит из следующих элементов:

$$C = \{C^{St_1} \cup C^{St_2} \cup \ldots \cup C^{St_k}\} \cup C^P,$$

где $C^{St_i}, i = \overline{1,k}$ – множество понятий, извлекаемых из стандартов i-ой серии, которые используются в проектной организации (например, ГОСТ 34.602-89, ГОСТ 19.201-78 и т. д.); C^P – множество понятий, которые извлекаются из текстовых технических документов (артефактов проектирования АС).

Бинарное отношение обобщения R_G связывает между собой пары понятий из множества C. Запись вида: $c_i R_G c_j, c_i, c_j \in C$ означает, что c_i является подпонятием (наследуемым понятием) c_j. Отношение «часть-целое (hasAPart)» R_P^{CL} будем определять между двумя понятиями таким образом, что $c_i R_P^{CL} c_j, c_i, c_j \in C$ означает, что сущность определяемая c_i имеет в качестве части (фрагмента) c_j.

Рассмотрим множество R^{dom} кортежа (2.7):

$$R^{dom} = \{R^I, R^A, R^R\}, \tag{2.4}$$

где R^I – множество отношений типа «includedIn (включен в)», связывающих технические документы электронного архива с проектами; R^A – множество отношений типа «associatedWith (ассоциирован с)», связывающих множество понятий с документами, в которых данные понятия встречаются (рис. 2.5).

В работе будем различать *априорные* и *апостериорные* отношения, отличие между которыми заключается в следующем: априорные отношения определяются независимо от содержимого информационной базы, в свою очередь, апостериорные отношения формируются по мере заполнения информационной базы электронными ресурсами (техническим документами и проектными диаграммами).

Дадим следующие определения.

Определение 1. *Априорное онтологическое отношение* – это такое отношение между двумя вершинами онтологического графа, которое определяется независимо от того набора информационных ресурсов, который формирует содержимое информационной базы.

Определение 2. *Апостериорное онтологическое отношение* – это такое отношение между двумя вершинами онтологического графа, для определения которого привлекается информация из информационных ресурсов, составляющих содержимое информационной базы.

В качестве апостериорных отношений в онтологии проектного репозитория рассматриваются отношения из множества R^{dom} (2.4), остальные отношения являются априорными.

Множество F^{dom} включает в себя следующие *интерпретирующие функции*:

$$F^{dom} = \{F^{dom}_{tp}, F^{dom}_{ptp}, F^{dom}_{td}, F^{dom}_{dtd}, F^{dom}_{cisc}, F^{dom}_{cpc}, F^{dom}_{dp}, F^{dom}_{cd}\},$$

где $F^{dom}_{tp} : TP \to TP$ – функция, сопоставляющая типу проекта его родительский тип; $F^{dom}_{ptp} : P \to TP$ – функция, сопоставляющая проекту его тип; $F^{dom}_{td} : TD \to TD$ – функция, сопоставляющая типу документа его родительский тип; $F^{dom}_{dtd} : D \to TD$ – функция, сопоставляющая документу его тип; $F^{dom}_{cisc} : C \to C$ – функция, сопоставляющая понятию (концепту) его родительское понятие; $F^{dom}_{cpc} : C \to C$ – функция, сопоставляющая понятию (концепту) другое понятие, связанное с первым отношением «part_of»; $F^{dom}_{dp} : D \to P$ – функция, сопоставляющая документу проект, которому он принадлежит; $F^{dom}_{cd} : C \to D$ – функция, сопоставляющая понятию документ, в котором оно упоминается.

2.3.2. Тезаурус проектной организации

Основой лингвистической компененты онтологии ИПР является *тезаурус проектной организации*, структура которого представлена на рисунке 2.6.

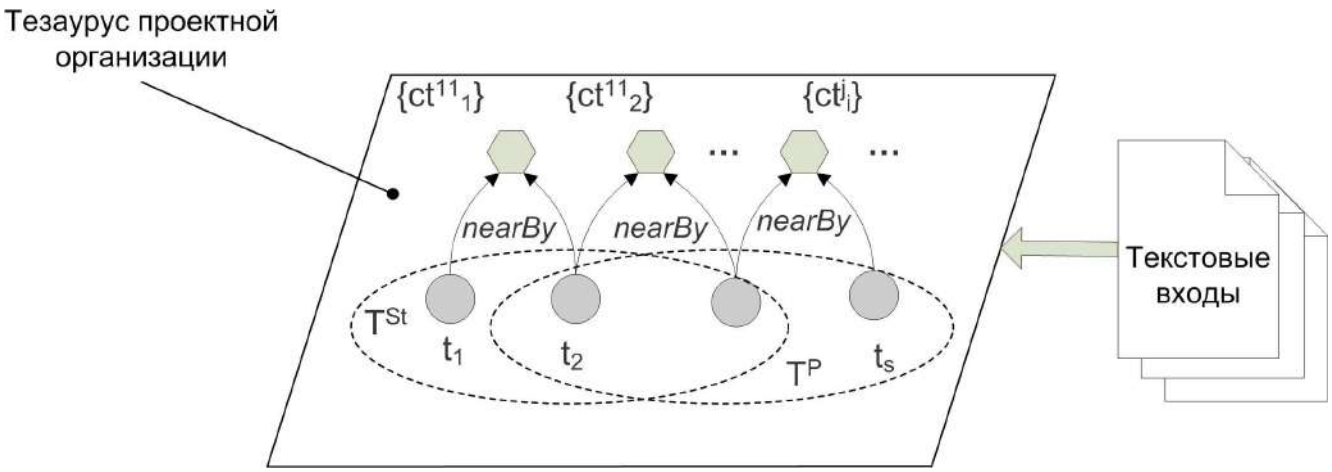


Рис. 2.6. Структура тезауруса проектной организации

Структуру тезауруса проектной организации запишем как кортеж вида:

$$O^{tz} = \langle CT, T, R^{tz}, F^{tz} \rangle. \tag{2.5}$$

Каждое понятие из множества CT фактически представляет собой одноэлементное множество $\{ct_j\}$ (или *номинал* в терминах дескриптивных логик). Индекс j указывает на на порядковый номер номинала. Фактически элемент ct_j является термином-дескриптором соответствующего понятия онтологии предметной области.

Множество T состоит из термов $t_i, i = \overline{1,s}$ – терминов, извлеченных из документальных информационных баз проектной организации и прошедших процедуру стемминга. Множество $T = T^{St} \cup T^{P}$ есть множество термов предметной области (T^{St} – множество термов в стандартах предприятия, T^{P} – множество термов в текстовых артефактах проектирования).

Отношение R^{tz} есть отношение вида «nearBy» между термом t_k и номиналом $\{ct_j\}$. Запись вида $t_k R^{tz} \{ct_j\}$ означает, что терм t_k находится близко с

точки зрения семантики относительно понятия c_j, выраженного в тексте в виде терма, представляемого номиналом $\{ct_j\}$. Далее в работе будет представлено формальное описание семантического расстояния между терминами (термами) текстовых технических документов.

Функция интерпретации $F^{tz} : T \times CT \to [0,1]$ – функция, сопоставляющая паре $\{t_i, \{ct_j\}\}$ вещественное число из диапазона $[0,1]$, которое определяет степень семантической близости термина t_i и понятия ct_j.

2.3.3. Онтология жизненного цикла

Онтологию жизненного цикла проектирования АС O^{lc} представим в виде кортежа (рисунок 2.7):

$$O^{lc} = \langle St, R^{lc}, F^{lc} \rangle,$$

где St – множество стадий проектирования АС, которое соответствует реализуемой модели жизненного цикла проектируемой АС; R^{lc} – множество семантических отношений онтологии O^{lc}; F^{lc} – множество функций интерпретации в O^{lc}.

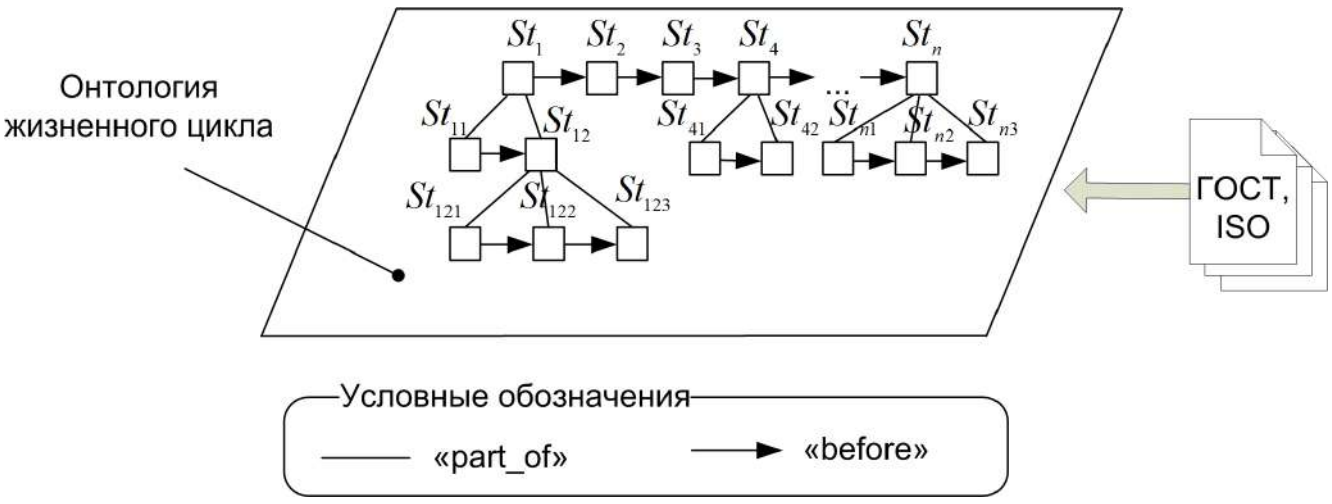


Рис. 2.7. Структура онтологии жизненного цикла объекта проектирования

Множество отношений R^{lc} включает в себя два класса отношений:

$$R^{lc} = \{R^{lc}_B, R^{lc}_{Pt}\},$$

где R_B^{lc} – бинарное семантическое отношение, которое связывает две стадии проектирования между собой, одна из которых предваряет другую; отношение R_{Pt}^{lc} определяет бинарное семантическое отношение типа «part_of» между стадиями и проектными процедурами. Формальная запись $St_{ijk}R_{Pt}^{lc}St_{ij}$ означает включение проектной процедуры St_{ijk} в состав стадии St_{ij}.

Интерпретирующая функция $F^{lc} : St \to St$ – есть функция, сопоставляющая стадии проектирования связанную с ней проектную процедуру.

2.3.4. Концептуальная сеть проекта

На начальных этапах проектирования нового изделия формируется концептуальная сеть проекта O^{cn}, которую представим в виде кортежа (рисунок 2.8):

$$O^{cn} = \langle CS, R^{cn}, F_{in}^{cn}, F_{out}^{cn} \rangle,$$

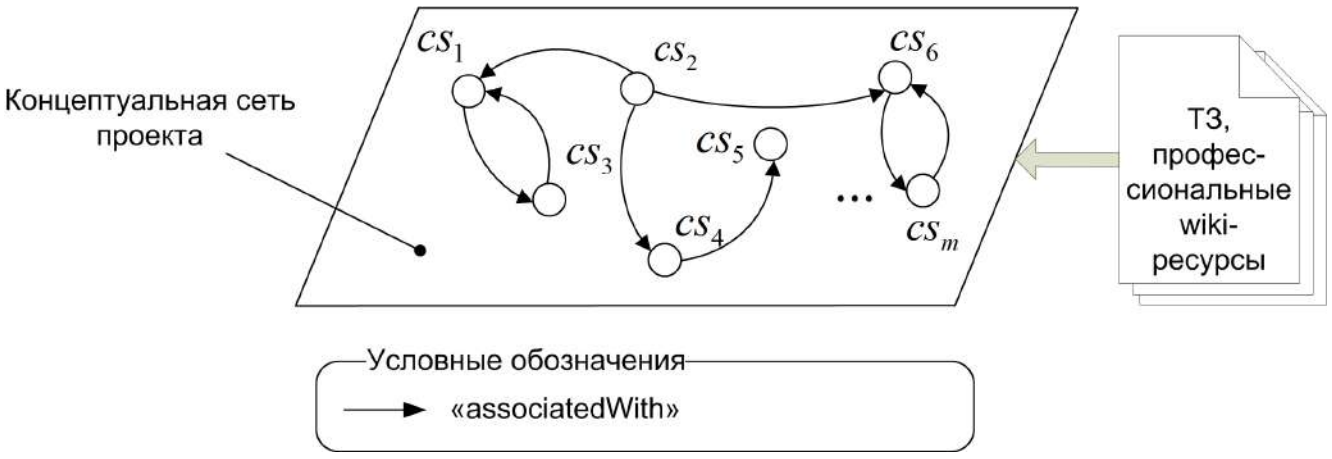


Рис. 2.8. Структура концептуальной сети проекта

где $CS = \{cs_1, \ldots, cs_m\}$ – множество понятий, имеющих отношение к проекту и извлекаемые из wiki-ресурсов; R^{cn} – направленное бинарное отношение «associatedWith», связывающее два понятия; $F_{in(out)}^{cn} : CS \to n$ – функция интерпретации, ставящее в соответствие понятию $cs_i \in CS$ натуральное число n, определяющее входящих (соответственно, исходящих) дуг-отношений для данного понятия.

Подробное описание алгоритма формирования концептуальной сети проекта АС представлено в следующей главе данной работы.

2.3.5. Онтология анализа технических временных рядов

Мониторинг показателей технических временных рядов позволяет накапливать в базах данных электронного архива проектной организации важную с точки зрения принятия проектных решений информацию. Анализ поведения во времени технических подсистем дает возможность проектировщику сделать осознанный и обоснованный выбор архитектурных решений, реализаций подсистем с учетом требований по производительности, надежности и других показателей качества.

Предлагаемая структура онтологии анализа технических временных рядов включает набор шаблонов различных тенденций (рисунок 2.9). Каждый шаблон соотносится с выделенным фрагментом временного ряда, который, в свою очередь, имеет отношение к анализируемому показателю. Кроме того, с каждым шаблоном связана лингвистическая метка, определяющая качественную характеристику соответствующей тенденции.

Формально, онтологию анализа технических временных рядов представим в виде кортежа:

$$O^{ts} = \langle V, TS, FT, TRG, R^{ts}, F^{ts} \rangle,$$

где $V = \{v_1, \ldots, v_m\}$ – множество понятий, определяющих анализируемые показатели АС; TS – понятие, определяющее множество временных рядов $(ts_{11}, ts_{12}, \ldots)$; FT – понятие, определяющее множество фрагментов временных рядов $(ft_{111}, ft_{112}, \ldots)$; TRG – понятие, определяющее множество шаблонов вида прямоугольного треугольника $(trg_{1111}, trg_{1112}, \ldots)$, которые сопоставляются с представлениями фрагментов временных рядов.

Множество отношений онтологии O^{ts} будем записывать следующим обра-

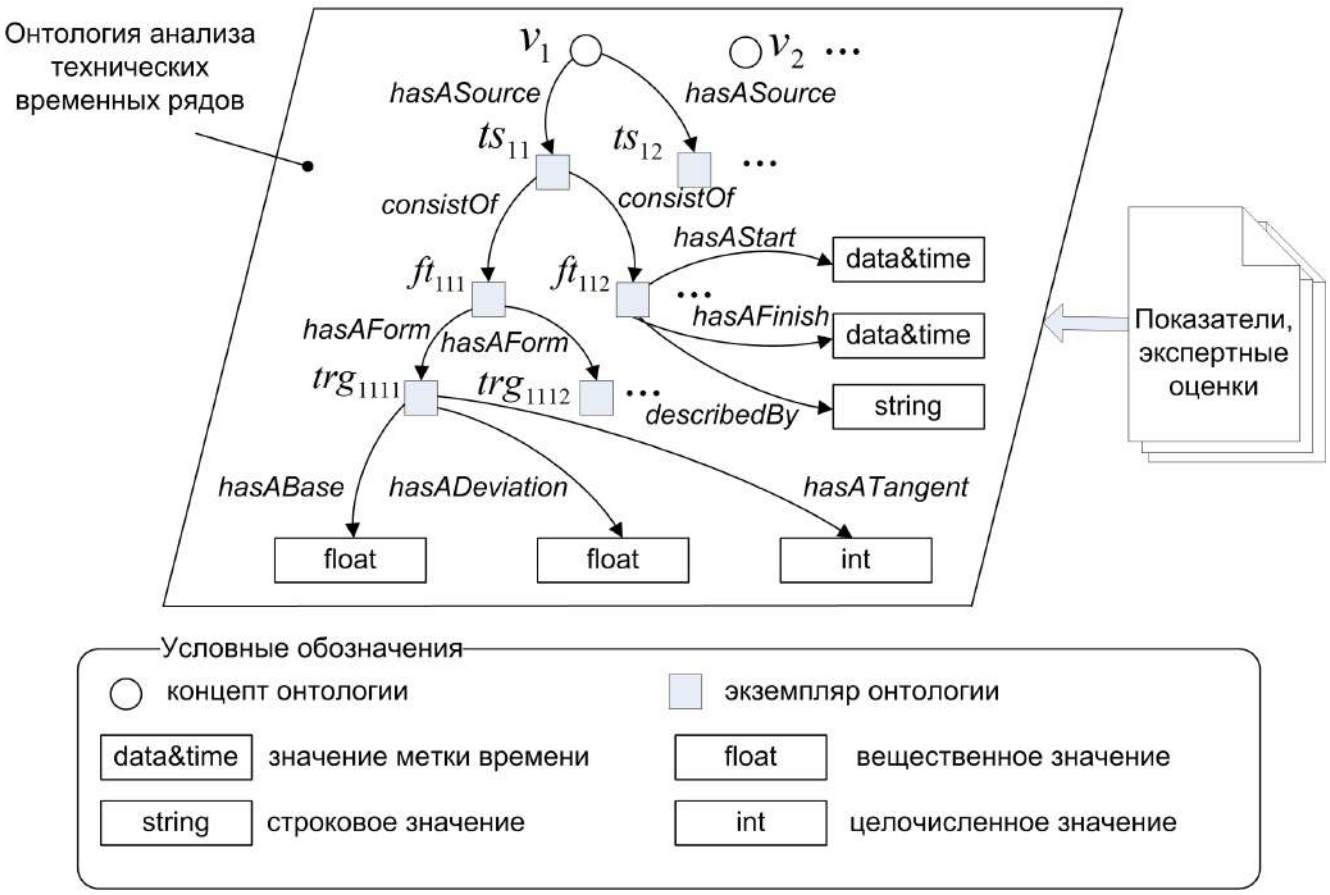


Рис. 2.9. Структура онтологии анализа технических временных рядов

зом:

$$R^{ts} = \{R^{ts}_{src}, R^{ts}_{cns}, R^{ts}_{str}, R^{ts}_{fin}, R^{ts}_{desc}, R^{ts}_{form}, R^{ts}_{base}, R^{ts}_{dev}, R^{ts}_{tan}\},$$

где R^{ts}_{src} – направленное бинарное отношение «hasASource», связывающее два индивида классов V и TS; R^{ts}_{cns} – направленное бинарное отношение «consistOf», связывающее два индивида классов TS и FT и позволяющее представить технический временной ряд как последовательный набор фрагментов; R^{ts}_{form} – направленное бинарное отношение «hasAForm», сопоставляющее каждому фрагменту временного ряда ft_i шаблон trg_j; $R^{ts}_{str}(R^{ts}_{fin})$ – отношение, связывающее фрагмент временного ряда с меткой времени, соответствующей его началу (соответственно, окончания); R^{ts}_{desc} – отношение, связывающее фрагмент временного ряда с лингвистической меткой, характеризующей тренд и имеющей строковое представление; отношения $R^{ts}_{base}, R^{ts}_{dev}, R^{ts}_{tan}$ связывают шаблон фрагмента временного ряда с числовыми значениями параметров шаблона: основания тре-

угольника, расхождения шаблона и представления фрагмента временного ряда, и тангенса наклона гипотенузы шаблона, соответственно.

Множество функций интерпретации F^{ts} включает в себя следующие компоненты:

$$F^{ts} = \{F^{ts}_{vts}, F^{ts}_{tsft}, F^{ts}_{str}, F^{ts}_{fin}, F^{ts}_{desc}, F^{ts}_{fttrg}, F^{ts}_{base}, F^{ts}_{dev}, F^{ts}_{tan}\},$$

где $F^{ts}_{vts} : V \to TS$ – функция сопоставления каждому показателю набор временных рядов; $F^{ts}_{tsft} : TS \to FT$ – отображение временного ряда на его фрагменты; $F^{ts}_{str(fin)} : FT \to type(data\&time)$ – функция интерпретации, определяющая для фрагмента временного ряда его начальную (конечную) временную метку; $F^{ts}_{desc} : FT \to type(string)$ – функция, задающая фрагменту временного ряда лингвистическую метку строкового типа; $F^{ts}_{fttrg} : FT \to TRG$ – функция, задающая фрагменту временного ряда набор шаблонов; $F^{ts}_{base} : TRG \to type(float)$ – функция определения основания для шаблона фрагмента временного ряда; $F^{ts}_{dev} : TRG \to type(float)$ – функция определения степени расхождения шаблона фрагмента временного ряда с представлением временного ряда; $F^{ts}_{tan} : TRG \to type(int)$ – функция определения тангенса угла наклона для шаблона фрагмента временного ряда.

2.3.6. Онтология проектных диаграмм

Для решения задачи интеллектуального анализа проектных диаграмм, входящих в состав проектной документации, необходимо обладать знаниями в области построения формализованных диаграмм (использования нотаций). На рисунке 2.10 представлена структура фрагмента онтологии проектных диаграмм, в частности, диаграммы классов языка UML (Unified Modeling Language) [127].

Такие знания позволяют выполнять идентификацию применяемых в различных проектах шаблонов проектирования и, следовательно, находить проекты со схожими архитектурными решениями и подходами к реализации программных подсистем АС. На рисунке 2.10 в качестве примера приведен шаблон

проектирования, который называется «Делегирование» [25].

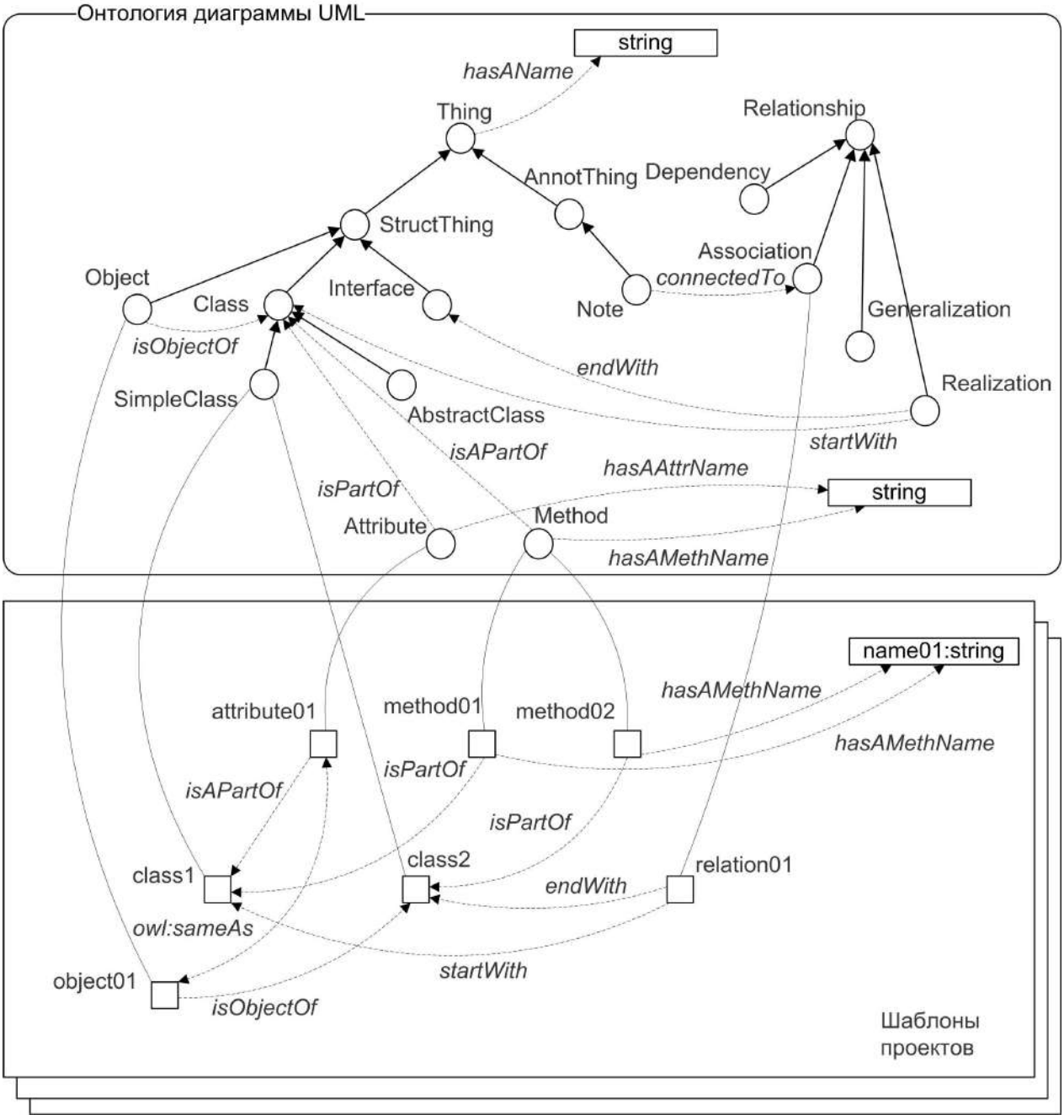


Рис. 2.10. Структура онтологии проектных диаграмм (включая пример шаблона проектирования)

Формально, онтологию проектных диаграмм представим как множество:

$$O^{prj} = \{O^{prj}_{dc}, O^{prj}_{tmp_1}, O^{prj}_{tmp_2}, \ldots, R^{prj}, F^{prj}\}, \tag{2.6}$$

где O^{prj}_{dc} – онтология диаграммы UML (в работе применяется диаграмма классов), $O^{prj}_{tmp_i}$ – онтологическое представление i-го шаблона проектирования программных систем; R^{prj} – отношение, связывающее концепт из O^{prj}_{dc} и экземпляр,

принадлежащий $O^{prj}_{tmp_i}$; F^{prj} – функция интерпретации, устанавливающая соответствия экземпляров из $O^{prj}_{tmp_i}$ и классов O^{prj}_{dc}.

Рассмотрим основные компоненты онтологии диаграммы UML:

$$O^{prj}_{dc} = \langle C^{prj}, R^{prj}, F^{prj} \rangle,$$

где $C^{prj} = \{c^{prj}_1, \ldots, c^{prj}_l\}$ – множество понятий, определяющих основные элементы диаграмм классов языка моделирования UML (например, такие как: «Thing», «Class», «Object», «Interface», «Relationship» и другие); R^{prj} – множество отношений между понятиями, позволяющие формировать онтологические представления проектных диаграмм, соблюдая соответствующие нотации (например, отношения: определение наименования элемента диаграммы строкового типа «hasAName», отношение наследования «isA», отношение, связывающее класс с объектом этого класса «isObjectOf» и другие); F^{prj} – множество функций интерпретации, определенных на отношениях R^{prj}.

Онтологическое представление i-го шаблона проектирования программных систем:

$$O^{prj}_{tmp_i} = \{instance(c^{prj}_1), \ldots, instance(r^{prj}_1), \ldots, r_{sameAs}\}.$$

Фактически онтологическое представление отдельно взятого шаблона проектирования представляет собой множество экземпляров понятий и отношений из онтологии проектных диаграмм с добавлением отношения r_{sameAs}, которое представляет собой встроенное в язык описания онтологий OWL отношение «owl:sameAs». Если указанное отношение связывает два экземпляра онтологического представления шаблона проектирования, то эти экземпляры считаются одним и тем же объектом.

На рисунке 2.10 отношением «owl:sameAs» связываются *attribute01* и *object01*. Данный факт означает, что указанный объект некоторого класса (*class2*) является атрибутом другого класса (*class1*).

2.3.7. Интегрированная модель системы онтологий ИПР

Рассмотренные ранее модели онтологий ориентированы на решение достаточно широкого спектра задач в контексте проектирования сложных АС – от содержательной интерпретации тенденций показателей технических временных рядов до выполнения семантических проектных запросов с целью нахождения аналогов проектных решений.

Интегрированная модель онтологий образует семантическое ядро интеллектуального проектного репозитория САПР АС и формально будем записывать ее в виде кортежа:

$$O = \langle O^{dom}, O^{tz}, O^{lc}, O^{cn}, O^{ts}, O^{prj}, R, F \rangle, \tag{2.7}$$

где $O^{dom}, O^{tz}, O^{lc}, O^{cn}, O^{ts}, O^{prj}$ – рассмотренные ранее в предыдущих разделах онтологии в составе интегрированной модели системы онтологий ИПР; R – множество отношений между концептами и/или экземплярами, относящимися к различным онтологиям (рисунок 2.11); F – множество функций интерпретации, заданное на множестве R.

Множество отношений интегрированной онтологии ИПР O будем записывать следующим образом:

$$R = \{R_{Pt}, R_{con}, R_{rep}, R_{sameAs}\},$$

где R_{Pt} – отношение «isAPartOf», связывающее экземпляр проектного шаблона онтологии O^{prj} и документ электронного архива d; R_{con} – отношение «connectedTo» между стадией жизненного цикла St онтологии O^{lc} и типом технического документа td электронного архива; R_{rep} – отношение вида «representsA», позволяющее связывать номинал ct тезауруса проектной организации с понятие предметной области онтологии O^{dom}; отношение R_{sameAs} вида «owl: sameAs» позволяет связать понятие онтологии предметной области c с понятием концептуальной сети проекта cs и с понятием онтологии анализа технических временных рядов v.

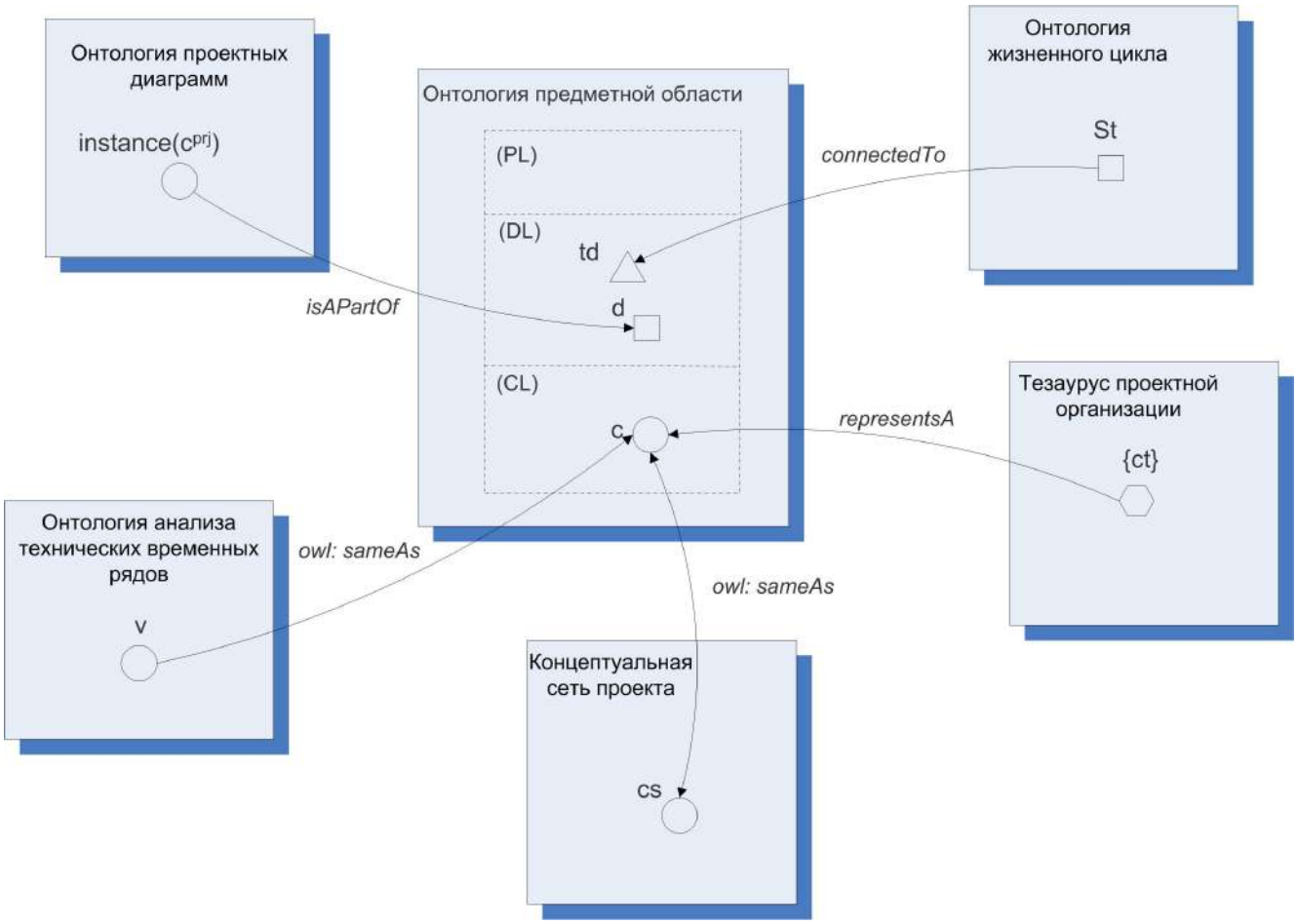


Рис. 2.11. Схема интеграции онтологий

Рассмотрим подробнее использование множества R_{con} анализа состояния текущего проекта электронного архива. Запись вида $St_i R_{con}\{td_j, td_k\}$ означает, что в качестве результатов стадии проектирования St_i понимаются документальные артефакты типов td_j и td_k. Более краткая запись будет иметь следующий вид: $St_i|td_j, td_k$.

Представим множество стадий проектирования АС St и множество возможных классов артефактов Td в виде четкого соответствия

$$\Gamma_{St/Td} = (St, Td, F_{St/Td}).$$

Данное соответствие запишем в виде матрицы инциденций $R_{\Gamma_{St/Td}}$, где строки составляют элементы $St_i \in St$, $i \in I = \{1, 2, \ldots, n\}$, а столбцы – элементы $td_j \in Td$, $j \in J = \{1, 2, \ldots, m\}$. На пересечении St_i строки и td_j столбца определяется элемент $r_{ij} = \gamma_{F_{St/Td}} < St_i, td_j >$, где $\gamma_{F_{St/Td}} = \{0, 1\}$ – характеристическая

функция, определяющая, существует ли отношение $St_i|td_j$: $\gamma_{F_{St/Td}} = 1$. Для отсутствующего отношения справедлива запись: $\gamma_{F_{St/Td}} = 0$.

Определим новое понятие *текущее состояние проекта* P. Формально, текущее состояние проекта есть множество типов документов:

$$Sm_t^P = \{td_1, td_2, \ldots, td_\lambda\},$$

где t – текущий момент времени, $td \in Td$ – тип технического документа, λ – количество типов документальных артефактов.

Введем интерпретирующую функцию $F_{tdst} : \{td\}_i \to St_i$ – функция, сопоставляющая множеству типов документов определенную стадию или проектную процедуру.

Алгоритм нахождения текущей стадии жизненного цикла АС, учитывая состояние проекта в некоторый момент времени t:

Шаг 1. Нахождение множества документов

$$D_t^P = \{d_1^P, d_2^P, \ldots, d_m^P\}_t$$

в электронном архиве проектной организации в момент времени t.

Шаг 2. Формирование представления о состоянии проекта P:

$$Sm_t^P = \{td_1, td_2, \ldots, td_l\}_t^P,$$

с использованием функции отображения D_t^P на множество возможных типов документов архива

$$\{td_1, td_2, \ldots, td_\lambda\}.$$

Шаг 3. Формирование множества $\{St\}_t^P$, элементы которого удовлетворяют условию:

$$St_{i(j)(k)}|\{td\} \supset \{td_1, td_2, \ldots, td_l\}_t^P.$$

Шаг 4. Используя отношения R_B^{ls} (из онтологии жизненного цикла) и R_{tdst}, определяется такая стадия или проектная процедура $St_{i(j)(k)}$, для которой не

является истинным следующее условие:

$$\forall St_{i(j)(k)}, St_{x(y)(z)} \in \{St\}_t^P : St_{i(j)(k)} R_B^{lc} St_{x(y)(z)}.$$

Шаг 5. Для найденной $St_{i(j)(k)}$ определяется последующая стадия или проектная процедура. При этом выполняется следующее условие:

$$\exists St_t^P : St_{i(j)(k)} R_B^{lc} St_t^P.$$

2.4. Формализация понятий предметной области проектной организации

В настоящее время существует большое количество определений *понятия*. Рассматривая понятие в онтологическом подходе к моделированию предметной области автоматизированного проектирования, наиболее близким можно считать определение, принятое в формальной логике [9]: «*понятие* – элементарная единица мыслительной деятельности, обладающая известной целостностью и устойчивостью и взятая в отвлечении от словесного выражения этой деятельности».

Содержание понятия развертывается с использованием специальной логической операции – *определения* [8]. На практике применяются как явные, так и неявные определения. Явные определения содержат указание на важные признаки, которые соответствуют обозначаемому предмету. Неявные определения подразделяются на: *остентивные* – раскрывающие содержание понятия через непосредственное ознакомление обучаемого с предметами, действиями и ситуациями, связанными с данным понятием, *контекстуальные* – определяющие содержание исследуемого понятия по смыслу целостного текста, образа или речи.

Поскольку в структуре онтологии предметной области множество понятий представляется связанным с множеством терминов текстовых документов и ат-

рибутов проектных диаграмм, в данном исследовании имеет смысл рассматривать возможность применения именно неявных определений понятий предметной области.

Учитывая ранее определенную структуру интегрированной онтологии ИПР определим формально понятие c_i онтологии следующим образом:

$$c_i = \langle c^i_{name}, \langle \{ct^i_1\}, \{t^i_1, \dots t^i_{s1}\}\rangle \dots, \langle \{ct^i_n\}, \{t^i_1, \dots t^i_{sn}\}\rangle\rangle. \quad (2.8)$$

Согласно выражению (2.8) каждое i-е понятие онтологии предметной области состоит из наименования понятия c^i_{name}, непустого множества номиналов с соответствующими терминами:

$$\{\langle \{ct^i_1\}, \{\dots\}\rangle \dots, \langle \{ct^i_n\}, \{\dots\}\rangle\}.$$

В лингвистических онтологиях для каждого понятия ставится в соответствие большое количество различных слов и выражений, значения которых связаны с данным понятием. Создаваемые таким образом языковые конструкции называются *текстовыми входами* понятия или терминами онтологии [28]. Согласно [10], [26], [104], «термин – это такое слово или словосочетание, которое способно выполнять функцию номинирования понятия определенной предметной области».

Если таксономию понятий в онтологии еще можно построить вручную (однако для широкой предметной области решение этой задачи становится весьма трудоемким процессом), то построение текстовых входов понятий необходимо автоматизировать. Далее рассматривается способ формирования текстовых входов понятий онтологии предметной области на основе имеющихся в распоряжении экспертов текстовых источников (специально подготовленных текстов), в которых представлены описания понятий предметной области.

Дадим определение *текстовому входу* понятия c_i онтологии предметной области автоматизированного проектирования.

Определение 3. Под *текстовым входом* понятия предметной области автоматизированного проектирования будем понимать множество терминов (слов), извлеченных из документальных баз проектной организации, которые семантически наиболее тесно связаны с данным понятием.

Феномен текстового входа понятия онтологии является важным по причине того, что именно посредством его анализа имеется возможность в онтологии переходить с уровня терминов на уровень понятий в процессе онтологического анализа технических документов. Особое внимание необходимо уделить вопросам обеспечения качества текстовых входов понятий, т. к. наличие ошибок на данном этапе формирования онтологии влечет снижение качества получаемых результатов анализа информационных ресурсов. Для построения текстовых входов понятий онтологии необходимо решить две задачи:

1. Определение метрики семантического расстояния между терминами, извлекаемыми из корпуса заранее подготовленных текстов.
2. Определение подмножества терминов для каждого понятия онтологии, которое включает только термины, образующие компактные группы.

Способ вычисления семантического расстояния между терминами в технических документах основывается на идее вычисления коэффициента семантической близости слов текста, представленной в работе [180]. Данный коэффициент определяется посредством нахождения расстояния между словами в текстовых документах.

В документе электронного архива отношение между терминами должно определяться тем, насколько удалены термины относительно друг друга по тексту документа. Данная компонента расстояния является внутритекстовой и зависит только от лексических и грамматических особенностей конкретного текстового документа. Если тема документа повторяется в нескольких абзацах, то степень ее важности для анализа текстовой информации возрастает.

Для внесения в метрику семантического расстояния специфики проектной деятельности необходимо построить набор словарей $L_p = \{L_{p_1}, L_{p_2}, \dots L_{p_n}\}$, где каждый из L_{p_i} представляет собой словарь терминов, извлекаемых из документов проекта p_i. Кроме того, будем применять словари, сформированные из применяемых в организации стандартов: $L_{st} = \{L_{st_1}, L_{st_2}, \dots L_{st_k}\}$, где L_{st_i} – словарь, построенный на основе i-го стандарта.

Семантический коэффициент отношения между номиналом понятия и термином будем определять следующим образом:

$$S\left(\{ct_j^i\}, t_k\right) = \frac{\sum_{occur(\{ct_j^i\},t_k)} \frac{1}{exp(sentence \cdot (paragraph+1))}}{num(occur(\{ct_j^i\}, t_k))} \times$$

$$\times max\left(\frac{num(prj - cooccur(\{ct_j^i\}, t_k))}{num(totalprj)}, \frac{num(stnd - cooccur(\{ct_j^i\}, t_k))}{num(totalstnd)}\right), \quad (2.9)$$

где $\{ct_j^i\}, t_k$ – j-й номинал i-го понятия онтологии и k-й термин соответственно; $sentence$ – расстояние, выраженное в количестве предложений между номиналом и термином; $paragraph$ – расстояние, выраженное в количестве абзацев между номиналом и термином; $num(occur(\{ct_j^i\}, t_k))$ – количество совпадений $\{ct_j^i\}$ и t_k; $num(prj - cooccur(\{ct_j^i\}, t_k)$ – количество словарей проектов, где существует совместная встречаемость $\{ct_j^i\}$ и t_k; $num(totalprj)$ – число словарей проектов; $num(stnd - cooccur(\{ct_j^i\}, t_k)$ – количество словарей стандартов, где существует совместная встречаемость $\{ct_j^i\}$ и t_k; $num(totalstnd)$ – число словарей стандартов.

Семантическое расстояние между понятием c_i и термином t_k определяется выражением:

$$\Delta_{sem}(c_i, t_k) = 1 - \max_j(\bar{S}(\{ct_j^i\}, t_k)), \quad (2.10)$$

где $\bar{S}(\{ct_j^i\}, t_k)$ – нормализованный семантический коэффициент отношения между номиналом и термином (приведенный к отрезку [0,1]).

После определения семантических расстояний между исследуемым понятием, для которого формируется текстовый вход, и терминами документа необхо-

димо определить термины, которое находятся на минимальном семантическом расстоянии от понятия. Экспериментальные исследования с определением меры семантического расстояния между терминами технических документов проектной организации, выполненные в рамках диссертации, показали, что, во-первых, необходимо выполнять редукцию терминов, связанных с понятием; во-вторых, распределение терминов по оси меры расстояния не является равномерным (рисунок 2.12).

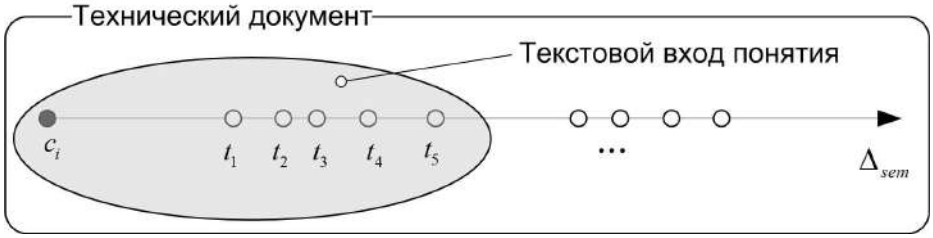


Рис. 2.12. Компактное подмножество терминов понятия онтологии

Учитывая вышесказанное, в данной работе функцию формирования текстового входа предлагается основывать на гипотезе λ-компактности [31], которая опирается на понятие λ-расстояния, учитывающего нормированное расстояние d между терминами и показатель τ локальной плотности терминов.

Если имеется возможность определить значения семантических коэффициентов между парами терминов (согласно выражению (2.10), где вместо номинала записывается обычный термин), которые являются претендентами для включения в текстовый вход, то можно построить граф, вершинами которого являются термины (включая номинал, связанный с исследуемым понятием онтологии), и найти самое длинное ребро – диаметр графа (D). Выделим номинал $\{ct_j^i\}$ и термин t_k. Обозначим длину ребра, связывающего номинал с термином, через $\alpha(\{ct_j^i\}, t_k)$. Нормированное расстояние между этими терминами определим как величину $d = \frac{\alpha}{D}$.

Теперь находится самое короткое ребро среди всех ребер, которые являются смежными ребру $(\{ct_j^i\}, t_k)$. Его длину обозначим через β_{min}. Отношение всех длин смежных ребер обозначим как $\tau^* = \frac{\alpha}{\beta_{min}}$. Для того, чтобы указанную

величину сделать нормированной, в полном графе находится наибольшее значение τ_{max}. Найденная величина $\tau = \frac{\tau^*}{\tau_{max}}$ характеризует степень локальной неоднородности плотности терминов онтологии относительно $\{ct_j^i\}$ и t_k. Величину $\lambda = f(\tau, d)$ назовем λ-расстоянием между $\{ct_j^i\}$ и t_k. Основываясь на результатах исследования, представленных в работе [31], в качестве меры расстояния будем применять величину $\lambda = \tau^2 \times d$.

Для нахождения группы терминов, включаемых в текстовый вход понятий онтологии необходимо определить такое ребро (t_i, t_j), которому соответствовала бы граница, разделяющая термины, относящиеся к выбранному понятию онтологии, и термины, которые не включаются в текстовый вход данного понятия. Применяя алгоритм λ-KRAB ([31]) определяемый критерий, характеризующий качество такого разделения терминов, выражается величиной:

$$F = h^4 \tau^2 d \to max.$$

При этом:

$$h = 2 \cdot \frac{m^+}{m} \cdot \frac{m^-}{m},$$

где m^+ – число терминов, включенных в текстовый вход понятия; m^- – число остальных терминов.

Каждый текстовый вход W^k понятия c_k представим следующим образом:

$$W^k = \{(t_1^k, w_1^k), (t_2^k, w_2^k), \ldots, (t_i^k, w_i^k), \ldots, (t_{l_k}^k, w_{l_k}^k)\}, \tag{2.11}$$

где t_i^k – i-й термин k-го понятия онтологии; l_k – общее количество терминов, которые наиболее близки с k-м понятием; w_i^k – нормализованный вес i-го термина в текстовом входе k-го понятия (семантическое расстояние между термином и понятием в пределах одного текстового входа).

Здесь следует отметить тот факт, что, определив текстовые входы (термины) для ряда понятий $c_{i1}, c_{i2}, \ldots, c_{il_i}$, которые являются дочерними относительно понятия c_i, нельзя сделать вывод о том, что определен текстовый вход для

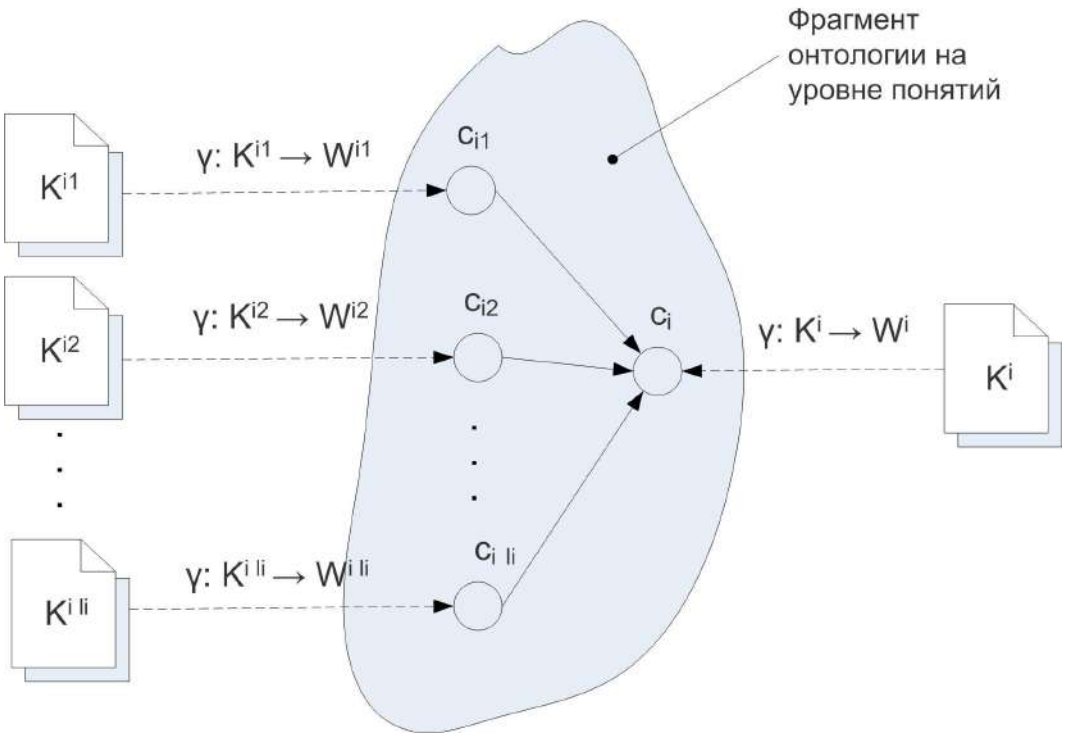


Рис. 2.13. Процесс формирования текстовых входов понятий онтологии

понятия c_i. Поэтому для родительского понятия c_i также формируется набор соответствующих ему терминов (рисунок 2.13). Причина такого подхода кроется в особенностях естественного языка, а именно, в его свойстве несогласованности данных (нарушении транзитивности). На рисунке 2.14 представлен алгоритм формирования текстовых входов понятий онтологии предметной области.

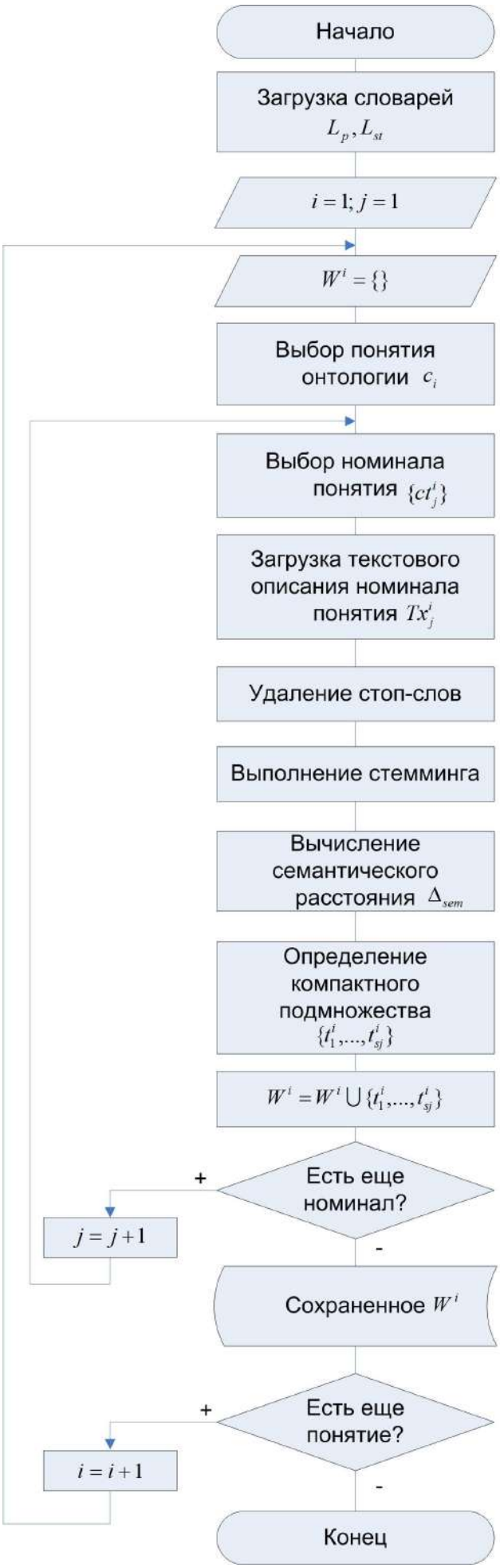


Рис. 2.14. Алгоритм формирования текстовых входов

2.5. Метод оценивания качества онтологии на основе нечетких соответствий

Рассмотренный подход к формированию текстовых входов понятий онтологии, с одной стороны, позволяет сократить трудоемкость рассматриваемой задачи, а с другой стороны, требует оценки качества ее решения. Поскольку, в общем случае, текстовый вход для одного понятия может основываться на нескольких документах, то возникает вопрос, касающийся оптимального набора таких документальных источников. Интуитивно понятно, что просто увеличением мощности текстового входа понятия онтологии не всегда возможно повышение качества описания такого понятия. Более того, возможна и обратная ситуация, когда добавление текстовых источников приведет к снижению качества текстового входа.

Формализацию оценки качества онтологии будем производить относительно выделенных фрагментов онтологии предметной области.

> **Определение 4.** *Группа однородных понятий* – это такое подмножество понятий онтологии, которые подчинены какому-либо одному понятию (являются дочерними понятиями) или находящихся на самом верхнем уровне иерархии понятий онтологии.

На рисунке 2.15 представлен иллюстративный пример групп однородных понятий онтологии предметной области. Такая декомпозиция метауровня понятий онтологии, с одной стороны, позволяет выделить в ней «плохие» и «хорошие» фрагменты, а с другой стороны, применить для формализации математическое моделирование. Для описания фрагментов онтологии будем использовать математический аппарат нечетких соответствий в терминах решения задачи оценивания качества онтологии [167], [55].

Семантическая связь между множеством терминов T и множеством понятий онтологии C формально можно представить как нечеткое соответствие,

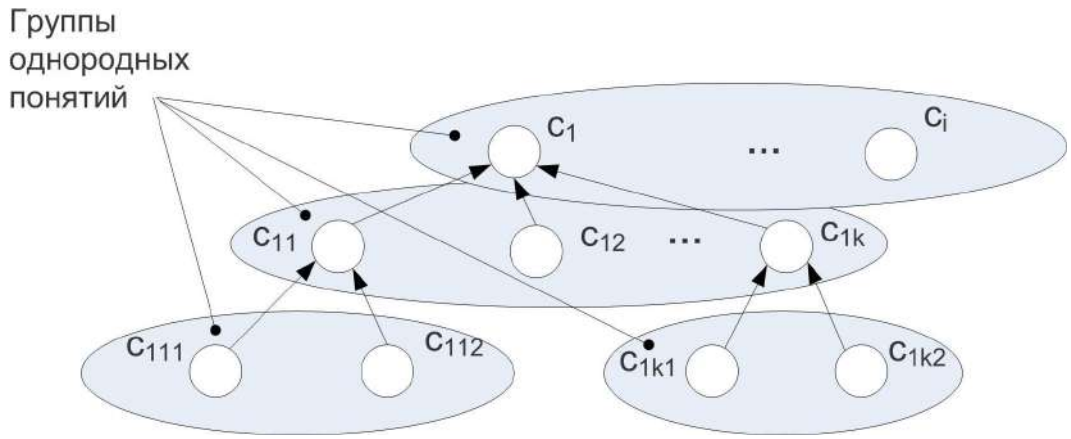


Рис. 2.15. Группы однородных понятий онтологии

которое будем обозначать через тройку множеств $\tilde{\Gamma}_{TC} = (T, C, \tilde{F}_{TC})$, где $\tilde{F}_{TC}$ – нечеткое множество в $T \times C$. Множество W есть область отправления, множество C – область прибытия, а $\tilde{F}_{TC}$ – нечеткий график нечеткого соответствия.

Нечеткое соответствие будем задавать в виде ориентированного графа с множеством вершин $T \cup C$. Каждая дуга $< t_k, c_j >$ данного графа которого соответствует значение функции принадлежности $\mu_{\tilde{F}_{TC}} < t_k, c_j >$. Значение $\mu_{\tilde{F}_{TC}} < t_k, c_j >$ определяется, принимая во внимание семантическое расстояние (2.10):

$$\mu_{\tilde{F}_{TC}} < t_k, c_j >= \underset{j}{max}(\bar{S}(\{ct_j^i\}, t_k)). \tag{2.12}$$

Определим образ множества терминов T при соответствии $\tilde{\Gamma}_{TC}$ как нечеткое множество $\tilde{\Gamma}(T)$ во множестве понятий C в виде следующего выражения:

$$\tilde{\Gamma}(T) = \{< \mu_{\tilde{\Gamma}(T)}(c), c > | c \in C\}, \tag{2.13}$$

где $\mu_{\tilde{\Gamma}(T)}(c) = \underset{t \in T}{\vee} \mu_{\tilde{F}_{TC}} < t, c >$.

Прообразом множества понятий C при соответствии $\tilde{\Gamma}_{TC}$ будем называть нечеткое множество $\tilde{\Gamma}^{-1}(C)$ во множестве терминов T, определяемом следующим выражением:

$$\tilde{\Gamma}^{-1}(C) = \{< \mu_{\tilde{\Gamma}^{-1}(C)}(t), t > | t \in T\}, \tag{2.14}$$

где $\mu_{\tilde{\Gamma}^{-1}(C)}(t) = \underset{c \in C}{\vee}(\mu_{\tilde{F}_{TC}} < t, c >)$.

Предлагаемая методика лексического описания понятий онтологии предметной области автоматизированного проектирования будет включать следующие этапы.

1. Формирование таксономии понятий предметной области проектной организации.
2. Определение набора текстовых документов для каждого понятия онтологии, на основе которого формируются текстовые входы понятий.
3. Формирование первоначального состава текстовых входов понятий (уровень терминов в онтологии).
4. Оптимизация набора текстовых документов, определяющих понятия, в рамках каждой группы однородных понятий онтологии.
5. Уточнение текстовых входов понятий, используя результаты оптимизации на предыдущем этапе.

В основу формального критерия качества онтологии положим свойства нечетких соответствий, представленных в работах [5], [6]: нечеткая функциональность, нечеткая инъективность и нечеткая всюду определенность.

Степень нечеткой функциональности фрагмента онтологии будем определять по формуле:

$$\beta(\tilde{\Gamma}_{TC})_{fon} = 1 - \alpha(\tilde{\Gamma}_{TC})_{fon}, \tag{2.15}$$

где $\alpha(\tilde{\Gamma}_{TC})_{fon} = \frac{1}{C^2_{|C|}} \sum\limits_{c_i, c_j \in C} (\frac{1}{|T|} \sum\limits_{t \in T} (\mu_{\tilde{\Gamma}^{-1}(c_i)}(t) \& \mu_{\tilde{\Gamma}^{-1}(c_j)}(t)))$.

Здесь под $|C|$ понимается количество понятий в группе однородных понятий онтологии (мощность множества C), под $|T|$ – количество терминов, ассоциированных с понятиями и под $C^2_{|C|}$ – число сочетаний из $|C|$ по два, соответствующее количеству всевозможных пар концептов.

Согласно выражению (2.15), качество фрагмента онтологии будет тем выше, чем больше значение нечеткой функциональности. Действительно, если у каждого концепта онтологии будут такие текстовые входы, которые мало пересекаются (имеют небольшое количество общих терминов), то такой фрагмент

онтологии будет считаться более правильным, чем в случае обнаружения одних и тех же терминов с в различных текстовых входах.

Степень неинъективности фрагмента онтологии формально будем представлять в следующем виде:

$$\alpha(\tilde{\Gamma}_{TC})_{inj} = \frac{1}{C^2_{|T|}} \sum_{t_i,t_j \in T} (\frac{1}{|C|} \sum_{c \in C} (\mu_{\tilde{\Gamma}(t_i)}(c) \& \mu_{\tilde{\Gamma}(t_j)}(c))). \quad (2.16)$$

Соответствующая ей степень инъективности: $\beta(\tilde{\Gamma}_{TC})_{inj} = 1 - \alpha(\tilde{\Gamma}_{TC})_{inj}$. Содержательно степень инъективности онтологии показывает встречаемость разных терминов в одном текстовом входе. Причем вычисление такой встречаемости выполняется попарно по всем терминам. Чем больше ассоциаций у понятия с различными терминами и выше вес таких ассоциаций, тем больше степень неинъективности и, соответственно, меньше степень инъективности.

Степень всюду определенности фрагмента онтологии будем вычислять по следующей формуле:

$$\beta(\tilde{\Gamma}_{TC})_{def} = \frac{1}{|T|} \sum_{t \in T} (\frac{1}{|C|} \sum_{c \in \Gamma(t)} \mu_{\tilde{\Gamma}(t)}(c)). \quad (2.17)$$

Соответствующая ей степень не всюду определенности: $\alpha(\tilde{\Gamma}_{TC})_{def} = 1 - \beta(\tilde{\Gamma}_{TC})_{def}$. Смысловое содержание показателя качества онтологии (2.17) заключается в том, что в чем большее количество текстовых входов входит каждый термин онтологии и чем выше весовые коэффициенты таких вхождений (близость термина к понятию в семантическом смысле), тем больше значение степени всюду определенности фрагмента онтологии. Если каждый термин фрагмента онтологии ассоциирован с каждым понятием, входящим в группу однородных понятий, и веса таких ассоциаций равны 1, то степень всюду определенности онтологии будет равна 1.

Произведем оценивание качества онтологии по вышеприведенным показателям, принимая во внимание следующие иллюстративные виды нечетких соответствий, описывающие различные фрагменты онтологий (рисунок 2.16):

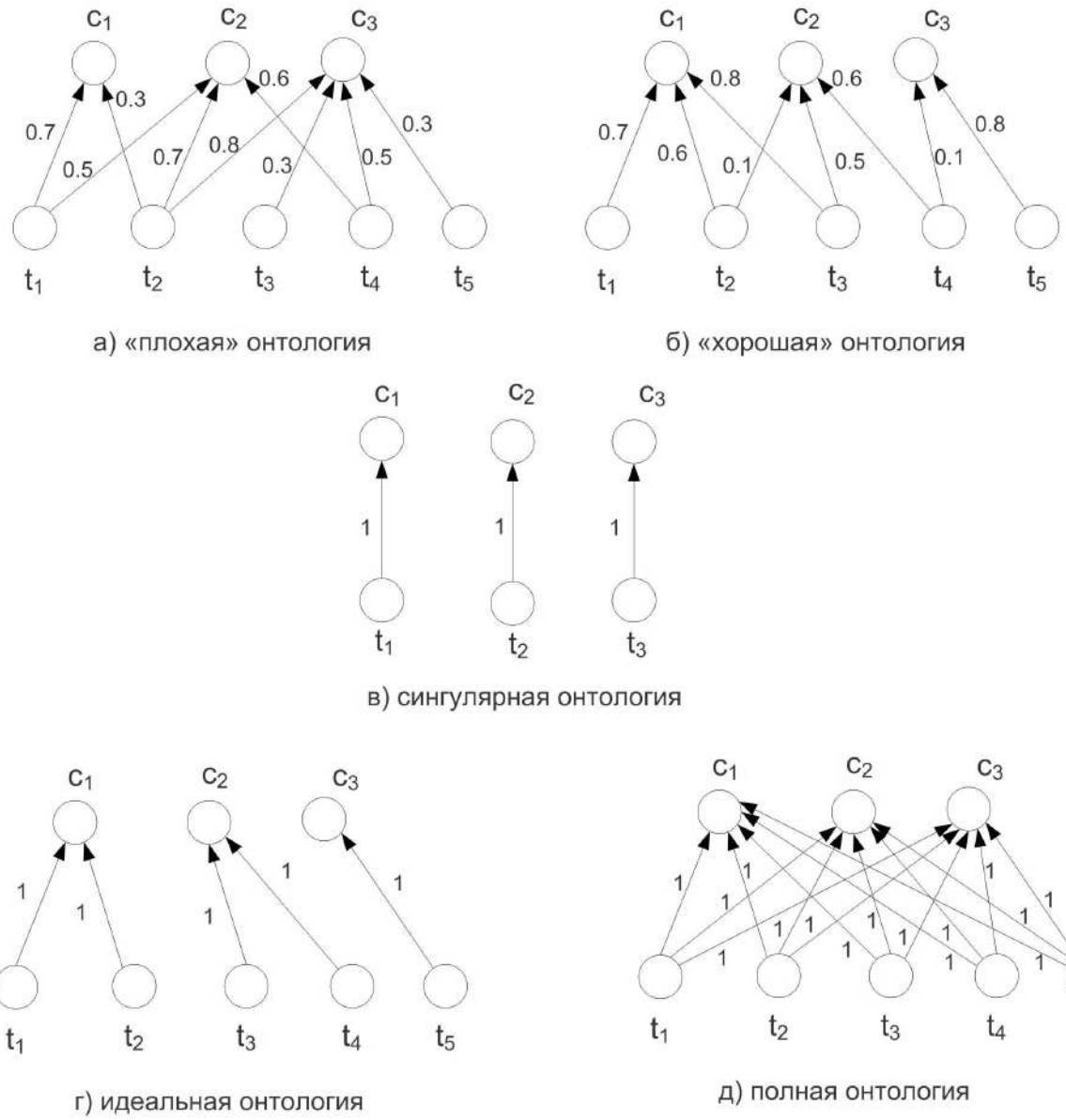


Рис. 2.16. Иллюстративные примеры нечетких соответствий для представления фрагментов онтологии

«плохая» онтология; «хорошая» онтология; сингулярная онтология; идеальная онтология и полная онтология. Под «плохой» онтологией будем понимать такую онтологию, у которой есть значительные пересечения текстовых входов концептов. У «хорошей» онтологии такие пересечения незначительны. Сингулярной онтологией будем называть гипотетическую онтологию, у которой текстовые входы концептов состоят из единственного термина. Такой термин по наименованию совпадает с соответствующим концептом (или номиналом, представляющим концепт). Идеальная онтология не содержит пересечений текстовых входов, а полная онтология предполагает такие ассоциации между концептами и терминами, при которых каждый термин входит во все текстовые входы с единичным весом.

Вычислим значения показателей качества фрагментов онтологий, приведенных на рисунке 2.16.

«Плохая» онтология (рисунок 2.16, а)

- *Степень нефункциональности*

  Используя формулу (2.14), получаем следующие множества:

$$\tilde{\Gamma}^{-1}(c_1) = \{0,7/t_1, 0,3/t_2\}, \tilde{\Gamma}^{-1}(c_2) = \{0,5/t_1, 0,7/t_2, 0,6/t_4\},$$

$$\tilde{\Gamma}^{-1}(c_3) = \{0,8/t_2, 0,3/t_3, 0,5/t_4, 0,3/t_5\}.$$

  Согласно выражению (2.15) получаем значение степени нефункциональности:

$$\alpha(\tilde{\Gamma}_{TC})_{fon} = 0,15.$$

- *Степень функциональности*

$$\beta(\tilde{\Gamma}_{TC})_{fon} = 1 - \alpha(\tilde{\Gamma}_{TC})_{fon} = 0,85.$$

- *Степень неинъективности*

  Используя формулу (2.13), получаем следующие множества:

$$\tilde{\Gamma}(t_1) = \{0,7/c_1, 0,5/c_2\}, \tilde{\Gamma}(t_2) = \{0,3/c_1, 0,7/c_2, 0,8/c_3\},$$

$$\tilde{\Gamma}(t_3) = \{0,3/c_3\}, \tilde{\Gamma}(t_4) = \{0,6/c_2, 0,5/c_3\}, \tilde{\Gamma}(t_5) = \{0,3/c_3\}.$$

Согласно выражению (2.16) получаем значение степени неинъективности:

$$\alpha(\tilde{\Gamma}_{TC})_{inj} = 0,13.$$

- *Степень инъективности*

$$\beta(\tilde{\Gamma}_{TC})_{inj} = 1 - \alpha(\tilde{\Gamma}_{TC})_{inj} = 0,87.$$

- *Степень всюду определенности*

Используя выражение (2.17), получаем:

$$\beta(\tilde{\Gamma}_{TC})_{def} \approx 0,313.$$

- *Степень не всюду определенности*

$$\alpha(\tilde{\Gamma}_{TC})_{def} = 1 - \beta(\tilde{\Gamma}_{TC})_{def} \approx 0,687.$$

«Хорошая» онтология (рисунок 2.16, б)

- *Степень нефункциональности*

$$\tilde{\Gamma}^{-1}(c_1) = \{0,7/t_1, 0,6/t_2, 0,8/t_3\},$$

$$\tilde{\Gamma}^{-1}(c_2) = \{0,1/t_1, 0,5/t_3, 0,6/t_4\}, \tilde{\Gamma}^{-1}(c_3) = \{0,1/t_4, 0,8/t_5\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{fon} \approx 0,047.$$

- *Степень функциональности*

$$\beta(\tilde{\Gamma}_{TC})_{fon} = 1 - \alpha(\tilde{\Gamma}_{TC})_{fon} \approx 0,953.$$

- *Степень неинъективности*

$$\tilde{\Gamma}(t_1) = \{0,7/c_1\}, \tilde{\Gamma}(t_2) = \{0,6/c_1, 0,1/c_2\},$$

$$\tilde{\Gamma}(t_3) = \{0,8/c_1, 0,5/c_2\}, \tilde{\Gamma}(t_4) = \{0,6/c_2, 0,1/c_3\},$$

$$\tilde{\Gamma}(t_5) = \{0,8/c_3\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{inj} \approx 0,09.$$

- *Степень инъективности*

$$\beta(\tilde{\Gamma}_{TC})_{inj} = 1 - \alpha(\tilde{\Gamma}_{TC})_{inj} \approx 0,91.$$

- *Степень всюду определенности*

$$\beta(\tilde{\Gamma}_{TC})_{def} \approx 0,28.$$

- *Степень не всюду определенности*

$$\alpha(\tilde{\Gamma}_{TC})_{def} = 1 - \beta(\tilde{\Gamma}_{TC})_{def} \approx 0,72.$$

Сингулярная онтология (рисунок 2.16, в)

- *Степень нефункциональности*

$$\tilde{\Gamma}^{-1}(c_1) = \{1/t_1\}, \tilde{\Gamma}^{-1}(c_2) = \{1/t_2\},$$

$$\tilde{\Gamma}^{-1}(c_3) = \{1/t_3\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{fon} = 0.$$

- *Степень функциональности*

$$\beta(\tilde{\Gamma}_{TC})_{fon} = 1 - \alpha(\tilde{\Gamma}_{TC})_{fon} = 1.$$

- *Степень неинъективности*

$$\tilde{\Gamma}(t_1) = \{1/c_1\}, \tilde{\Gamma}(t_2) = \{1/c_2\},$$

$$\tilde{\Gamma}(t_3) = \{1/c_3\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{inj} = 0.$$

- *Степень инъективности*

$$\beta(\tilde{\Gamma}_{TC})_{inj} = 1 - \alpha(\tilde{\Gamma}_{TC})_{inj} = 1.$$

- *Степень всюду определенности*

$$\beta(\tilde{\Gamma}_{TC})_{def} \approx 0,33.$$

- *Степень не всюду определенности*

$$\alpha(\tilde{\Gamma}_{TC})_{def} = 1 - \beta(\tilde{\Gamma}_{TC})_{def} \approx 0,67.$$

Идеальная онтология (рисунок 2.16, г)

- *Степень нефункциональности*

$$\tilde{\Gamma}^{-1}(c_1) = \{1/t_1, 1/t_2\}, \tilde{\Gamma}^{-1}(c_2) = \{1/t_3, 1/t_4\},$$

$$\tilde{\Gamma}^{-1}(c_3) = \{1/t_5\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{fon} = 0.$$

- *Степень функциональности*

$$\beta(\tilde{\Gamma}_{TC})_{fon} = 1 - \alpha(\tilde{\Gamma}_{TC})_{fon} = 1.$$

- *Степень неинъективности*

$$\tilde{\Gamma}(t_1) = \{1/c_1\}, \tilde{\Gamma}(t_2) = \{1/c_1\},$$

$$\tilde{\Gamma}(t_3) = \{1/c_2\}, \tilde{\Gamma}(t_4) = \{1/c_2\},$$

$$\tilde{\Gamma}(t_5) = \{1/c_3\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{inj} \approx 0,067.$$

- *Степень инъективности*

$$\beta(\tilde{\Gamma}_{TC})_{inj} = 1 - \alpha(\tilde{\Gamma}_{TC})_{inj} \approx 0,933.$$

- *Степень всюду определенности*

$$\beta(\tilde{\Gamma}_{TC})_{def} \approx 0,333.$$

- *Степень не всюду определенности*

$$\alpha(\tilde{\Gamma}_{TC})_{def} = 1 - \beta(\tilde{\Gamma}_{TC})_{def} \approx 0,667.$$

Полная онтология (рисунок 2.16, д)

- *Степень нефункциональности*

$$\tilde{\Gamma}^{-1}(c_1) = \{1/t_1, 1/t_2, 1/t_3, 1/t_4, 1/t_5\},$$

$$\tilde{\Gamma}^{-1}(c_2) = \{1/t_1, 1/t_2, 1/t_3, 1/t_4, 1/t_5\},$$

$$\tilde{\Gamma}^{-1}(c_3) = \{1/t_1, 1/t_2, 1/t_3, 1/t_4, 1/t_5\},$$

$$\tilde{\Gamma}^{-1}(c_4) = \{1/t_1, 1/t_2, 1/t_3, 1/t_4, 1/t_5\},$$

$$\tilde{\Gamma}^{-1}(c_5) = \{1/t_1, 1/t_2, 1/t_3, 1/t_4, 1/t_5\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{fon} = 1.$$

- *Степень функциональности*

$$\beta(\tilde{\Gamma}_{TC})_{fon} = 1 - \alpha(\tilde{\Gamma}_{TC})_{fon} = 0.$$

- *Степень неинъективности*

$$\tilde{\Gamma}(t_1) = \{1/c_1, 1/c_2, 1/c_3\}, \tilde{\Gamma}(t_2) = \{1/c_1, 1/c_2, 1/c_3\},$$

$$\tilde{\Gamma}(t_3) = \{1/c_1, 1/c_2, 1/c_3\}, \tilde{\Gamma}(t_4) = \{1/c_1, 1/c_2, 1/c_3\},$$

$$\tilde{\Gamma}(t_5) = \{1/c_1, 1/c_2, 1/c_3\}.$$

$$\alpha(\tilde{\Gamma}_{TC})_{inj} = 1.$$

- *Степень инъективности*

$$\beta(\tilde{\Gamma}_{TC})_{inj} = 1 - \alpha(\tilde{\Gamma}_{TC})_{inj} = 0.$$

- *Степень всюду определенности*

$$\beta(\tilde{\Gamma}_{TC})_{def} = 1.$$

- *Степень не всюду определенности*

$$\alpha(\tilde{\Gamma}_{TC})_{def} = 1 - \beta(\tilde{\Gamma}_{TC})_{def} = 0.$$

Таблица 2.1. Значения показателей качества онтологии

Показатель	**«Плох.» онт.**	**«Хор.» онт.**	**Синг. онт.**	**Идеал. онт.**	**Полн. онт.**	**Тенденция**
$\alpha(\tilde{\Gamma}_{TC})_{fon}$	0,15	0,047	0	0	1	$\downarrow$
$\beta(\tilde{\Gamma}_{TC})_{fon}$	0,85	0,953	1	1	0	$\uparrow$
$\alpha(\tilde{\Gamma}_{TC})_{inj}$	0,13	0,09	0	0,067	1	$\downarrow$
$\beta(\tilde{\Gamma}_{TC})_{inj}$	0,87	0,91	1	0,933	0	$\uparrow$
$\alpha(\tilde{\Gamma}_{TC})_{def}$	0,687	0,72	0,67	0,667	0	$\uparrow\downarrow$
$\beta(\tilde{\Gamma}_{TC})_{def}$	0,313	0,28	0,33	0,333	1	$\uparrow\downarrow$

Сводный результат вычислений показателей качества фрагментов онтологий представлен в таблице 2.1.

Интегральным критерием качества фрагмента онтологии будем считать следующий показатель:

$$\tilde{\delta}_{TC} = 0.3 \cdot \beta(\tilde{\Gamma}_{TC})_{fon} + 0.2 \cdot \beta(\tilde{\Gamma}_{TC})_{inj} + 0.5 \cdot \alpha(\tilde{\Gamma}_{TC})_{def} \rightarrow max, \tilde{\delta}_{TC} \in [0,1]. \quad (2.18)$$

Принцип оптимальности текстовых входов понятий онтологии предметной области будет иметь следующую формулировку: множество документов $\hat{K} \subset K$ будет оптимальным с точки зрения формирования текстовых входов понятий, принадлежащих группе однородных понятий, тогда, когда целевая функция (2.18) принимает свое максимальное значение.

Сформулированный принцип оптимальности находит свое применение в двух случаях:

- на этапе создания онтологии, когда аналитик при определении лингвистической части онтологии для каждого понятия формирует первоначальный набор текстовых документов, описывающих понятия;
- на этапе модификации лингвистической части онтологии, когда нужно принять решение о целесообразности добавления или удаления документа, определяющего текстовый вход какого-либо понятия.

2.6. Логическое представление онтологии интеллектуального проектного репозитория

В обзорной главе были рассмотрены различные языки представления прикладных онтологий. С учетом поддержки на уровне стандартов консорциумом W3C группы языков OWL (OWL Light, OWL DL, OWL Full) будем в качестве логической основы языка описания онтологии ИПР рассматривать формализм $\mathcal{SHOIN}(\mathcal{D})$ [117], [126], [155]. Дескриптивная логика $\mathcal{SHOIN}(\mathcal{D})$ обладает достаточными выразительными возможностями для представления фрагментов модели предметной области, относящихся к верхним метауровням онтологии, структура которой приведена на рисунке 2.5. На метауровне понятий и атомарном метауровне данного формализма уже не достаточно для представления неполной информации о свойствах концептов. Особенности естественного языка и неполнота в описании классов, сущностей и отношений между ними в проектных диаграммах требуют использования формализмов, способных работать с нечеткой, неполной информацией[89]. Одним из расширений $\mathcal{SHOIN}(\mathcal{D})$ является формализм $fuzzy\mathcal{SHOIN}(\mathcal{D})$, сочетающий языковые возможности базовой дескриптивной логики и развитый математический аппарат теории нечетких множеств (fuzzy sets theory) [188], [189].

В контексте дескрипционной логики $\mathcal{SHOIN}(\mathcal{D})$ онтология представляет собой базу знаний следующего вида [79], [146], [183]:

$$KB = \{TBox, ABox\},$$

где $TBox$ – набор терминологических аксиом, представляющие общие знания о понятиях электронного архива проектной организации и их взаимосвязях; $ABox$ – набор утверждений (фактов) об индивидах.

Принимая во внимание структуру онтологии предметной области (рисунок 2.5), тезауруса проектной организации (рисунок 2.6) и онтологию проектных диаграмм (рисунок 2.10) будем различать $TBox^{arch}$ – терминологию про-

ектного архива, $TBox^{dom}$ – терминологию предметной области проектной организации, $TBox^{prj}$ – терминологию шаблонов проектирования и $ABox^{arch}$, $ABox^{dom}$, $ABox^{prj}$ – соответствующие множества фактов:

$$TBox = TBox^{arch} \cup TBox^{dom} \cup TBox^{prj},$$
$$ABox = ABox^{arch} \cup ABox^{dom} \cup ABox^{prj}.$$

$TBox^{arch}$ (соответственно, $ABox^{arch}$) включают в себя терминологию (факты), принадлежащие метауровням проектов и документов онтологии (2.7). Метауровень понятий онтологии предметной области и тезаурус проектной организации определяют $TBox^{dom}$ и $ABox^{dom}$. Онтологию проектных диаграмм составляют $TBox^{prj}$ и $ABox^{prj}$.

Терминологию будем записывать в виде:

$$\mathcal{T} \models A \sqsubseteq B, \tag{2.19}$$
$$\mathcal{T} \models A \equiv B, \tag{2.20}$$

где A и B – концепты предметной области, причем A – атомарный концепт, а B – может представлять собой атомарный концепт или сложный концепт с применением символов $\sqcap$, $\sqcup$, $\forall R.C$, $\exists R.C$ и $\leq 1R.C$.

Выражение (2.19) определяет систему вложенности концептов вида «подконцепт $\sqsubseteq$ надконцепт», а выражение (2.20) представляет собой систему определений концептов с необходимыми и достаточными условиями. Запишем содержание терминологии $TBox^{arch}$, опираясь на структуру онтологии (рисунок 2.5).

Терминология $\mathbf{TBox^{arch}}$

$$tp_{11} \sqsubseteq tp_1 \qquad tp_1 \sqsubseteq tp$$
$$tp_{12} \sqsubseteq tp_1 \qquad tp_2 \sqsubseteq tp$$
$$tp_{21} \sqsubseteq tp_2 \qquad \vdots$$
$$tp_{22} \sqsubseteq tp_2 \qquad tp_l \sqsubseteq tp$$
$$tp \equiv \top \sqcap \leq 1 hasATypePrjName.String,$$

где $hasATypePrjName$ – наименование функциональной роли *имеет наименование типа проекта*, $String$ – конкретный домен (concrete domain) строкового типа [162], [163].

Концепт «Проект» запишем в виде:

$$P \equiv \top \sqcap \leq 1 hasAPrjName.String \sqcap \leq 1 hasADeveloperName.String \sqcap$$
$$\sqcap \exists hasAInitialDate.Date \sqcap \exists hasAType.tp,$$

где $hasAPrjName$, $hasADeveloperName$, $hasAInitialDate$, $hasAType$ – наименования ролей *имеет наименование проекта*, *имеет имя разработчика*, *имеет дату начала проекта*, *имеет тип*, соответственно; $Date$ – конкретный домен типа «Дата».

$$\begin{array}{ll} td_{11} \sqsubseteq td_1 & td_1 \sqsubseteq td \\ td_{12} \sqsubseteq td_1 & td_2 \sqsubseteq td \\ \vdots & \vdots \\ td_{n1} \sqsubseteq td_n & td_n \sqsubseteq td \\ td_{n2} \sqsubseteq td_n & \end{array}$$
$$td \equiv \top \sqcap \leq 1 hasATypeDocName.String,$$

где $hasATypeDocName$ – наименование функциональной роли *имеет наименование типа документа*.

Концепт «Документ» запишем в виде:

$$D \equiv \top \sqcap \leq 1 hasADocDecimal.String \sqcap \exists hasAAuthor.String \sqcap$$
$$\sqcap \exists hasADate.Date \sqcap \exists hasAType.td \sqcap \forall includedIn.P,$$

где $hasADocDecimal$, $hasAAuthor$, $hasADate$, $hasAType$, $includedIn$ – наименования ролей *имеет децимальный номер*, *имеет автора*, *имеет дату*, *имеет тип* и *включен в*, соответственно.

Набор фактов ABox$^{\mathbf{arch}}$

$$p_1^{11} : P \qquad \langle p_1^{11}, tp_{11} \rangle : hasAType$$
$$p_1^{12} : P \qquad \langle p_1^{12}, tp_{12} \rangle : hasAType$$
$$p_2^{12} : P \qquad \langle p_2^{12}, tp_{12} \rangle : hasAType$$
$$p_1^{22} : P \qquad \langle p_1^{22}, tp_{22} \rangle : hasAType$$

$$d_1^{11} : D \qquad \langle d_1^{11}, td_{11} \rangle : hasAType \qquad \langle d_1^{11}, p_1^{11} \rangle : includedIn$$
$$d_2^{11} : D \qquad \langle d_2^{11}, td_{11} \rangle : hasAType$$
$$d_1^{12} : D \qquad \langle d_1^{12}, td_{12} \rangle : hasAType$$
$$d_1^{2} : D \qquad \langle d_1^{2}, td_{2} \rangle : hasAType \qquad \langle d_1^{2}, p_2^{12} \rangle : includedIn$$
$$d_1^{n1} : D \qquad \langle d_1^{n1}, td_{n1} \rangle : hasAType$$
$$d_2^{n1} : D \qquad \langle d_2^{n1}, td_{n1} \rangle : hasAType$$
$$d_1^{n2} : D \qquad \langle d_1^{n2}, td_{n2} \rangle : hasAType$$
$$d_2^{n2} : D \qquad \langle d_2^{n2}, td_{n2} \rangle : hasAType \qquad \langle d_2^{n2}, p_1^{22} \rangle : includedIn$$

Терминология TBox$^{\mathbf{dom}}$

При определении терминологии $TBox^{dom}$ применение только конкретных доменов не является достаточным. Проблема состоит в определении *степени выраженности понятий* онтологии (в рамках метауровня понятий) в документах электронного архива проектной организации. Отдельно взятое понятие c_i может с различной степенью принадлежности иметь отношение к некоторому фрагменту документа d_j. С данной целью будем применять в терминологии нечеткие предикаты с заранее заданными функциями принадлежности (рисунок 2.17).

Трапезоидные, триангулярные, L-функции и R-функции являются не только достаточно простыми с вычислительной точки зрения, но и самыми рас-

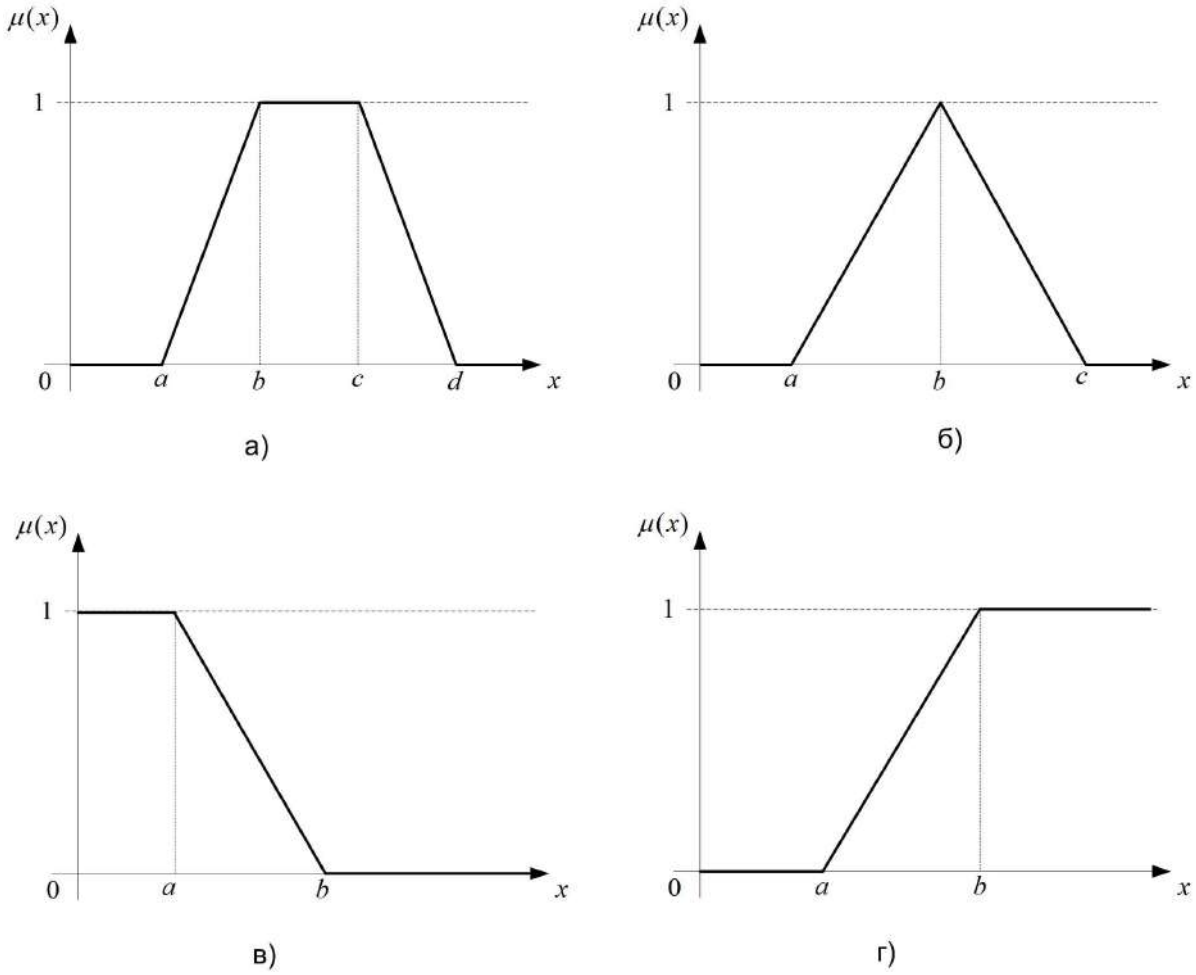


Рис. 2.17. а) Трапезоидная функция, б) Триангулярная функция, в) L-функция, R-функция

пространенными при определении функций принадлежности нечетких переменных. В данной работе указанные функции определены на множестве $[0, 1]$. *Трапезоидная функция* $trz(x; a, b, c, d)$ определена следующим образом: пусть $a < b \leq c < d$ числа из диапазона $[0, 1]$, тогда (рисунок 2.17, а):

$$trz(x; a, b, c, d) = \begin{cases} 0, & \text{если } x \leq a; \\ (x - a)/(b - a), & \text{если } x \in [a, b]; \\ 1, & \text{если } x \in [b, c]; \\ (d - x)/(d - c), & \text{если } x \in [c, d]; \\ 0, & \text{если } x \geq d. \end{cases}$$

Триангулярную функцию $tri(x; a, b, c)$ будем определять в виде (рисунок 2.17, б):

$$tri(x;a,b,c) = \begin{cases} 0, & \text{если } x \leq a; \\ (x-a)/(b-a), & \text{если } x \in [a,b]; \\ (c-x)/(c-b), & \text{если } x \in [b,c]; \\ 0, & \text{если } x \geq c. \end{cases}$$

L-функцию $L(x;a,b)$ определим, как (рисунок 2.17, в):

$$L(x;a,b) = \begin{cases} 1, & \text{если } x \leq a; \\ (b-x)/(b-a), & \text{если } x \in [a,b]; \\ 0, & \text{если } x \geq b. \end{cases}$$

Наконец, *R-функцию* $R(x;a,b)$ определим в виде (рисунок 2.17, г):

$$R(x;a,b) = \begin{cases} 0, & \text{если } x \leq a; \\ (x-a)/(b-a), & \text{если } x \in [a,b]; \\ 1, & \text{если } x \geq b. \end{cases}$$

Запишем терминологию метауровня понятий, структура которого приведена на рисунке 2.5.

$$\begin{array}{ll} c_{11} \sqsubseteq c_1 & c_1 \sqsubseteq c \\ c_{12} \sqsubseteq c_1 \sqcap \exists hasAPart.c_{k1} & \vdots \\ c_{k1} \sqsubseteq c_k & c_k \sqsubseteq c \\ c_{k2} \sqsubseteq c_k & \end{array}$$

$$c \sqsubseteq \top \sqcap \forall associationWith.D \sqcap$$
$$\sqcap (\exists hasAExpValue.High \sqcup \exists hasAExpValue.Middle \sqcup \exists hasAExpValue.Low)$$

$$c^{exp}_{high} \sqsubseteq c \sqcap \exists hasAExpValue.High$$
$$c^{exp}_{middle} \sqsubseteq c \sqcap \exists hasAExpValue.Middle$$
$$c^{exp}_{low} \sqsubseteq c \sqcap \exists hasAExpValue.Low,$$

где $hasAPart$ – наименования роли *имеет часть*, $hasAExpValue$ – наименования роли *имеет значение степени выраженности*. При этом, $High$, $Middle$ и Low – нечеткие конкретные предикаты, определяемые как:

$$High, Middle, Low : [0,1] \to [0,1].$$

Отдельно были выделены понятия c^{exp}_{high}, c^{exp}_{middle} и c^{exp}_{low}, которые означают понятия онтологии с высокой, средней и низкой степенью выраженности, соответственно.

Параметрически нечеткие предикаты определены следующим образом:

$$Low(x) = L(x; 0.2, 0.4);$$
$$Middle(x) = trz(x; 0.2, 0.4, 0.6, 0.8);$$
$$High(x) = R(x; 0.6, 0.8).$$

Запишем терминологию $TBox^{dom}$, относящуюся к тезаурусу проектной организации и связанную с терминологией метауровня понятий:

$$\{ct^{11}_1\} \equiv \top \sqcap \leq 1 represents A.c_{11}$$
$$\{ct^{11}_2\} \equiv \top \sqcap \leq 1 represents A.c_{11}$$
$$T \equiv \top \sqcap (\exists nearBy.\{ct^{11}_1\} \sqcup \exists nearBy.\{ct^{11}_2\})$$

Набор фактов $\mathbf{ABox^{dom}}$

$$
\begin{array}{ll}
ct_1^{11} : \{ct_1^{11}\} & \langle t_1, ct_1^{11}\rangle : nearBy \\
ct_2^{11} : \{ct_2^{11}\} & \langle t_2, ct_1^{11}\rangle : nearBy \\
t_1 : T & \langle t_2, ct_2^{11}\rangle : nearBy \\
t_2 : T & \langle t_s, ct_2^{11}\rangle : nearBy \\
\vdots & \langle ct_1^{11}, c_1^{11}\rangle : representsA \\
t_s : T & \langle ct_2^{11}, c_1^{11}\rangle : representsA \\
\vdots & ct_1^{11} : c_{high}^{exp} \leq 0.75 \\
 & ct_2^{11} : c_{high}^{exp} \leq 0.6
\end{array}
$$

Факты вида $a : C \leq \eta$ интерпретируются как: экземпляр a принадлежит концепту C со степенью принадлежности не выше порогового значения η. В следующей главе, посвященной концептуальному индексированию ресурсов электронного архива проектной организации, будет приведен метод определения численного значения степени выраженности концепта в фрагменте технического документа.

Терминология $\mathbf{TBox^{prj}}$

Запишем терминологию проектных диаграмм, разделяя логическое представление нотации языка UML и логическое представление шаблонов проектирования. Фрагмент терминологии нотации языка UML, соответствующий рисунку 2.10 и определяющий отношения имеет следующий вид:

$Relationship \sqsubseteq \top$

$Dependency \sqsubseteq Relationship$

$Association \sqsubseteq Relationship$

$Generalization \sqsubseteq Relationship$

$Realization \sqsubseteq Relationship \sqcap \exists startWith.Class \sqcap \exists endWith.Interface,$

где $startWith$ и $endWith$ – наименование ролей *исходит от* и *подходит к*, соответственно.

Представление основных классов запишем в виде:

$$Thing \sqsubseteq \top \sqcap \exists hasAName.String$$

$$StructThing \sqsubseteq Thing$$

$$AnnotThing \sqsubseteq Thing$$

$$Note \sqsubseteq AnnotThing \sqcap \exists connectedTo.Association$$

$$Class \sqsubseteq StructThing$$

$$Object \sqsubseteq StructThing \sqcap \forall isObjectOf.Class$$

$$Interface \sqsubseteq StructThing$$

$$SimpleClass \sqsubseteq Class$$

$$AbstractClass \sqsubseteq Class,$$

где $hasAName$ – наименование роли *имеет имя*, $isObjectOf$ – наименование роли *является объектом*, $connectedTo$ – наименование роли *связан с*, $String$ – конкретный домен (concrete domain) строкового типа.

Атрибуты и методы классов представляются следующим образом:

$$Attribute \sqsubseteq \top \sqcap \exists hasAAttrName.String \sqcap \exists isAPartOf.Class$$

$$Method \sqsubseteq \top \sqcap \exists hasAMethName.String \sqcap \exists isAPartOf.Class,$$

где $hasAAttrName$ – наименование роли *имеет наименование атрибута*, $hasAMethName$ – наименование роли *имеет наименование метода*, $isAPartOf$ – наименование роли *является частью*.

Рассмотрим терминологию проектных шаблонов, связанных с логическим представлением нотации проектных диаграмм (на примере диаграммы классов

языка UML):

$$Template \sqsubseteq \top \sqcap \exists hasATempName.String \sqcap$$
$$\sqcap (\exists hasAExpValue.High \sqcup \exists hasAExpValue.Middle \sqcup \exists hasAExpValue.Low)$$
$$Temp_{deleg} \sqsubseteq Template,$$

где $hasATempName$ – наименование роли *имеет имя шаблона*.

Отдельный шаблон проектирования в каждом конкретном проекте может иметь определенную степень выраженности:

$$Temp^{exp}_{high} \sqsubseteq Template \sqcap \exists hasAExpValue.High$$
$$Temp^{exp}_{middle} \sqsubseteq Template \sqcap \exists hasAExpValue.Middle$$
$$Temp^{exp}_{low} \sqsubseteq Template \sqcap \exists hasAExpValue.Low.$$

Набор фактов ABox$^{\text{prj}}$

На рисунке 2.10 приведен пример шаблона «Делегирование», который в виде набора фактов $ABox^{prj}$ выглядит следующим образом:

$class1 : SimpleClass$ $class2 : SimpleClass$

$attribute01 : Attribute$ $object01 : Object$

$method01 : Method$ $method02 : Method$

$\langle method01, name01 : string\rangle : hasAMethName$

$\langle method02, name01 : string\rangle : hasAMethName$

$\langle attribute01, class1\rangle : isAPartOf$ $\langle object01, class2\rangle : isObjectOf$

$\langle object01, attribute01\rangle : owl : sameAs$ $\langle method01, class1\rangle : isPartOf$

$\langle method02, class2\rangle : isPartOf$ $\langle relation01, class1\rangle : startWith$

$relation01 : Association$ $\langle relation01, class2\rangle : endWith.$

Фактически, в онтологии ИПР в набор фактов $ABox^{prj}$ включены все факты по используемым шаблонам проектирования. Далее в процессе концептуаль-

ного индексирования (рассматривается в следующей главе) происходит сопоставление фактов из $ABox^{prj}$ с фактами, извлекаемыми из проектных диаграмм и определяется степень выраженности для каждого шаблона онтологии.

2.7. Выводы по второй главе

1. Особенности проектной деятельности накладывают определенные ограничения на структуру и логико-аксиоматическое наполнение онтологии ИПР. Совершенно недостаточно в современных условиях рассматривать онтологию только как тезаурус предметной области. Представление онтологии ИПР как интегрированной онтологической системы позволяет, с одной стороны, учесть различные аспекты гетерогенных знаний проектной организации от абстракции проектов до уровня лексиконов, а с другой стороны, сократить трудоемкость построения такой онтологии.
2. Основными видами информационных ресурсов, которые отображаются в онтологические представления, являются текстовые технические документы и проектные диаграммы (например, UML-диаграммы совместно с шаблонами проектирования). Нижний уровень онтологии ИПР должен предусматривать возможность концептуализации составных элементов указанных ресурсов с целью их последующего концептуального индексирования.
3. Нечеткое расширение формализма дескриптивной логики $\mathcal{SHOIN}(\mathcal{D})$ позволяет представить расплывчатые характеристики информационных объектов, которые не могут быть выражены с использованием четких предикатов, определенных на конкретных доменах. Указанные характеристики являются неотъемлемой чертой информационных объектов электронных архивов проектной организации и следствием неформализованности естественного языка.

Глава 3

Концептуальный индекс интеллектуального проектного репозитория

3.1. Понятие концептуального индекса. Его структура

Потребность в концептуальном индексировании информационных ресурсов электронных архивов проектных организаций вызвана следующими причинами:

1. Разнородный характер видов информационных ресурсов (текстовые документы и модели в различных нотациях) усложняет их анализ на синтаксическом уровне и не позволяет выполнять содержательную интерпретацию.
2. Слабоформализованных характер как текстовых документов, так и моделей в графических нотациях снижает эффективность применения поисковых моделей, классификационных моделей и моделей кластерного анализа для задач структуризации электронных архивов из-за высокой степени неопределенности представления информационных ресурсов (любое понятие может быть представлено с применением различных терминов (слов) и различными сущностями, классами и т.д.).

Концептуальное индексирование позволяет получить сжатое (компактное) представление различных по своей структуре и содержанию информационных ресурсов электронного архива проектной организации на семантическом уровне [83], [84]. Структурная схема процесса формирования концептуального индекса проектной организации представлена на рисунке 3.1.

Основу концептуального индекса составляет хеш-таблица, состоящая из идентификаторов терминов лексиконов, экземпляров понятий онтологии проектных диаграмм и идентификаторов концептов (понятий) онтологии. Идентификаторы терминов и экземпляров понятий являются ключами, а идентифи-

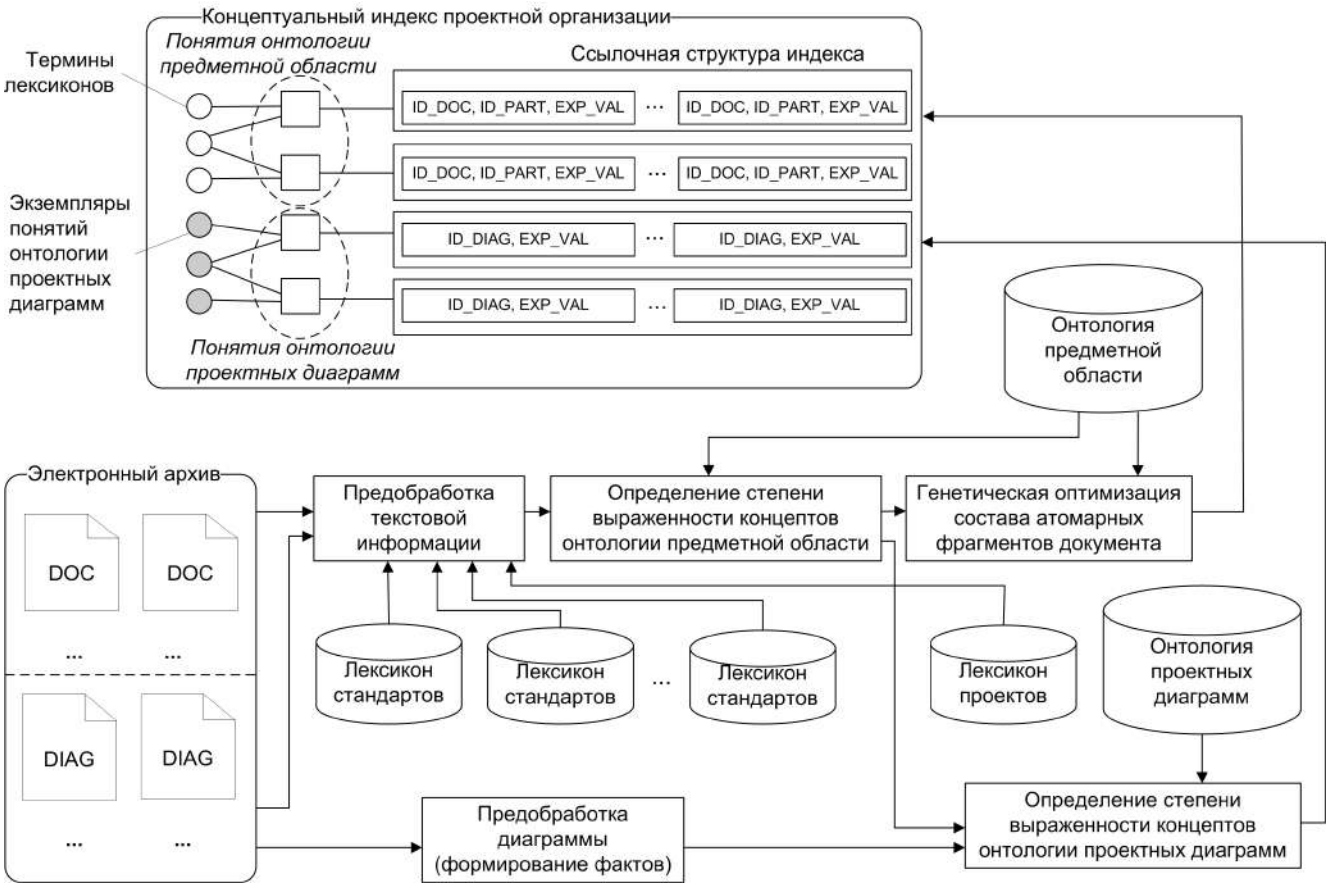


Рис. 3.1. Формирование концептуального индекса проектной организации

каторы концептов – значениями. С каждым идентификатором концепта связан список троек $[ID\_DOC, ID\_PART, EXP\_VAL]$ если речь идет о текстовых технических документах и список $[ID\_DIAG, EXP\_VAL]$ – для проектных диаграмм. Множество таких списков определяет ссылочную структуру концептуального индекса. Списки состоят из следующих элементов: ID_DOC – идентификатор технического документа из электронного архива, ID_DIAG – идентификатор проектной диаграммы; ID_PART – идентификатор раздела технического документа, EXP_VAL – вещественное число, соответствующее степени выраженности понятия в документе (или шаблона проектирования в проектной диаграмме).

3.2. Метод концептуального индексирования текстовых ресурсов проектного репозитория

Индексирование технических документов (ТД) как информационных ресурсов документальных баз данных включает в себя следующие функции [158]:

- формирование множества технических документов;
- удаление нерепрезентативных слов;
- выполнение функции стемминга (формирование термов);
- вычисление частоты встречаемости термов (абсолютной и относительной);
- определение степени выраженности концептов онтологии предметной области электронного архива.

В основе концептуального индексирования ТД лежит следующая функция [68], [69]:

$$F_{oV} : ch_j^d \to oV_j^d, \tag{3.1}$$

где ch_j^d – j-ый раздел технического документа d, oV_j^d – онтологическое представление j-го раздела документа d.

Рассмотрим процедуру выполнения концептуального индексирования (3.1) [56], [59], [108], [110]. Центральным элементом данной процедуры является понятие степени выраженности концепта онтологии.

Определение 5. *Степень выраженности* концепта онтологии проектного репозитория есть степень совпадения текстового входа концепта (понятия) онтологии с множеством терминов фрагмента текстового технического документа.

На синтаксическом уровне j-й раздел документа записывается в виде множества пар «термин-частота»:

$$\{(t_{1j}^d, f_1^j), (t_{2j}^d, f_2^j), \ldots, (t_{ij}^d, f_i^j), \ldots, (t_{l_j j}^d, f_{l_j}^j),$$

где l_j – количество терминов в j-м разделе текстового документа после реализации функции удаления стоп-слов (нерепрезентативных слов документа).

Нормализованный вес термина t_{ij}^d в составе j-го раздела ТД вычисляется по следующей формуле:

$$f_i^j = 1 + log(tf_{t_{ij}^d}) \cdot log\left(\frac{N}{dt}\right) \cdot \frac{1}{\sqrt{tf_{t_{1j}^d}^2 + tf_{t_{2j}^d}^2 + \ldots + tf_{t_{nj}^d}^2}}, 1 \leq i \leq n,$$

где f_i^j – нормализованный вес термина t_{ij}^d в j-м разделе документа; $tf_{t_{ij}^d}$ – частота встречаемости термина t_{ij}^d; N – количество текстовых документов в архиве; dt – количество ТД, в которых встречается термин t_{ij}^d; n – общее число терминов в j-м разделе технического документа.

Вычисление значения степени выраженности отдельного концепта онтологии предметной области автоматизированного проектирования выполняется для каждого раздела ТД. Данная процедура использует в своей основе формализм нечетких соответствий [5]. *Нечеткое соответствие* между множеством терминов T, которые образуют текстовые входы понятий и множеством понятий онтологии предметной области C есть тройка множеств $\tilde{\Gamma} = (T, C, \tilde{O})$, где T и C – обычные (четкие) множества, а $\tilde{O}$ определяется как нечеткое множество в $T \times C$. Множество T есть область отправления, множество C – область прибытия, а $\tilde{O}$ – нечеткий график нечеткого соответствия.

Носителем нечеткого соответствия $\tilde{\Gamma} = (T, C, \tilde{O})$ является четкое соответствие $\Gamma = (T, C, O)$, у которого график O есть носитель нечеткого графика $\tilde{O}$.

Образом множества $\tilde{T^d}$ (множество терминов текстового документа d, определяемое через нечеткое множество) при соответствии $\tilde{\Gamma}$ будем называть нечеткое множество $\tilde{\Gamma}(\tilde{T^d})$ в C, представляемое в виде выражения:

$$\tilde{\Gamma}(\tilde{T^d}) = \{\langle \mu_{\Gamma(T^d)}(c), c\rangle \mid c \in C\}, \tag{3.2}$$

где $\mu_{\Gamma(T^d)}(c) = \vee_{t^d \in T^d} \left(\mu_{\Gamma(T^d)}(t^d) \wedge \mu_O\langle t, c\rangle\right)$ [5].

Каждый элемент $c \in C$ может соответствовать нескольким элементам $t^d \in T^d$. Следовательно, значение функции принадлежности элемента c нечеткому множеству $\tilde{\Gamma}(\tilde{T^d})$ определяется как максимальное из значений, получаемых с помощью выбора минимального между значением функции принадлежности каждого $t^d \in T^d$ нечеткому множеству $\tilde{T}^d$ и значением функции принадлежности пары $\langle t, c\rangle$ нечеткому графику $\tilde{O}$. Следует уточнить, что $\mu_{T^d}(t^d)$ есть нормализованный вес термина индексируемого ТД. Кроме того, $\mu_O\langle t, c\rangle$ определяется выражением (3.2) и формирует множество степеней выраженности понятий онтологии $\langle \mu_{\Gamma(T^d)}(c), c\rangle$.

При составлении содержания текстового ТД, являющегося артефактом проектирования АС, его автор вынужден использовать специальный понятийный аппарат и в различных разделах документа акцентировать свое внимание на определенных темах. В связи с этим возникает задача учета данного факта в процессе концептуального индексирования ТД, принимая во внимание онтологию как формализованное описание состояния предметной области проектного репозитория.

Предположим, что текст документа, который подвергается концептуальному индексированию, уже прошел предобработку и представляет собой последовательность терминов, являющихся элементами предложений документа [56], [59], [64], [65], [108], [110]. В качестве минимального фрагмента анализируемого текста будем считать отдельное предложение, а максимальным фрагментом – тематический раздел, как структурную часть документа.

Определение доминирующего понятия в текстовом фрагменте технического документа

Пусть задан текстовый технический документ d, который состоит из следующей последовательности терминов:

$$S^d = t^d_{11}, t^d_{21}, \ldots, t^d_{i_1 1}, \ldots, t^d_{n_1 1}, t^d_{12}, \ldots, t^d_{i_2 2}, \ldots, t^d_{n_2 2}, \ldots, t^d_{i_j j}, \ldots, t^d_{n_m m}, \tag{3.3}$$

где – i_j номер термина в j-м предложении, $j = \overline{1,m}$; $i_j = \overline{1,n_j}$, где n_j – количество терминов в j-м предложении.

Для k-го понятия соответствующий текстовый вход запишем в виде множества:

$$\{\left(t_1^k, f_1^k\right), \left(t_2^k, f_2^k\right), \ldots, \left(t_i^k, f_i^k\right), \ldots, \left(t_{l_k}^k, f_{l_k}^k\right)\},$$

где l_k – общее количество терминов, наиболее близких в семантическом смысле с k-м понятием; t_i^k – i-ый термин k-го понятия онтологии; f_i^k – семантический вес i-го термина в текстовом входе k-го понятия (семантическое расстояние между термином и понятием для одного текстового входа, прошедшего нормализацию).

Пусть S_p^d будет обозначать часть последовательности S^d, которая, в свою очередь, определяется выражением (3.3). Запишем указанный фрагмент последовательности следующим образом:

$$S_p^d = t_{1p}^d, t_{2p}^d, \ldots, t_{j_p p}^d, \ldots, t_{n_p p}^d, p = \overline{1,s}.$$

Истинным является следующее выражение:

$$S_1^d, S_2^d, \ldots, S_s^d = S^d. \tag{3.4}$$

Выполняя процедуру концептуального индексирования электронного архива проектной организации необходимо найти подмножество понятий онтологии предметной области, которые соответствуют тематике анализируемых документов. Предлагается использовать следующую гипотезу.

Гипотеза: *Текстовый документ электронного архива из ограниченной предметной области можно разделить на фрагменты, которые не пересекаются и в каждом из которых можно определить доминирующий концепт предметной области.*

Для определения численных значений степеней доминирования понятий применяется следующий способ: производится сравнение текстовых входов по-

нятий в онтологии предметной области с фрагментами анализируемых текстовых документов [56], [59], [64], [65], [108], [110].

Степень выраженности $\mu_{S_p^d}(c)$ понятий $c \in C$ в p-м фрагменте ТД d будем определять, используя представление образа множества терминов согласно выражению (3.2). На рисунке 3.2 представлены две различные ситуации, соответствующие двум иллюстративным текстовым фрагментам S_1 и S_2. Будем предполагать, что извлечение из текстового ТД фрагмента преследовало цель нахождения доминирующего концепта. Очевидно, что во фрагменте S_1 (рисунок 3.2, а) указанная операция была выполнена успешно, чем во фрагменте S_2 (рисунок 3.2, б). На рисунке 3.2, а имеет место доминирование концепта c_1 относительно других концептов, основываясь на значении степени его выраженности в текстовом фрагменте S_1, что нельзя утверждать по фрагменту S_2.

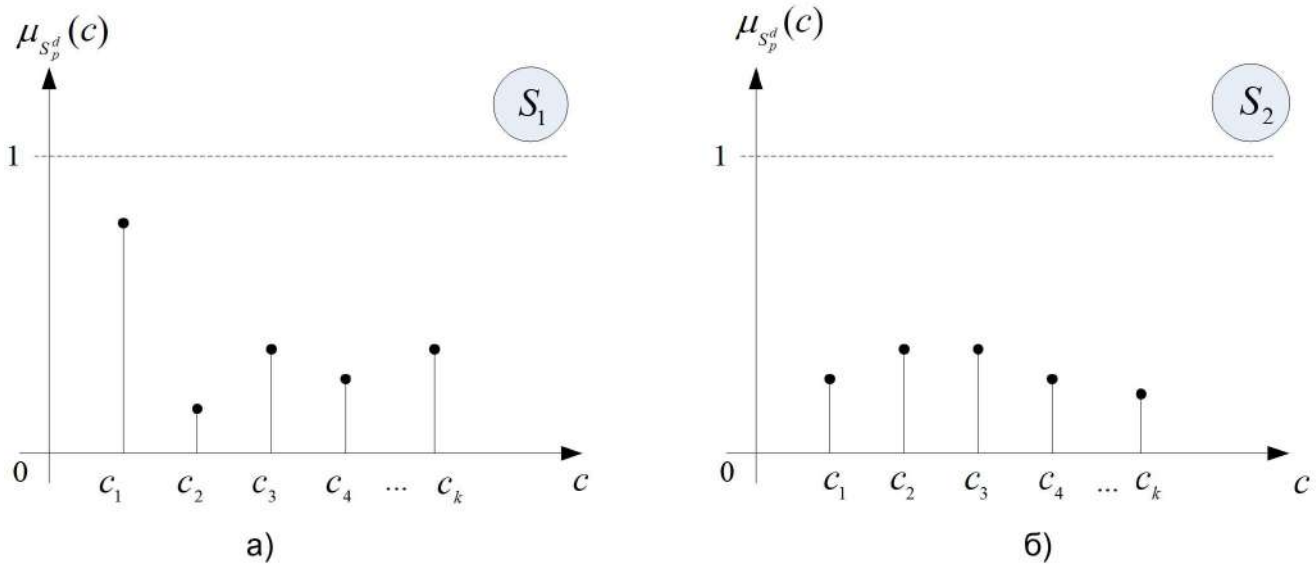


Рис. 3.2. Степени выраженности понятий онтологии в текстовом фрагменте технического документа

Алгоритм нахождения доминирующих понятий в текстовом фрагменте технического документа включает в себя следующие шаги.

Шаг 1. Вычисление наибольшей степени выраженности понятия (рисунок 3.3):

$$\hat{\mu}_{S_p^d}(c) = \max_c \left(\mu_{S_p^d}(c) \right).$$

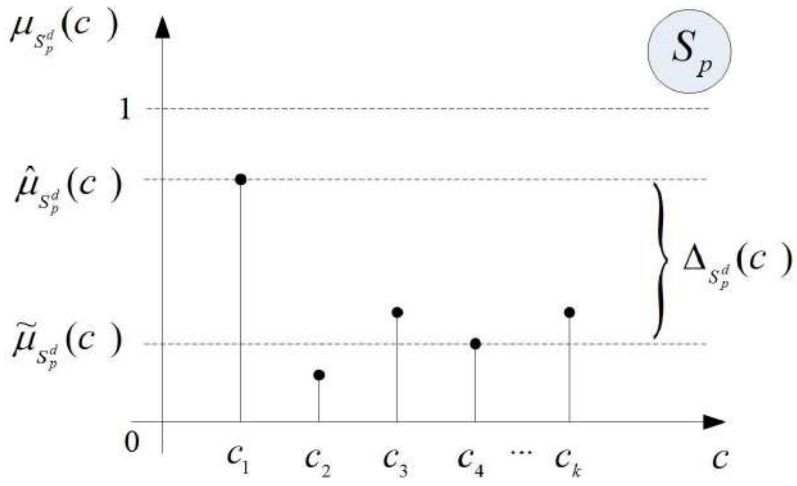


Рис. 3.3. Доминирование понятия в документе

Шаг 2. Вычисление среднего значения по всем степеням выраженности понятий в документе за исключением найденного понятия на предыдущем шаге алгоритма:

$$\tilde{\mu}_{S_p^d}(c) = \frac{1}{n-1} \sum_{i=1}^{n-1} \mu_{S_p^d}(c_i),$$

где $c_i \in c - c_k$; $c_k = \arg\max_c \left(\mu_{S_p^d}(c) \right)$; n – количество понятий со степенью выраженности для текстового фрагмента S_p^d больше нуля.

Шаг 3. Вычисление результата – степени доминирования понятия в текстовом фрагменте S_p^d:

$$\Delta_{S_p^d}(c) = \hat{\mu}_{S_p^d}(c) - \tilde{\mu}_{S_p^d}(c). \tag{3.5}$$

Необходимо отметить тот факт, что формула (3.5) представляет критерий качества нахождения текстового фрагмента в документе. При этом целью является явное ограничение в тексте понятия предметной области из онтологии проектного репозитория.

Адаптация генетического алгоритма к задаче концептуального индексирования

Целью генетической оптимизации в процессе концептуального индексирования документальной базы является определение последовательности (3.4) текстовых фрагментов документа, как информационного ресурса. Такая оптимальная последовательность предполагает нахождение минимального значения целевой функции:

$$F\left(S^d\right) = \frac{1}{s}\sum_{p}\left(1 - \Delta_{S_p^d}(c)\right) \to \min, \tag{3.6}$$

$p = \overline{1,s}$, где s – количество фрагментов ТД; $s = \overline{1,m}$, где m – количество предложений в обрабатываемом документе. Следовательно, минимальный текстовый фрагмент будет состоять из единственного предложения ТД, а максимальный – включать в себя весь ТД.

Определим генетический алгоритм в соответствии со следующими ограничениями [98].

1. Формируется целевая функция для определения критерия качества решений.
2. Формируется исходная популяция потенциальных решений, каждое из которых кодируется как вектор (хромосома).
3. Каждая хромосома должна иметь такую структуру, которая позволяет выполнять ее количественную оценку (значение эффективности) в соответствии с заданной целевой функцией.
4. Для каждой хромосомы вычисляется вероятность воспроизведения, которая зависит от значения целевой функции генетической оптимизации.
5. С учетом вероятностей воспроизведения формируется новая популяция хромосом. С большей вероятностью переходят в следующий пул наиболее эффективные решения. Хромосомы воспроизводятся, используя операцию кроссинговера (скрещивание хромосом) и мутацию (случайное изменение фрагментов хромосомы).

Структуру генетического алгоритма оптимизации текстовых фрагментов представим следующим образом [98]:

$$\Gamma A = \left(P^0, \lambda, l, v, \rho, F, \tau\right), \tag{3.7}$$

где $P^0 = (a_1^0, \ldots, a_\lambda^0)$ – исходное множество решений, где a_i^0 – отдельное решение задачи, представленное в виде хромосомы; λ – целое число, определяющее размер популяции потенциальных решений алгоритма; l – число, задающее длину хромосомы в популяции; v – функция, выполняющая отбор хромосом в следующий пул; ρ – отображение, определяющее кроссинговер и мутацию; F – целевая функция генетического алгоритма; τ – критерий остановки генетического алгоритма.

Для решения задачи оптимизации текстовых фрагментов ТД необходимо выполнить следующее:

- определить способ кодирования хромосом (потенциальных решений);
- составить целевую функцию, которую алгоритм будет минимизировать;
- определиться с реализацией операций кроссинговера и мутации.

Для реализации функции концептуального индексирования потенциальное решение представляется в виде вектора:

$$a_i^t = \left(\langle p, j\rangle\right), p = \overline{1, s}, j = \overline{1, m}, 1 \leq s \leq m, \tag{3.8}$$

где p – текущий номер фрагмента документа; j – порядковый номер предложения фрагмента документа; s – общее количество фрагментов; m – общее количество предложений; i – порядковый номер хромосомы; t – текущий номер пула эволюционного цикла. Структура потенциального решения генетической оптимизации, определяемая выражением (3.8), представляет собой некоторую последовательность фрагментов ТД (3.4).

Целевая функция позволяет выполнять отображение хромосомы как потенциального решения задачи оптимизации на единичный отрезок:

$$F : a_i^t \to [0, 1].$$

В качестве целевой функции F используется функция, определяемая выражением (3.6).

Выполнение генетического алгоритма начинается с формирования начальной популяции $P^0 = \left(a_1^0, \ldots, a_\lambda^0\right)$. В созданной популяции для каждой хромосомы a_i^0 вычисляется значение целевой функции $F(a_i^0)$. Далее выполняется операция ранжирования хромосом. При этом, ранг $rank$ задается следующим образом:

$$\forall i \in \{1, \ldots, \lambda\} : rank\left(a_i^t\right) = i,$$

если для $\forall j \in \{1, \ldots, \lambda - 1\} : F(a_j^t) < F(a_{j+1}^t)$.

Лучшие хромосомы (первые g) перемещаются в следующее поколение без изменений. Остальные хромосомы (их количество – $(\lambda - g)$) создаются посредством операции кроссинговера. Оператор кроссинговера определяется таким образом, что не нарушается последовательность предложений в тексте документа и их количество не изменяется в процессе формирования хромосом. Место скрещивания хромосом (точка кроссинговера) выбирается случайным образом на границе соседних текстовых фрагментов:

$$a_i^0 = (\ldots, \langle p, j\rangle, \langle p+1, j+1\rangle, \ldots).$$

Таким образом точка кроссинговера определяется для первой хромосомы. Необходимо учитывать тот факт, что в процессе данной операции происходит обмен частями хромосом. Кроме того, необходимо принимать во внимание ранее сформулированные ограничения. Выбор точки кроссинговера для второй хромосомы происходит таким образом, чтобы в левой части остались такое же количество j первых предложений, как и у первой хромосомы.

Способ формирования текстовых фрагментов может быть различным (с точностью до предложения), поэтому нельзя исключать ситуацию, когда в процессе кроссинговера во второй хромосоме будет добавлен еще один текстовый фрагмент. Это возможно по причине сохранения равного количества предложений в хромосомах, и, как следствие, всегда имеется ненулевая вероятность

разбиения одного текстового фрагмента второй хромосомы на два фрагмента. Указанная ситуация представлена на рисунке 3.4.

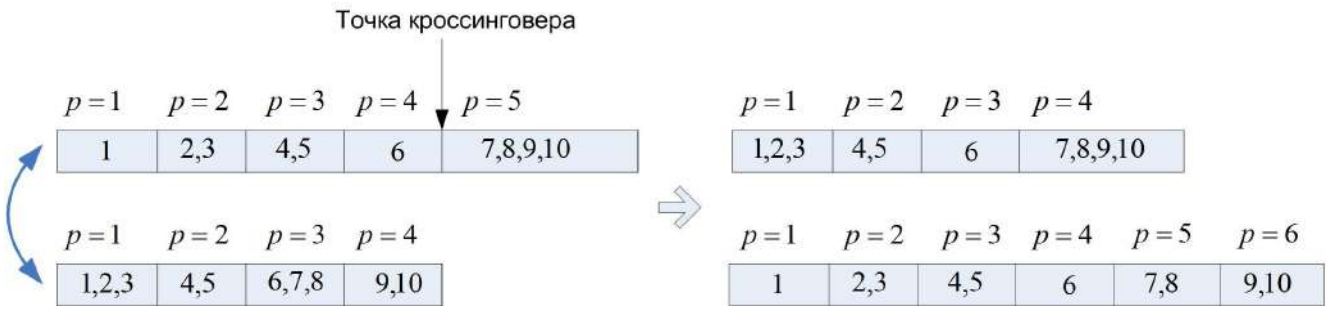


Рис. 3.4. Пример выполнения операции кроссинговера

Отбор хромосом, которые участвуют в кроссинговере, производится посредством вычисления значений вероятностей $p_v(a_i^t)$ для каждого потенциального решения, применяя известный метод пропорционального отбора:

$$p_v(a_i^t) = \frac{F(a_i^t)}{\sum_{j=1}^{\lambda} F(a_j^t)}.$$

На последнем этапе создания новой популяции сформированный пул хромосом подвергается действию оператора мутации. В решаемой задаче построения концептуального индекса электронного архива проектной организации используются две разновидности оператора мутации хромосом: 1) смещение границы фрагмента текстового документа и 2) соединение фрагментов текстового документа (рисунок 3.5).

Вариант оператора мутации, предполагающий смещение границы фрагмента текстового документа производит выбор границы между двумя текстовыми фрагментами ТД с некоторой вероятностью. Далее необходимо принять решение о том, в каком направлении будет выполняться смещение границы (в правую или левую сторону). В этом случае критерием выбора является количество предложений в соседних фрагментах ТД. Граница перемещается на одно предложение в ту сторону, где количество предложений больше. Если количество предложений в соседних текстовых фрагментов равно, то направление

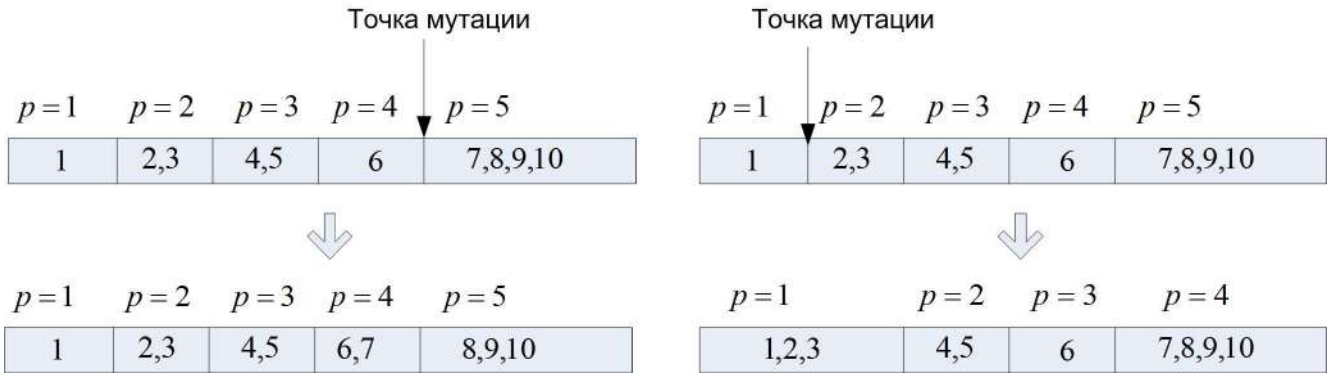


Рис. 3.5. Пример операторов мутации хромосом

смещения выбирается случайно. Если текстовый фрагмент содержит одно предложение, то смещения границы не происходит. На рисунке 3.5, а) представлен иллюстративный пример выполнения оператора мутации, который осуществляет смещение границы между текстовыми фрагментами №4 и № 5 вправо.

Оператор мутации, который выполняет соединение двух соседних фрагмента, позволяет сокращать количество найденных фрагментов в документе благодаря их укрупнению. На рисунке 3.5, б) показан иллюстративный пример выполнения оператора мутации, целью которого является объединение первого и второго фрагмента ТД.

Для практического применения генетического алгоритма с целью оптимизации концептуальных представлений ТД необходимо решить задачу подбора параметров алгоритма. В частности, необходимо настроить значения вероятностей мутации смещения и соединения текстовых фрагментов для того, чтобы:

- разработанный алгоритм генетической оптимизации обладал свойством сходимости;
- выполнялась генерация большого количества хромосом из разных подобластей допустимых решений (для обеспечения возможности выхода из локальных экстремумов целевой функции генетического алгоритма);
- сохранялся устойчивый баланс между скоростью увеличения количества фрагментов ТД в процессе кроссинговера и скоростью их уменьшения в

процессе реализации оператора мутации.

В главе, посвященной практической реализации, представлены результаты экспериментов, которые позволяют определить те значения параметров генетической оптимизации, при которых наблюдается удовлетворительная сходимость алгоритма.

Корректировка множества понятий в концептуальном представлении технического документа

Первоначальное онтологическое представление каждого раздела документа в формируется в виде:

$$o\hat{V}_j^d = \langle ch_j, \{\hat{C}_j^P \cup \hat{C}_j^{St}\}\rangle, \hat{C}_j^P \subseteq C^P, \hat{C}_j^{St} \subseteq C^{St},$$

где $\hat{C}_j^P$ – множество понятий, извлекаемых из проектов электронных архивов, $\hat{C}_j^{St}$ – множество понятий, извлекаемых из различных серий стандартов, используемых на предприятии.

Эксперименты по созданию концептуального индекса проектной организации, использующие метод генетической оптимизации, показали, что только небольшое число понятий в каждом текстовом фрагменте дает существенный вклад в общую сумму степеней выраженности понятий в рамках выделенного фрагмента. В различных фрагментах это соотношение изменялось в диапазоне от 60/40 до 80/20. В среднем около 30% понятий в сумме дают примерно 70% от общей степени выраженности всех понятий фрагмента ТД. В этой связи в работе предлагается первоначальное множество понятий $\hat{C}_j$ j-го раздела технического документа дополнять наиболее значимыми понятиями каждого фрагмента.

3.3. Метод концептуального индексирования проектных диаграмм

Целью концептуального индексирования проектных диаграмм является формирование индекса, структура которого представлена на рисунке 3.1. Для решения этой задачи необходимо для индексируемой диаграммы определить экземпляры классов онтологии проектных диаграмм и вычислить степень соответствия проектной диаграмме шаблонов проектирования, определенных в онтологии.

Исходными данными будем считать:

- $\{\langle cs_1, dc_1\rangle, \langle cs_2, dc_2\rangle, \ldots, \langle cs_n, dc_n\rangle\}$ – множество анализируемых проектов электронного архива, каждый из которых включает исходный код cs_i и диаграмму классов dc_i, i – номер проекта;
- онтологию $O^{prj} = \{\langle C^{prj}, R^{prj}\rangle, \{tmp_1, tmp_2, \ldots, tmp_m\}\}$, включающую в качестве элементов множество концептов нотации проектных диаграмм C^{prj} (диаграммы классов языка UML), множество отношений между классами R^{prj} и множество шаблонов проектирования $\{tmp_1, tmp_2, \ldots, tmp_m\}$;
- $Tz^p = \{\langle t_1^p, f_1^p\rangle, \langle t_2^p, f_2^p\rangle, \ldots, \langle t_l^p, f_l^p\rangle\}$ – техническое задание на новый проект АС p, прошедший предобработку и представленный в виде набора терминов $t_1^p, \ldots, t_l^p$ с соответствующими частотами $f_1^p, \ldots, f_l^p$;
- dc^p – проектная диаграмма как составная часть нового проекта АС, соответствующая техническому заданию Tz^p.

Согласно схеме схеме формирования концептуального индекса проектной организации (рисунок 3.1) концептуальное индексирование проектных диаграмм выполняется посредством следующих шагов:

1. Определение контекста проекта.
2. Определение подмножества диаграмм электронного архива, соответствующих контексту проекта.
3. Определение степени соответствия шаблонов проектирования из онтоло-

гии O^{prj} проектной диаграмме dc^p относительно каждого элемента найденного подмножества диаграмм электронного архива.

Определение контекста проекта выполняется на основе рассмотренного ранее метода концептуального индексирования текстовых информационных ресурсов:

$$oV_{tz} = F_{oV}(Tz, O^{dom}, O^{tz}).$$

На вход функции концептуального индексирования текстовой информации F_{oV} поступает техническое задание Tz, прошедшее предобработку, онтология предметной области O^{dom} и тезаурус O^{tz}.

Результатом концептуального индексирования является множество

$$oV_{tz} = \{\mu(c_1^{tz})/c_1^{tz}, \mu(c_2^{tz})/c_2^{tz}, \ldots, \mu(c_k^{tz})/c_k^{tz}\} = \mu_{oV}(c^{tz}),$$

включающее понятие $c_i^{tz} \in C^{dom}$ с соответствующим значением функции принадлежности i-го понятия $\mu_i(c_i^{tz})$ техническому заданию Tz (степень выраженности понятия в тексте технического задания). Полученное множество oV_{tz} будем считать контекстом реализуемого проекта.

Аналогичным образом концептуальное индексирование выполняется для множества проектов электронного архива $\{\langle Sc, Dc\rangle\}$. Сначала для каждого текста исходного кода $sc_i \in Sc$ извлекается текст комментария:

$$\forall i : tc_i = F_{extcomm}(sc_i),$$

где tc_i – текстовое представление комментария программного модуля sc_i, прошедшее предобработку:

$$tc_i = \{\langle t_1^{sc_i}, f_1^{sc_i}\rangle, \langle t_2^{sc_i}, f_2^{sc_i}\rangle, \ldots, \langle t_s^{sc_i}, f_s^{sc_i}\rangle\}.$$

В результате выполнения функции концептуального индексирования получаем онтологическое представление комментариев исходного кода для каждого

программного модуля:

$$oV_{sc_i} = F_{oV}(tc_i, C^{dom}, C^{tz}),$$
$$oV_{sc_i} = \{\mu(c_1^{sc_i})/c_1^{sc_i}, \mu(c_2^{sc_i})/c_2^{sc_i}, \ldots, \mu(c_l^{sc_i})/c_l^{sc_i}\} = \mu_{oV}(c^{sc_i}).$$

Множество проектных диаграмм электронного архива проектной организации, соответствующее контексту oV_{tz} будем определять следующим образом:

$$Dc|_{oV_{tz}} = \{dc_i : \&(\mu_{oV}(c^{sc_i}) \leftrightarrow \mu_{oV}(c^{tz})) \geq 0.5\},$$

где «$\leftrightarrow$» – операция эквивалентности нечетких множеств, а «&» – операция конъюнкции по всем $c^{sc_i}, c^{tz} \in C^{dom}$.

Другими словами, в указанное множество входят проектные диаграммы, для которых выполняется условие нечеткой эквивалентности онтологических представлений исходных текстов программ и онтологического представления технического задания.

Рассмотрим процесс определения степени выраженности шаблонов онтологии проектных диаграмм, позволяющий сформировать концептуальный индекс проектной организации по представленным в электронном архиве проектным диаграммам (на языке UML). В основе концептуального индекса проектных диаграмм лежит понятие нечеткой меры степени соответствия элементов проектной диаграммы шаблону онтологии (2.6).

Будем обозначать через $\mu_{tmp_j}(dc_i)$ степень принадлежности проектной диаграммы dc_i шаблону tmp_j. Аналитически $\mu_{tmp_j}(dc_i)$ будем определять по следующей формуле:

$$\mu_{tmp_j}(dc_i) = \frac{N(ABox_{dc_i}^{prj})}{N(ABox_{tmp_j}^{prj})},$$

где $N(ABox_{dc_i}^{prj})$ – количество фактов, которые являются истинными при условии истинности терминологии $TBox^{prj}$ и соответствуют базе фактов $ABox_{tmp_j}^{prj}$; $N(ABox_{tmp_j}^{prj})$ – количество фактов шаблона tmp_j.

Если количество фактов $N(ABox^{prj}_{tmp_j})$ некоторого шаблона tmp_j определяется достаточно просто (суммированием количества фактов j-го шаблона проектирования), то для вычисления $N(ABox^{prj}_{dc_i})$ необходимо использовать следующий разработанный алгоритм:

Шаг 1. Преобразование проектной диаграммы dc_i электронного архива в набор фактов $ABox^{prj}_{dc_i}$ вида:

$$elem^{dc_i}_k : Concept$$
$$\langle elem^{dc_i}_k, elem^{dc_i}_s \rangle : Role,$$

где $Concept$ – понятие, определенное в $TBox^{prj}$ и $Role$ – роль, определенная в $TBox^{prj}$; $elem^{dc_i}_k, elem^{dc_i}_s$ – экземпляры понятий, извлеченные из проектной диаграммы dc_i.

Шаг 2. Определение набора базовых классов из $ABox^{prj}_{dc_i}$ относительно шаблона tmp_j.

Базовым классом будем называть такой экземпляр $elem^{dc_i}_k$ понятия «Class» (или его дочернего понятия «Subclass») из $ABox^{prj}_{dc_i}$, который соответствует некоторому экземпляру $cls^{tmp_j}_l \in Class$ из $ABox^{prj}_{tmp_j}$ и для которого в шаблоне tmp_j имеет место максимальное количество фактов вида:

$$elem^{dc_i}_k : Concept$$
$$\langle elem^{dc_i}_k, * \rangle : Role, \qquad \langle *, elem^{dc_i}_k \rangle : Role.$$

Определенный набор базовых классов проектной диаграммы dc_i относительно шаблона tmp_j запишем в виде множества:

$$\{\langle elem^{dc_i}_1, cls^{tmp_j}_1 \rangle, \langle elem^{dc_i}_2, cls^{tmp_j}_2 \rangle, \dots\}$$

где кортеж $\langle elem^{dc_i}_k, cls^{tmp_j}_k \rangle$ означает, что экземпляр понятия проектной диаграммы $elem^{dc_i}_k$ эквивалентен экземпляру класса проектного шаблона $cls^{tmp_j}_k$.

Шаг 3. Вычисление количества истинных фактов, выполняя попарную за-

мену экземпляров классов j-го шаблона tmp_j и i-ой проектной диаграммы dc_i:

$$\forall k : cls_k^{tmp_j} \leftrightarrow elem_k^{dc_i}. \tag{3.9}$$

Факт шаблона проектирования является истинным относительно проектной диаграммы, если удается найти ему соответствие в наборе фактов проектной диаграммы с учетом того, что различные имена экземпляров понятий обозначают различные индивиды. Различные имена экземпляров понятий относятся к одному и тому же экземпляру онтологии только в том случае, если явно данные экземпляры связаны отношением «owl:sameAs».

Указанные шаги алгоритма концептуального индексирования проектной диаграммы dc_i выполняются для каждого шаблона проектирования, доступного в онтологии проектных диаграмм. В результате, онтологическое представление проектной диаграммы электронного архива имеет следующий вид:

$$oV_{dc_i} = \{\mu_{tmp_1}(dc_i)/tmp_1, \mu_{tmp_2}(dc_i)/tmp_2, \ldots, \mu_{tmp_s}(dc_i)/tmp_s\}. \tag{3.10}$$

Фактически, выражение (3.10) представляет собой нечеткое множество, построенное на множестве шаблонов проектирования онтологии проектных диаграмм, где $\mu_{tmp_j}(dc_i)$ – есть степень принадлежности проектной диаграммы dc_i шаблону проектирования tmp_j.

3.4. Формальная модель концептуального индекса

Для решения задач анализа большой совокупности текстовых документов традиционно применяются индексы в виде инвертированных списков и их различные модификации [45]. Для задач интеллектуального анализа информационных ресурсов проектных репозиториев САПР АС, решения которых основываются на состоянии предметной области в виде онтологии, представление проектного индекса в виде инвертированного списка уже не может считаться адекватным. Действительно, в основе любого индекса репозитория текстовых

документов лежит учет только внутренних свойств информационных ресурсов. К таким свойствам можно отнести частоты встречаемости терминов в документах.

Концептуальное индексирование ТД и проектных диаграмм предполагает представление информационного ресурса в системе координат, которая задается онтологией ИПР. Данное положение легло в основу построения формального описания концептуального индекса документальной информационной базы САПР АС.

При формализации концептуального индекса речь идет о моделировании трудноформализуемой совокупности документов, содержащей качественные взаимосвязи. Следовательно, инцидентность между элементами данной совокупности можно считать нечеткой. В этом случае в качестве моделей концептуального индекса могут применяться нечеткие гиперграфы, сочетающие в себе достоинства как нечетких, так и графовых моделей, и позволяющие строить удобно программируемые формальные оптимизационные и поисковые процедуры. Это позволяет сократить время поиска решений при автоматизированном проектировании сложных систем.

3.4.1. Формализация концептуального индекса текстовых технических документов

Будем использовать определение нечеткого неориентированного гиперграфа из работ [122], [5], применив его к задаче интеллектуального анализа проектно-технической документации.

Пусть $C = \{c_i\}$, $i \in I = \{1, 2, 3, ..., n\}$ – полное множество понятий из онтологии предметной области проектного репозитория; $D = \{\tilde{d}_j\}$, $j \in J = \{1, 2, 3, ..., m\}$ – множество нечетких подмножеств, определенных на множестве C. Пара множеств $\widetilde{CI}_{doc} = (C, D)$ будет определять нечеткий неориентированный гиперграф, если $\tilde{d}_j \neq \varnothing, j \in J$ и $\underset{j \in J}{\cup} \tilde{d}_j = C$. В этом случае $c_1, c_2, ..., c_n \in C$ являются вершинами гиперграфа. Множество концептуальных представлений

документов D, которое состоит из $\tilde{d}_1, \tilde{d}_2, ..., \tilde{d}_m$, есть множество нечетких ребер гиперграфа.

Метод концептуального индексирования предполагает, что каждый документ электронного архива имеет концептуальное представление. Тогда множество $D = \{\tilde{d}_j\}$ будет представлять множество ТД в концептуальном индексе проектной организации, а $\tilde{d}_j$ – концептуальное представление j-го ТД. Следовательно, нечеткий неориентированный гиперграф

$$\widetilde{CI}_{doc} = (C, D) \tag{3.11}$$

формально определяет концептуальный индекс документальной базы (рисунок 3.6).

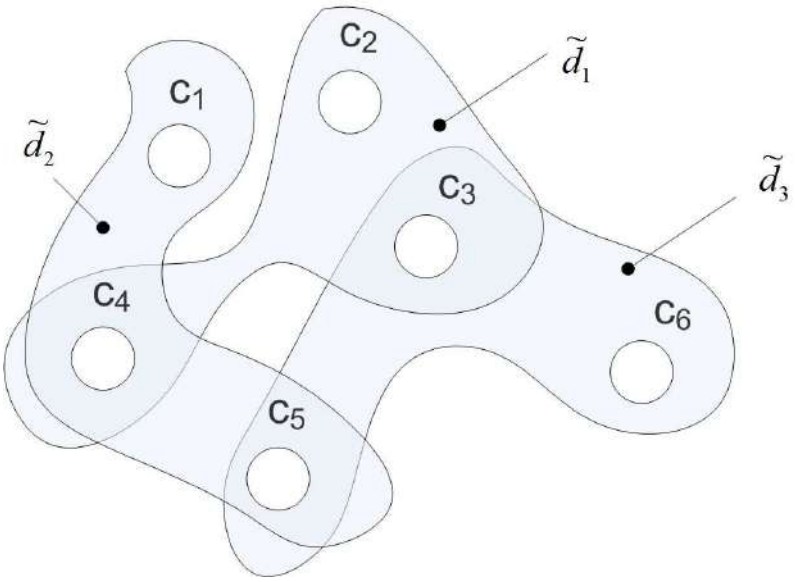


Рис. 3.6. Иллюстративный пример фрагмента концептуального индекса документальной базы электронного архива

Понятия c_α и c_β из состава концептуального индекса являются нечетко смежными, если существует технический документ (формально определяемый как нечеткое ребро гиперграфа концептуального индекса), который включает указанные понятия. При этом величина

$$\mu(c_\alpha, c_\beta) = \bigvee_{d_i \in D} \mu_j(c_\alpha, c_\beta), \text{ где} \tag{3.12}$$

$$\mu_j(c_\alpha, c_\beta) = \mu_{d_j}(c_\alpha) \& \mu_{d_j}(c_\beta)$$

будет определять степень смежности понятий c_α и c_β. Значение $1 - \mu(c_\alpha, c_\beta)$ – есть не что иное, как расстояние между понятиями c_α и c_β, которое учитывает семантическое содержание текстового ТД.

Семантическое расстояние между понятиями может использоваться при уточнении контекстно-ориентированного запроса пользователя к электронному архиву документов для улучшения показателей точности и полноты.

Два технических документа $\tilde{d_\gamma}$ и $\tilde{d_\delta}$ будем считать нечетко смежными при условии, если $\tilde{d_\gamma} \cap \tilde{d_\delta} \neq \varnothing$. Величина

$$\mu(\tilde{d_\gamma}, \tilde{d_\delta}) = \bigvee_{c \in (d_\gamma \cap d_\delta)} \mu_{d_\gamma \cap d_\delta}(c) \tag{3.13}$$

определяет степень смежности документов $\tilde{d_\gamma}$ и $\tilde{d_\delta}$. Величина $1 - \mu(\tilde{d_\gamma}, \tilde{d_\delta})$ формально представляет расстояние между документами в информационной базе, основываясь на содержании документов и онтологии проектного репозитория. Вычисляемое семантическое расстояние между документами применяется в задаче формирования навигационной структуры содержимого электронного архива. Элементом навигационной структуры может быть определен кластер документов. При этом, важным компонентом целевой функции в задаче кластеризации является расстояние между центром кластера и анализируемыми документами архива.

Основываясь на результатах исследований, приведенных в работе [5], определим операцию транспонирования матрицы инциденций нечеткого гиперграфа. Представим двойственный концептуальный индекс в следующем виде:

$$\widetilde{CI}^*_{doc} = (D, \tilde{C}_1, \tilde{C}_2, ..., \tilde{C}_n).$$

Вершинами данного гиперграфа являются документы $d_1, d_2, ..., d_m$, соответственно, нечеткими ребрами являются нечеткие подмножества $\tilde{C}_1, \tilde{C}_2, ..., \tilde{C}_n$ в множестве D, где

$$\tilde{C}_i = \{< \mu_{C_i}(d_j)/d_j > / \mu_{C_i}(d_j) = \mu_{d_j}(c_i), j \leq m\}, i = 1, 2, ..., n.$$

Если два понятия c_α и c_β (вершины гиперграфа) концептуального индекса $\widetilde{CI}_{doc}$ смежны со степенью $\mu(c_\alpha, c_\beta)$, то им соответствуют в $\widetilde{CI}^*_{doc}$ смежные понятия c_α и c_β (ребра гиперграфа) с той же степенью смежности. Если документы $\tilde{d}_\alpha$ и $\tilde{d}_\beta$ (ребра гиперграфа) концептуального индекса $\widetilde{CI}_{doc}$ смежны со степенью $\mu(\tilde{d}_\alpha, \tilde{d}_\beta)$, то им соответствуют документы $\tilde{d}_\gamma$ и $\tilde{d}_\delta$ (вершины), смежные в $\widetilde{CI}^*_{doc}$ с той же степенью.

Индекс $\widetilde{CI}^*_{doc}$ получается из $\widetilde{CI}_{doc}$ путем транспонирования матрицы инциденций, поэтому можно записать следующее выражение:

$$(\widetilde{CI}^*_{doc})^* = \widetilde{CI}_{doc}.$$

С помощью нечетких неориентированных гиперграфов можно представить концептуальные индексы документальных баз данных с учетом того, что в процессе индексирования определяются значимые понятия предметной области безотносительно к порядку их следования в тексте индексируемого документа. В этом случае можно лишь сделать вывод о том, каким образом ТД отображается на множество понятий прикладной онтологии в целом в процессе концептуального индексирования. В том случае, когда концептуальное индексирование выполняется по методу, предполагающему выделение в ТД фрагментов, для каждого из которых ставится в соответствие доминирующее понятие из онтологии, имеет важное значение порядок следования определенных во время индексирования текстовых фрагментов.

Для формализации представления такого рода концептуального индекса (будем называть такой индекс последовательным) применим модель нечеткого ориентированного гиперграфа [5].

Последовательным концептуальным индексом будем называть и через

$$\widetilde{CS}_{doc} = (C, S) \tag{3.14}$$

обозначать пару множеств, в которой $C = \{c_i\}, i \in I = \{1, 2, ..., n\}$ – множество понятий онтологии ИПР; $S = \{\tilde{s_j}\}, j \in J = \{1, 2, ..., m\}$ – множество нечетких

ориентированных онтологических представлений документов, причем каждый документ

$$\tilde{s_j} =<< \mu_{s_j}(c_{i_1})/c_{i_1} >, < \mu_{s_j}(c_{i_2})/c_{i_2} >, ..., < \mu_{s_j}(c_{i_k})/c_{i_k} >>$$

есть нечеткий кортеж с C. Здесь $c_{i_1}, c_{i_2}, ..., c_{i_k} \in C$, а $\mu_{s_j}(c_i)$ – функция принадлежности, определяющая степень инцидентности документа $\tilde{s_j}$ и понятия c_i для всех $c_i \in C$. Следует уточнить, что элементы $< \mu_{s_j}(c_\alpha) >, c_\alpha \in C$, для которых $\mu_{s_j}(c_\alpha) = 0$, в кортеж $\tilde{s_j}$ не входят.

Заметим, что некоторое понятие $c_\gamma \in C$ может встречаться в документе $\tilde{s_j}$ неоднократно в полном соответствии с методом формирования последовательности текстовых фрагментов, который приведен выше в данной работе. В этом случае $\mu_{s_j}(c_\gamma)$ может иметь различные значения в зависимости от места c_γ в документе (номера текстового фрагмента, где понятие c_γ определено как доминирующее). Каждый кортеж $\tilde{s_j}$ в последовательном концептуальном индексе имеет конечную длину, принимая во внимание конечное количество найденных в процессе концептуального индексирования текстовых фрагментов для каждого документа.

3.4.2. Формализация концептуального индекса проектных диаграмм

Формальная структура концептуального индекса проектных диаграмм является более сложной, чем для текстовых технических документов. Причина этого состоит в том, что при концептуальном индексировании проектных диаграмм, являющихся составными элементами информационных ресурсов электронного архива, учитывается как текстовая составляющая (комментарии в программном коде, различные инструкции и т.д.), так и элементы слабоформализованных нотаций языка представления проектных диаграмм.

Также, как и в случае документальных ресурсов $C = \{c_i\}$, $i \in I = \{1, 2, 3, ..., n\}$ – конечное множество понятий предметной области, зафиксированных в онтологии. Множество шаблонов проектных диаграмм в онтологии

обозначим как $T = \{tmp_k\}$, $k \in K = \{1, 2, 3, ..., l\}$. Множество проектных диаграмм обозначим как $Dc = \{\tilde{dc}_j\}$, $j \in J = \{1, 2, 3, ..., m\}$ – семейство нечетких подмножеств в $C \cup T$. Тройка $\widetilde{CI}_{prj} = (C, T, Dc)$ называется нечетким неориентированным гиперграфом при условии, когда $\tilde{dc}_j \neq \varnothing, j \in J$ и $\underset{j \in J}{\cup} \tilde{dc}_j = C \cup T$. При этом концепты онтологии предметной области $c_1, c_2, ..., c_n \in C$ и концепты онтологии проектных диаграмм $tmp_1, tmp_2, ..., tmp_l \in T$ являются вершинами гиперграфа, а множество концептуальных представлений проектных диаграмм Dc, состоящее из $\tilde{dc}_1, \tilde{dc}_2, ..., \tilde{dc}_m$, есть множество нечетких ребер гиперграфа.

Отдельная проектная диаграмма в результате концептуального индексирования преобразуется в онтологическое представление. Следовательно множество $Dc = \{\tilde{dc}_j\}$ будем понимать как результат концептуального отображения совокупности проектных диаграмм, а $\tilde{dc}_j$ – как концептуальное представление j-й проектной диаграммы электронного архива. Получаем, что нечеткий неориентированный гиперграф

$$\widetilde{CI}_{prj} = (C, T, Dc) \tag{3.15}$$

формально определяет концептуальный индекс базы проектов (рисунок 3.7).

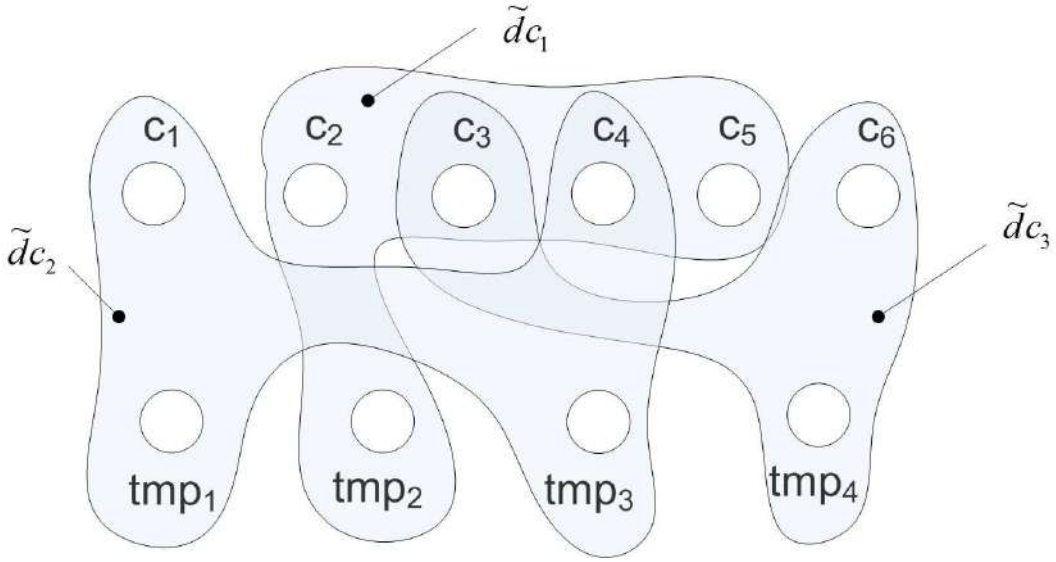


Рис. 3.7. Иллюстративный пример фрагмента концептуального индекса проектных диаграмм

Две проектных диаграммы $\tilde{dc}_\gamma$ и $\tilde{dc}_\delta$ на концептуальном уровне называются нечетко смежными, если $\tilde{dc}_\gamma \cap \tilde{dc}_\delta \neq \varnothing$.

При этом, величина

$$\mu(\tilde{dc_\gamma}, \tilde{dc_\delta}) = \bigvee_{c \in (dc_\gamma \cap dc_\delta)} \mu_{dc_\gamma \cap dc_\delta}(c) \& \bigvee_{tmp \in (dc_\gamma \cap dc_\delta)} \mu_{dc_\gamma \cap dc_\delta}(tmp) \qquad (3.16)$$

есть степень смежности проектных диаграмм $\tilde{dc_\gamma}$ и $\tilde{dc_\delta}$. Величина $1-\mu(\tilde{dc_\gamma}, \tilde{dc_\delta})$ представляет меру расстояния между проектными диаграммами в информационной базе, основываясь на содержании контекстов проектов и степени принадлежности проектной диаграммы шаблонам проектирования из онтологии ИПР. Данный показатель может применяться в задаче нечеткой кластеризации проектных диаграмм электронного архива.

3.5. Выводы по третьей главе

1. Концептуальный индекс ИПР можно рассматривать как семантическое ядро электронного архива проектной организации. Основная цель формирования концептуального индекса состоит в реализации принципа семантической интероперабельности процедур анализа ресурсов проектного репозитория вне зависимости от вида ресурсов: текстовые технические документы или проектные диаграммы.
2. Предложенный метод концептуального индексирования позволяет представить слабоструктурированный ресурс электронного архива САПР АС как множество отображений фрагментов ресурса на прикладную онтологию ИПР. Онтология выполняет роль согласованной модели представления знаний в проектной организации.
3. Применение математических моделей формирования онтологического представления информационного ресурса ИПР на основе нечетких соответствий и нечетких гиперграфов позволяет учесть принципиальную неполноту описания ресурсов электронного архива.

Глава 4

Интеллектуальный анализ информационных ресурсов проектной организации

4.1. Структуризация документальных информационных баз

4.1.1. Онтологическая модель технического документа

Технический документ (ТД) будем рассматривать в контексте интеллектуального анализа слабоструктурированных информационных ресурсов электронного архива проектной организации. Анализируя любой ТД, с одной стороны, приходится иметь дело с текстом на естественном языке, а с другой стороны, ТД имеет структуру, которая определяется стандартами проектирования и некоторыми нормативными документами [70], [73].

При рассмотрении ТД с концептуальной точки зрения отдельно взятый документ включает в себя некоторый набор данных. В этой связи онтология предметной области может быть рассмотрена, как носитель метаданных документальной базы. Способ взаимодействия ТД с онтологией показан на рисунке 4.1. Множество связей ТД с концептами онтологии рассматриваются относительно двух основных онтологических ресурсах: метауровень понятий онтологии предметной области и множество терминов тезауруса проектной организации.

Рассмотрим основные связи между документом и концептами онтологии:

- семантическая связь «connect_to» определяет в проектном репозитории документальных информационных ресурсов принадлежность документа подмножеству понятий онтологии, извлекаемых из применяемых в проектной организации стандартов, и из реализованных проектах (фиксируется

основная тематика технического документа);

- семантическая связь «contain» между разделом ТД и терминами тезауруса определяет набор и частоту тех терминов, которые содержатся в разделе ТД.

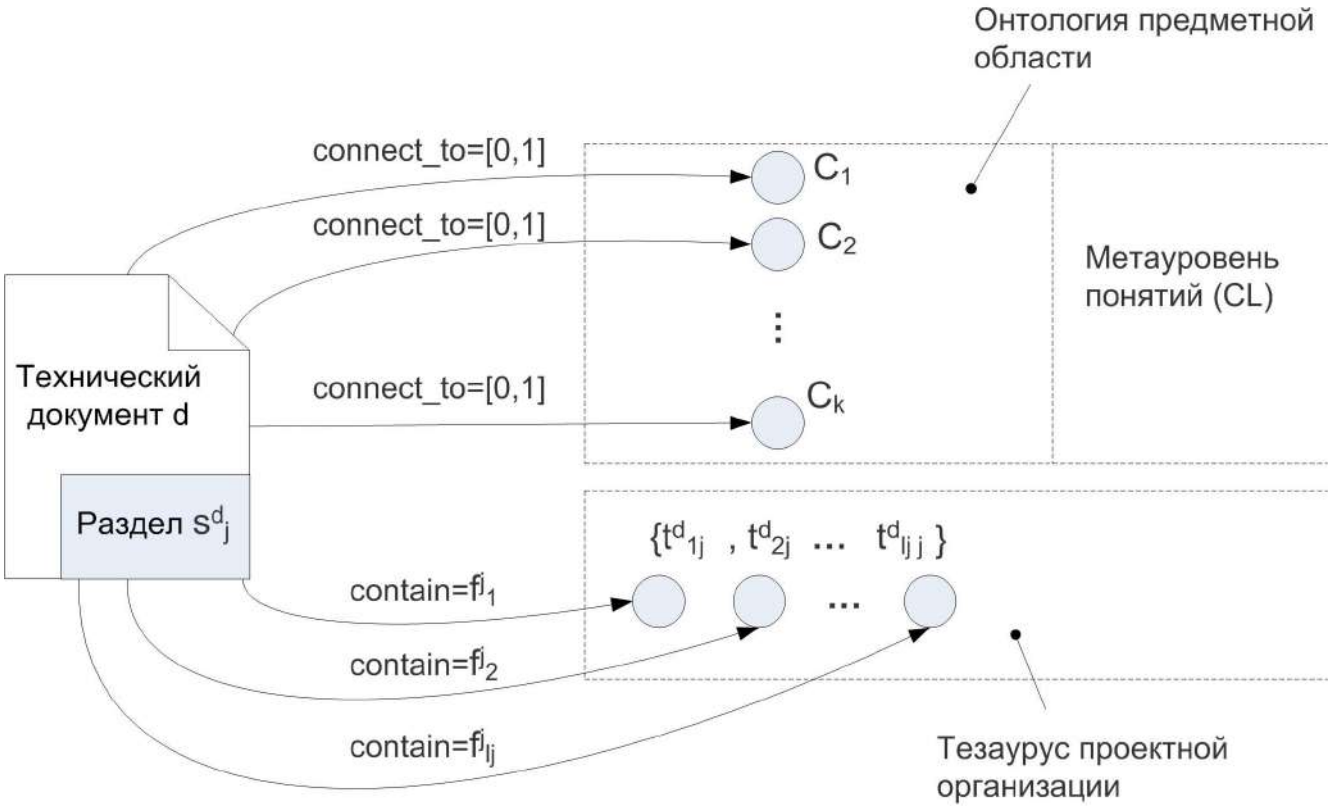


Рис. 4.1. Семантические связи между техническим документом и онтологией

Структурное представление ТД формируется совокупностью его разделов и подразделов, порядком их чередования и иерархией. Весовой коэффициент семантической связи «contain» будем обозначать как f_i^j – частоту встречаемости i-го термина в j-м разделе документа. При формировании оценочных значений терминов будем использовать следующие ограничения [49], [55]:

- часто встречающиеся в тексте термины не являются специфическими (однако, дают большой вклад в число совпадений при сравнении терминов запроса и документа);
- термины, которые редко встречаются в тексте, вносят очень небольшой вклад в качество поиска документов электронного архива (так как тер-

мины с невысокой частотой дают сравнительно небольшое число совпадений);

- наиболее полезными с точки зрения достижения показателей точности и полноты терминами являются такие термины, которые и не частые и не редкие.

Частота встречаемости терминов в документе сравнивается с частотой появления точно таких же терминов во всем электронном архиве. В том случае, если частоты терминов в анализируемом документе намного превосходят частоты терминов по всему электронному архиву, то делается вывод о том, что ценность данных терминов высока. Формализация данного ограничения выполняется следующим образом:

$$f_i = tfidf_i = tf_i \cdot log\left(\frac{N}{df(t_i)}\right),$$

где $tfidf_i$ – относительный вес термина t_i в документе; tf_i – нормализованная частота термина t_i; N – количество документов электронного архива; $df(t_i)$ – количество документов, в которых встречается термин t_i.

Структура ТД в проектном репозитории определяется составом его разделов и подразделов, месторасположением их относительно друг друга в документе. Следовательно, можно рассматривать раздел текстового технического документа как основной структурный элемент.

Под онтологической (или концептуальной) моделью текстового ТД будем считать формализованное представление документа, которое определяется состоянием множества артефактов проекта, определяемого текущей стадией жизненного цикла проектируемой системы. Аналогичная модель в работах Загорулько, включающая в себя семантический индекс, носит название электронного паспорта документа [33], [34].

Формально, структурную единицу ТД запишем в следующем виде:

$$oV_j^d = \langle ch, C_s^P, C_s^{St} \rangle, s \in S^d,$$

где ch – уникальное наименование раздела ТД; $C_s^P \cup C_s^{St}$ – подмножество понятий предметной области автоматизированного проектирования АС, ассоциированных с разделом ch.

Дадим формальное описание ТД с учетом предложенной структуры онтологии (2.7). Предположим, что для документа d выполняются следующие условия:

$$\begin{gathered} connect\_to(d, C_1^{P(St)}) = 1, \\ connect\_to(d, C_2^{P(St)}) = 1, \\ \dots \\ connect\_to(d, C_k^{P(St)}) = 1. \end{gathered} \tag{4.1}$$

Набор условий (4.1) означает, что документ d отображается в пространство понятий $C_1^{P(St)}, C_2^{P(St)}, \dots, C_k^{P(St)}$.

Пусть t_{ij}^d – i-й термин j-го раздела документа d. В этом случае, множество терминов j-го раздела текстового документа d запишем следующим образом:

$$T_j^d = \{t_{1j}^d, t_{2j}^d, \dots, t_{l_j j}^d\},$$

где l_j – количество терминов j-го раздела ТД d.

Предположим, что для j-го раздела документа d выполняются следующие ограничения:

$$\begin{cases} contain\left(s_j^d, t_{1j}^d\right) = f_1^j, \\ contain\left(s_j^d, t_{2j}^d\right) = f_2^j, f_1^j, f_2^j, \dots, f_{l_j}^j > 0, \\ \vdots \\ contain\left(s_j^d, t_{l_j j}^d\right) = f_{l_j}^j. \end{cases}$$

Тогда, применяя функцию интерпретации $F_{TC} : \{T\} \to \{C\}, C = C^P \cup C^{St}$ онтологии на этапе концептуального индексирования ТД, получаем онтологическое представление раздела документа в виде:

$$oV_j^d = \langle ch_j, C_j^P, C_j^{St} \rangle, C_j^P \subseteq C^P, C_j^{St} \subseteq C^{St} \mid_{St_k^{LC}} . \tag{4.2}$$

В выражении (4.2) запись $C_j^{St} \subseteq C^{St} \mid_{St_k^{LC}}$ означает, что в онтологическое представление ТД включены только те понятия онтологии предметной области, принадлежащие множеству C^{St} (извлекаемых из стандартов), которые соответствуют k-й стадии (этапу) проектирования St_k^{LC}.

Формальную онтологическую модель ТД будем записывать следующим образом:

$$oV^d = \langle S^d, \{C_d^P \cup C_d^{St}\}\rangle,$$

в которой можно выделить следующие части: структурную (S^d) и содержательную ($\langle C_d^P \cup C_d^{St}\rangle$) с учетом реализованных проектов из электронного архива и стандартов, применяемых в процессе автоматизированного проектирования АС.

4.1.2. Семантическая мера расстояния между документами

Для определения расстояния между отдельными ТД в электронном архиве необходимо измерить степень схожести между онтологическими представлениями ТД [60], [61], [63], [66], [67], [110]. В данной работе формальная мера семантического расстояния между документами рассматривается в контексте, который формируют понятий онтологии проектного репозитория, извлекаемые из стандартов.

Определив множество понятий $C_d^S \subseteq C^S$, извлекаемых из стандартов и выявленных в анализируемом документе d, а также имея их отображение на структуру понятий онтологии проектного репозитория, появляется возможность отдельное онтологическое представление документа преобразовать в дерево (иерархию) понятий предметной области. Такая иерархия определяется через процедуру нахождения такого минимального дерева, которое включает только те понятия онтологии, которые присутствуют в онтологическом представлении документа.

Онтологическое представление ТД будем рассматривать как иерархию. В

свою очередь, расстояние между содержанием двух ТД будем находить, применяя метод вычисления сложности преобразования одной иерархии в другую. Данный метод основывается на вычислении степени различия совокупности дуг, соединяющих понятия [37], [86].

В рамках данного исследования будем использовать определение «иерархии» так, как оно было сформулировано в работе [31]. Обозначим через C то множество понятий, которое включается в онтологическое представление анализируемого документа, $C = c_1, c_2, \ldots, c_l, \ldots, c_q$, а H есть совокупность *таксонов*, каждый из которых обозначим h.

Иерархией H понятий C называется множество подмножеств C таких, что выполняются следующие ограничения:

- $\forall c \in C, \{c\} \in H$ (терминальные вершины (листья) – одноэлементные множества понятий, для которых не существует семантических отношений с понятиями более низкого уровня);
- $C \in H$ (таксон, соответствующий корневому понятию онтологического представления ТД, и поглощающий все концепты C);
- для любых двух вершин $h, h' \in H$ имеется возможность записать: $h \cap h' = 0$, или $h \subset h'$, или $h' \subset h$.

Таким образом, иерархическая модель онтологического представления документа – это многоуровневая модель, которая предполагает, что понятия онтологии, включаемые в некоторый таксон на уровне j, также остаются в одном и том же таксоне на $j + 1$-м и далее на всех других более высоких уровнях.

На первом уровне располагаются терминальные вершины (соответствуют п. 1 в определении иерархии). На последнем, максимальном, уровне (обозначается как m) располагается наибольший таксон, который содержит все понятия, включаемые в документ. Данный таксон можно обозначить таким же символом C (соответствует п. 2 в определении иерархии). Объединение таксонов происходит или не происходит на каждом уровне (соответствует п. 3 в определении иерархии).

Для удобства графического представления обозначим каждый таксон иерархии отдельной точкой. Вершины иерархии будут соответствовать именам понятий c_{jd} в онтологическом представлении документов электронного архива. Понятия, включаемые в иерархию, связаны семантическими отношениями вида «is_A» (обобщение) (рисунок 4.2). В качестве примера рассмотрим задачу определения расстояния между двумя документами, которые имеют представления в виде иерархий H^1 (первый документ) и H^2 (второй документ).

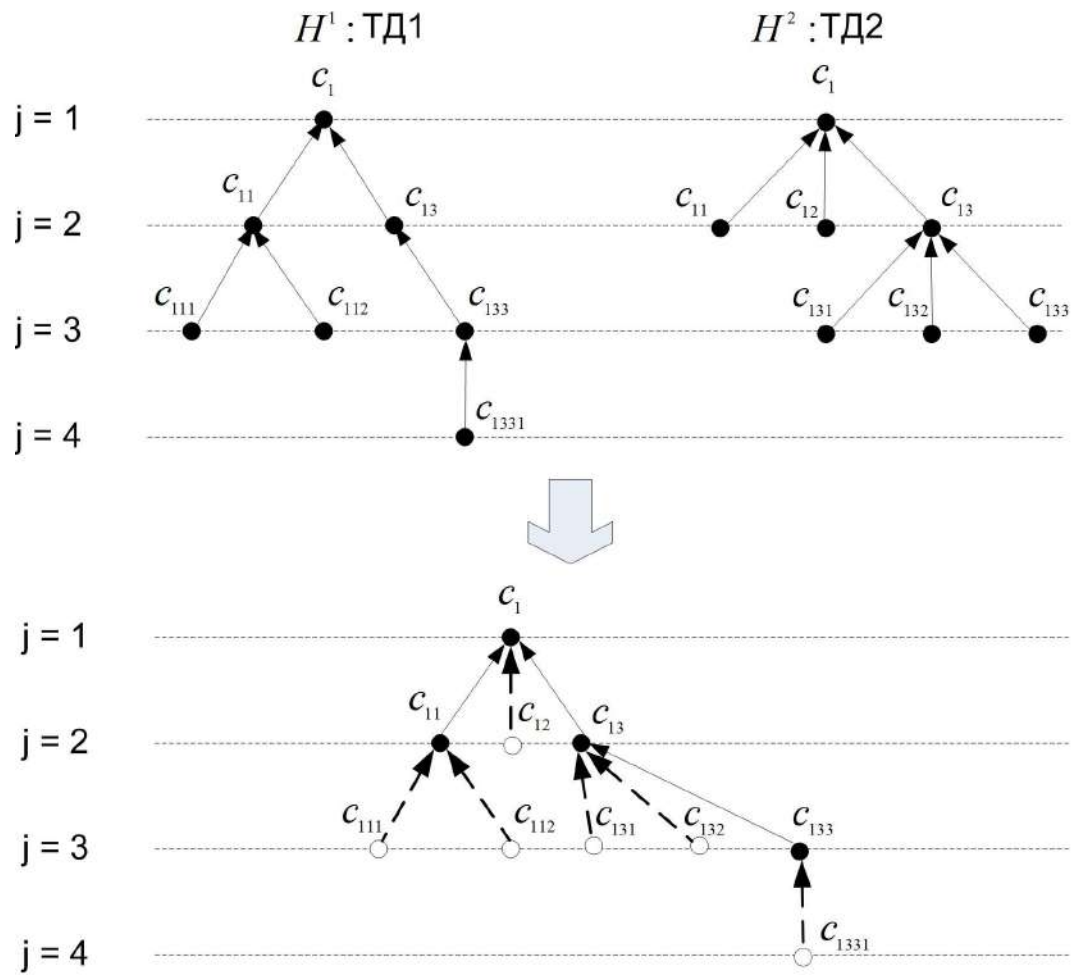


Рис. 4.2. Иерархическое представление двух документов

Для определения редакционного расстояния между иерархиями производится вычисление стоимости так называемой редакционной операции, которую будем записывать как $\varphi_{S_i}(R_G)$. Индекс S_i определяет принадлежность значения редакционной операции к i-ой серии стандартов. Таким образом, в задаче структуризации содержимого электронного архива ТД в качестве результата

редакционной операции считается число, которое принимает значение на числовом отрезке.

На рисунке 4.2 схематично представлен процесс нахождения множества отношений для двух сравниваемых онтологических представлений документов репозитория. Отношения, которыми дополняется иерархия при определении значения семантического расстояния между ТД, показаны на рисунке штриховыми линиями.

В результате редакционное расстояние между иерархиями определяется по следующей формуле:

$$\tau_{oV}^{*} = \max_{i} \left(\sum_{s=1}^{m} \varphi_{S_i}(R_G)_s \right), \tag{4.3}$$

где i – индекс серии используемых на предприятии стандартов проектирования; s – индекс отношения обобщения, добавляемого к иерархии. Итоговое редакционное расстояние, как видно из выражения (4.3), вычисляется как наибольшее из редакционных расстояний, которые определяются для каждой серии стандартов отдельно.

Коэффициент нормализации T_{oV} определяется путем суммирования всех весовых коэффициентов отношений результирующей иерархии (рисунок 4.2). Таким образом, семантическое расстояние между двумя онтологическими представлениями текстовых ТД определяется с помощью следующего выражения:

$$\left\| oV^{d_1} - oV^{d_2} \right\| = \frac{\tau_{oV}^{*}}{T_{oV}} \tag{4.4}$$

4.1.3. Метод построения навигационной структуры на основе нечеткой кластеризации

В основе метода структуризации документальных информационных баз проектной организации лежит алгоритм нечеткой кластеризации (Fuzzy C-Means), особенность которого состоит в том,что он допускает ситуацию, что один объект может принадлежать сразу нескольким кластерам с различной степенью

принадлежности [115]. В данной работе предлагается использовать модифицированный алгоритм нечеткой кластеризации документальной базы информационных ресурсов проектного репозитория, который минимизирует целевую функцию следующего вида:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| oV_i - oV_j^c \right\|^2, 1 \leq m < \infty,$$

где C – количество кластеров навигационной структуры на одном уровне иерархии; m – действительное число больше 1; N – количество текстовых ТД (их онтологических представлений) для кластеризации; u_{ij} – степень принадлежности онтологического представления oV_i кластеру j; oV_i – i-ое онтологическое представление; $\| * \|$ – значение нормализованного расстояния между центром кластера и онтологическим представлением документа; oV_j^c – центр j-го кластера.

Функционирование алгоритма нечеткой кластеризации предполагает выполнения следующих шагов.

Шаг 1. Задают параметры кластеризации и инициализируют первоначальную матрицу принадлежности онтологических представлений ТД кластерам $U = [u_{ij}]$.

Шаг 2. Вычисляют последующие (новые) значения центров кластеров:

$$oV_j^c = \frac{\sum_{i=1}^{N} u_{ij}^m \cdot oV_i}{\sum_{i=1}^{N} u_{ij}^m}.$$

Шаг 3. Формируется новое состояние матрицы принадлежности:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\left\| oV_i - oV_j^c \right\|}{\| oV_i - oV_l^c \|} \right)^{\frac{2}{m-1}}},$$

где u_{ij} – степень принадлежности i-го ТД кластеру j; oV_j^c – онтологическое представление центра j-го кластера; oV_l^c – онтологическое представление центра l-го кластера навигационной структуры.

Шаг 4. Вычисляют значение целевой функции и результат сравнивают со значением, которое получено на предыдущей итерации цикла. В том случае, если значение разности не превышает некоторого порога, задаваемого в параметрах функции кластеризации, делается вывод о том, что для текущего иерархического уровня навигационной структуры кластеризация завершена. Иначе осуществляется переход на второй шаг алгоритма кластеризации.

В процессе выполнения структуризации массивов технических документов проектных организаций создаются такие структуры содержимого архивов, которые позволяют выполнять навигацию по документам. В рамках первого шага процесса кластеризации электронного архива анализируется полное множество ТД. После формирования первой группы кластеров документов, каждая группа в отдельности определяется алгоритмом как объект выполнения процедуры кластеризации на следующем уровне. Алгоритм останавливается тогда, когда нет необходимости в дальнейшей декомпозиции содержимого электронного архива (количество документов в кластерах становится таким, что у проектировщика есть возможность их просмотра и изучения в сжатые временные сроки).

Множество кластеров, которые организованы в иерархическую структуру, фактически, представляют собой навигационную структуры электронного архива проектной организации. Для того, что бы такая структура была оптимальной, необходимо решить задачу определения значений весовых коэффициентов для семантических отношений между понятиями онтологии предметной области, которые соответствуют стандартам на предприятии. Выше в работе данные весовые коэффициенты были заданы как $\varphi_{S_i}(R_G)$ для вида семантического отношения «is_A» (обобщение).

Так как данный вид семантического отношения используется между понятиями в онтологии для различных серий стандартов, возникает задача нахождения оптимальных значений весовых коэффициентов для каждой такой серии. Сформулируем *принцип оптимальности* для множества весовых коэффициентов семантических отношений онтологии.

Допустим, что $\{oV^d\}^*$ – есть множество онтологических представлений документов, входящих в заранее определенное множество документов. Для данного множества известно экспертное разделение документов на классы такое, что:

$$\{oV^d\}^* \subset \{oV^d\},$$

где $\{oV^d\}$ – множество онтологических представлений всех ТД из электронного архива. Существует онтология, которая определяется выражением (2.7), где заданы отношения обобщения на множестве понятий с соответствующими весовыми коэффициентами $\varphi_{S_i}(R_G)$, где S_i – i-я серия стандартов, из которой извлекаются понятия онтологии.

Множество $\{oV^d\}^*$ включает в себя два подмножества: $\{oV^d\}^*_+ \cup \{oV^d\}^*_-$, которые создаются априорно экспертами в предметной области. Решение задачи оптимизации весовых коэффициентов семантических отношений нацелена на поиск множества таких коэффициентов:

$$\{\varphi^*_{S_1}(R_G), \varphi^*_{S_2}(R_G), \ldots, \varphi^*_{S_i}(R_G), \ldots, \varphi^*_{S_n}(R_G)\},$$

при которых определяемое выражением

$$F^* = \frac{\max\left(\bar{K}_+ + \bar{K}_-, \hat{K}_+ + \hat{K}_-\right)}{N} \to \min \tag{4.5}$$

качество структуризации является оптимальным. В выражении (4.5) $\bar{K}_-$ и $\hat{K}_-$ – множества документов, которые отсутствуют в первом кластере и во втором кластере, соответственно; $\bar{K}_+$ и $\hat{K}_+$ – множества документов, которые не должны присутствовать в первом кластере и во втором кластере, соответственно; N – количество документов.

При реализации генетического алгоритма оптимизации весовых коэффициентов необходимо уточнить его параметры и структурные элементы. При формализации целевой функции применяется оценка качества структуризации

множества $\{oV^d\}^*$ (4.5). Хромосома, которая представляет потенциальное решение задачи оптимизации, имеет вид, представленный на рисунке 4.3, где представлен иллюстративный пример операции кроссинговера (4.3, а) и пример операции мутации (4.3, б).

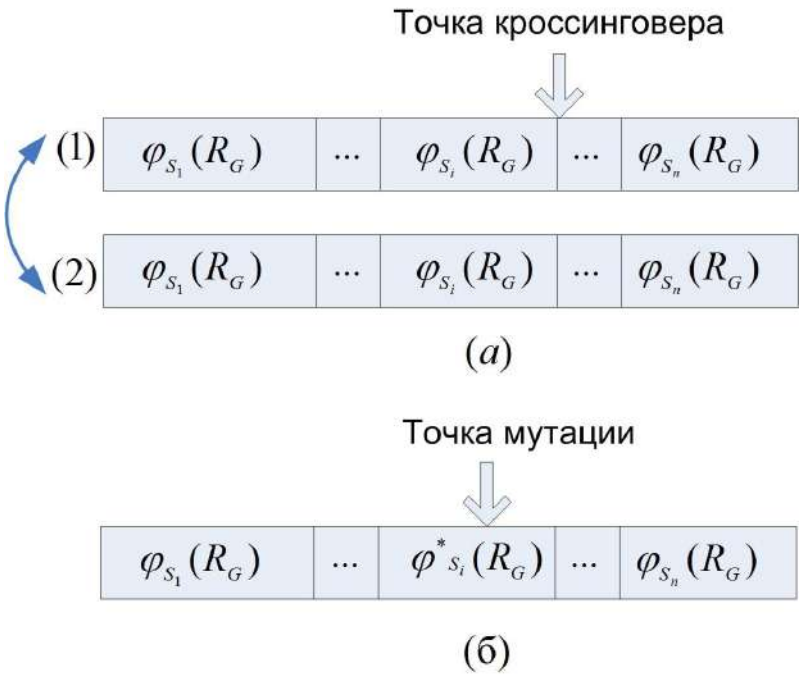


Рис. 4.3. Иллюстративные примеры операций генетического алгоритма оптимизации семантических коэффициентов

Ряд параметров генетического алгоритма, такие как вероятность мутации участков хромосом (случайной модификации значений весов коэффициентов семантических отношений), количество хромосом, которые переходят в следующий пул без каких-либо изменений(число элитизма), размер популяции, определяются во время выполнения вычислительных экспериментов при настройке алгоритма под рассматриваемую задачу.

4.2. Модели содержательной интерпретации ресурсов интеллектуального проектного репозитория

4.2.1. Интерпретация кластеров информационных ресурсов

В процессе решения задачи иерархической кластеризации документов электронного архива возникает новая проблема определения обобщенного описания группы документов, расположенных в одном кластере. Другими словами, как можно выполнить *содержательную интерпретацию* сформированных кластеров?

Каждый кластер содержит документы, так или иначе похожие друг на друга. В свою очередь, теория приближенных множеств Павлака (Rough Sets) основывается на понятии *отношения неразличимости* между объектами некоторого универсума [173], [174], [175]. Разные сочетания атрибутов, которые характеризуют объекты, в общем случае приводят к различным отношениям неразличимости.

Содержимое кластера технических документов можно представить в виде неопределенного понятия, которое сложно охарактеризовать, используя информацию об только отдельных его элементах [175]. Согласно подходу приближенных множеств Павлака, будем описание кластера представлять в виде набора ограничивающих его условий.

Таким образом будем различать экстенсиональное и интенсиональное описание кластера технических документов электронного архива. Экстенсиональное описание кластера формируется с помощью перечисления элементов (в данном случае – технических документов), которые принадлежат кластеру. Интенсиональное описание представляет собой краткую, компактную форму представления содержимого кластера в терминах предметной области.

Каждый ТД, как было определено выше, будем записывать в виде нечеткого вершинного подграфа дерева онтологии ИПР. Процесс отображения ТД на

концептуальную плоскость позволяет выполнять интеллектуальной анализ документов, основываясь не на используемой в документах лексике, а используя систему понятий предметной области автоматизированного проектирования из онтологии. Иллюстративный пример содержимого репозитория из нескольких упрощенных онтологических представлений приведен на рисунке 4.4.

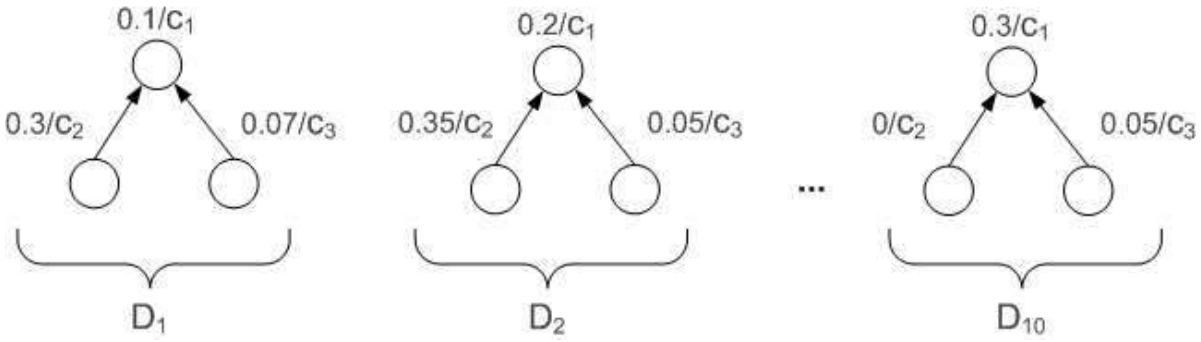


Рис. 4.4. Пример онтологических представлений документов репозитория

Для того, чтобы иметь возможность определить отношения неразличимости на полном множестве ТД, построим терм-множество лингвистической переменной «степень выраженности понятия онтологии». На рисунке 4.5 представлена структура данной лингвистической переменной.

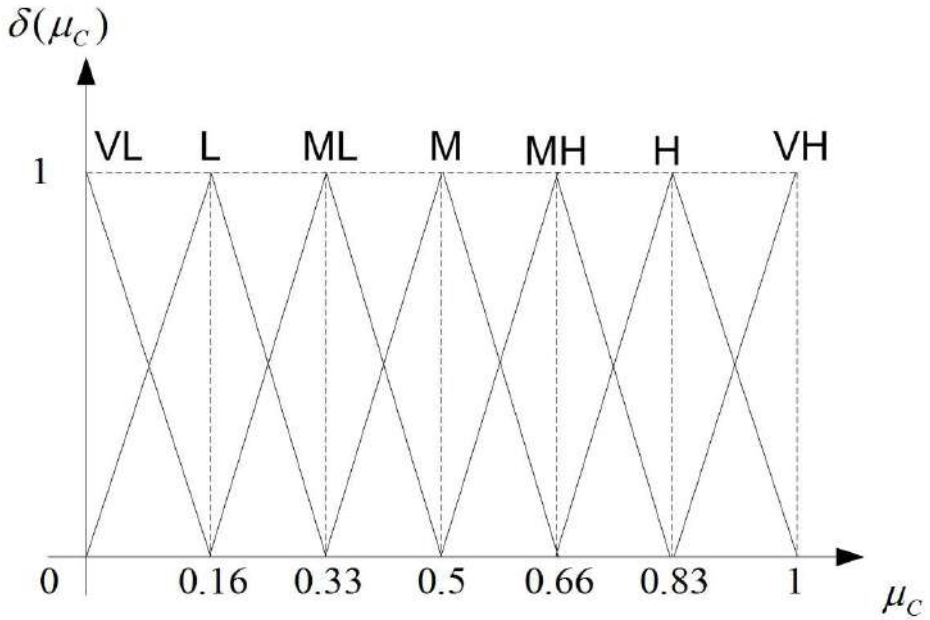


Рис. 4.5. Терм-множество лингвистической переменной «степень выраженности понятия онтологии»

Будем использовать следующие условные обозначения:

- μ_{C_i} – степень выраженности понятия C_i онтологии, включенного в концептуальный индекс;
- $\delta(\mu_{C_i})$ – функция принадлежности лингвистической переменной;
- very low (VL), low (L), middle low (ML), middle (M), middle-high (MH), high (H), very high (VH) – набор термов лингвистической переменной.

Для примера сделаем предположение, что в проектном репозитории находятся 10 ТД, фрагменты онтологических представлений которых показаны на рисунке 4.6. Все множество документов разделено на два класса ($K = \{1, 2\}$). Пусть в результате концептуального индексирования были получены значения степеней выраженности понятий C_1, C_2 и C_3 для указанного множества документов, так как показано на рисунке 4.6, а.

D	C_1	C_2	C_3	K
D_1	0,1	0,3	0,07	1
D_2	0,2	0,35	0,05	1
D_3	0,33	0,02	0	1
D_4	0	0	0,15	2
D_5	0,4	0,18	0,01	2
D_6	0	0,03	0,1	2
D_7	0,28	0,02	0	1
D_8	0,01	0,15	0,3	2
D_9	0,39	0,19	0	2
D_{10}	0,3	0	0,05	1

а)

D	C_1	C_2	C_3	K
D_1	L	ML	VL	1
D_2	L	ML	VL	1
D_3	ML	VL	VL	1
D_4	VL	VL	L	2
D_5	ML	L	VL	2
D_6	VL	VL	L	2
D_7	ML	VL	VL	1
D_8	VL	L	ML	2
D_9	ML	L	VL	2
D_{10}	ML	VL	VL	1

б)

Рис. 4.6. Значения степеней выраженности понятий иллюстративного примера

На рисунке 4.6, б представлены значения степеней выраженности понятий

для тех же самых документов, что и на рисунке 4.6, а, но не в виде числовых значений, а как термы лингвистической переменной, приведенной на рисунке 4.5. В алгоритме перевода степени выраженности понятий онтологии из числовой формы в лингвистическую применяется правило: в случае равенства значений принадлежности для двух соседних термов, окончательный выбор делается в пользу большего терма (на рисунке 4.6, б, например, среди термов H и VH будет выбран терм VH).

Для формирования интенсионального описания кластеров используется система правил, которые фактически формируют границы классов. На рисунке 4.6, б) представлен результат перевода множества правил в логическую форму, где множество документов (как элементов кластеров) записано в дизъюнктивной нормальной форме:

$$\begin{cases} [(C_1 = L) \wedge (C_2 = ML) \wedge (C_3 = VL) \wedge (K = 1)] \vee \\ [(C_1 = L) \wedge (C_2 = ML) \wedge (C_3 = VL) \wedge (K = 1)] \vee \\ [(C_1 = ML) \wedge (C_2 = VL) \wedge (C_3 = VL) \wedge (K = 1)] \vee \\ [(C_1 = VL) \wedge (C_2 = VL) \wedge (C_3 = L) \wedge (K = 2)] \vee \\ [(C_1 = ML) \wedge (C_2 = L) \wedge (C_3 = VL) \wedge (K = 2)] \vee \\ [(C_1 = VL) \wedge (C_2 = VL) \wedge (C_3 = L) \wedge (K = 2)] \vee \\ [(C_1 = ML) \wedge (C_2 = VL) \wedge (C_3 = VL) \wedge (K = 1)] \vee \\ [(C_1 = VL) \wedge (C_2 = L) \wedge (C_3 = ML) \wedge (K = 2)] \vee \\ [(C_1 = ML) \wedge (C_2 = L) \wedge (C_3 = VL) \wedge (K = 2)] \vee \\ [(C_1 = ML) \wedge (C_2 = VL) \wedge (C_3 = VL) \wedge (K = 1)]. \end{cases}$$

Такая база правил границ кластеров может быть записана более компактно, представляя конъюнкцию как алгебраическое произведение:

$$(C_1^L C_2^{ML} C_3^{VL} K^1) \vee (C_1^L C_2^{ML} C_3^{VL} K^1) \vee (C_1^{ML} C_2^{VL} C_3^{VL} K^1) \vee$$

$$\vee (C_1^{VL} C_2^{VL} C_3^L K^2) \vee (C_1^{ML} C_2^L C_3^{VL} K^2) \vee (C_1^{VL} C_2^{VL} C_3^L K^2) \vee$$

$$\vee (C_1^{ML} C_2^{VL} C_3^{VL} K^1) \vee (C_1^{VL} C_2^L C_3^{ML} K^2) \vee (C_1^{ML} C_2^L C_3^{VL} K^2) \vee$$

$$\vee(C_1^{ML} C_2^{VL} C_3^{VL} K^1).$$

Теперь необходимо найти минимальное множество непротиворечивых правил (логических импликаций), которые характеризуют множество ТД, представленных в дизъюнктивной нормальной форме. Для множества условных атрибутов $C = \{C_1, C_2, C_3\}$ и решающего атрибута K данные правила будут иметь вид $C_i^a C_j^b \dots C_k^c \to K^d$ или более подробно:

$$(C_i = a) \wedge (C_j = b) \wedge \dots \wedge (C_k = c) \to (K = d),$$

где $\{a, b, c, \dots\}$ – допустимые значения из домена, который определяется значениями заранее определенной экспертом лингвистической переменной «степень выраженности понятий онтологии».

Метод построения логических правил основывается на формировании «решающей матрицы» (decision matrix) отдельно для каждого значения d решающего атрибута K. Элементами решающей матрицы для отдельного значения d решающего атрибута K являются списки пар «атрибут-значение», которые различаются у документов, имеющих $K = d$ и $K \neq d$.

В качестве примера рассмотрим данные в таблице на рисунке 4.6, б. Атрибут K (номер кластера) будет переменной, значений которой нужно определить, а $\{C_1, C_2, C_3\}$ – условные переменные. Искомая переменная может принимать два значения: $\{1, 2\}$.

Пусть, в качестве примера, необходимо определить правила для первого кластера (когда $K = 1$). Концептуальные представления полного множества документов разделим на представления, для которых $K = 1$, и представления, для которых $K \neq 1$. В нашем случае концептуальные представления документов, для которых $K = 1$: $\{D_1, D_2, D_3, D_7, D_{10}\}$, в то время, как $K \neq 1$ соблюдается для $\{D_4, D_5, D_6, D_8, D_9\}$. Решающая матрица для $K = 1$ содержит все различия между ТД, для которых $K = 1$, и теми, для которых $K \neq 1$. Другими словами, решающая матрица включает в себя различия между $\{D_1, D_2, D_3, D_7, D_{10}\}$ и $\{D_4, D_5, D_6, D_8, D_9\}$. «Положительные» представления документов ($K = 1$)

расположим по строкам, а «отрицательные» ($K \neq 1$) – по столбцам указанной матрицы:

	D_4	D_5	D_6	D_8	D_9
D_1	$C_1^L, C_2^{ML}, C_3^{VL}$	C_1^L, C_2^{ML}	$C_1^L, C_2^{ML}, C_3^{VL}$	$C_1^L, C_2^{ML}, C_3^{VL}$	C_1^L, C_2^{ML}
D_2	$C_1^L, C_2^{ML}, C_3^{VL}$	C_1^L, C_2^{ML}	$C_1^L, C_2^{ML}, C_3^{VL}$	$C_1^L, C_2^{ML}, C_3^{VL}$	C_1^L, C_2^{ML}
D_3	C_1^{ML}, C_3^{VL}	C_2^{VL}	C_1^{ML}, C_3^{VL}	$C_1^{ML}, C_2^{VL}, C_3^{VL}$	C_2^{VL}
D_7	C_1^{ML}, C_3^{VL}	C_2^{VL}	C_1^{ML}, C_3^{VL}	$C_1^{ML}, C_2^{VL}, C_3^{VL}$	C_2^{VL}
D_{10}	C_1^{ML}, C_3^{VL}	C_2^{VL}	C_1^{ML}, C_3^{VL}	$C_1^{ML}, C_2^{VL}, C_3^{VL}$	C_2^{VL}

Данная система выражений означает, что для переменной (номера кластера) $K = 1$, например, концептуальное представление документа D_3 отличается от представления D_6 атрибутами C_1 и C_3.

Каждая решающая матрица является источником формирования булевы выражения, каждое из которых соответствует отдельной строке матрицы. Например, для представленной выше матрице, будут сформированы следующие пять булевых выражений:

$$\begin{cases} (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge (C_1^L \vee C_2^{ML}) \wedge (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge \\ \wedge (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge (C_1^L \vee C_2^{ML}), \\ (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge (C_1^L \vee C_2^{ML}) \wedge (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge \\ \wedge (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge (C_1^L \vee C_2^{ML}), \\ (C_1^{ML} \vee C_3^{VL}) \wedge (C_2^{VL}) \wedge (C_1^{ML} \vee C_3^{VL}) \wedge C_1^{ML} \vee C_2^{VL} \vee C_3^{VL}) \wedge (C_2^{VL}), \\ (C_1^{ML} \vee C_3^{VL}) \wedge (C_2^{VL}) \wedge (C_1^{ML} \vee C_3^{VL}) \wedge C_1^{ML} \vee C_2^{VL} \vee C_3^{VL}) \wedge (C_2^{VL}), \\ (C_1^{ML} \vee C_3^{VL}) \wedge (C_2^{VL}) \wedge (C_1^{ML} \vee C_3^{VL}) \wedge C_1^{ML} \vee C_2^{VL} \vee C_3^{VL}) \wedge (C_2^{VL}) \end{cases}$$

В полученном результате имеет место некоторая избыточность. В этой связи, следующим шагом будет его упрощение на основе использования традиционной булевой алгебры. Следовательно, утверждение

$$(C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge (C_1^L \vee C_2^{ML}) \wedge (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge$$
$$\wedge (C_1^L \vee C_2^{ML} \vee C_3^{VL}) \wedge (C_1^L \vee C_2^{ML}),$$

соответствующее представлениям документов $\{D_1, D_2\}$, упрощается до $C_1^L \vee C_2^{ML}$. Данное выражение соответствует импликации:

$$(C_1 = L) \vee (C_2 = ML) \rightarrow (K = 1).$$

По аналогии выражение:

$$(C_1^{ML} \vee C_3^{VL}) \wedge (C_2^{VL}) \wedge (C_1^{ML} \vee C_3^{VL}) \wedge C_1^{ML} \vee C_2^{VL} \vee C_3^{VL}) \wedge (C_2^{VL}),$$

которое соответствует концептуальным представлениям документов $\{D_3, D_7, D_{10}\}$, упрощается до выражения $C_1^{ML} C_2^{VL} \vee C_3^{VL} C_2^{VL}$, которое, в итоге, позволяет получить импликацию следующего вида:

$$(C_1 = ML \wedge C_2 = VL) \vee (C_3 = VL \wedge C_2 = VL) \rightarrow (K = 1).$$

В итоге получаем следующее множество правил:

$$\begin{cases} (C_1 = L) \rightarrow (K = 1) \\ (C_2 = ML) \rightarrow (K = 1) \\ (C_1 = ML) \wedge (C_2 = VL) \rightarrow (K = 1) \\ (C_3 = VL) \wedge (C_2 = VL) \rightarrow (K = 1). \end{cases}$$

Интерпретировать полученное множество правил можно следующим образом.

Кластер №1 электронного архива включает в себя ТД, у которых:

- степень выраженности понятия C_1 низкая (L) ИЛИ
- степень выраженности понятия C_2 средняя-низкая (ML) ИЛИ
- степень выраженности понятия C_1 средняя-низкая (ML) И степень выраженности понятия C_2 очень низкая (VL) ИЛИ

- степень выраженности понятия C_3 очень низкая (VL) И степень выраженности понятия C_2 очень низкая (VL).

Предложенный способ содержательной интерпретации кластеров документальных информационных баз САПР АС предназначен для использования в системах выработки рекомендаций проектировщикам в тех случаях, когда необходимо автоматически сформировать в компактной форме описание фрагмента содержимого больших информационных документальных ресурсов проектной организации.

4.2.2. Интерпретация технических временных рядов

В процессе автоматизированного проектирования АС часто возникает необходимость в анализе не только статических показателей, но и показателей, имеющих динамическую природу. Такие технические показатели нередко представлены в виде временных рядов. *Временной ряд* (или ряд динамики, или динамический ряд) – ряд последовательных значений, характеризующих изменение показателя во времени [116].

Для получения информации, необходимой для принятия управленческих решений, необходимо заключение человека-эксперта или экспертной системы, содержащей эти знания. Процесс получения такой информации на основании динамики временного ряда носит название содержательной интерпретации временного ряда.

В данном разделе рассмотрим возможность применения генетического алгоритма для разбиения временного ряда параметров аппаратных подсистем проектируемых АС на фрагменты, которые могут быть интерпретированы в терминах соответствующей предметной области. Знания экспертов, представляемых в базе знаний, будем описывать в виде онтологической модели [77].

Исходными данными для содержательной интерпретации технического временного ряда являются сведения о временных интервалах, в течение которых выполняется анализ исследуемого показателя. К таким сведениям будем отно-

сить: абсолютную величину изменения, длину временного интервала и функцию изменения значения показателя. В самом простом случае указанная функция является линейной. В предлагаемом методе фрагментирования временной ряд аппроксимируется множеством прямоугольных треугольников, у которых начальная и конечная точки отрезка гипотенузы соответствуют значениям временного ряда на начало и конец определенного отрезка времени. Сумма абсолютных значений разности между реальными значениями временного ряда и полученными в ходе аппроксимации для всех шаблонов (треугольников) представляет собой абсолютное значение ошибки фрагментирования временного ряда шаблонами, заданными в онтологии анализа временных рядов.

В ходе процесса фрагментирования следует решить две задачи, которые взаимоисключают друг друга: сведение к минимуму суммарной ошибки аппроксимации и минимизация количества фрагментов временного ряда, каждому из которых ставится в соответствие шаблон из онтологии. Пример исходного технического временного ряда, соответствующий значениям трафика вычислительной сети, представлен на рисунке 4.7, временной ряд после разбиения на фрагменты показан на рисунке 4.8.

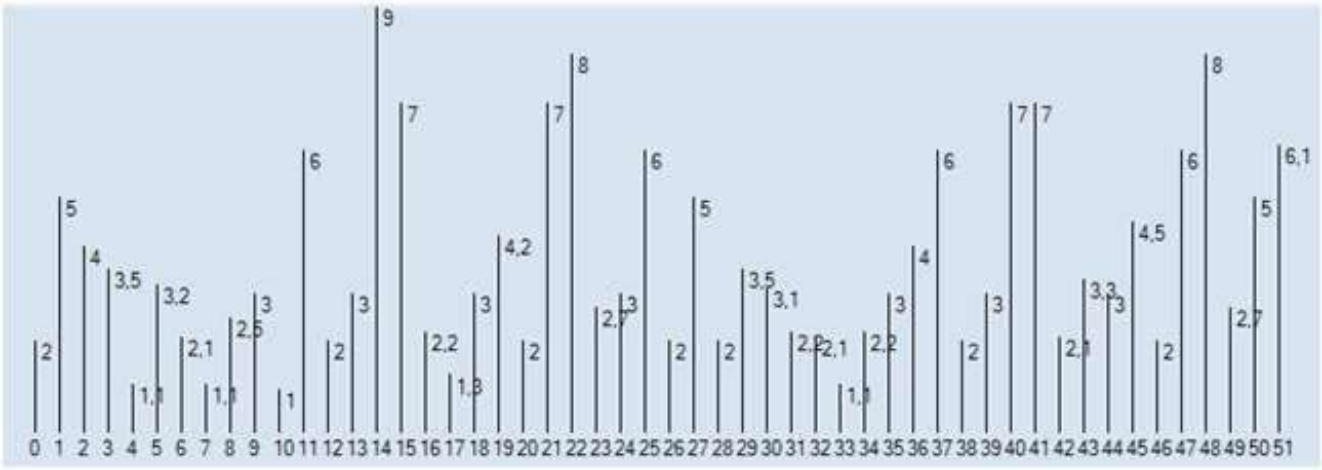


Рис. 4.7. Исходный временной ряд

При решении задачи аппроксимации будем использовать формальное представление генетического алгоритма (3.7) [98]. Для решения конкретной задачи

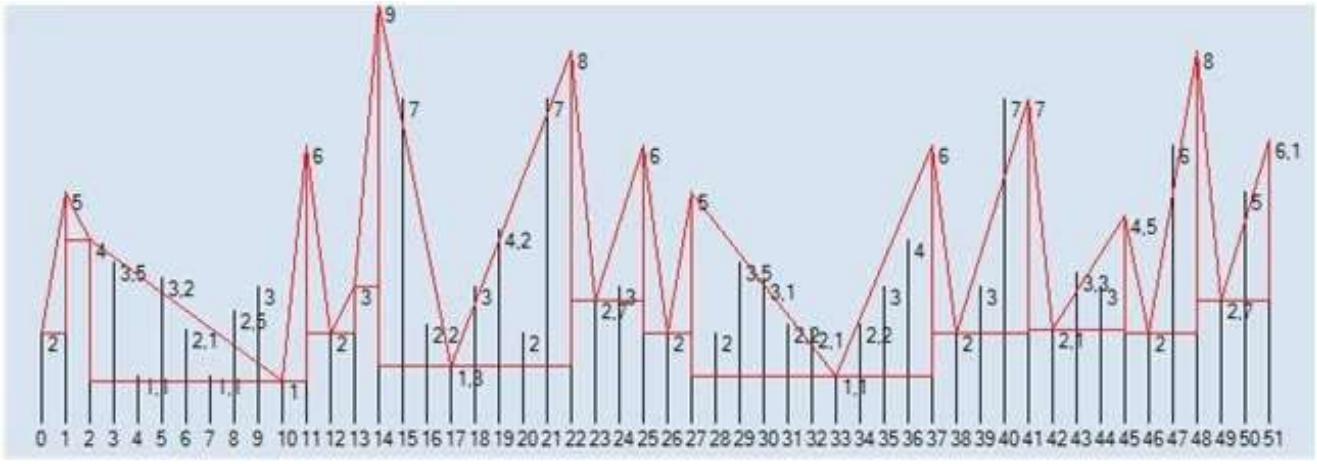


Рис. 4.8. Фрагментированный временной ряд

оптимизации фрагментов временного ряда алгоритм требует следующих уточнений:

- структуры представления хромосом в алгоритме оптимизации;
- формализация целевой функции;
- способы реализации генетических операций: кроссинговер и мутация.

Целевую функцию, которая подвергается минимизации, будем представлять следующей формулой:

$$F = \alpha a + (1 - \alpha)b,$$

где a и b – параметры, определяющие суммарную ошибку аппроксимации и количество интервалов разбиения временного ряда соответственно,

$$a = \frac{\sum_{i=1}^{L} \epsilon_i}{\sum_{i=1}^{L} y_i}, b = \frac{N}{L-1},$$

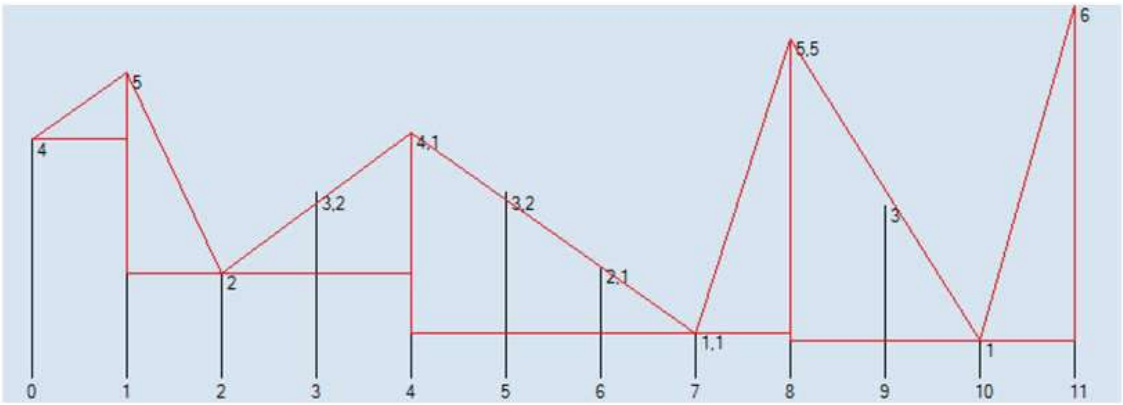
L – длина временного ряда, N – количество треугольников (интервалов разбиения).

Хромосома (как потенциальное решение оптимизационной задачи) представляется как последовательность чисел. При этом, длина последовательности определяется длиной временного ряда. Отдельный элемент данной последова-

тельности (ген) определяет, какому треугольнику соответствует определенное значение временного ряда. Фрагмент хромосомы (ген) или может быть равен целому числу n (анализируемое значение временного ряда соответствует n-ому треугольнику) или $n + 0,5$ (анализируемая вершина является общей для соседних (n-го и $(n+1)$-ого) треугольников). Следовательно, значение гена в первой позиции равно единице, а в последней позиции соответствует количеству интервалов декомпозиции временного ряда.

При определении структуры хромосом необходимо принимать во внимание, что все треугольники (интервалы декомпозиции) являются смежными. Следовательно, переход от вершины с целочисленным значением гена к вершине последующего треугольника осуществляется исключительно через общую для этих интервалов вершину, ген которой имеет дробное значение.

Графически способ кодирования хромосомы представим в следующем виде:



{ 1 1,5 2,5 3 3,5 4 4 4,5 5,5 6 6,5 7 }

Реализацию кроссовера будем выполнять по следующей схеме, представленной на рисунке 4.9.

Оператор мутации предполагает прохождение следующих шагов (рисунок 4.10):

- случайный выбор номера мутирующей вершины;

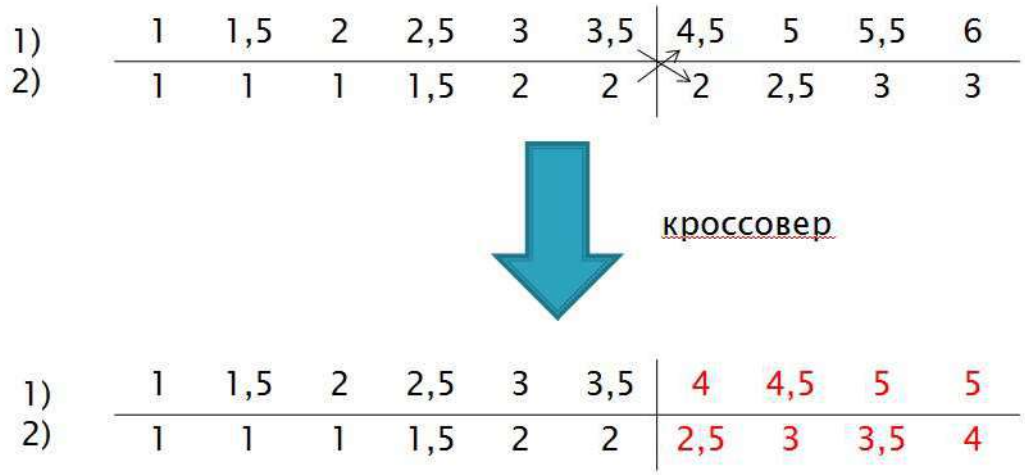


Рис. 4.9. Реализация оператора кроссовера

- присвоение случайного значения из интервала [1; n-0,5] мутирующему гену, где n – порядковый номер вершины;
- выполнение корректировки хромосомы правее мутирующей вершины;
- выполнение корректировки хромосомы левее мутирующей вершины.

Рис. 4.10. Реализация оператора мутации

Методика содержательной интерпретации.

1. Формирование набора эталонных временных рядов выбранной предметной области.
2. Определение фрагментов эталонных временных рядов экспертом и присвоение каждому фрагменту лингвистической метки.
3. Сохранение полученной информации в форме прикладной онтологии анализа временных рядов.
4. Определение фрагментов анализируемого временного ряда.
5. Определение степени совпадения описаний фрагментов ряда с описания-

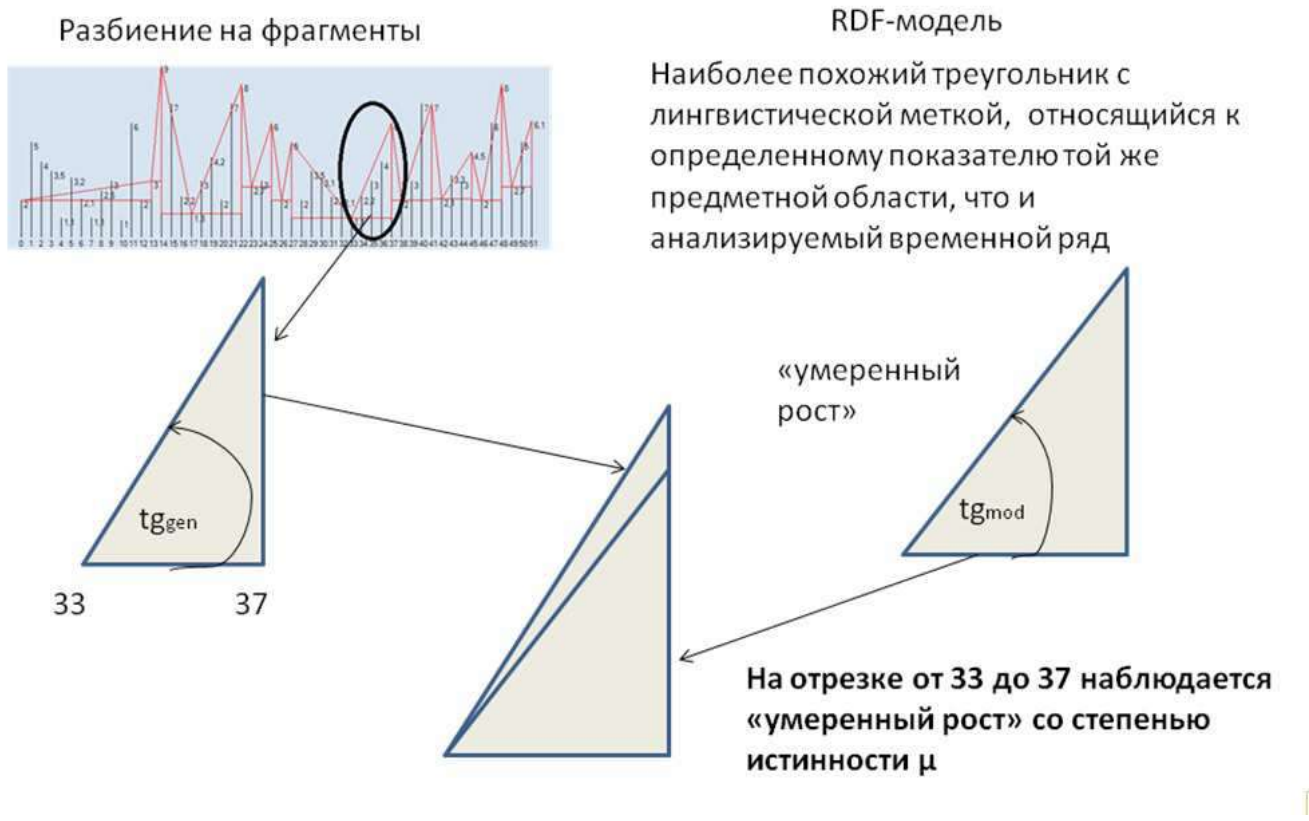


Рис. 4.11. Процесс сопоставления представлений фрагментов временного ряда с лингвистическими метками

ми, представленными в онтологии.

6. Назначение каждому фрагменту ряда лингвистической метки с вычисленной степенью совпадения ее с экспертной.

Структура онтологии показана на рисунке 2.9. На рисунке 4.11 представлен процесс сопоставления представлений фрагментов временного ряда в виде треугольников с описаниями лингвистических меток, зафиксированных в онтологии предметной области.

Для определения степени совпадения фрагмента, который получен в результате декомпозирования анализируемого временного ряда алгоритмом генетической оптимизации, и шаблонного значения (паттерна) поведения временного ряда, взятого из онтологии, применяется функция $\mu(tg_{gen}, tg_{mod}) \in [0, 1]$, где tg_{gen} – тангенс угла наклона гипотенузы треугольника, который вычисля-

ется в процессе работы генетического алгоритма. Значение tg_{mod} есть тангенс угла треугольника из описания лингвистической метки онтологии. Если значение функции μ близко к единице, то степень совпадения между фактическим треугольником и треугольником из онтологии наивысшее.

Значение функции μ вычисляется по формуле:

$$\mu(tg_{gen}, tg_{mod}) = \begin{cases} 1 & tg_{gen} = 0 \text{ и } tg_{mod} = 0 \\ 1 - \frac{|tg_{gen} - tg_{mod}|}{tg_{gen} + tg_{mod}} & \text{при равенстве знаков } tg_{gen} \text{ и } tg_{mod} \\ 0 & \text{в остальных случаях} \end{cases}$$

Предложенный способ содержательной интерпретации технических временных рядов позволяет извлекать и аккумулировать знания эксперта с целью их использования для решения задачи идентификации различного рода ограничений проектируемых АС. В частности, идентификация фрагментов временного ряда, представляющего собой изменение показателя трафика между узлами вычислительной сети, позволит интеллектуальной системе сделать вывод о целесообразности использования определенного вида коммуникационного оборудования.

4.3. Формализация контекстно-ориентированных запросов к электронному архиву проектной организации

4.3.1. Понятие информационной потребности проектировщика

Традиционным является представление документальных информационных баз проектной организации в виде электронных архивов технической документации. Основными задачами таких электронных архивов являются обеспечение возможности совместной работы проектных групп над общим проектом, добавление, хранение и поиск технических документов, являющихся артефактами проектной деятельности, в электронном архиве. Для формирования узкоспециализированных запросов к электронному архиву проектной организации тре-

буется привлечение соответствующих знаний предметной области. Фактически речь идет об интерактивном поиске, основанном на знаниях.

Многие участники проекта посылают узкоспециализированные проектные запросы к электронному архиву организации, преследуя удовлетворение *информационных потребностей*, которые предполагают наличие как объективных факторов, так и носят субъективный характер [142], [151], [182].

Известно большое количество определений понятия «информационная потребность», сформулированных различными исследователями. Согласно [24], *информационная потребность* – «необходимость в информации, требующая удовлетворения и обычно выражаемая в информационном запросе, одно из центральных понятий в информатике».

Если рассматривать потребность как функциональную систему, то: «информационная потребность – потребность в информационной деятельности, устраняющей дисбаланс (рассогласование между наличным и нормальным состоянием) информационной сферы субъекта» [99]. Часто в контексте автоматизированного проектирования информационная деятельность понимается как «совокупность процессов создания, сбора, понимания, переработки, хранения, поиска и распространения информации».

ГОСТ 7.73 - 96 «Поиск и распространение информации» предлагает еще одно определение *информационной потребности*: «информационные потребности – характеристики предметной области, значения которых необходимо установить для выполнения поставленной задачи в практической деятельности» [23].

Ориентация только на запросы пользователей не позволяет получить всестороннее и достаточно надежное представление об информационных потребностях [17].

Проектная деятельность по разработке сложных АС имеет некоторые специфические особенности [105].

1. Результатом проектной деятельности является организованное множество

сведений, которые служат знаковой моделью объекта, который в момент проектирования пока еще не существует.

2. Проектные процедуры реального объекта соответствуют преобразованию его исходного описания, учитывая ряд ограничений.
3. Способы преобразования информации при проектировании нельзя отразить в виде математических соотношений, т. е. невозможно в принципе построить строгую математическую модель данного процесса.
4. Поскольку проектируемые объекты являются сложными системами, на каждом этапе разработки принимают участие различные специалисты. Это придает процессу проектированию характер коллективной деятельности.
5. Как правило, проектирование имеет итерационный и многовариантный характер. Поэтому для принятия проектных решений используются различные научно-технические знания.

Основной целью формирования профессиональных запросов к электронному архиву проектной организации является удовлетворение некоторой информационной потребности проектировщика. Корректно выразить информационную потребность с помощью ограниченных возможностей набора ключевых слов информационного запроса является в определенной степени искусством, которое сложно формализовать. Правильный набор ключевых слов предполагает у проектировщика наличие хороших знаний предметной области. Кроме того, необходимо обладать обширными знаниями о содержимом электронного архива, который может в проектных организациях достигать сотен тысяч документов, накопленных за весьма продолжительный промежуток времени [72], [74], [75], [76].

4.3.2. Система контекстов проектирования автоматизированных систем

Рассмотрим систему контекстов (рисунок 4.12), в которых происходит проектирование современной АС. *Контекст процесса проектирования* является внешним контекстом относительно проектной организации. Информационные ресурсы данного контекста определяются на основе государственных стандартов, отраслевых стандартов и т.д. и описывают методологию проектирования, а также понятийный аппарат в самом общем виде (например, понятия «проект», «технический документ», «стадия проектирования» и другие) [82].

Контекст проектной организации определяется системой понятий и терминов, которые используются внутри организации в процессе проектной деятельности. Данный контекст формируется на основе существующих в организации терминологических словарей и (или) термины, и понятия извлекаются из технических документов электронных архивов. Отношение вложенности контекста проектной организации в контекст процесса проектирования определяет зависимость внутреннего контекста от внешнего. Внешний контекст выступает в роли ограничений, которым должен удовлетворять внутренний контекст (например, если контекст процесса проектирования формируется с использованием ГОСТ 27.002-2015 «Надежность в технике (ССНТ). Термины и определения» [21], тогда в состав понятий контекста проектной организации может быть включен концепт «Внезапный отказ»).

Контекст проекта образуют термины документа, на основе которого проектируется АС (техническое задание и (или) технико-экономическое обоснование).

Контекст проекта будем определять как граф вида:

$$G^{PT} = (C^{PT}, R^{PT}),$$

где C^{PT} – множество вершин-понятий проекта, R^{PT} – множество дуг, соединяющих вершины-понятия.

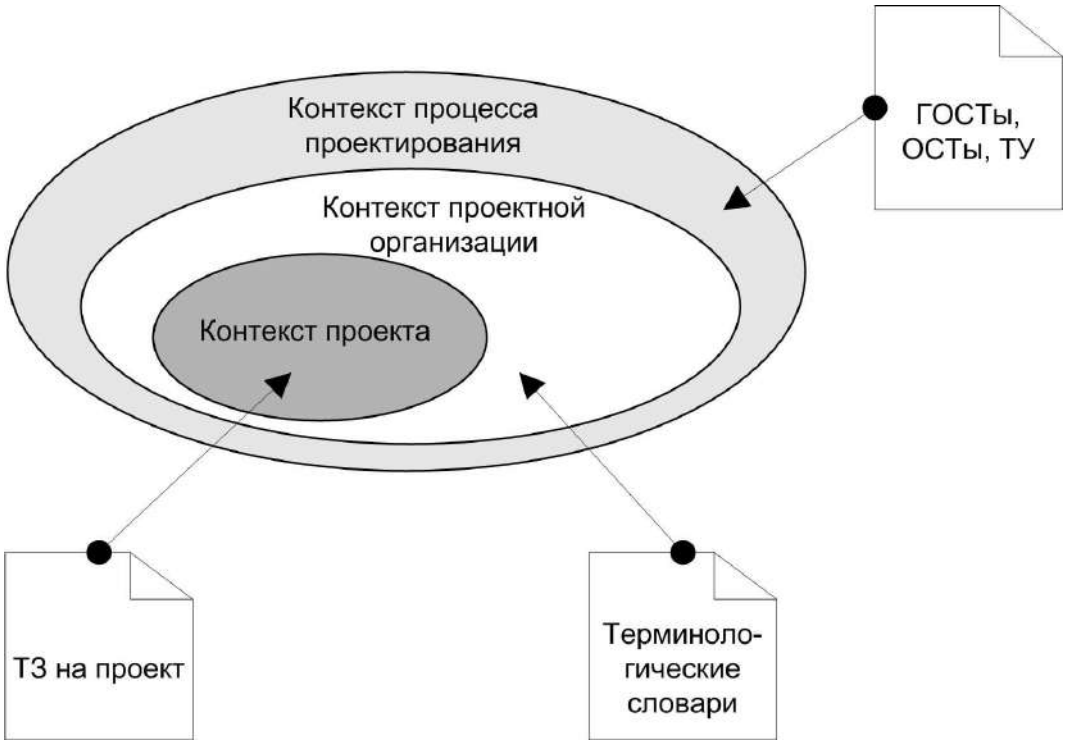


Рис. 4.12. Система вложенных контекстов проектной организации

Множество понятий проекта определяется как результат функции концептуального индексирования технического задания (Tz) на реализуемый проект ($F_{cI}(Tz)$) и функции концептуального доопределения множества C^{Tz} как результата $F_{cI}(Tz)$ с использованием wiki-ресурсов сети Internet ($F_{cAdd}(C^{Tz})$). Алгоритм формирования контекста проекта будем представлять в виде следующих шагов.

Шаг 1. Загрузка файла технического задания (Tz).

Шаг 2. Концептуальное индексирование технического задания:

$$C^{Tz} = F_{cI}(Tz).$$

Шаг 3. Доопределение множества C^{Tz}.

На данном шаге выполняется анализ wiki-ресурса Internet и определяется множество дополнительных понятий, имеющих связи с понятиями множества C^{Tz}. Идентификация связей между понятиями определяется на основе существующих гиперссылок на соответствующие страницы сети, содержащие текстовые описания понятий.

Шаг 4. Загрузка словаря технических терминов Dic.

Словарь Dic формируется на основе технической документации электронного архива проектной организации и является разделяемым ресурсом информационного обеспечения автоматизированного проектирования.

Шаг 5. Сравнение текстовых входов ($T_{sur}(\hat{C}^{Tz})$) понятий $\hat{C}^{Tz} = F_{cAdd}(C^{Tz})$ с терминами из Dic.

Если $\forall \hat{w} \in T_{sur}(\hat{C}^{Tz})$ выполняется условие $\hat{w} \notin Dic$, тогда необходимо удалить понятие $\hat{c} \in \hat{C}^{Tz}$.

Шаг 6. Проверка очередного $\hat{c} \in \hat{C}^{Tz}$.

Если сравнение текстовых входов понятий со словарем выполнено не для всех элементов множества $\hat{C}^{Tz}$, тогда выполняется переход к шагу 5.

Шаг 7. Определение множества дуг R^{PT} на основе анализа гиперссылок страниц wiki-ресурса.

Шаг 8. Сохранение графа G^{PT}.

4.3.3. Формирование концептуальной сети проекта из wiki-ресурса

Метод формирования концептуальной сети проекта из внешних wiki-ресурсов основывается на идеи алгоритма Ли для трассировки печатных плат [87]. На первом шаге происходит инициализация понятий в wiki-сети, определенные в техническом задании, как наиболее выраженные (рисунок 4.13). Страница wiki-сети понимается как концепт (понятие), а гиперссылки – отношения, связывающие понятия.

На втором шаге работает цикл распространения волны. На каждой итерации цикла происходит последовательное разворачивание концептуальной сети (рисунок 4.13).

После прохождения заданного количества итераций разворачивания сети происходит восстановление пути, соединяющего исходные понятия (рисунок 4.14). На последнем шаге формируется результирующая концептуальная сеть, включающая понятия, определенные на предыдущем шаге и понятия,

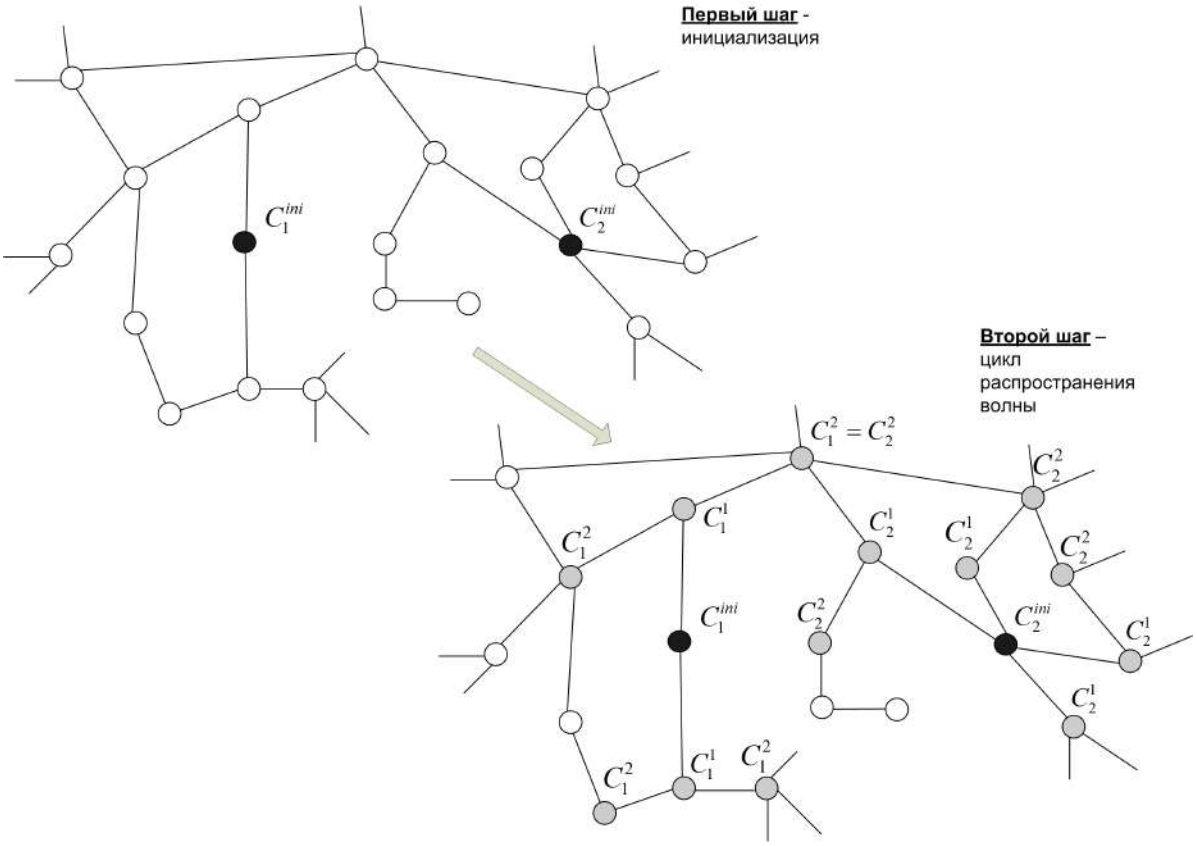


Рис. 4.13. Инициализация понятий и цикл распространения волны

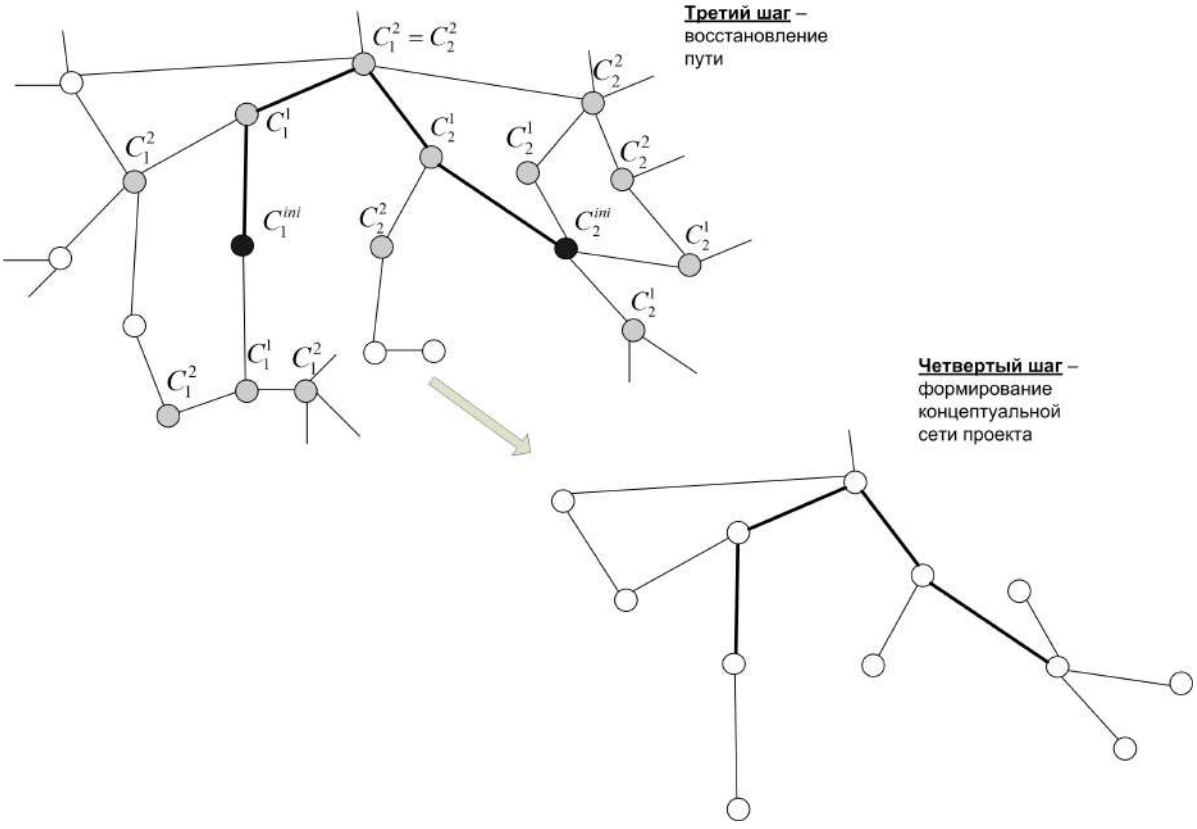


Рис. 4.14. Восстановление пути и формирование концептуальной сети

непосредственно связанные с ними.

4.3.4. Формализация опыта взаимодействия проектировщика с электронным архивом

Формализация *опыта проектировщика* осуществляется на основе предположения о том, что имеется возможность фиксировать результаты проектных запросов к электронному архиву в виде множества документов, удовлетворяющих информационной потребности, и множества документов, которые текущей информационной потребности не удовлетворяют. Учитывая выражение (3.15), каждой информационной потребности In_j^i ставится в соответствие пара классов понятий онтологии ИПР $K^+ = \{c_1^+, c_2^+, \ldots, c_n^+\}$, $K^- = \{c_1^-, c_2^-, \ldots, c_m^-\}$, определяющие положительные и отрицательные подмножества понятий онтологии соответственно [78].

Положительные K^+ и отрицательные K^- классы понятий формируются следующим образом:

- В процессе выполнения проектировщиком информационных запросов к электронному архиву определяется набор ТД, которые соответствуют его информационной потребности (D^+) и ТД, не соответствующие ей (D^-), с учетом текущей стадии (этапа) проектирования.
- Для каждого документа определяется его концептуальное представление. Другими словами, производится концептуальное индексирование. Запишем нечеткое соответствие между множеством $K^{+(-)}$ и множеством $T^D = T_{in}^D \bigcup T_{ext}^D$, как $\tilde{\Gamma}_{KT} = (K^{+(-)}, T^D, \tilde{F}_{KT})$, где $\tilde{F}_{KT}$ – нечеткое множество в $K^{+(-)} \times T^D$. Определим нечеткое соответствие $\tilde{\Gamma}_{KT}$ в виде ориентированного двудольного графа с множеством вершин $K^{+(-)} \bigcup T^D$, каждой дуге $\langle c_i^+, t_j\rangle(\langle c_i^-, t_j\rangle)$ которого приписываем значение функции принадлежности $\mu_{F_{KT}}\langle c_i^+, t_j\rangle(\mu_{F_{KT}}\langle c_i^-, t_j\rangle)$. Указанное значение функции принадлежности вычисляется на основе нормализованной частоты встречаемости термина в текстовом входе понятия, которые формируются из внутреннего ис-

точника (терминологических словарей) и из внешних источников (профессиональных wiki-ресурсов) (рисунок 4.15), а также $T^D = T^D_{in} \bigcap T^D_{ext} = \varnothing$. Образ множества T^D, при соответствии $\tilde{\Gamma}_{KT}$, фактически представляет собой нечеткое множество, элементами которого являются концепты с соответствующими степенями выраженности:

$$\tilde{\Gamma}_{KT}(T^D) = \{\mu_{\Gamma_{KT}}(c^{+(-)})/c^{+(-)}\},$$

где $\mu_{\Gamma_{KT}}(c^{+(-)}) = \vee_{t \in T^D} \mu_{F_{KT}} \langle c_i^{+(-)}, t_j \rangle$.

- В положительные и отрицательные подмножества понятий онтологии предметной области включаются такие понятия из онтологических представлений, степень выраженности которых наибольшая.

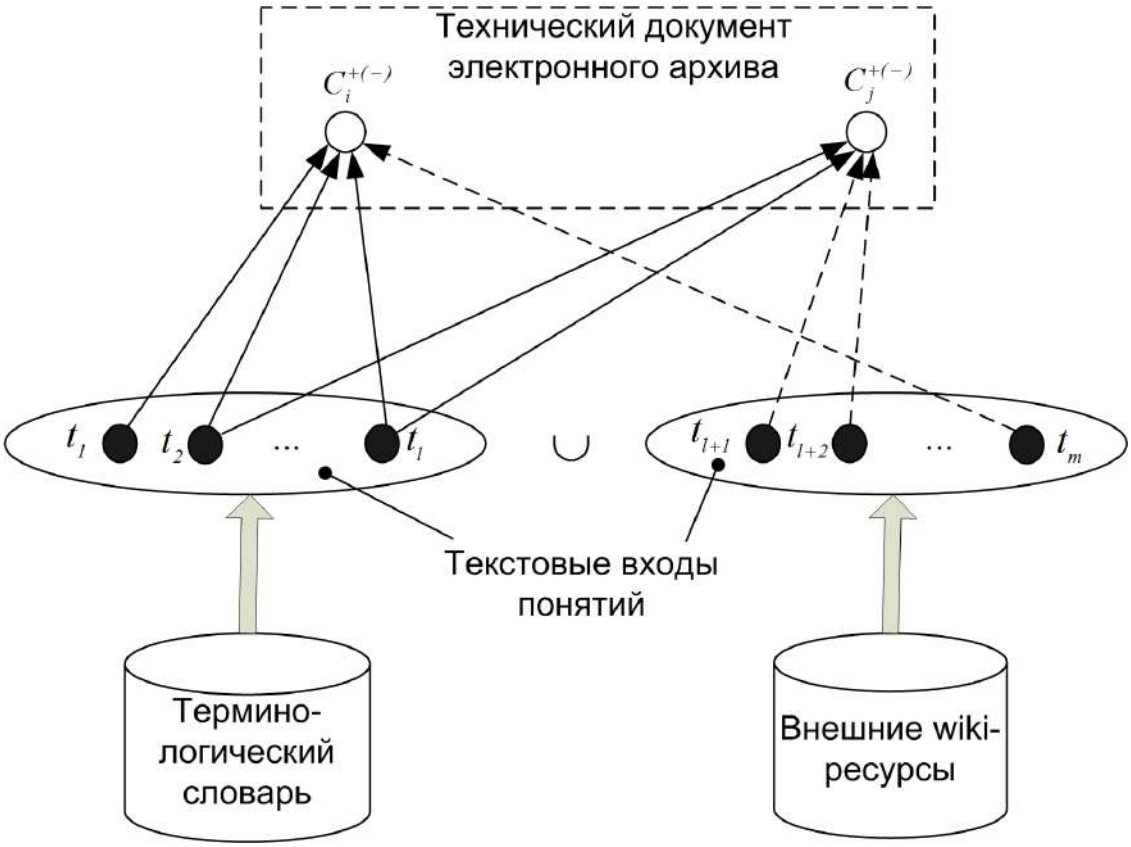


Рис. 4.15. Структура текстовых входов понятий при формализации опыта проектировщика

При получении результатов запросов к электронному архиву проектировщик отмечает те разделы документов, которые соответствуют его информационной потребности и которые явно ей не соответствуют. Таким образом накапливается обучающая выборка, состоящая из положительных и отрицательных

разделов документов, которые проходят процедуру концептуального индексирования. В результате формируются вышеуказанные классы понятий онтологии K^+ и K^-, с привязкой к идентификатору проектировщика и номеру стадии жизненного цикла проектируемой АС.

Рассмотрим модель классификации разделов документов, которые являются результатами выполнения проектных запросов. Будем принимать во внимание, что документы электронного архива имеют концептуальный вид и представляются в виде множества понятий онтологии, имеющих высокую степень выраженности в разделах ТД. Данная модель основана на применении наивного байесовского классификатора. Вероятность того, что j-й раздел i-го документа s_j^i принадлежит классу $k \in \{K^+, K^-\}$, будем определять по формуле Байеса:

$$P(k|s_j^i) = \frac{P(s_j^i|k) \cdot P(k)}{P(s_j^i)},$$

где $P(s_j^i|k)$ – вероятность встретить раздел s_j^i среди всех документов класса k; $P(k)$ – безусловная вероятность класса k в обучающей выборке; $P(s_j^i)$ – безусловная вероятность раздела документа s_j^i в корпусе документов обучающей выборки.

Наиболее вероятный класс для полученного раздела документа определяется, используя оценку апостериорного максимума:

$$k_{map} = \arg\max_{k \in K} \frac{P(s_j^i|k) \cdot P(k)}{P(s_j^i)}.$$

Поскольку $P(s_j^i) = const$ в рамках корпуса документов и учитывая, что

$$P(s_j^i|k) \approx P(c_1|k) \cdot P(c_2|k) \cdot \ldots \cdot P(c_n|k) = \prod_{s=1}^{n} P(c_s|k),$$

получаем:

$$k_{map} = \arg\max_{k \in K} [P(k) \cdot \prod_{s=1}^{n} P(c_s|k)]. \tag{4.6}$$

В выражении (4.6) $P(c_s|k)$ определяет вероятность встретить понятие c_s среди всех документов класса k;

Для разделов документов количество множителей $P(c_s|k)$ в выражении (4.6) может быть большим, а следовательно, возникает проблема исчезновения порядка вследствие перемножения большого количества малых чисел. Перепишем выражение (4.6) с учетом свойств логарифма:

$$k_{map} = \arg\max_{k \in K}[\log P(k) + \sum_{s=1}^{n} \log P(c_s|k)]. \quad (4.7)$$

Оценка вероятностей $P(k)$ и $P(c_s|k)$ выполняется на основе обучающей выборки, сформированной для каждой информационной потребности In_j^i. Вероятность класса будем записывать как:

$$P(k) = \frac{D_k}{D},$$

где D_k – количество разделов документов, принадлежащих классу k и определяемое на основе результатов выполнения запросов проектировщика к электронному архиву; D – общее количество разделов документов в обучающей выборке.

Величина $P(c_s|k)$ определяет вероятность встретить понятие онтологии ИПР c_s среди понятий разделов документов, принадлежащих классу k. Значение данной величины будем определять с учетом того, что данное понятие может отсутствовать в документах анализируемого класса. Применяя метод аддитивного сглаживания (сглаживания Лапласа), получаем:

$$P(c_s|k) = \frac{f_{sk} + 1}{\sum_{s' \in V}(f_{s'k} + 1)},$$

где f_{sk} – частота встречаемости s-го понятия в документах класса k; V – множество всех понятий онтологии ИПР.

В результате классификации результатов запросов вычисленные значения вероятностей попадания раздела технического документа в «положительный» или «отрицательный» классы учитываются при корректировке рангов документов в выборке результатов.

4.3.5. Формирование контекстно-ориентированных проектных запросов

Рассмотрим процесс формирования проектных запросов, который используется интеллектуальным агентом проектировщика и позволяет применять его опыт для улучшения показателей точности и полноты запросов к электронным архивам технических документов проектной организации.

Пусть множество $\hat{t} = \{\hat{t_1}, \hat{t_2}, \ldots, \hat{t_n}\}$ есть множество ключевых слов проектного запроса к электронному архиву. Рассмотрим задачу уточнения исходного множества ключевых слов запроса на основе *контекста проекта* и *индивидуального профиля проектировщика* (рисунок 4.16).

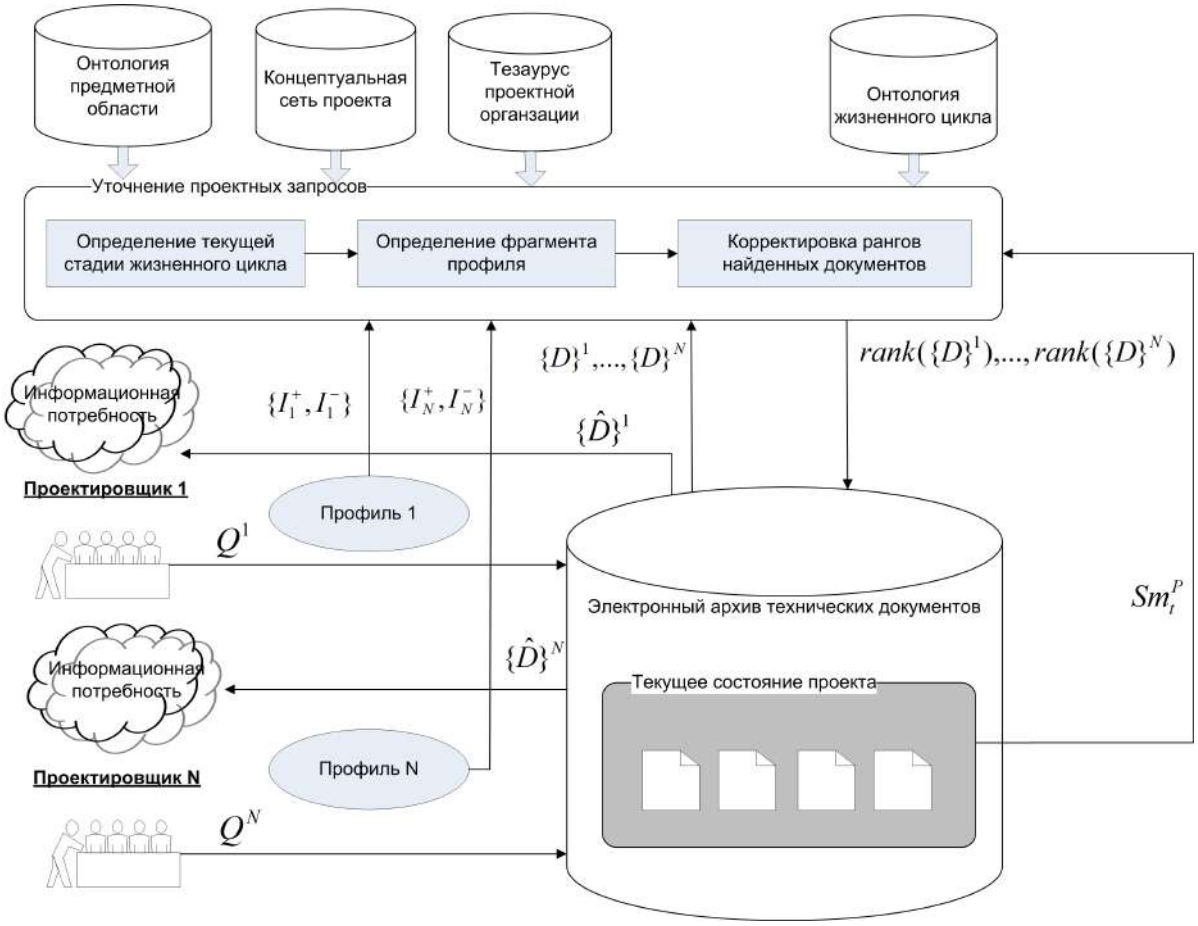


Рис. 4.16. Структурная схема уточнения проектных запросов

Определение 6. *Индивидуальный профиль проектировщика* есть структура, заданная на множестве понятий онтологии проектной ор-

ганизации и выражающая интересы проектировщика на определенной стадии жизненного цикла разрабатываемой системы.

Формально индивидуальный профиль проектировщика будем представлять в виде кортежа:

$$Ex^i =< \{In_1^i, K^+, K^-\}, \ldots, \{In_j^i, K^+, K^-\}, \ldots, \{In_n^i, K^+, K^-\} >, \qquad (4.8)$$

где i – индекс проектировщика, j – индекс стадии жизненного цикла.

Поскольку, в общем случае, у разных проектировщиков в различные моменты времени возникают разные информационные потребности, то их учет при формировании проектных запросов способен привести к повышению показателей качества получаемых результатов (повышение точности и полноты откликов на проектные запросы). Для этого будем применять систему индивидуальных профилей проектировщиков.

Основная цель применения профиля проектировщика при формировании запросов к электронному архиву состоит в определении подходящего контекста решения проектной задачи в соответствии с системой вложенных контекстов проектной организации (рисунок 4.16).

При уточнении проектных запросов $\{Q\}^i$ используются соответствующие знания из онтологии ИПР. Стадия проектирования определяется на основе анализа текущего состояния Sm_t^P реализуемого проекта P, используя знания о том, какие типы документов формируются в рамках каждой конкретной проектной стадии.

Онтология ИПР определяет контекст проектной организации. Контекст проекта формируется на основе концептуального анализа технического задания на проектирования изделия. При формировании профиля проектировщика указанные контексты используются для определения понятий предметной области, которые явно соответствуют или явно не соответствуют текущей информационной потребности (множество $\{I_i^+, I_i^-\}$ на рисунке 4.16).

Процедура уточнения проектных запросов включает в себя определение множества понятий онтологии предметной области, соответствующих текущей информационной потребности. Используя соответствующие текстовые входы понятий, происходит уточнение набора ключевых терминов запросов $\{Q\}^i$. В результате получаем уточненное множество $\{\hat{Q}\}^i$.

4.3.6. Метод редукции понятий проектного запроса

Если имеет место онтология с большим количеством понятий и сформированный запрос неявно соответствует некоторому фрагменту предметной области проектирования, в этом случае мощность множества C^q (количество понятий, включенных в нечеткое представление проектного запроса) может быть большой.

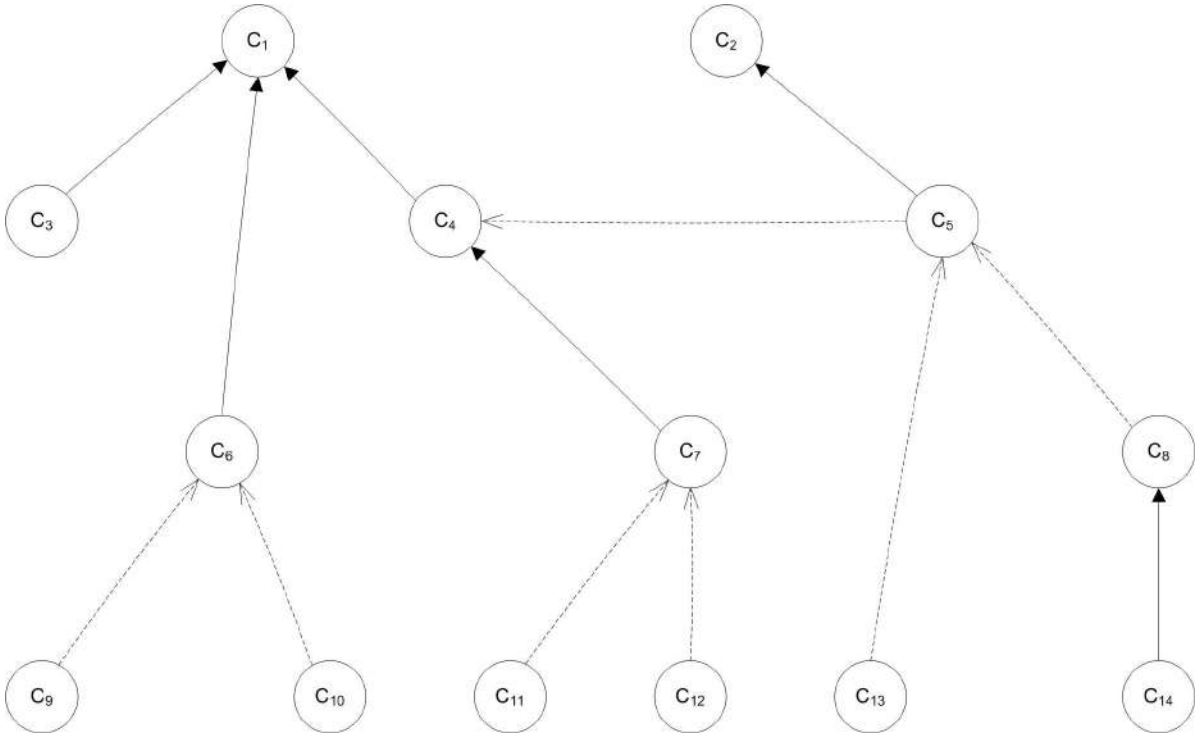


Рис. 4.17. Исходная структура проектного запроса

Метод редукции понятий контекстного запроса базируется на способе разделения исходного графа $G^q = \{C^q, R^q\}$ на несколько подграфов. Каждый подграф содержит только те вершины, которые соединены дугами одной семантической категории. В исследовании используются две семантические категории:

«обобщение» («is_A») и «часть-целое» («part_of»).

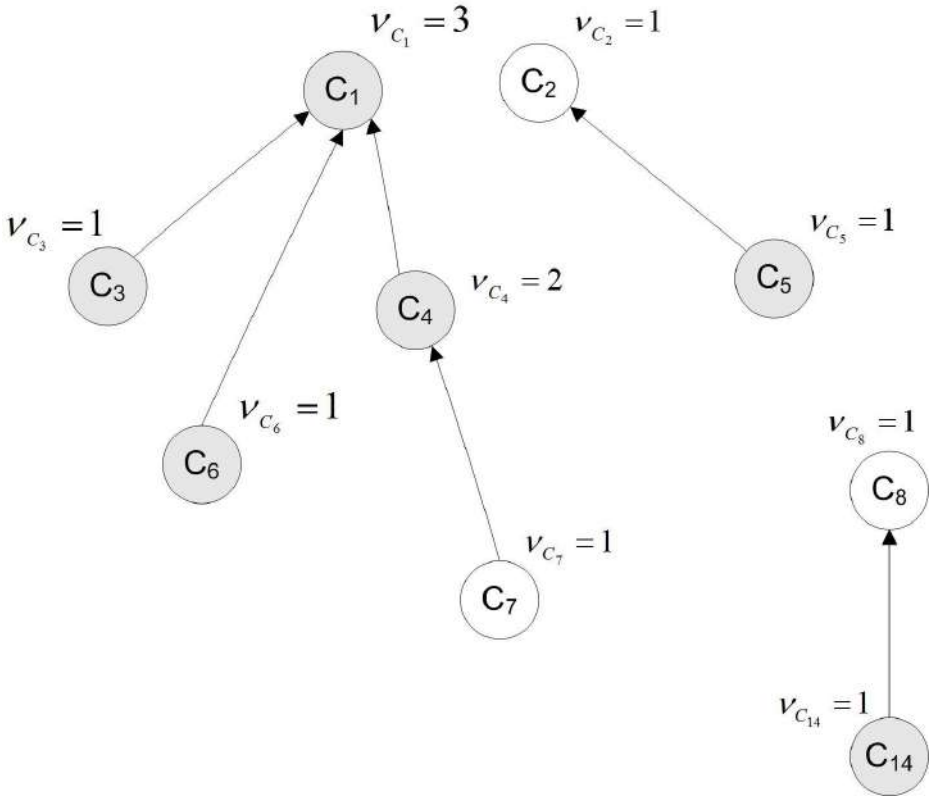


Рис. 4.18. Подграф «isA»

Пусть исходная структура проектного запроса будет иметь вид, представленный на рисунке 4.17. В данном иллюстративном примере присутствуют вершины – понятия онтологии предметной области и два типа дуг: дуги «is_A» (штриховая стрелка) и дуги «part_of» (сплошная стрелка).

Редукция множества понятий, включаемых в граф контекстного запроса, предполагает выполнение следующих шагов:

Шаг 1. Разбиение графа проектного запроса на несколько подграфов с учетом семантических категорий («is_A» или «part_of»). На рисунке 4.18 представлен подграф «is_A» графа исходного проектного запроса $G^q_{isA} = \{C^q_{isA}, R^q_{isA}\}$. Соответственно, на рисунке 4.19 представлен подграф $G^q_p = \{C^q_p, R^q_p\}$ с отношениями «part_of».

Шаг 2. Определяются значения степени вершин в каждом подграфе v_{c_j} (число входящих и исходящих дуг). Из рисунков видно, что наибольшие степени имеют понятие C_1 в подграфе «is_A» и понятия C_5, C_6 и C_7 в подграфе

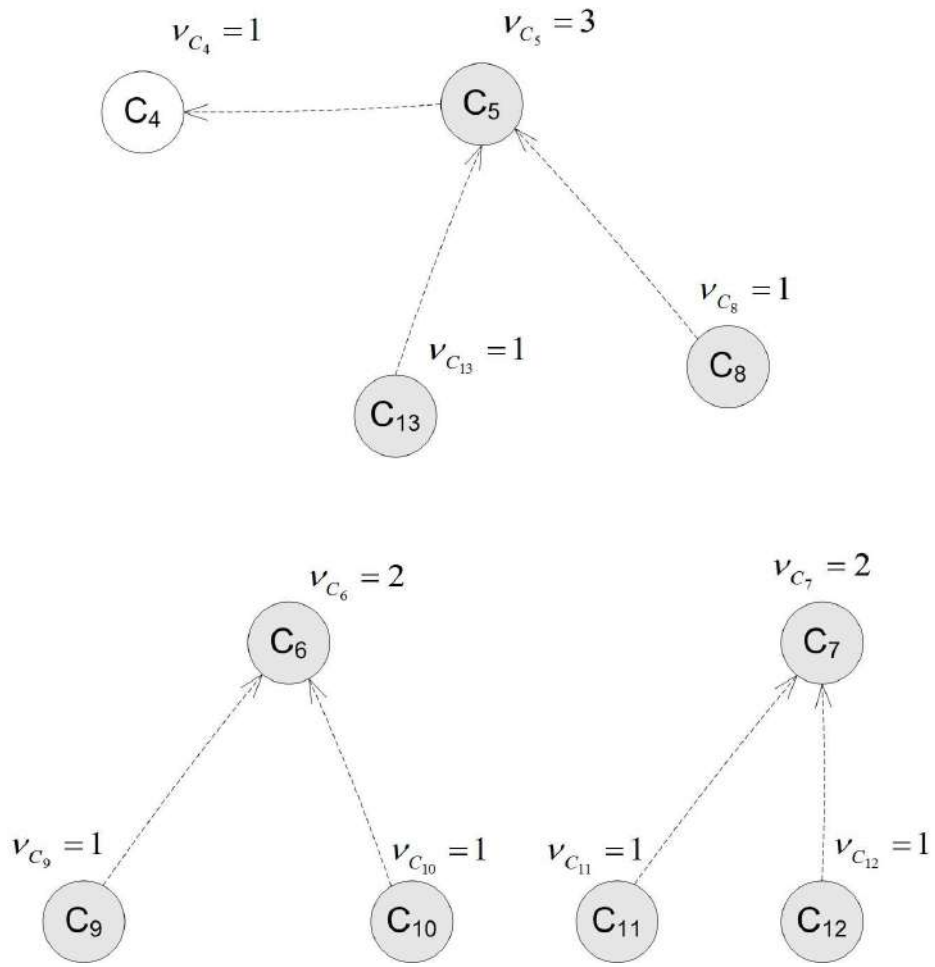


Рис. 4.19. Подграф «part_of»

«part_of».

Шаг 3. Включение понятий в редуцированное множество отдельно взятого подграфа проектного запроса происходит по следующим правилам:

1. Если в подграфе существует вершина с максимальной степенью, то в результирующее множество включается данная вершина и связанные с ней вершины, дуги от которых направлены к этой вершине.
2. Если подграф содержит две вершины, соединенные дугой, то в результирующее множество включается вершина с исходящей дугой.
3. Если подграф состоит из одной изолированной вершины, то данная вершина включается в результирующее множество.

На рисунке 4.18 и рисунке 4.19 темным фоном отмечены те вершины подграфов, которые включены в результирующие множества понятий C_{isA}^{q+} и C_{p}^{q+}

с использованием вышеприведенных правил.

Шаг 4. В редуцированное множество понятий проектного запроса включаются понятия, которые включены как во множество C_{isA}^{q+}, так и во множество C_p^{q+}: $C^{q+} = C_{isA}^{q+} \bigcap C_p^{q+}$.

Анализ проведенных вычислительных экспериментов по редукции понятий в проектных запросах к электронному архиву проектной организации позволяет сделать вывод о сокращении количества понятий, связанных с запросом до 30%.

4.4. Выводы по четвертой главе

1. Применение онтологического подхода позволяет по новому решить задачу структуризации информационных ресурсов электронного архива проектной организации, наделив его свойствами интеллектуальной системы. Описание содержимого технических документов на концептуальном уровне в терминах предметной области и используемых стандартов является компактным и адекватным для иерархической кластеризации и построения навигационной структуры электронного архива.
2. Разработанные модели содержательной интерпретации кластеров информационных ресурсов электронного архива проектной организации отличаются возможностью описания фрагментов электронного архива с использованием нечетких лингвистических шкал. Это позволяет применять такие модели в нечетких интеллектуальных системах в качестве объяснительной компоненты. Применение формализма приближенных множеств Павлака (rough sets) учитывает принципиальную невозможность построения четких границ между кластерами информационных ресурсов проектных архивов.
3. Предложенная модель контекстно-ориентированных проектных запросов базируется на системе контекстов проектной организации и учитывает

различия в информационных потребностях проектировщиков на разных стадиях реализации проекта. Используемая модель байесовского классификатора в алгоритме уточнения проектных запросов позволяет интеллектуальному репозиторию накапливать опыт взаимодействия субъекта проектирования с электронным архивом.

4. Способ онтологически-ориентированной редукции понятий проектных запросов нацелен на сокращение размера запроса при сохранении полноты представления информационной потребности после уточнения запроса на основе контекста проекта.

Глава 5

Архитектура и структуры данных интеллектуального проектного репозитория

5.1. Обобщенное представление архитектуры репозитория

Под интеллектуальным проектным репозиторием (ИПР) будем понимать программную систему, которая предназначена для систематизации и автоматизации взаимодействия с электронным архивом технических документов (ТД) и формализованных проектных диаграмм, учитывающая контекст проектирования и индивидуальные предпочтения проектировщика.

ИПР включает в себя следующие подсистемы:

1. подсистема кластеризации и формирования навигационной структуры электронного архива (Приложение 1);
2. подсистема визуализации и оценки качества онтологии ИПР;
3. подсистема информационной поддержки автоматизированного проектирования АС.

На рисунке 5.1 представлена обобщенная структура интеллектуального проектного репозитория.

Электронный архив проектной организации является составной частью (подсистемой) интеллектуального проектного репозитория. На каждой стадии жизненного цикла проектируемого изделия формируются артефакты проектирования и возникает информационная потребность, реализуемая посредством выполнения проектных запросов проектировщика к электронному архиву.

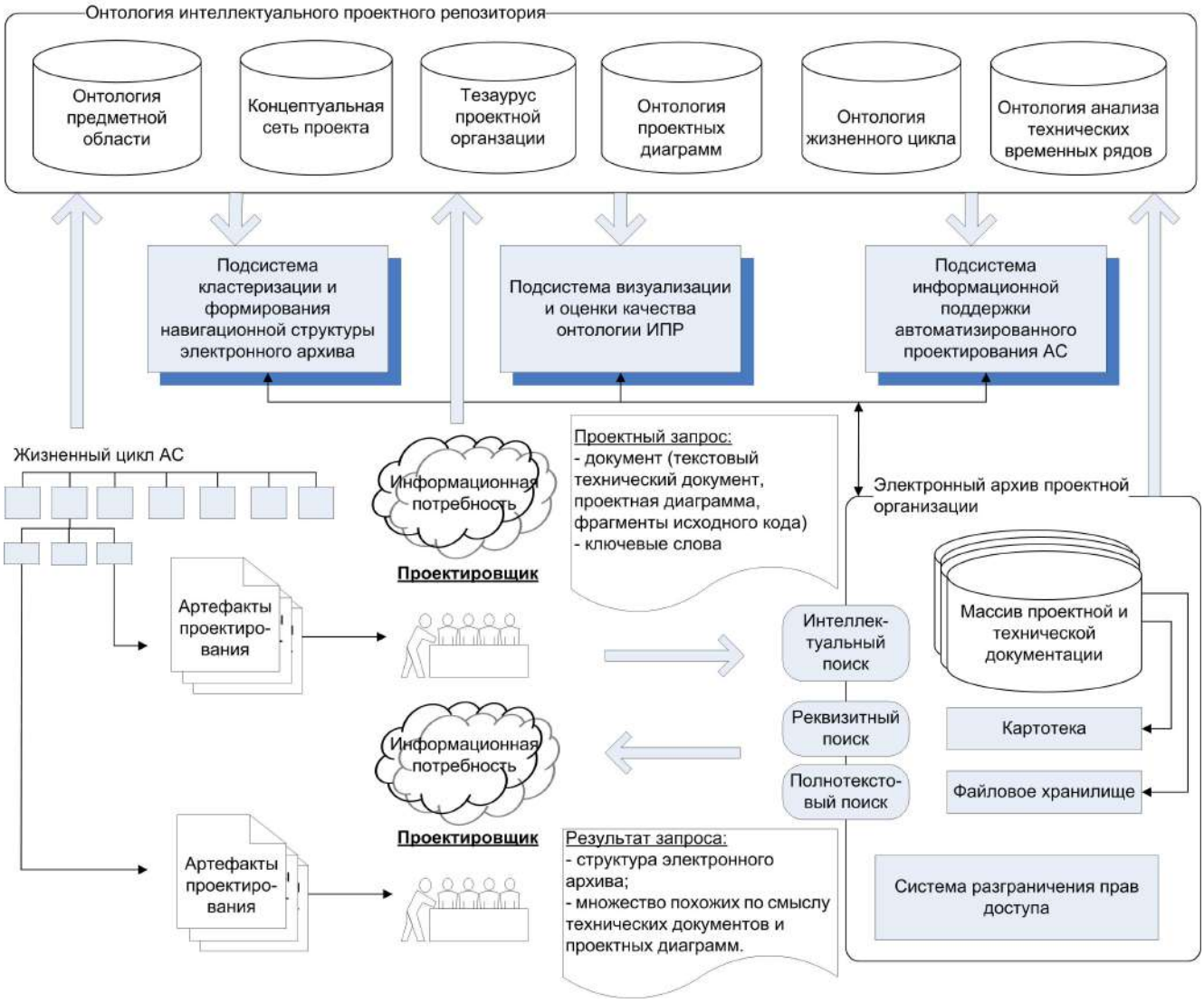


Рис. 5.1. Архитектура интеллектуального проектного репозитория

5.2. Подсистема кластеризации и формирования навигационной структуры электронного архива

Подсистема кластеризации и формирования навигационной структуры электронного архива реализована как модуль ИПР (Приложение 1).

Основные функции подсистемы [62], [64], [109], [111]:

1. Формирование концептуального индекса технических документов электронного архива.
2. Управление системой онтологий репозитория и хранение онтологических

представлений ТД.

3. Построение навигационной структуры электронного архива, которая учитывает модели жизненного цикла (ЖЦ) проектируемой системы. Пример фрагмента навигационной структуры показан на рисунке 5.2.
4. Выполнение контекстного поиска в электронном архиве, принимая во внимание семантическую структуру документов.

На рисунке 5.3 показана структура подсистемы кластеризации и формирования навигационной структуры электронного архива.

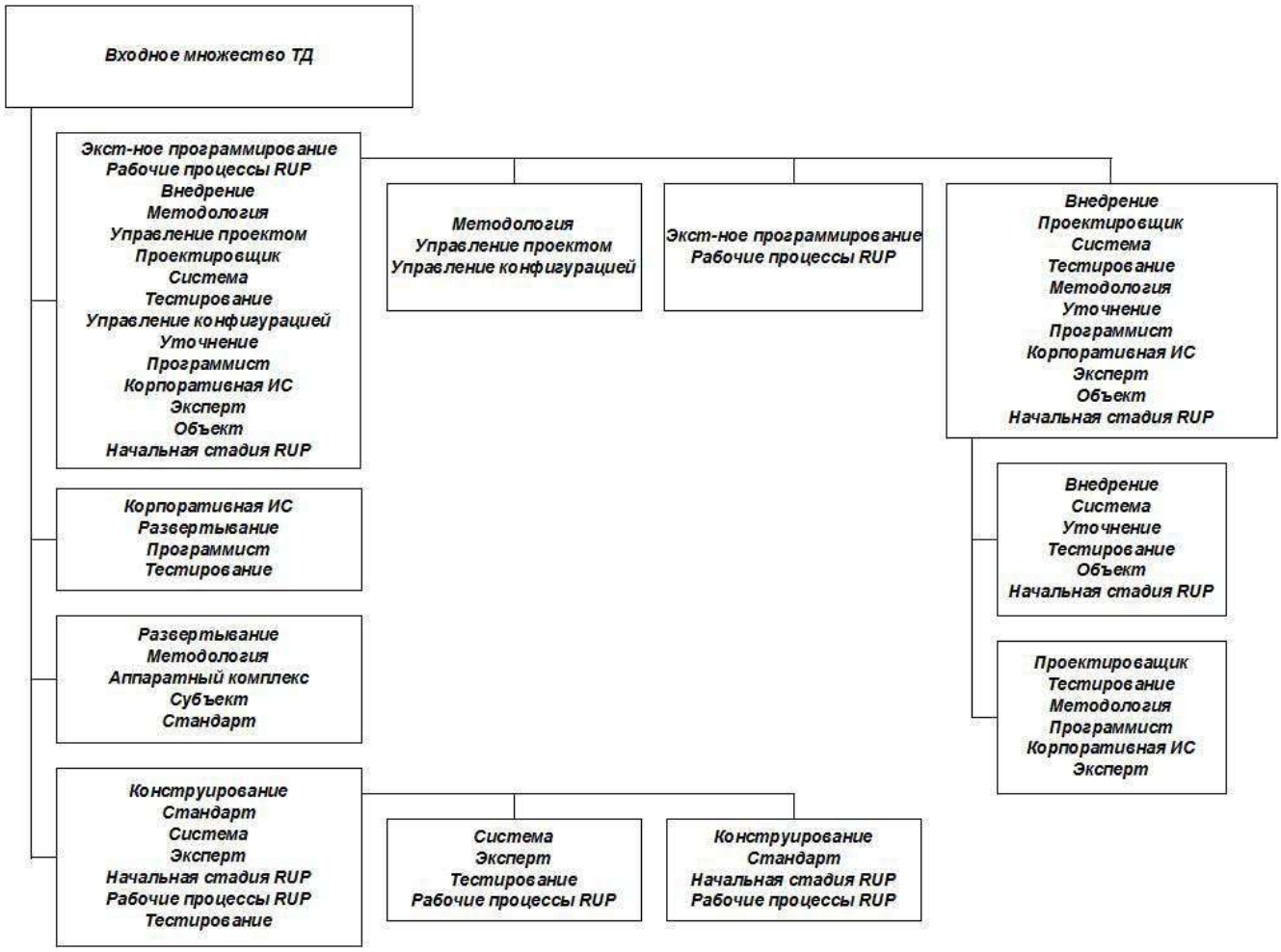


Рис. 5.2. Пример фрагмента навигационной структуры электронного архива

ИПР использует документальную базу данных САПР для получения электронных информационных ресурсов, которые в дальнейшем используются в процессе индексирования. Полученные наборы индексов хранятся в ИПР сред-

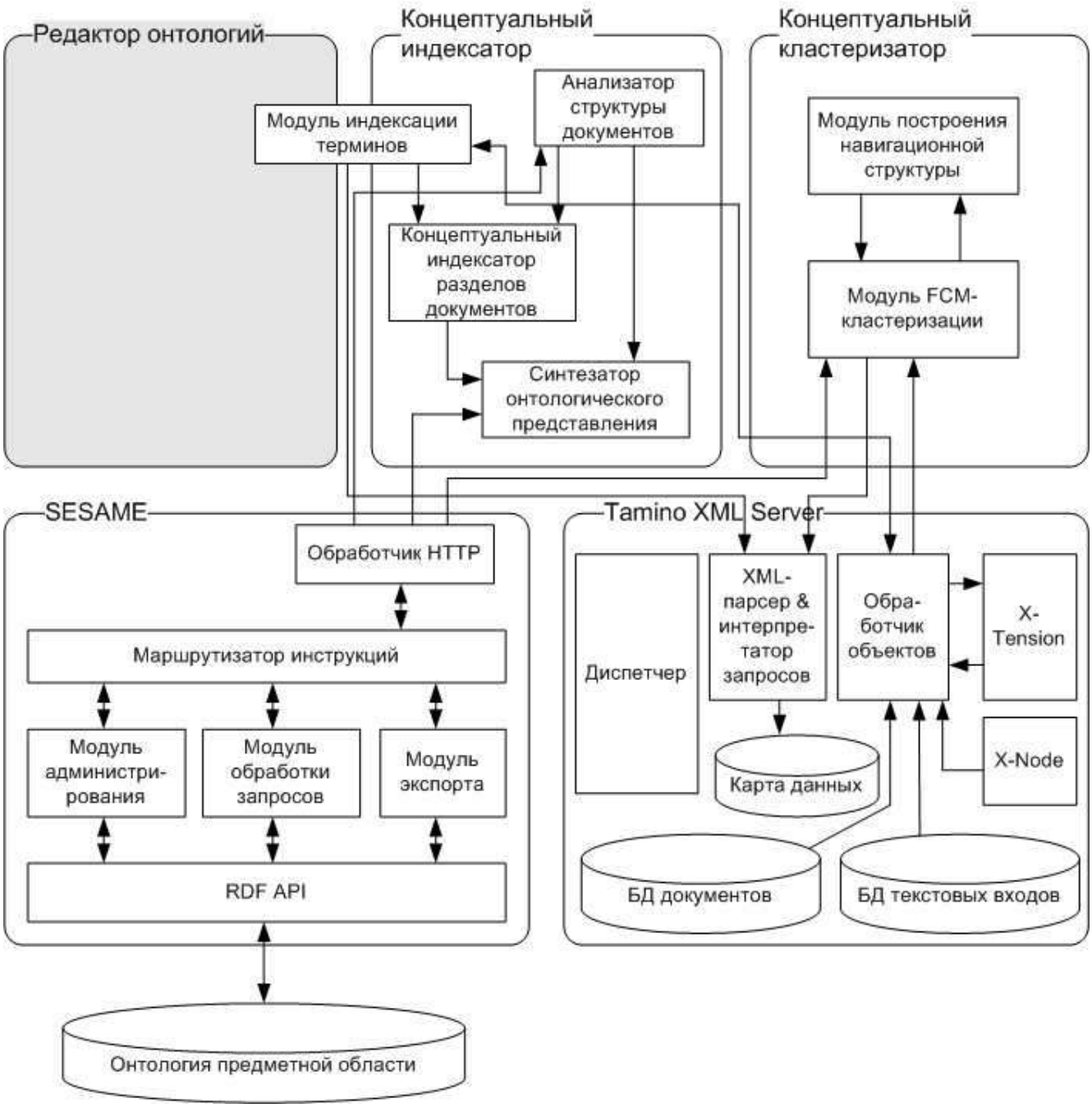


Рис. 5.3. Структура подсистемы кластеризации и формирования навигационной структуры архива

ствами Tamino XML Server. Процесс структуризации электронного архива выполняется для каждой стадии ЖЦ. Полученная навигационная структура электронного архива, соответствующая стадиям ЖЦ проектируемой системы, применяется для нахождения ТД.

XML-ориентированная СУБД Tamino – база данных исходных (пре-

добработанных) и документов, прошедших процедуру концептуального индексирования.

Система управления базами данных Tamino (Transaction Architecture for the Management of INternet Objects) представляет из себя информационный сервер, которые является продуктом компании Software AG. Он является информационным XML-сервером, предназначенным для обмена данными и интеграции приложений и реализующим технологию превращения данных, обрабатываемых существующими приложениями, в объекты сети Интернет [97].

Интегрированное решение Tamino включает в себя следующие компоненты [97].

1. Подсистема X-Machine представляет собой инструмент для хранения XML-данных в их оригинальном виде, используя возможность добавления функциональных расширений сервера для выполнения различных операций преобразования документов.
2. Tamino SQL Engine. С его помощью средства отображения данных позволяют автоматически решать задачу их представления в виде объектов сети Интернет. Кроме того, информационные объекты сети Интернет могут стать доступными в виде реляционных данных для приложений, которые ориентированы на SQL.
3. Диспетчер Tamino является инструментарием для администрирования объектов Tamino.
4. Расширения сервера. Архитектура Tamino позволяет пользователям встраивать специализированные функции для дополнительной обработки информации, обеспечивающие возможности работы с данными, хранящимися в Tamino, которые, в свою очередь, могут быть представлены как XML, так и не-XML структуры.
5. Компонент описания схемы является элементом базы знаний Tamino, который содержит правила отображения, хранения и конструирования объектов XML. Правила построения объектов основаны на информации о

схеме данных, поддерживаемой администратором сервера.

6. С помощью компонента X-Node Tamino пользователь получает доступ к гетерогенным и распределенным источникам данных. В качестве источников данных могут выступать базы данных, файловые системы или данные, полученные из систем передачи сообщений.
7. Комплект инструментальных средств разработки приложений Tamino SDK обеспечивает взаимодействие Tamino с XQL, SQL или объектно-ориентированными приложениями (DOM).

Выбор XML-сервера Tamino в качестве хранилища информационных ресурсов ИПР был обусловлен изначальной ориентацией на хранение XML-документов в их оригинальном виде, в отличие от реляционных СУБД и объектно-ориентированных СУБД, оснащенных XML-преобразователем [57], [58].

Преимущество языка XML для онтологического представления документов состоит в отсутствии заранее определенного набора элементов разметки (тэгов) [112], [16]. Дополнительные тэги могут включаться в процессе создания документа. При этом нет необходимости вносить изменения в программный код. Элементы XML разметки могут иметь неограниченное число свойств (например, автор или номер версии). Тэги, свойства и структурные элементы XML позволяют формировать информацию о контексте и интерпретировать значение элемента XML, что «открывает новые возможности для построения интеллектуальных поисковых машин, средств многомерного анализа данных, агентов и т. п.» [97].

Фрагмент XML-файла ТД после предобработки:

```
<structure>
  <vertex name="chapter1" relation="chapter"
    value="1,0000000000000000" parent="ROOT" />
  ...
  <vertex name="chapter15" relation="chapter"
    value="1,0000000000000000" parent="ROOT" />
</structure>
<content>
```

```
  <term paragraph="0" sentence="0" term="задан" value="1,0"/>
  ...
  <term paragraph="115" sentence="0" term="программисту" value="1,0"/>
</content>
```

Фрагмент XML-файла как элемента концептуального индекса электронного архива:

```
<structure>
  <vertex name="chapter1" relation="chapter"
    value="1,0000000000000000" parent="ROOT" />
  ...
  <vertex name="chapter15" relation="chapter"
    value="1,0000000000000000" parent="ROOT" />
</structure>
<content>
  <instance name="Информац" value="0,0044204320722766" />
  ...
  <instance name="Этап" value="0,0103495718735234" />
</content>
...
<icontent>
  <term name="celeron" value="0,0079821059246524" />
  ...
  <term name="языке" value="0,0127210234731346" />
</icontent>
```

Онтологический сервер Sesame – репозиторий онтологий предметных областей.

Sesame является онтологическим хранилищем с возможностью выполнения логического вывода по RDF-тройкам и построения запросов на языках SPARQL и SeRQL. Технология Sesame предлагает обширный перечень инструментов для работы с информацией, представленной в формате RDF.

Язык RDF является спецификацией представления информации об объектах и их отношениях друг с другом. Часто язык RDF применяется для определения метаданных Web-ресурсов, например, название Web-страницы, ее автор, дата изменений и т. д. Тем не менее, нотация RDF также может применяться

для описания объектов реального мира. Это дает возможность использовать данную информацию не только людям, но и вычислительным системам. Особенности языка RDF обеспечивают возможность обмена информацией между различными программными системами без трансформации изначального смысла.

Основная идея языка RDF состоит в обозначении объектов с помощью идентификаторов URL (Uniform Resource Identifiers). Контекстная информация объекта представляется с помощью тройки: объект-свойство-значение.

Реализация Sesame предполагает применения сервлета (Java Servlet Application) и работает внутри контейнера сервлетов Apache Tomcat. Связь с ним осуществляется по протоколу HTTP (HyperText Transfer Protocol). Одной из основных подсистем в Sesame является репозиторий, который представляет собой контейнер для RDF. Данный контейнер может быть представлен как объект или множество объектов на языке Java или как реляционная база данных.

Фрагмент онтологии предметной области в формате RDF:

```
<rdf:RDF
<!-- Life Circle Ontology -->
  <Stage rdf:ID="Разработка концепцииАС "/>
  <Stage rdf:ID="Изучениеобъекта_">
    <PartOfStage rdf:resource="Разработка концепцииАС "/>
  </Stage>
  ...
<!-- Life Circle Ontology End -->
<!-- Domain Ontology -->
  <Concept rdf:ID="Серия стандартов 34"/>
  <Concept rdf:ID="Общетехнические термины">
    <PartOf rdf:resource="Серия стандартов 34"/>
  </Concept>
  ...
<!-- Domain Ontology End -->
<!-- Project Ontology -->
  <Instance rdf:ID="Прибор"/>
  ...
  <Term rdf:ID="документ"/>
```

```
  ...
  <ConceptInstance rdf:ID="CInst1">
    <ConnectToCIConcept rdf:resource="Программно техническийкомплексАС  "/>
    <ConnectToCIInstance rdf:resource="Прибор"/>
  </ConceptInstance>
  ...
  <InstanceTerm rdf:ID="CIndex1">
    <ConnectToInstance rdf:resource="Прибор"/>
    <ConnectToTerm rdf:resource="документ"/>
    <ConnectToFreq rdf:datatype="float">
      0,0483945306899893
    </ConnectToFreq>
  </InstanceTerm>
  ...
<!-- Project Ontology End -->
</rdf:RDF>
```

На рисунках 5.4 и 5.5 представлены графические описания фрагмента RDF-схемы и фрагмента онтологии предметной области ИПР, соответственно. В приложении 2 представлены фрагменты онтологии предметной области ИПР.

Модуль концептуального индексирования технических документов

Процесс концептуального индексирования технических документов состоит из следующих шагов:

- загрузка документов;
- анализ структуры документов;
- удаление стоп-слов;
- выделение основы слова, получение термов – стемминг;
- выполнение расчета относительной частоты встречаемости термов;
- выполнение расчета степени выраженности понятий;
- построение концептуального индекса для множества документов.

В качестве входных данных для концептуального индексирования выступают предобработанные документы в формате XML, которые включают информацию о содержании и структуре разделов исходных документов.

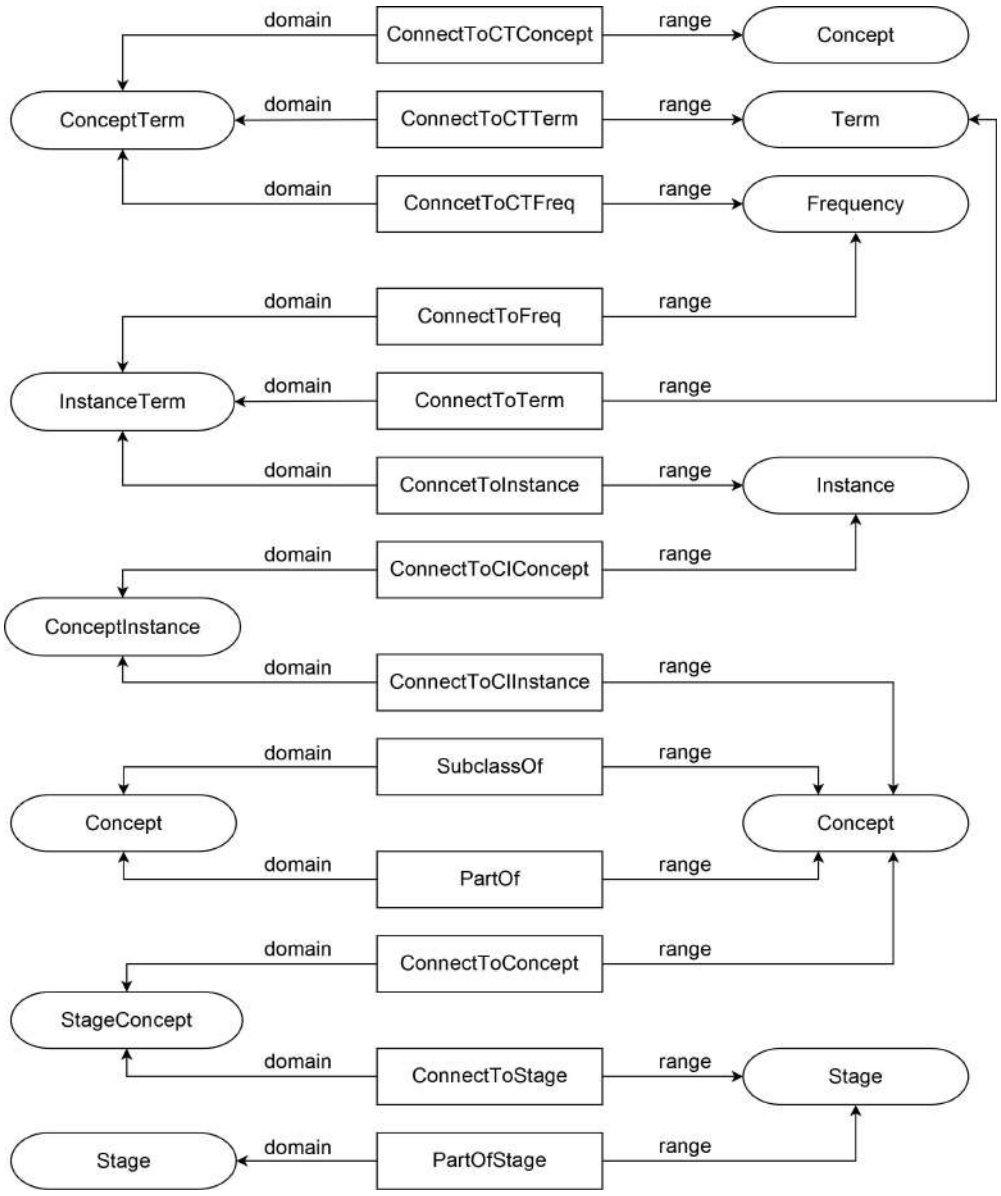


Рис. 5.4. RDF-схема онтологии предметной области

Перед началом индексирования определяются необходимые для функции индексирования классы онтологии предметной области – понятия из проектов электронного архива, текстовые входы указанных понятий, класс, определяющий связь понятия с его текстовым входом.

Параметризация генетического алгоритма оптимизации онтологических представлений ТД предполагает определение: размера популяции, доли элитных хромосом, значений вероятности мутации, критерия завершения алгоритма, максимального числа итераций, которые выполнит алгоритм перед завершением.

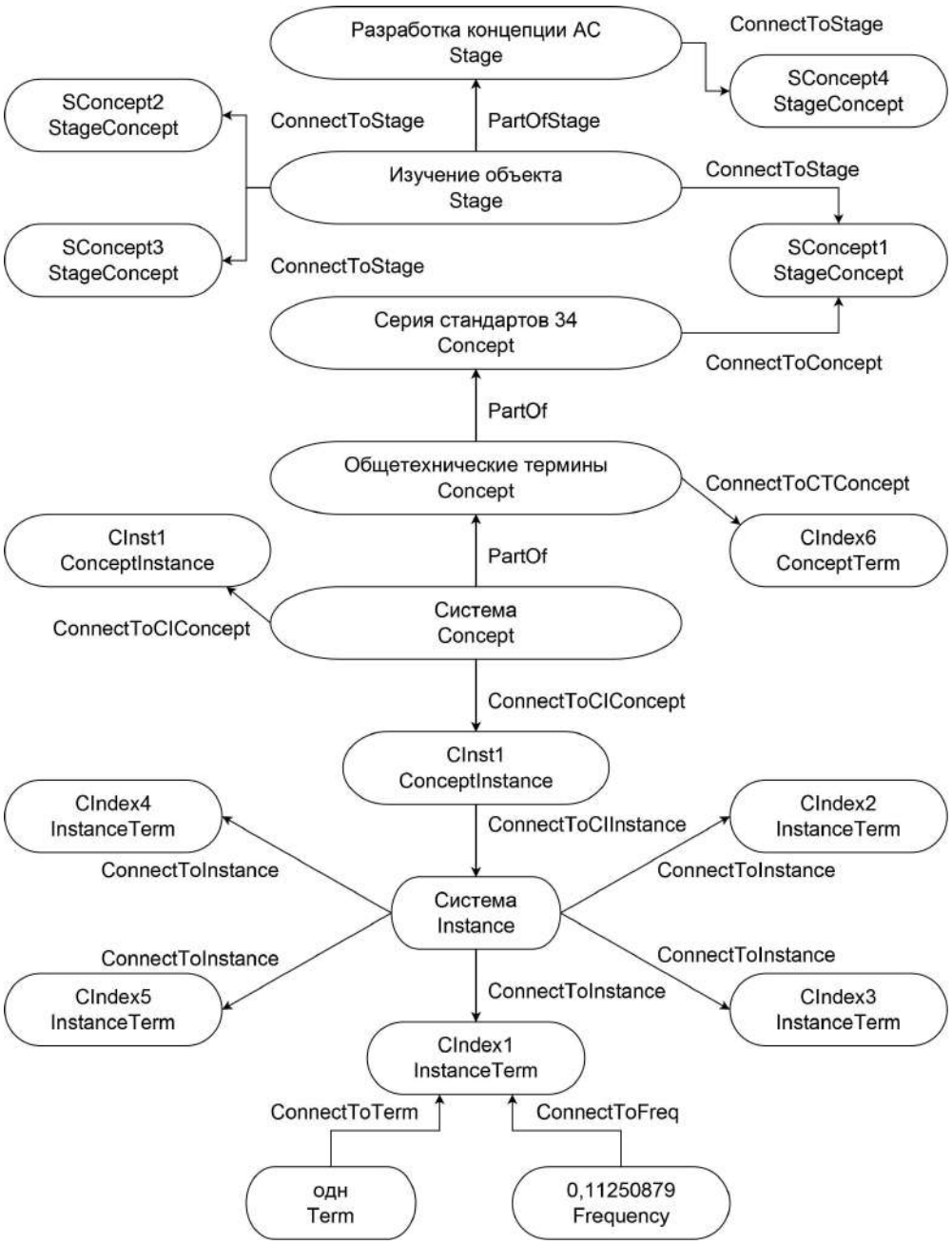


Рис. 5.5. Фрагмент онтологии предметной области, согласно схеме RDF

Модуль концептуальной кластеризации технических документов

В основе концептуальной кластеризации лежит модифицированный алгоритм нечеткой кластеризации Fuzzy-C-Means, с учетом формирования онтологического представления ТД как иерархии. Исходными данными являются представления ТД в концептуальном индексе, полученные на этапе индексирования.

При настройке параметров концептуального кластеризатора указываются:

- веса отношений между понятиями онтологии. Реализована возможность расчета указанных весов в автоматическом режиме с помощью генетического алгоритма при наличии данных с результатами экспертного разбиения исходного множества ТД;
- подмножество понятий онтологии, используемые при структуризации исходного множества ТД и/или расчете весов отношений между понятиями онтологии;
- определяются необходимые для процесса структуризации классы онтологии предметной области: понятия и классы отношений между понятиями онтологии.

Модуль концептуальной кластеризации предполагает функционирование с несколькими типами представлений ТД.

1. Онтологическое представление (после операции концептуального индексирования).
2. Онтологическое представление без использования генетической оптимизации.
3. Традиционное представление (множество пар терм-частота).
4. Структурное представление (с учетом структуры разделов документов).

У пользователя есть возможность выбора режима построения навигационной структуры: автоматический или ручной. Отличие данных режимов заключается в способе задания числа кластеров. В автоматическом режиме число кластеров рассчитывается с помощью эмпирической закономерности $M \approx \sqrt{N/2}$ (N – количество ТД) [159]. В ручном режиме количество кластеров определяется пользователем. При использовании автоматического режима построения навигационной структуры параметры FCM-алгоритма задаются перед началом процесса построения, в ручном режиме данные параметры можно указывать перед началом каждого шага. Максимальное количество шагов автоматического режима указывается пользователем.

5.3. Подсистема визуализации и оценки качества онтологии

Общепризнанным методом, обеспечивающим восприятие и понимание больших объемов информации, является визуализация информации с применением графовых моделей. Онтология предметной области может быть представлена в виде орграфа, вершины которого изображают сущности, такие как классы, объекты и атрибуты онтологии, а ребра изображают отношения между этими сущностями. Эксперт, работающий с графическим представлением онтологии, намного лучше понимает ее структуру и благодара этому повышается качество работы эксперта.

Особенностями подсистемы визуализации является возможность ее использования как компоненты интеллектуального электронного архива, работающего с хранилищем Sesame, а также реализация в ней функции оценки качества онтологии с применением математического аппарата, основанного на нечетких соответствиях.

На рисунке 5.6 представлено структурно-функциональное решение подсистемы визуализации.

Описание онтологии состоит из двух частей – RDF и RDF Schema. С помощью модели RDF непосредственно описывается сама онтология, а с помощью RDF Schema описывается ее структура. Для обработки и хранения RDF-документов используется API Sesame. Модуль извлечения триплетов онтологии подключается к Sesame и с помощью специального языка запросов SeRQL извлекает данные в виде RDF-триплетов «субъект-предикат-объект».

Модуль настроек визуализации обрабатывает данные из RDF Schema и позволяет создать настройки репозитория, которые определяют, что будет изображаться в качестве вершин и ребер графа.

С помощью модуля преобразования данных из извлеченных данных в виде RDF триплетов создается множество java-объектов, необходимых для дальней-

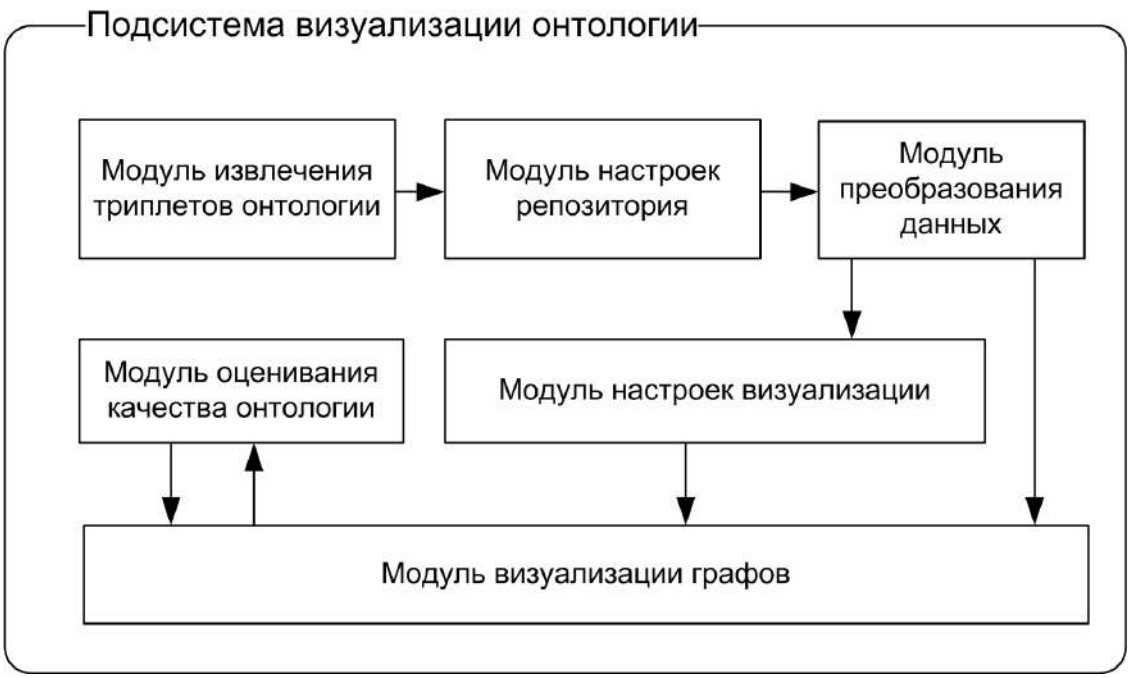


Рис. 5.6. Структура подсистемы визуализации онтологии САПР

шей работы системы. Модуль настроек визуализации определяет внешний вид графа, в частности, позволяет определить цвет для каждого элемента графа.

Для создания модели графа и ее визуализации используется свободно распространяемый фреймворк JUNG версии 2.0.1. Данный фреймворк представляет собой пакет java-классов, необходимых для работы с графом.

Модуль оценивания качества онтологии выделяет из модели графа группы однородных понятий и производит оценку качества каждой группы.

На рис. 5.7 отображен результат работы подсистемы визуализации, на котором представлен онтологический граф с цветовым выделением вершин графа для оценки качества фрагментов онтологии.

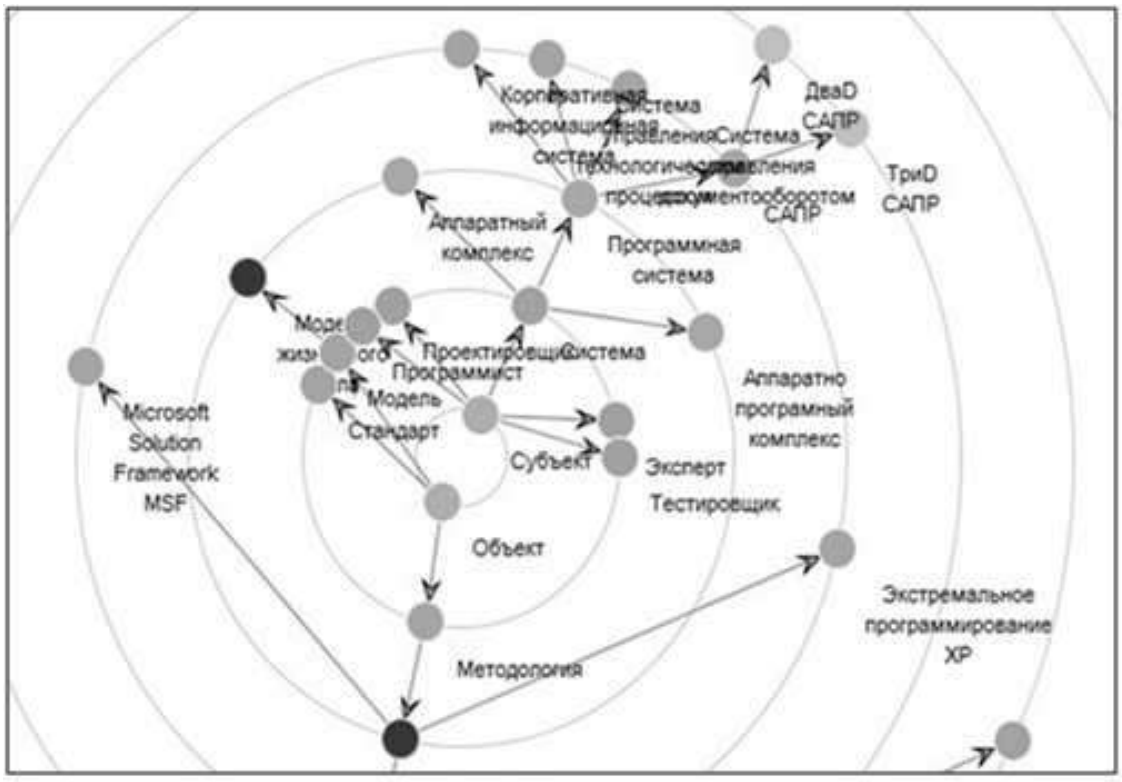


Рис. 5.7. Результат визуализации фрагмента онтологии САПР

5.4. Подсистема информационной поддержки автоматизированного проектирования АС

5.4.1. Структурно-функциональное решение подсистемы информационной поддержки

Интеллектуальная подсистема информационной поддержки проектирования автоматизированных систем – это программная система, которая предназначена для обеспечения контекстного взаимодействия проектировщика с электронным архивом ТД. Важными функциями указанной системы являются:

- Концептуальное индексирование текстовых технических документов и проектных диаграмм.
- Сохранение и управление системой онтологий и концептиуальных представлений ТД.
- Формирование и хранение индивидуальных профилей пользователей, которые учитывают информационные потребности субъектов проектирова-

ния на различных стадиях жизненного цикла АС.

- Формирование контекстно-ориентированных проектных запросов к электронному архиву.
- Поиск текстовых технических документов и проектных диаграмм из электронного архива с использованием профилей проектировщиков.

Подсистема информационной поддержки проектирования входит в состав электронного архива ФНПЦ АО «НПО «Марс», структурная схема которого представлена на рисунке 5.8. Для работы с указанным электронным архивом определены несколько ролей:

- Архивариус – задачами пользователя является заполнение архива электронными копиями технических документов;
- Аналитик – пользователь ведет учет статистической информации о работе электронного архива;
- Пользователь – конечный потребитель содержимого электронного архива.

В составе электронного архива НПО «Марс» есть система разграничения прав доступа, которая определяет группы и/или отдельные документы, выдаваемые системой информационной поддержки.

Электронный архив ФНПЦ АО «НПО «Марс» хранит большое количество ТД, которые используются подсистемой информационной поддержки в качестве входных данных. Технические документы используются в процессе индексирования [65], [66], полученные наборы индексов хранятся в электронном архиве. Онтологические ресурсы записываются в информационную базу сервера онтологий Sesame. Задаваемые пользователем запросы обрабатываются подсистемой информационной поддержки и используются в процессе поиска в электронном архиве.

Рис. 5.8. Основные подсистемы электронного архива ФНПЦ АО «НПО «Марс»

5.4.2. Модуль формирования индивидуального профиля проектировщика

Реализация программного модуля формирования профиля проектировщика предполагает использование модели интеллектуального агента [91]. В качестве интеллектуального агента понимается система, которая через набор датчиков получает информацию о среде и воздействующая через исполнительные механизмы на данную среду. Под свойством интеллектуальности следует понимать наличие у агента модели пользовательских потребностей и механизма их удовлетворения. Таким образом, интеллектуальный агент должен обладать следующими характеристиками.

- Автономность – агент выполняет большую часть своей работы автономно, без взаимодействия с человеком или другими агентами.
- Коммуникабельность – агент должен уметь общаться с пользователем, получая от него задания и предоставляя результаты.
- Адаптируемость и адаптивность – в процессе общения с пользователем агент должен уметь подстраиваться под привычки и методы работы конкретного пользователя.
- Восприимчивость – агент в окружающей его информационной среде должен воспринимать определенным образом изменения, происходящие в окружающей среде и реагировать на данные изменения.
- Проактивность – агент не только должен формально выполнять поставленную задачу поиска, но и должен собирать при этом полезную для пользователя информацию, относящуюся к запросу пользователя.

В данной работе определены следующие задачи интеллектуального агента: доопределение знаний о предметной области; выполнение анализа пользовательских потребностей; формирование ранжированного списка документа на основе анализа потребностей пользователя в поисковой системе. Для данных задач будем применять интеллектуального агента, который имеет архитектуру, пред-

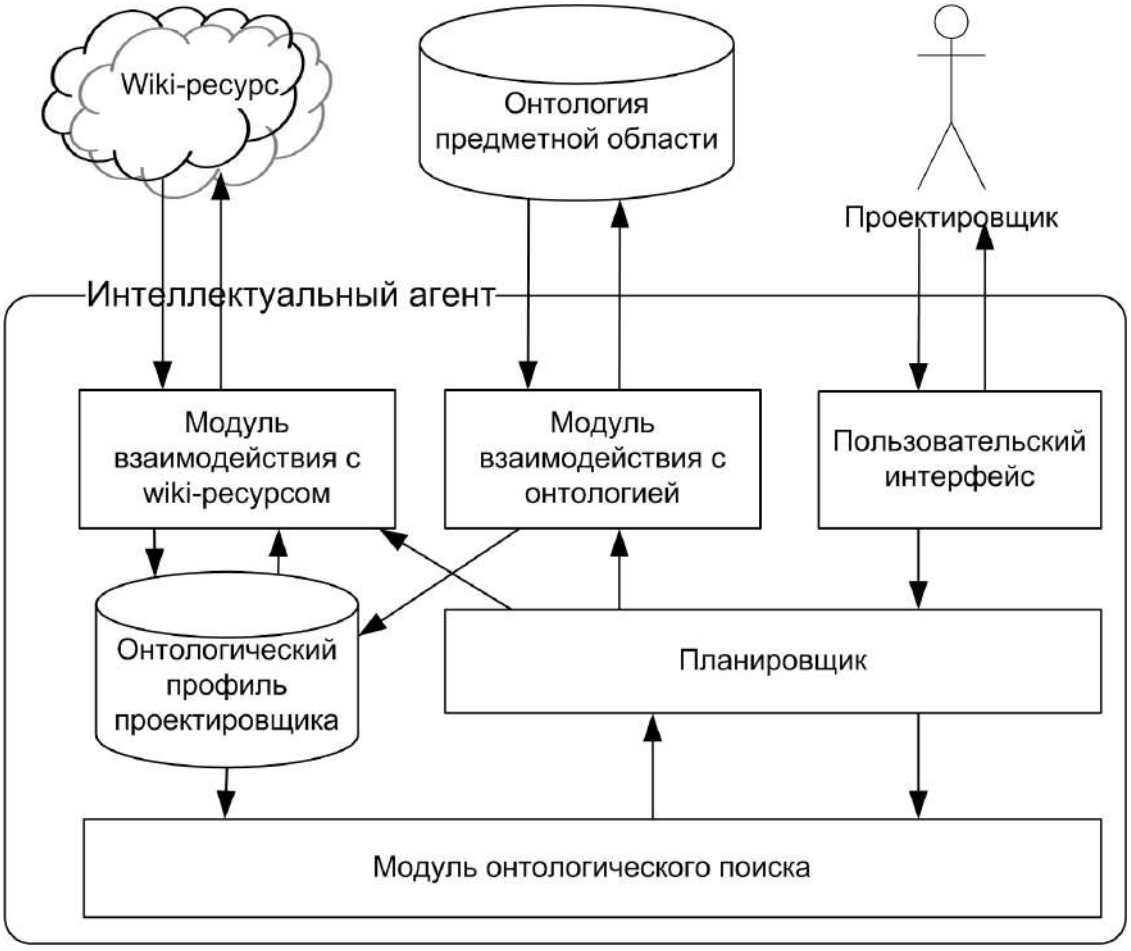


Рис. 5.9. Структура интеллектуального агента, формирующего профиль проектировщика

ставленную на рисунке 5.9.

Пользовательский интерфейс осуществляет фоновое взаимодействие проектировщика с интеллектуальным агентом. Пользователь вводит контекстные запросы к электронному архиву, выраженные в виде набора терминов, получает список релевантных документов и оценивает документы на степень удовлетворенности своих потребностей. Планировщик осуществляет следующие функции: преобразует пользовательский запрос к онтологическому виду, выраженный в виде множества концептов предметной области; передает преобразованный запрос модулю онтологического поиска; получает список релевантных документов и выводит их в модуле пользовательского интерфейса; пополняет пользовательский профиль знаниями о предметной области.

Индивидуальный профиль проектировщика содержит знания и опыт о взаимодействии пользователя с электронным архивом. Данные знания выражены

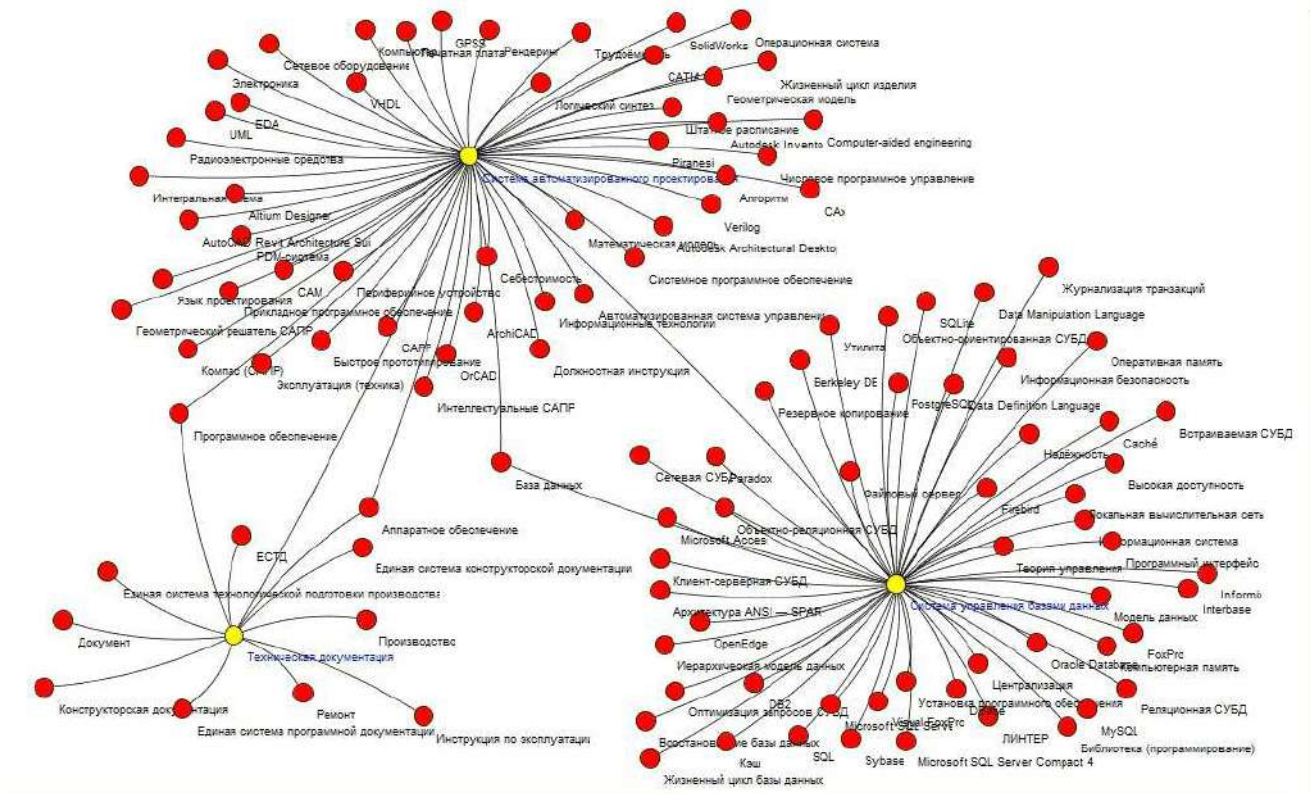


Рис. 5.10. Структура семантической сети, извлеченной из wiki-ресурсов

в виде множества концептов онтологии предметной области и концептов, извлеченных из wiki-ресурсов. В дальнейшем пользовательский профиль используется в процессе формирования контекстных запросов.

Модуль онтологического поиска получает от планировщика онтологический запрос, выполняет поиск релевантных документов и возвращает набор ранжированных документов планировщику. В модуле взаимодействия с wiki-ресурсом реализованы функции извлечения концептов [80]. Пример результата извлечения концептов из wiki-ресурса (в качестве которого выступала Википедия) представлен на рисунке 5.10.

Основные программные модули контекстного поиска были разработаны в процессе выполнения научно-исследовательской работы №1167 в рамках базовой части государственного задания в сфере научной деятельности по Заданию №2014/232 за 2015 год (Приложение №3).

5.5. Выводы по пятой главе

1. Разработанная архитектура интеллектуального проектного репозитория отличается от известных включением онтологических подсистем, позволяющих при анализе информационных ресурсов архива и выполнении проектных запросов учитывать семантический контекст выполняемых процедур и опыт проектировщика.
2. Формирование навигационной структуры в виде иерархии кластеров технических документов позволяет сократить пространство поиска документов в архиве. Сгруппированные в соответствии с мерой семантического расстояния информационные ресурсы электронного архива определяют иерархическую модель репозитория в терминах предметной области проектной организации.
3. Сложности синтаксиса моделей представления прикладных онтологий проектных организаций и большая трудоемкость построения онтологических ресурсов приводит к необходимости разработки специализированных средств поддержки онтологического инжиниринга. В составе архитектуры ИПР предлагается использовать подсистему визуализации и оценки качества онтологии предметной области проектной организации, позволяющую оптимальным образом определять состав текстовых входов понятий онтологии при ее создании и реконструкции.
4. Разработанная подсистема информационной поддержки автоматизированного проектирования АС позволяет уточнять проектные запросы пользователей к проектному архиву, что особенно важно в случае неявного представления информационной потребности в виде набора ключевых слов, что характерно для начальных этапов проектирования АС.

Глава 6

Анализ результатов вычислительных экспериментов по эксплуатации интеллектуального репозитория

6.1. Анализ качества структуризации электронного архива ФНПЦ АО «НПО «Марс»

Согласно [55], будем считать структуризацию тем более качественной, чем ближе разбиение массива документов, полученное в результате работы алгоритма структуризации, к разбиению того же массива документов, полученному в результате проведения экспертной кластеризации.

В результате имеем целевую функцию, формализующую качество структуризации, используя два критерия – отсутствие документов в кластере и наличие «лишних» документов в кластере:

$$\hat{f}_i = 1 - \frac{\max(\sum_{i=1}^{M} \bar{K}^i, \sum_{i=1}^{M} \hat{K}^i)}{N}, \tag{6.1}$$

где $\bar{K}^i$ – множество отсутствующих документов, входящих в i-й кластер согласно сопоставлению результатов экспертного и автоматического разбиений, $\hat{K}^i$ – множество «лишних» документов, входящих в i-й кластер согласно сопоставлению результатов экспертного и автоматического разбиений, $i = \overline{1, M}$ – номер кластера, M – количество кластеров, N – количество документов.

Основные вычислительные эксперименты проводились на базе электронного архива проектно-технической документации ФНПЦ АО «НПО «Марс» (Приложение 4). Полученные результаты легли в основу разработанной программной системы интеллектуального анализа текстовой информации (Приложение 5). Руководители основных подразделений и инженерно-технические работники

ФНПЦ АО «НПО «Марс» прошли обучение по программам повышения квалификации, в рамках которых преподавались элементы методологии построения онтологических систем для электронных архивов (Приложение 6).

Состав онтологии предметной области

При анализе результатов работы концептуального индексатора и концептуального кластеризатора на множестве документов электронного архива ФНПЦ АО «НПО «Марс» была использована онтология предметной области, содержащая различные серии стандартов, применяемые в НПО «Марс». Среди них можно выделить следующие.

1. Информационные технологии. Взаимосвязь открытых систем (ГОСТ 34). Содержит 108 понятий онтологии.
2. Единая система программной документации (ГОСТ 19). Содержит 111 понятий онтологии.

Другая часть онтологии сформирована на основе выборки документов электронного архива НПО «Марс», состоящей из 5017 ТД. Данный уровень содержит 81 понятие и 10 078 уникальных терма, составляющих терминологическое окружение понятий.

Таким образом, онтология предметной области имеет в своем составе 300 понятий: 219 понятий из стандартов и 81 понятие из проектов, а также 10 078 уникальных термов.

Описание множества документов электронного архива ФНПЦ АО «НПО «Марс»

Для осуществления анализа результатов работы системы на множестве документов электронного архива НПО «Марс» экспертом была подготовлена выборка, состоящая из 5017 ТД и сгруппированная по разным основаниям классификации:

- по классу документации – 4 группы (ЕСКД, ЕСПД, ЕСТД, Нормативные);

- по виду документации – 52 группы (ГОСТ 2.601, 2.602, 2.102, 2.701 и 3.1201);
- по разделу документации – 27 групп (ГОСТ 2.106 и сложившаяся на предприятии система присвоения документам децимальных номеров);
- по тематике работ – 28 групп (изделия, рассматриваемые в документах).

При этом данная выборка ТД является смешанной, и примерно на 1/20 часть состоит из нормативных документов.

Оценка качества структуризации множества документов электронного архива ФНПЦ АО «НПО «Марс»

Для проведения экспериментов по оценке качества структуризации множества документов ЭА НПО «Марс» был построен индекс, содержащий в своем составе онтологические и традиционные представления ТД. На следующем шаге полученный индекс был подвергнут различным вариантам структуризации с последующим расчетом качества структуризации:

- структуризация средствами системы Oracle Text (FCM-алгоритм) традиционных представлений ТД;
- структуризация средствами FCM-алгоритма кластеризации представлений ТД в форме «термин-частота»;
- структуризация средствами модифицированного FCM-алгоритма кластеризации онтологических представлений ТД;
- структуризация средствами модифицированного FCM-алгоритма кластеризации онтологических представлений ТД с учетом моделей жизненного цикла (ЖЦ).

Оценка качества структуризации рассматриваемой выборки представлена в таблице 6.1. Лучшие значения оценочной функции представлены для онтологических представлений с учетом моделей ЖЦ, лишь для структуризации по виду документа система Oracle Text показала лучшие результаты (рисунок 6.1). Отдельное внимание следует уделить разбиению тестовой выборки ТД по

тематике работ, так как данный вариант структуризации наиболее близко соответствует основной задаче ИПР – структуризации электронного архива ТД по содержанию отдельных документов. Согласно значениям целевой функции, результаты структуризации онтологических представлений ТД с учетом моделей ЖЦ примерно на 40% лучше по сравнению с результатами системы Oracle Text.

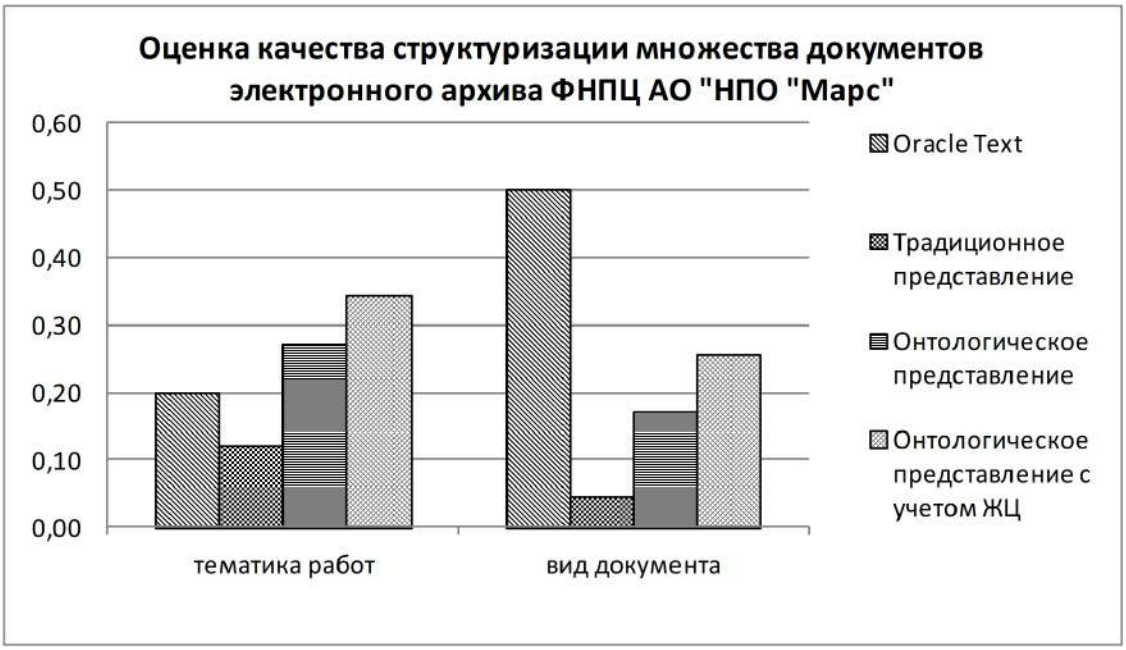


Рис. 6.1. Оценка качества структуризации множества документов электронного архива ФНПЦ АО «НПО «Марс»

Таблица 6.1. Оценка качества структуризации множества документов электронного архива ФНПЦ АО «НПО «Марс»

Вид разбиения	Oracle Text	Традиционное представление	Онтологическое представление	Онтологическое представление с учетом ЖЦ
Класс документации	0,45	0,40	0,39	**0,48**

продолжение таблицы 6.1

Вид разбиения	Oracle Text	Традиционное представление	Онтологическое представление	Онтологическое представление с учетом ЖЦ
Раздел документации	0,33	0,25	0,33	**0,39**
Тематика работ	0,20	0,12	0,27	**0,34**
Вид документа	**0,50**	0,05	0,17	0,26

Анализ временных затрат на процессы индексирования и структуризации множества документов электронного архива ФНПЦ АО «НПО «Марс»

Для оценки временных затрат на процессы индексирования и структуризации множества документов электронного архива НПО «Марс» необходимо рассмотреть результаты выполненных экспериментов с помощью реализованной системы.

На процесс индексирования множества документов электронного архива НПО «Марс» было затрачено 41 656,58 секунд. Таким образом, индексирование крупных выборок ТД требует значительных временных затрат, однако в силу специфики процесса индексирования данный результат можно считать приемлемым.

В таблице 6.2 содержатся сведения о временных затратах на одну итерацию процесса структуризации множества документов электронного архива НПО «Марс». При увеличении количества документов и кластеров время структуризации традиционных представлений ТД значительно возрастает. Разница в

скорости структуризации между традиционными и онтологическими представлениями ТД является значительной и зависит от размера выборки ТД, количества понятий онтологии предметной области, количества терминов в составе редуцированного множества терминов всей выборки ТД и количества кластеров (рисунок 6.2).

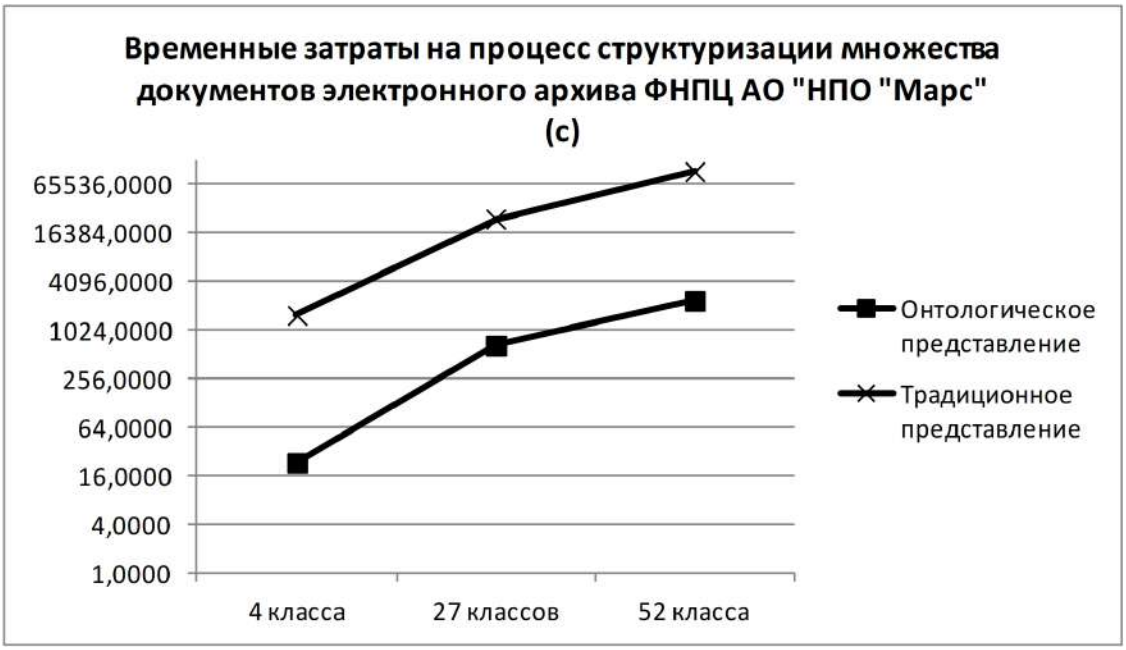


Рис. 6.2. Временные затраты на процесс структуризации множества документов электронного архива ФНПЦ АО «НПО «Марс»

Таблица 6.2. Временные затраты на одну итерацию процесса структуризации множества документов электронного архива ФНПЦ АО «НПО «Марс»

Тип представления	Класс документации	Вид документации	Раздел документации	Тематика работ
Онтологическое, сек	23,6430	2393,2295	670,0300	813,4364
Традиционное, сек	1577,3659	95232,4200	24659,5578	23118,5248

Оценка снижения времени выполнения проектных процедур с использованием навигационной структуры электронного архива

Для проведения экспериментов по оценке снижения времени выполнения проектных процедур с использованием навигационной структуры электронного архива в рамках опытно-конструкторской работы ФНПЦ АО «НПО «Марс» был использован существующий проект. В выборке, состоящей из 5017 ТД электронного архива НПО «Марс», осуществлялся поиск ТД, похожего по содержанию на пояснительную записку из состава рассматриваемого проекта (рисунок 6.3), с последующим подсчетом суммарного времени поиска документа (рисунок 6.4). В таблице 6.3 представлены зависимости суммарного времени поиска документа от вида разбиения множества ТД, при $\Delta t = 40$ секундам и $k_n = 0.15$.

Таблица 6.3. Суммарное время поиска документа в зависимости от вида разбиения множества ТД и используемых методов (сек)

Вид разбиения	Oracle Text	Кластеризация традиционных представлений	Навигационная структура ЭА	Навигационная структура ЭА с учетом ЖЦ
Класс документации	11060	54380	**4260**	4680
Раздел документации	1420	17660	2780	**700**
Тематика работ	1380	27640	1600	**700**
Вид документа	1280	14260	1600	**400**

Использование навигационной структуры электронного архива с учетом моделей ЖЦ вместо традиционных методов разбиения множества ТД на определенное число кластеров сократило суммарное время поиска документов в среднем на 13%.

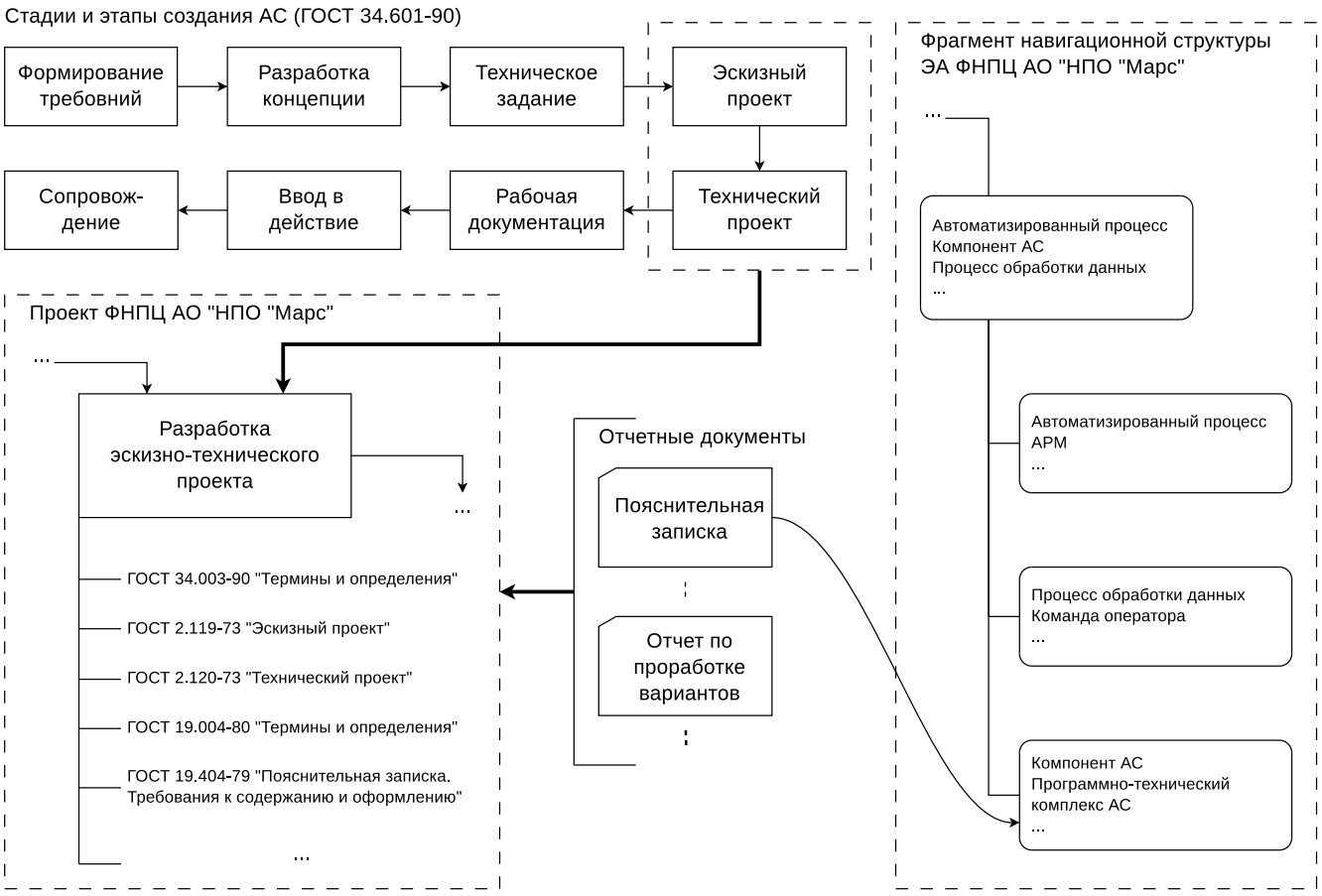


Рис. 6.3. Процесс поиска документа, похожего по содержанию на пояснительную записку из состава проекта ФНПЦ АО «НПО «Марс» с использованием навигационной структуры электронного архива

Навигационная структура электронного архива строится на основе данных о содержимом документов, тем самым необходимо использовать данные, содержащиеся в существующей системе управления электронным архивом НПО «Марс», для улучшения качества поиска необходимых документов. Использование дополнительной информации о содержимом электронного архива, при построении навигационной структуры данного архива, позволит сократить пространство поиска и повысить качество структуризации.

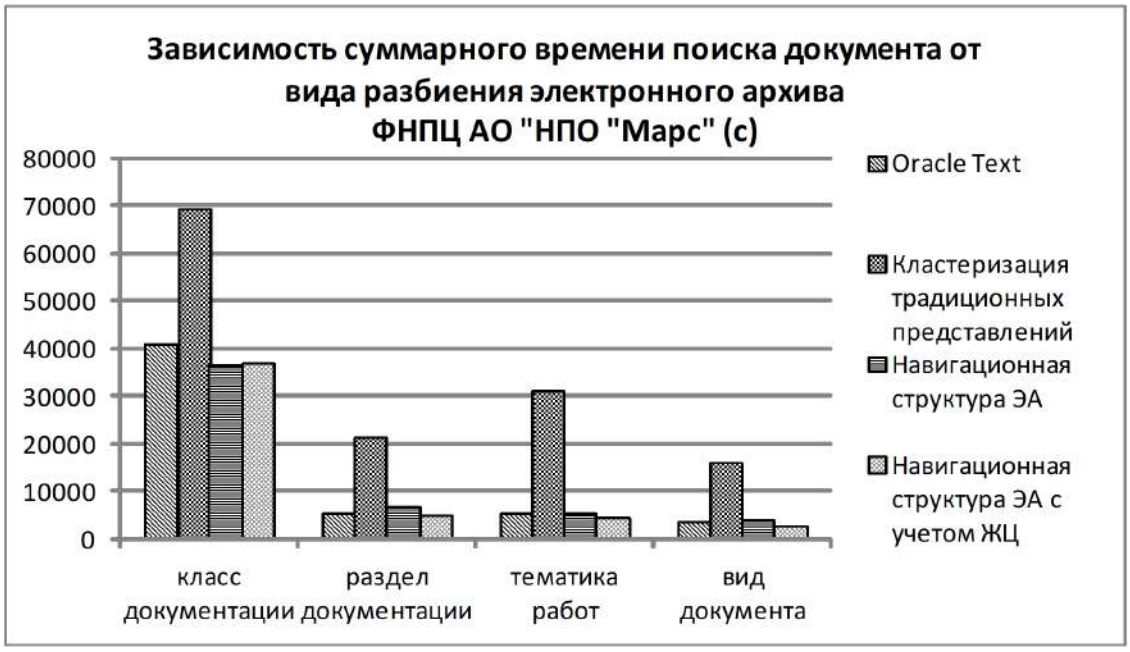


Рис. 6.4. Зависимость суммарного времени поиска документов от вида разбиения множества ТД

6.2. Исследование параметров генетической оптимизации в процессе концептуального индексирования

В процессе настройки параметров генетического алгоритма необходимо правильным образом подобрать значения размера популяции, процента элитных хромосом, вероятностей мутации сдвига текстового фрагмента и мутации объединения текстовых фрагментов для того, чтобы:

- обеспечивалась удовлетворительная сходимость генетического алгоритма;
- происходило формирование достаточного разнообразия вариантов хромосом (с целью выхода или непопадания в локальные экстремумы целевой функции);
- обеспечивался баланс между ростом количества текстовых фрагментов в процессе кроссинговера и их уменьшением благодаря оператору мутации с объединением текстовых фрагментов.

Для определения оптимальных значений параметров генетического алгоритма было проведено около 160 экспериментов и получены следующие резуль-

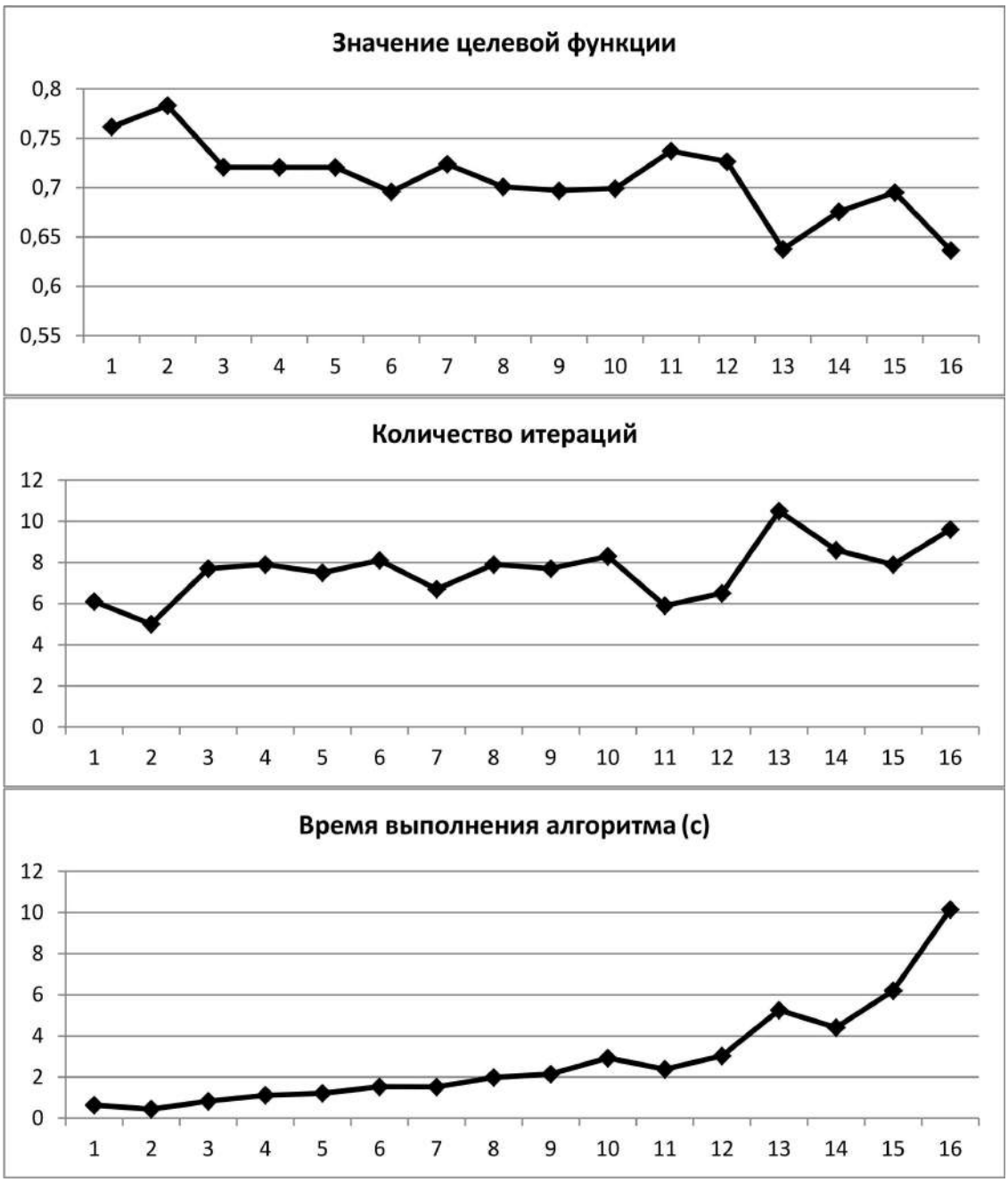


Рис. 6.5. Зависимость значения целевой функции, количества итераций и времени выполнения алгоритма от размера популяции

таты (каждая точка на графике соответствует среднему значению показателя по десяти экспериментам).

- Зависимость найденного значения целевой функции, количества итераций и времени выполнения алгоритма от размера популяции (рисунок 6.5). Размер популяции изменялся от 100 до 500 с шагом 50, от 500 до 1000 с шагом 100 и от 1000 до 2000 с шагом 500.

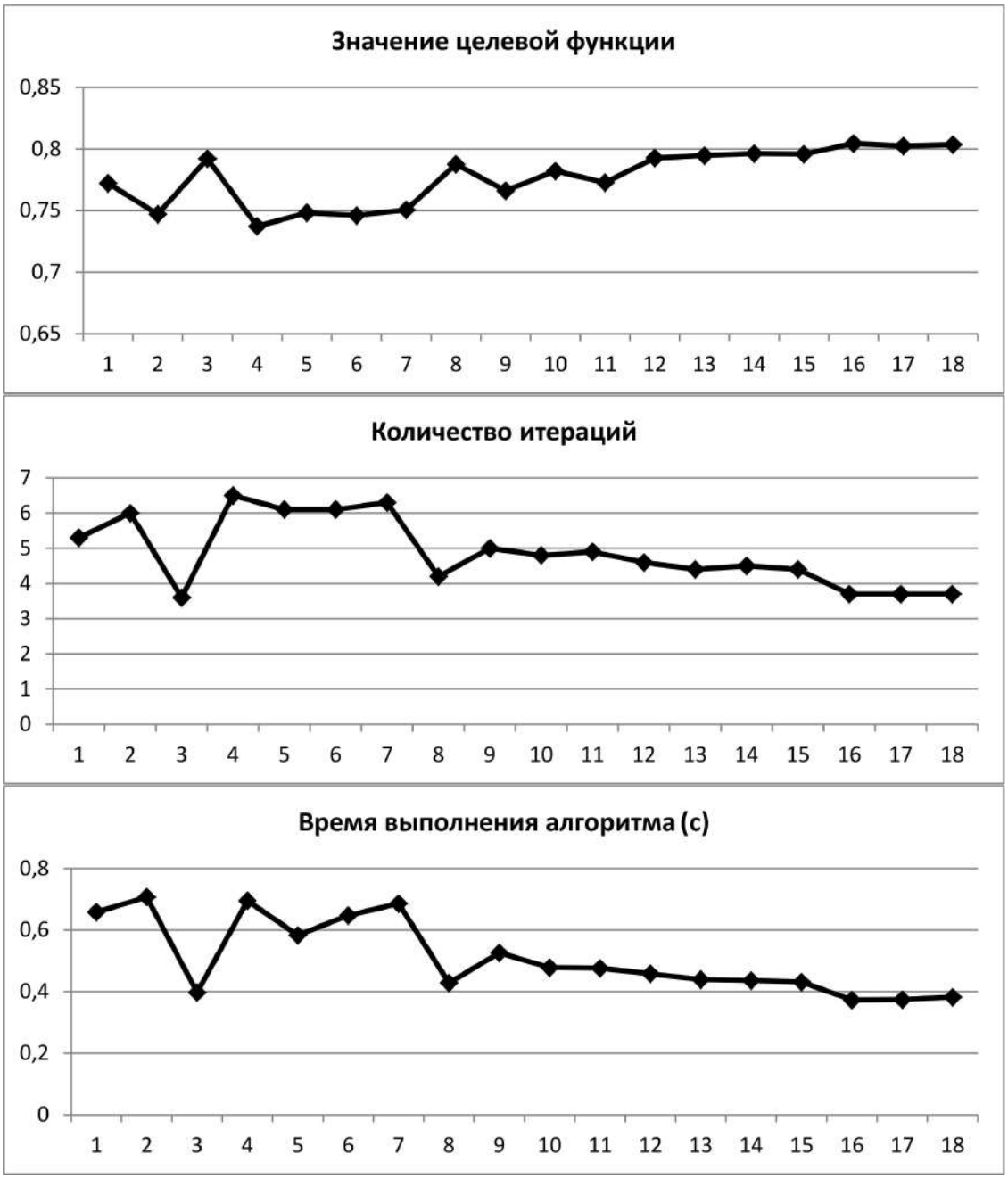


Рис. 6.6. Зависимость значения целевой функции, количества итераций и времени выполнения алгоритма от процента элитных хромосом

- Зависимость найденного значения целевой функции, количества итераций и времени выполнения алгоритма от процента элитных хромосом (рисунок 6.6). Значение данного параметра изменялось от 10% до 95% с шагом 5%.
- Зависимость найденного значения целевой функции, количества итераций и времени выполнения алгоритма от вероятности мутации сдвига тексто-

вого фрагмента (рисунок 6.7). Значение вероятности мутации сдвига текстового фрагмента изменялось от 10% до 100% с шагом 5%.

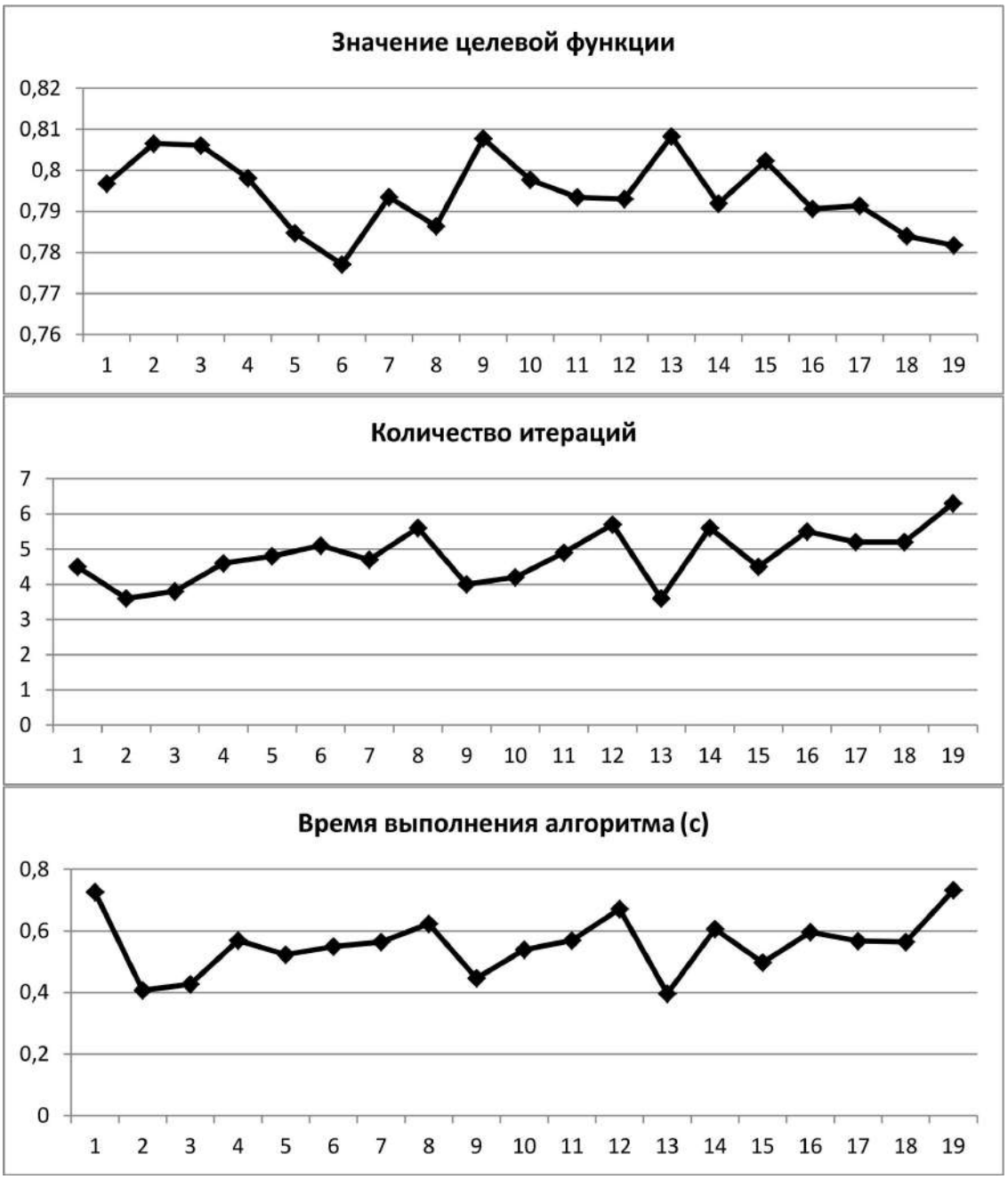


Рис. 6.7. Зависимость значения целевой функции, количества итераций и времени выполнения алгоритма от вероятности мутации сдвига текстового фрагмента

- Зависимость найденного значения целевой функции, количества итераций и времени выполнения алгоритма от вероятности мутации объединения текстовых фрагментов (рисунок 6.8). Значение вероятности мутации объединения текстовых фрагментов изменялось от 10% до 100% с шагом 5%.

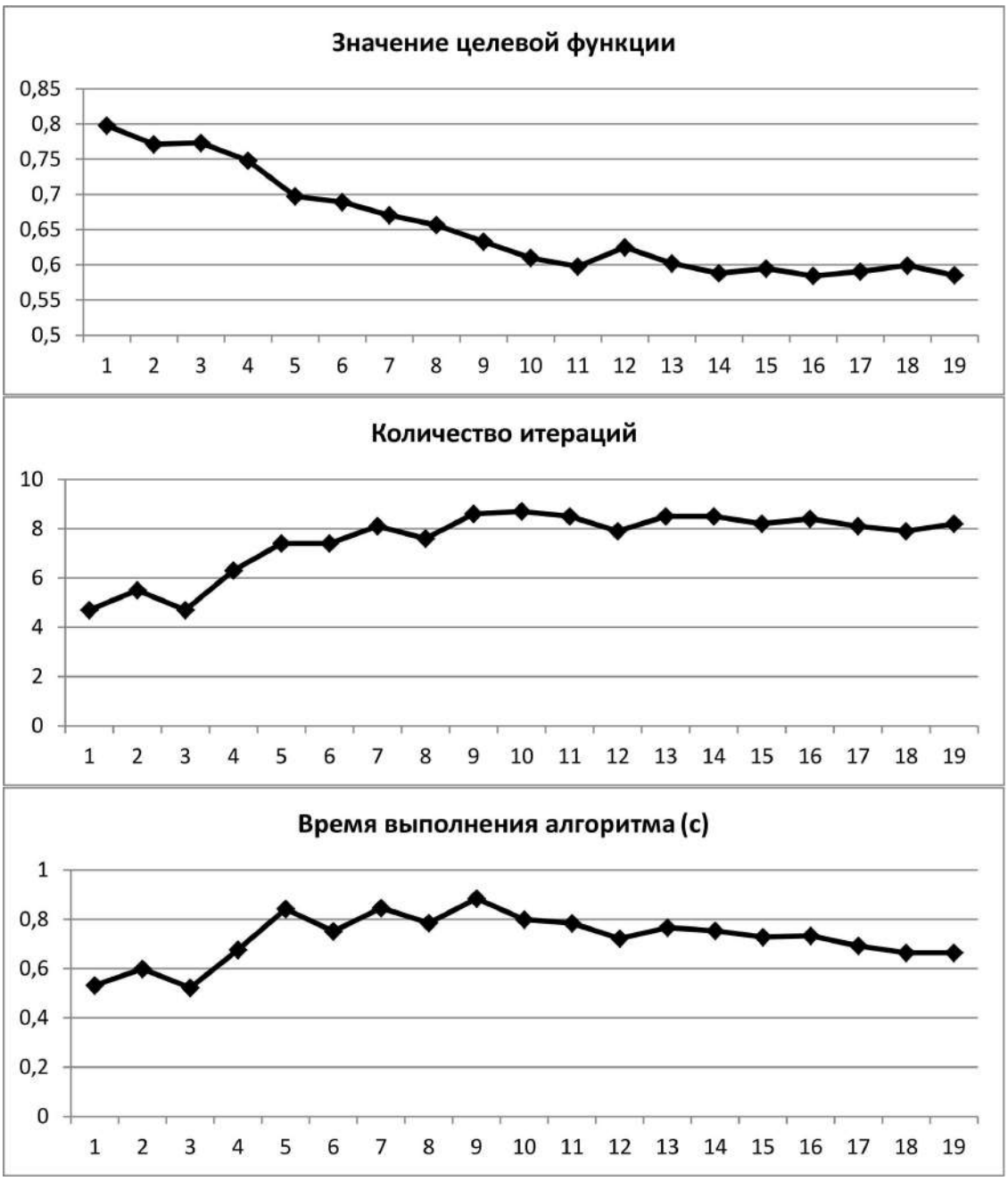


Рис. 6.8. Зависимость значения целевой функции, количества итераций и времени выполнения алгоритма от вероятности мутации объединения текстовых фрагментов

Определены следующие рекомендуемые параметры генетического алгоритма: размер популяции – от 300 до 400, процент элитных хромосом – 35%, вероятность мутации сдвига текстового фрагмента – 0,45, вероятность мутации объединения текстовых фрагментов – 0,55.

6.3. Результаты вычислительных экспериментов по формированию контекстно-ориентированных проектных запросов

На первой стадии вычислительных экспериментов рассматривался вид запроса, в котором явно или не явно определялась предметная область проекта. Сравнительные результаты по каждому профилю пользователя представлены в виде гистограмм. В качестве итоговых величин точности, полноты и F-меры использовались значения, которые наиболее часто встречались в ходе экспериментов. В результате экспериментов с применением профиля «Программист»

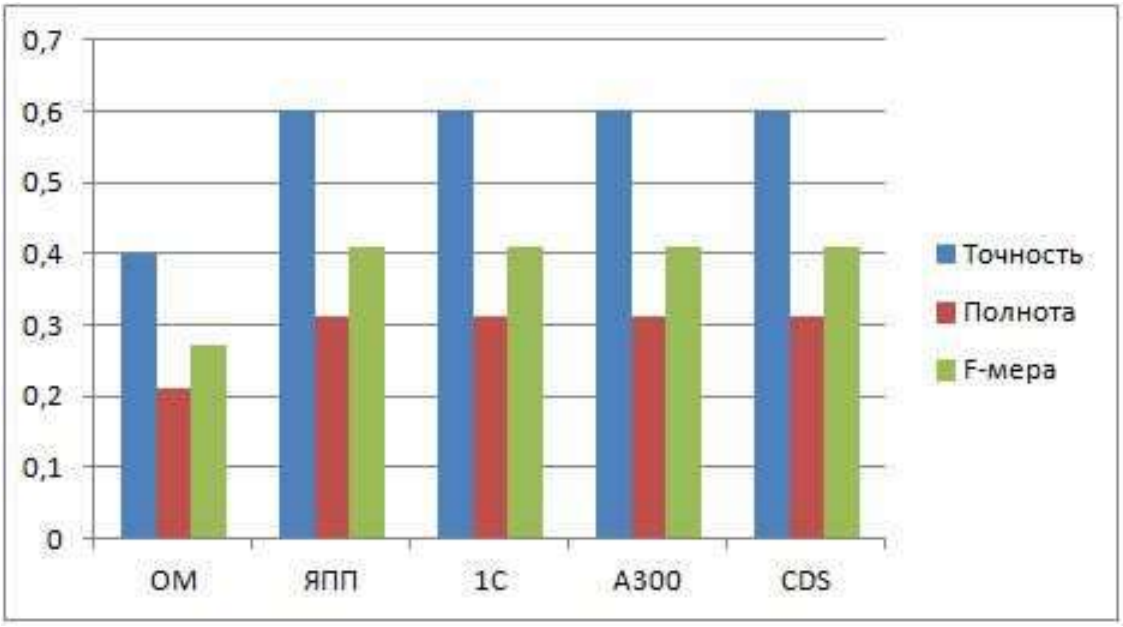


Рис. 6.9. Сравнение результатов экспериментов для профиля программиста с запросами, семантически явно определяющих предметную область

были получены следующие результаты для запросов, которые семантически явно определяют предметную область (рисунок 6.9) и для запросов, которые не явно идентифицируют предметную область (рисунок 6.10).

Следующий набор запросов определялся количеством терминов. На рисунке 6.11 и рисунке 6.12 представлены гистограммы, в которых отражена оценка влияния размера запроса на качество поиска ТД.

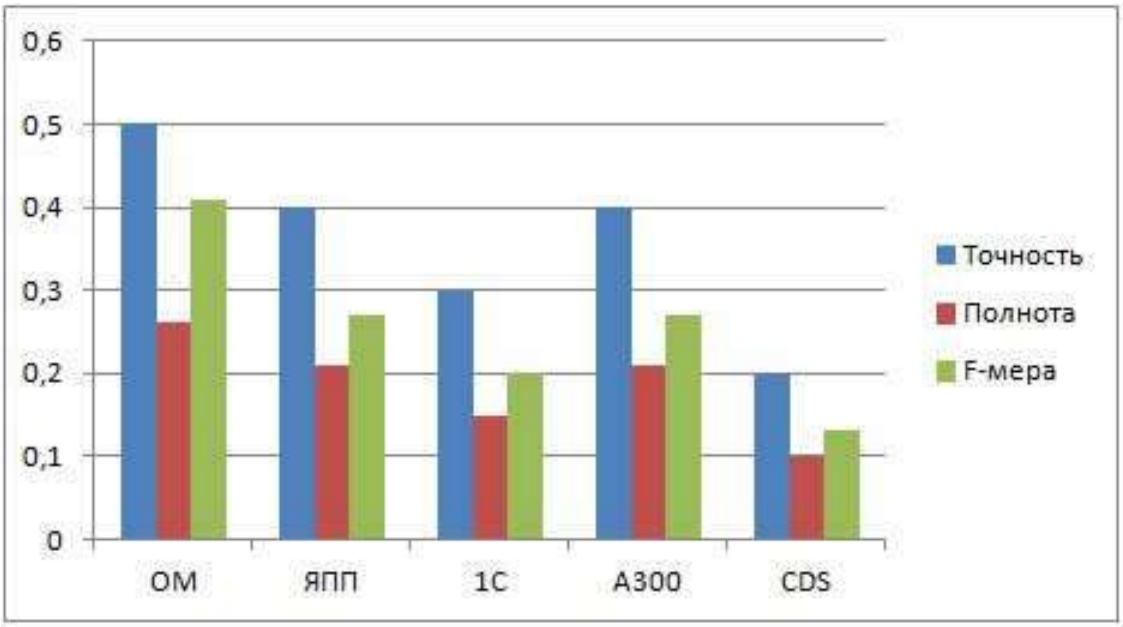


Рис. 6.10. Сравнение результатов экспериментов для профиля программиста с запросами, семантически неявно определяющих предметную область

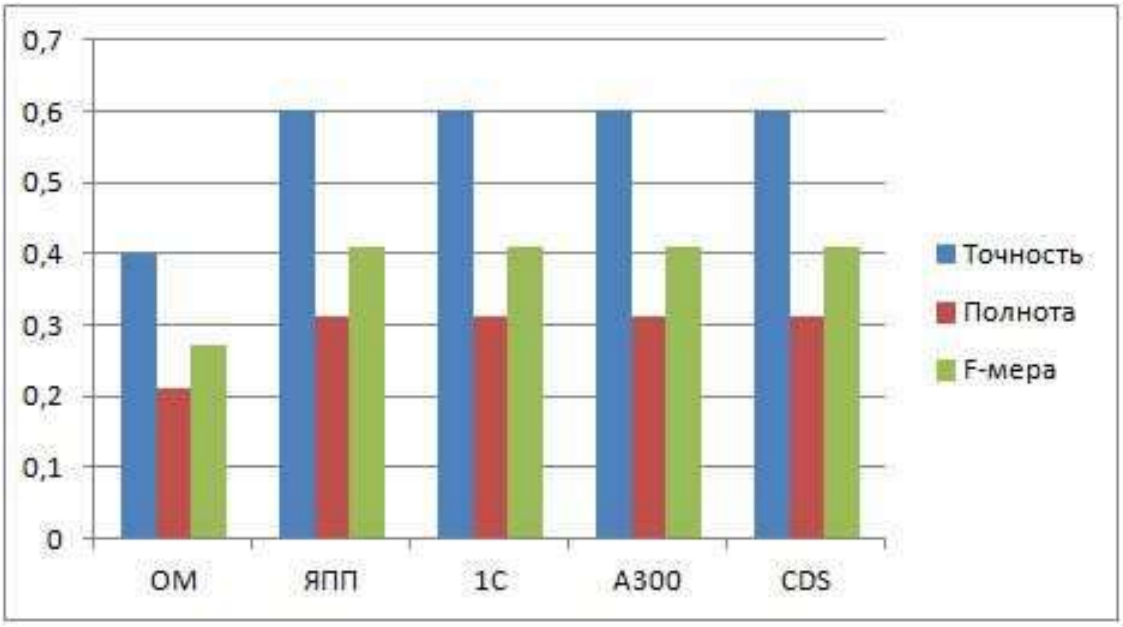


Рис. 6.11. Сравнение результатов экспериментов для профиля программиста с короткими запросами

В результате экспериментов с применением профиля «Инженер» получены следующие результаты для запросов, которые семантически явно определяют предметную область (рисунок 6.13) и для запросов, которые неявно определяют предметную область (рисунок 6.14).

Следующий набор запросов определялся количеством терминов. На ри-

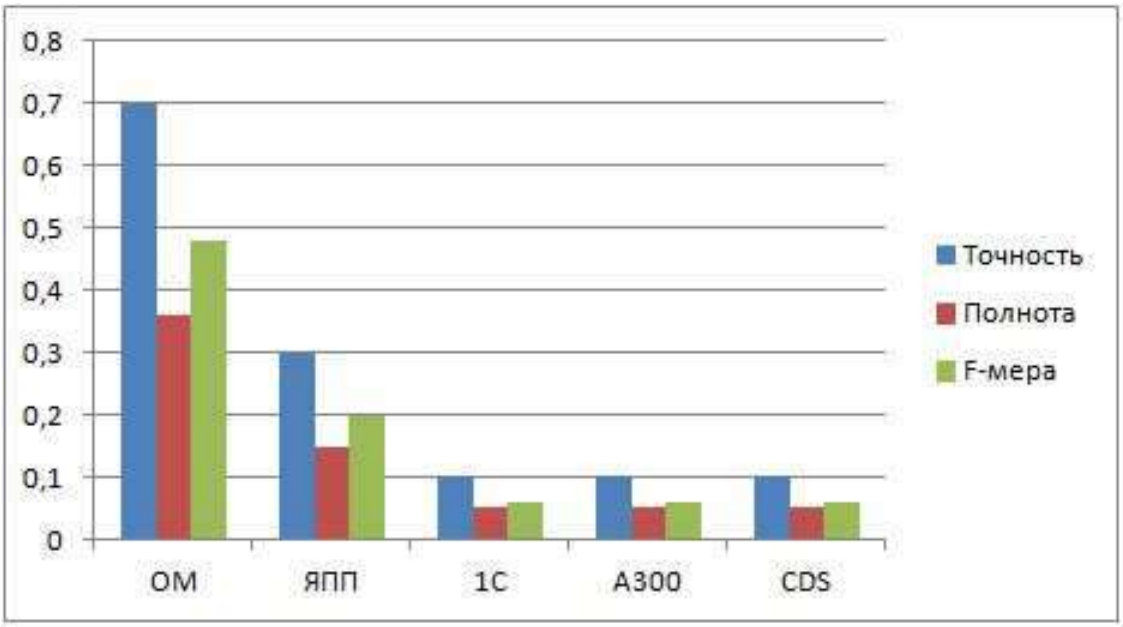


Рис. 6.12. Сравнение результатов экспериментов для профиля программиста с длинными запросами

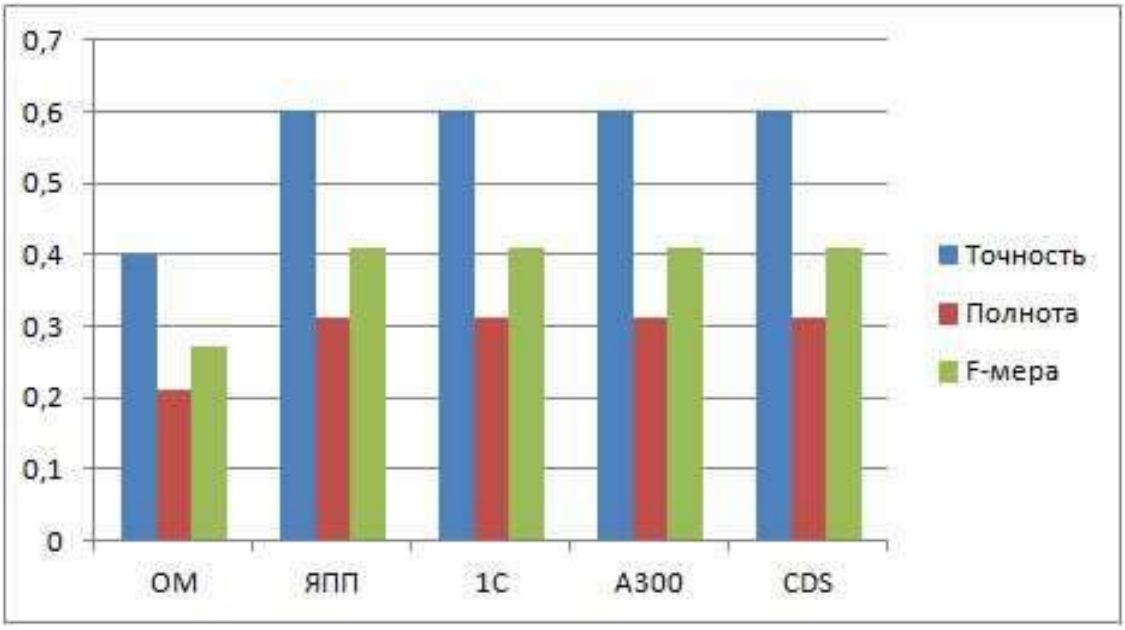


Рис. 6.13. Сравнение результатов экспериментов для профиля инженера с запросами, семантически явно определяющих предметную область

сунке 6.15 и рисунке 6.16 представлены гистограммы, на которых отражается влияние размера запроса на поиск ТД для профиля инженера.

В результате экспериментов с применением профиля «Проектировщик» получены следующие результаты для запросов, которые семантически явно определяют предметную область (рисунок 6.17) и для запросов, которые неявно

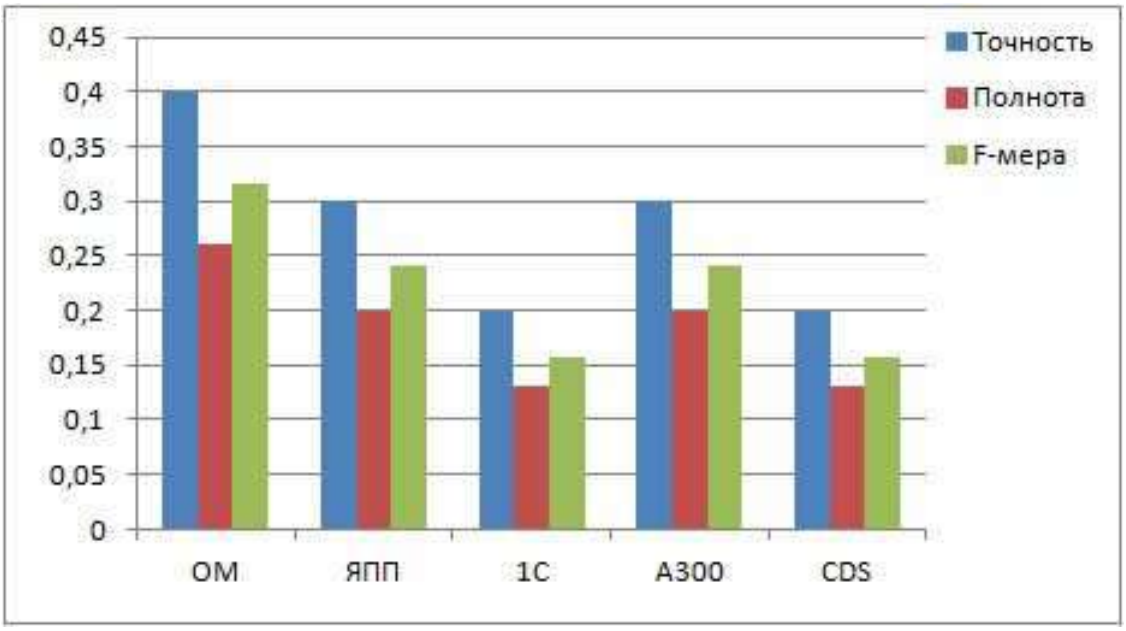


Рис. 6.14. Сравнение результатов экспериментов для профиля инженера с запросами, семантически неявно определяющих предметную область

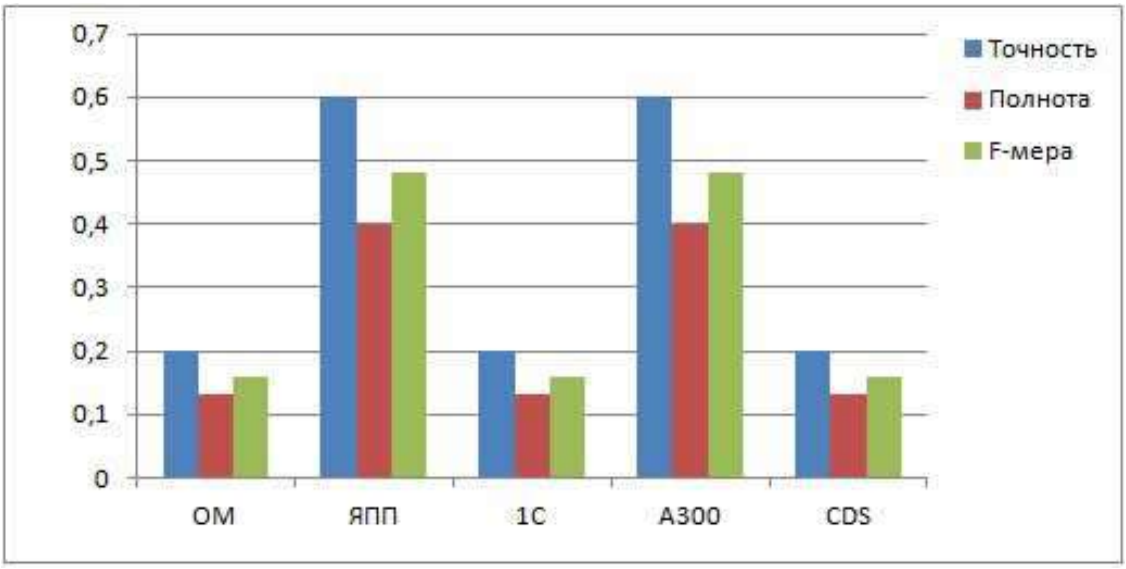


Рис. 6.15. Сравнение результатов экспериментов для профиля инженера с короткими запросами

идентифицируют предметную область (рисунок 6.18).

Следующий набор запросов определялся количеством терминов. На рисунке 6.19 и рисунке 6.20 представлены гистограммы, которые иллюстрируют влияние размера запроса на качество поиска ТД для профиля проектировщика.

На рисунке 6.21 представлен сравнительный анализ двух способов фор-

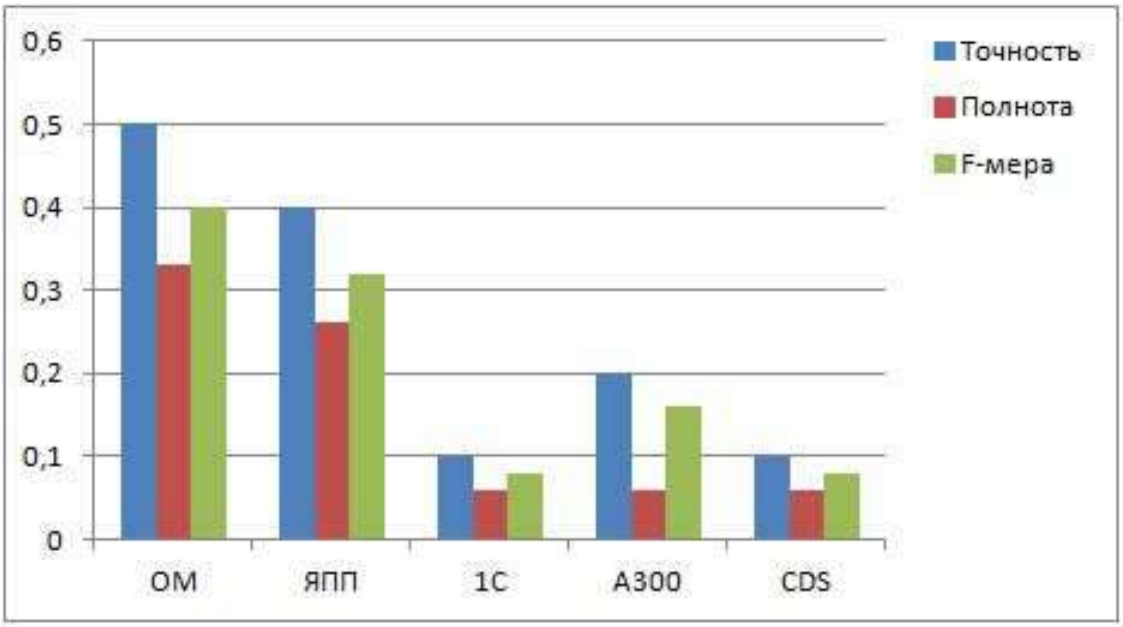


Рис. 6.16. Сравнение результатов экспериментов для профиля инженера с длинными запросами

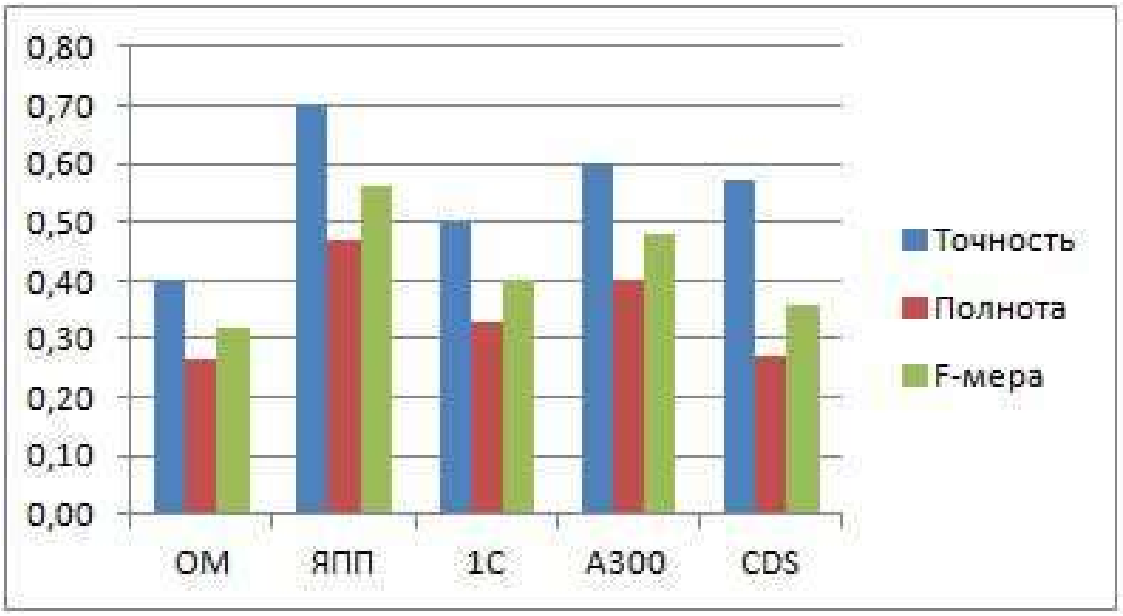


Рис. 6.17. Сравнение результатов экспериментов для профиля проектировщика с запросами, семантически явно определяющих предметную область

мирования контекстно-ориентированного поиска: с использованием профилей пользователей и с онтологическим поиском, в котором не используется информация о содержимом профилей пользователей.

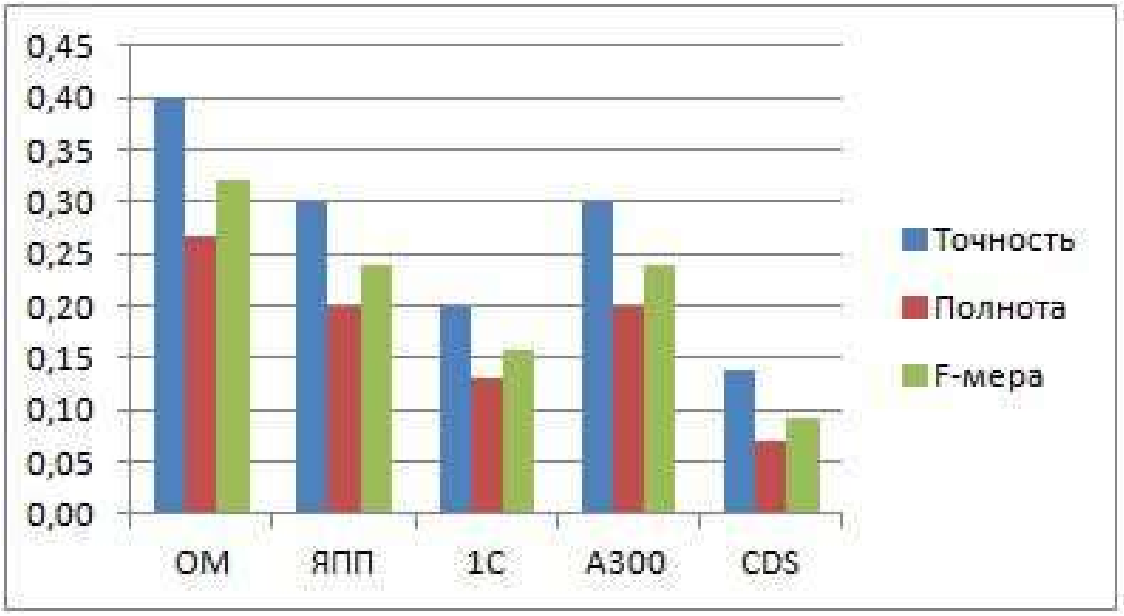


Рис. 6.18. Сравнение результатов экспериментов для профиля проектировщика с запросами, семантически неявно определяющих предметную область

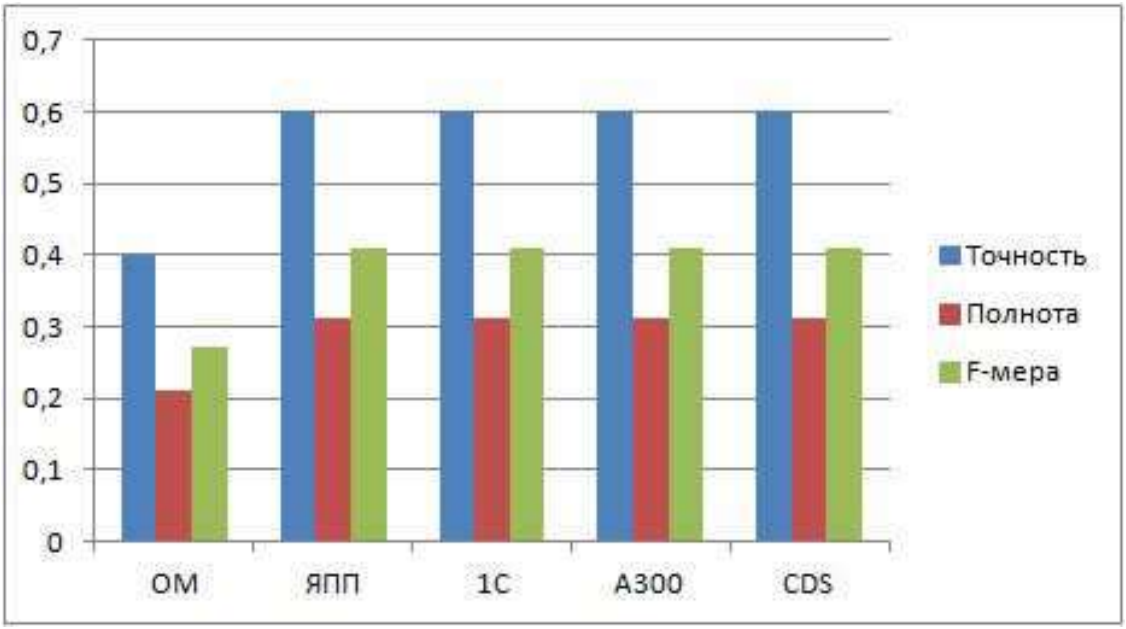


Рис. 6.19. Сравнение результатов экспериментов для профиля проектировщика с короткими запросами

В ходе проведенных экспериментов было определено, что онтологическая модель, использующая индивидуальные профили пользователей, показывает более качественный результат, чем поиск, который не учитывает информационную потребность конкретного специалиста. Применение профилей позволяет достигнуть показателей качества, представленные в таблице 6.4.

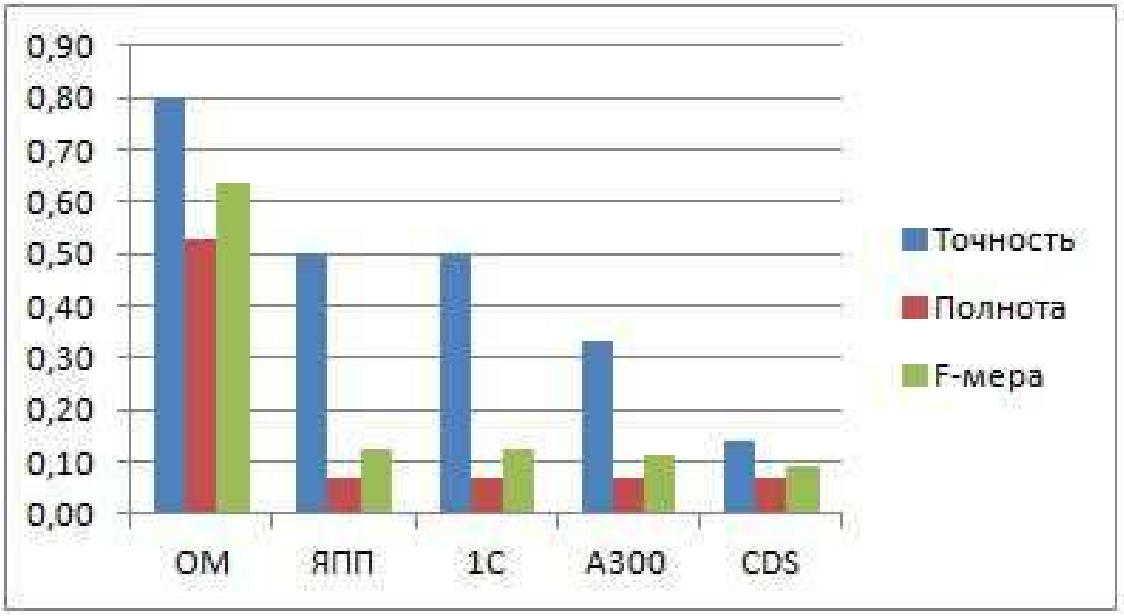


Рис. 6.20. Сравнение результатов экспериментов для профиля проектировщика с длинными запросами

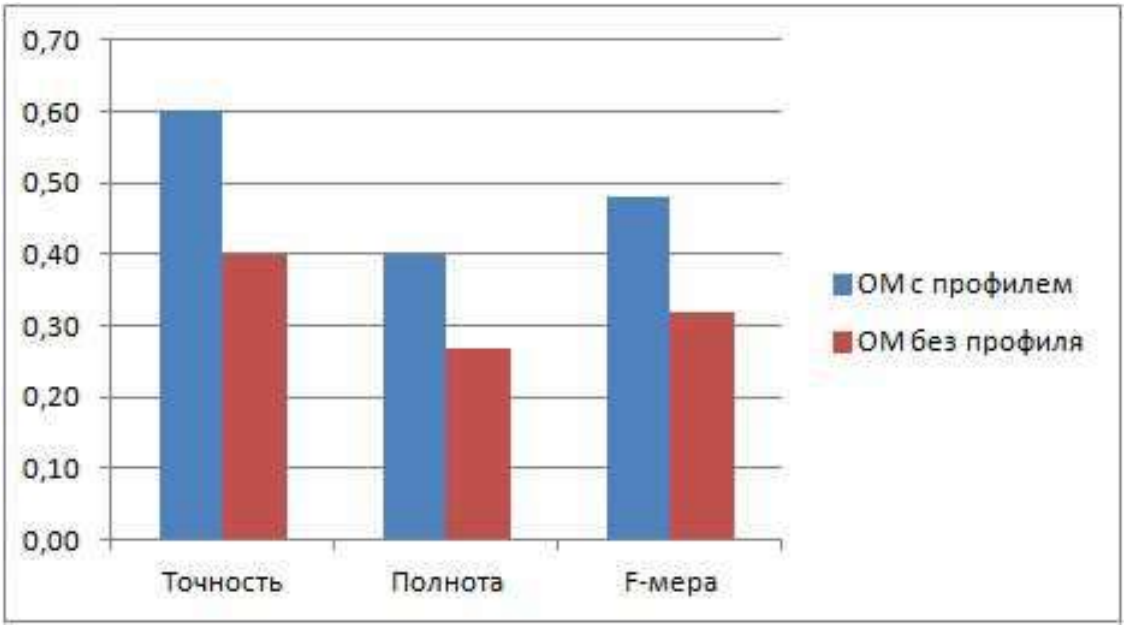


Рис. 6.21. Сравнение результатов экспериментов онтологических моделей поиска

Следующим этапом экспериментов стало сравнение качества результатов контекстно-ориентированного поиска ТД в электронном архиве, которые используют различные онтологии, отличающиеся способом формирования. Первая онтология содержит концептуальную сеть, концепты которой автоматизированным способом были извлечены из электронной библиотеки. Вторая он-

Таблица 6.4. Улучшение качества запросов при использовании профилей

Характеристики	Оценка улучшения поиска в интервальной форме
Точность	от 30% до 60%
Полнота	от 30% до 50%
F-мера	от 30% до 60%

тология содержит концептуальную сеть, составленную экспертом предметной области. На рисунке 6.22 представлен результат сравнения данного этапа вычислительных экспериментов.

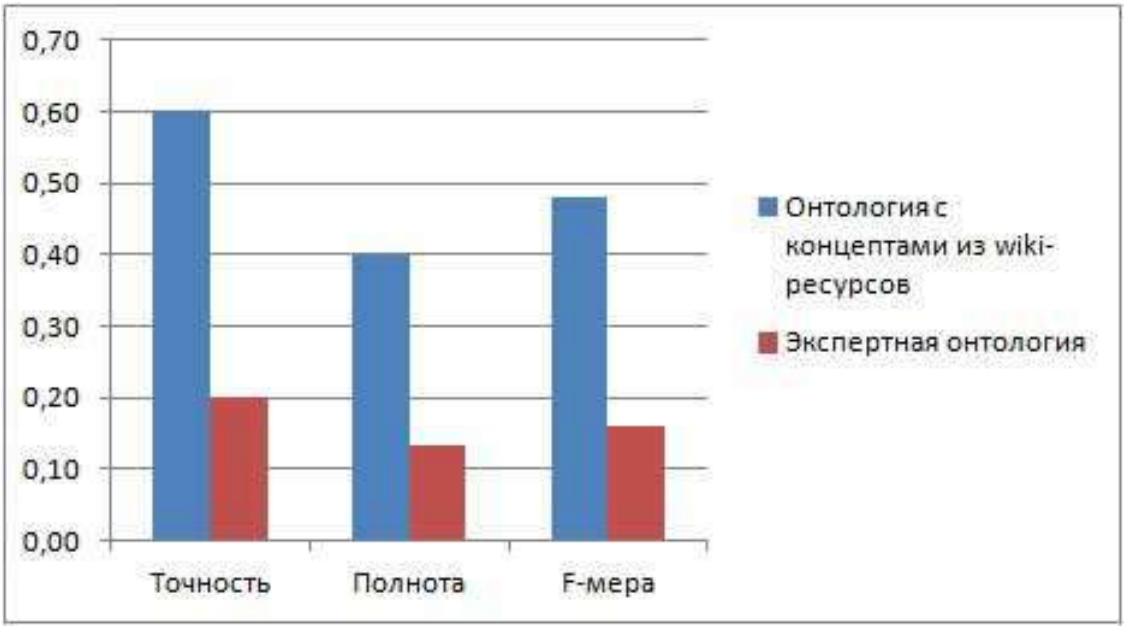


Рис. 6.22. Сравнение результатов экспериментов онтологических моделей поиска, использующие различные онтологии

Результаты вычислительных экспериментов показывают, что построение концептуальной сети автоматизированным способом и применение ее в процессах информационной поддержки позволяет улучшить качество поиска по сравнению с экспертной онтологией. В таблице 6.5 представлены сравнительные характеристики качества поиска.

Отдельный этап вычислительных экспериментов выполнялся с рабочим

Таблица 6.5. Улучшение качества запросов при использовании дополнительной терминологической сети

Характеристики	Оценка улучшения поиска в интервальной форме
Точность	от 30% до 70%
Полнота	от 40% до 70%
F-мера	от 35% до 70%

проектом, реализуемым коллективом работников ФНПЦ АО «НПО «Марс». Процесс проектирования сопровождается многочисленными проектными запросами к электронному архиву и применением опыта предыдущих разработок. Использование ИПР не нарушает общепринятых этапов проектирования, но

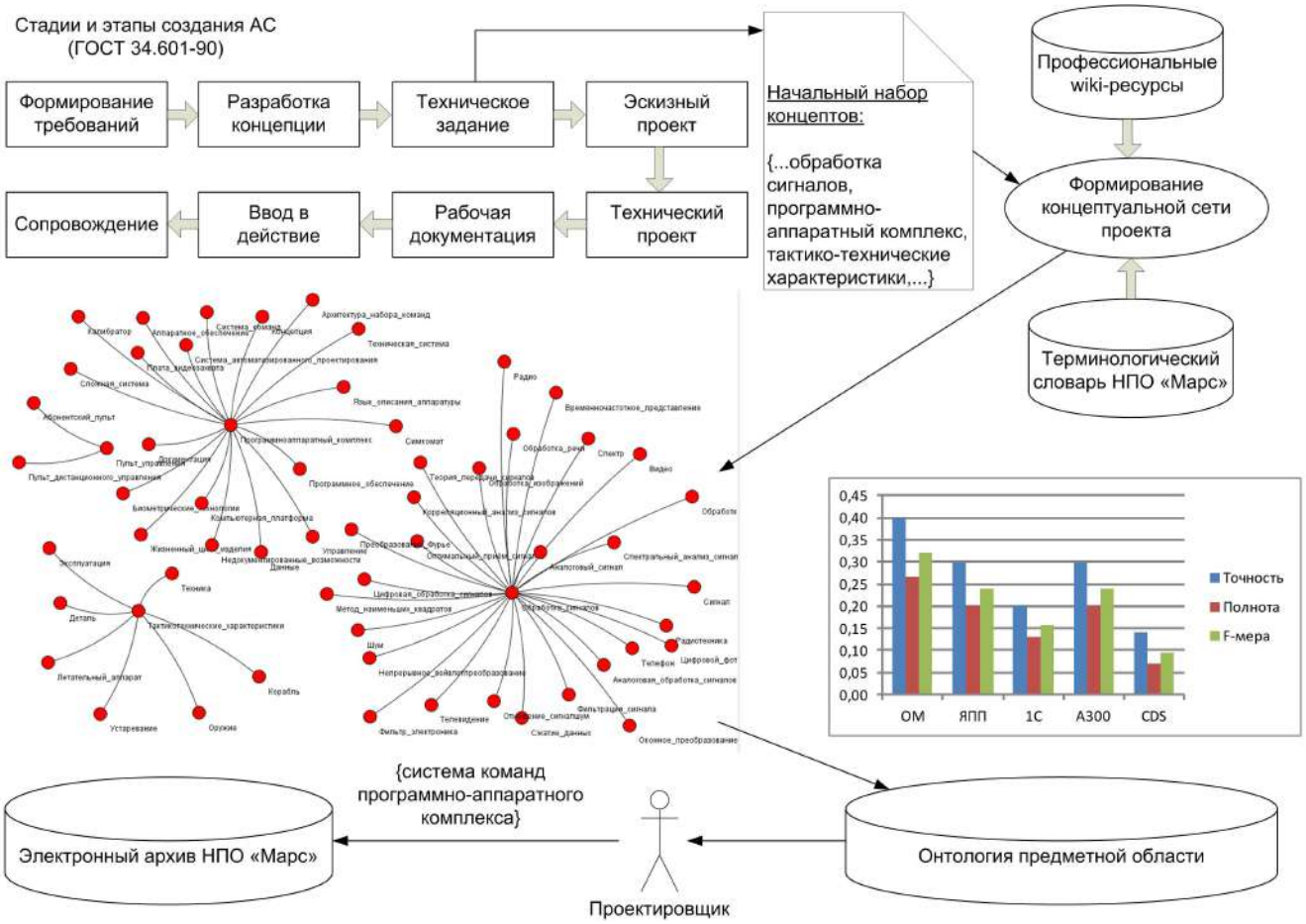


Рис. 6.23. Интеграция онтологической модели информационной поддержки в ИПР

способствует повышению скорости реализации проекта. Интеллектуальная компонента информационной поддержки включает в себя формирование онтологии, разработку индивидуальных профилей проектировщиков и интеграцию разработанных онтологических ресурсов в систему электронного архива проектной организации. Применение данной компоненты на первоначальном этапе минимально, особенно активно ее использование происходит на стадии анализа технического задания (рисунок 6.23).

На рисунке 6.23 представлены этапы проектирования с применением интеллектуальной компоненты информационной поддержки. Как видно из рисунка активная фаза использования возникает в процессе анализа технического задания и сопровождается извлечением понятий, которые используются как начальный набор концептов для построения концептуальной сети проекта. Для данного эксперимента из технического задания рабочего проекта были извлечены следующие понятия: «волоконно-оптическая связь», «средства преодоления противоракетной обороны», «программно-аппаратный комплекс», «радиоэлектронная борьба», «надводный корабль», «обработка сигналов», «тактико-технические характеристики», «пульт управления», «система автоматизированного проектирования», «электромагнитное излучение», «техника».

Помимо концептов технического проекта в процессе формирования концептуальной сети проекта используются понятия из терминологического словаря НПО «Марс». Таким образом, сформированная концептуальная сеть проекта включала в себя концепты технического задания, концепты терминологического словаря НПО «Марс» и концепты, извлеченные из wiki-ресурсов. Концептуальная сеть проектов, состоящая примерно из 3 00 понятий, была включена онтологию предметной области, фрагмент концептуальной сети представлена рисунке 6.24

Сформированная онтология предметной области применялась в задачах информационной поддержки по рабочему проекту, выполняемая коллективом. Для оценки качества информационной поддержки проводились аналогичные

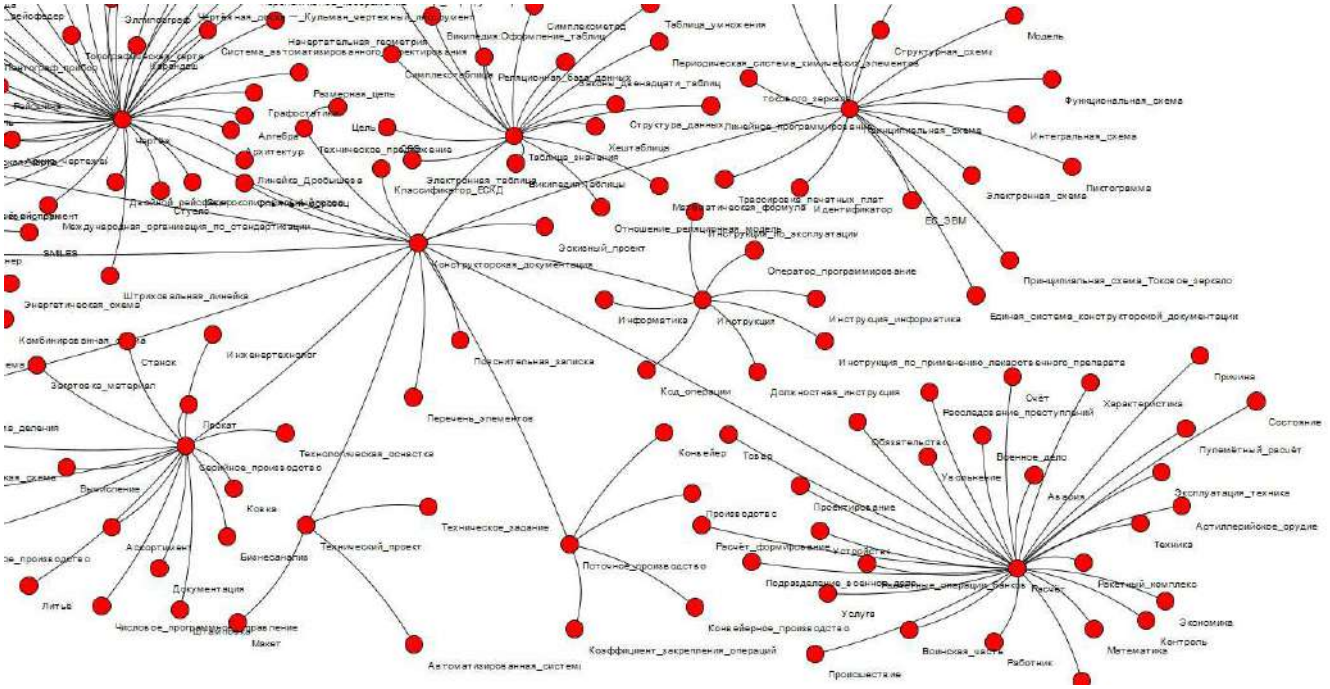


Рис. 6.24. Фрагмент концептуальной сети рабочего проекта НПО «Марс»

опыты, которые рассматривались выше, а именно сравнение онтологической модели поиска с существующими системами информационного поиска. На рисунке 6.25 и рисунке 6.26 представлены гистограммы, которые показывают, что с запросами, которые слабо выражают семантику предметной области и состоящие из большого количества терминов, онтологическая модель показала более качественные результаты.

В таблице 6.6 представлены сравнительные характеристики качества поиска.

Таблица 6.6. Улучшение качества запросов, которые слабо выражают семантику предметной области

Характеристики	Оценка улучшения поиска в интервальной форме
Точность	от 20% до 60%
Полнота	от 30% до 65%
F-мера	от 25% до 60%

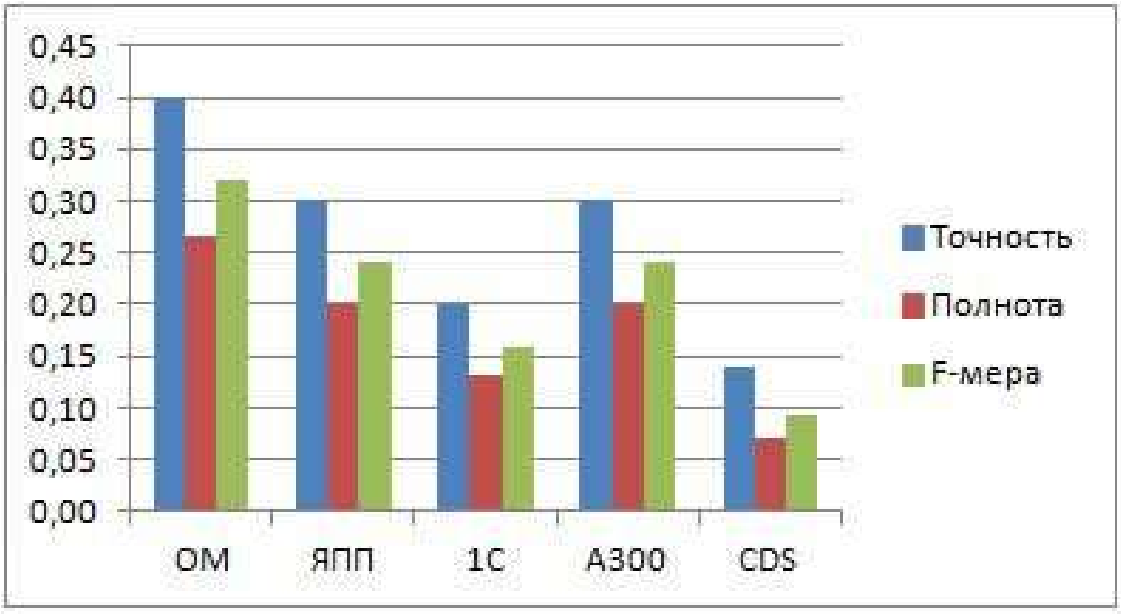


Рис. 6.25. Сравнение результатов экспериментов с запросами, слабо выражающими семантику предметной области

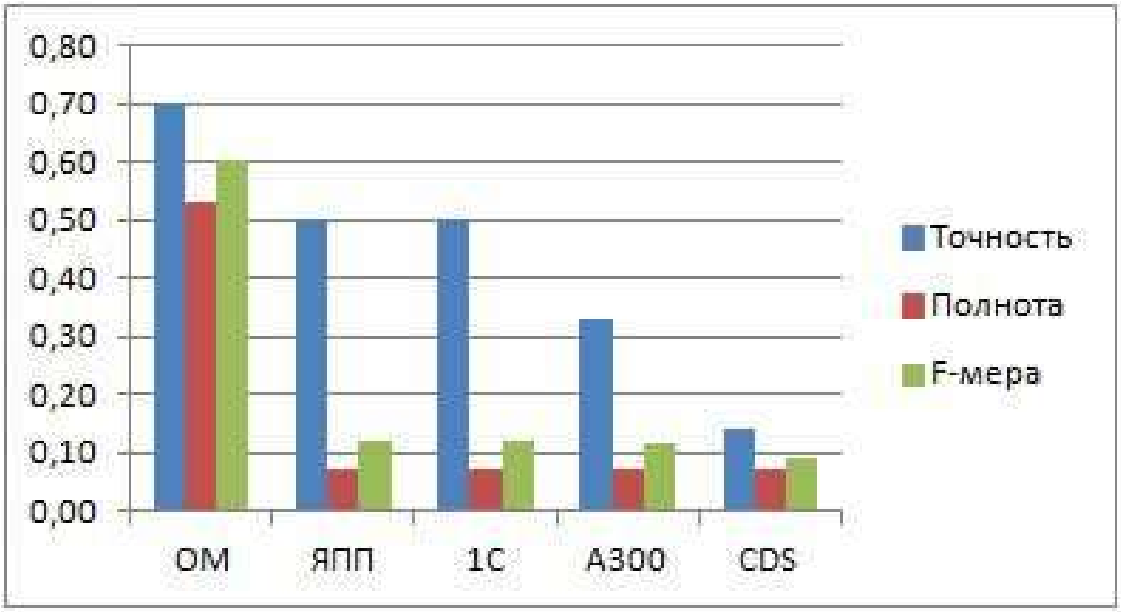


Рис. 6.26. Сравнение результатов экспериментов с длинными запросами

Полученные результаты эксперимента с рабочим проектом показали, что онтологическая модель поиска ТД, показывает более качественный результат, чем традиционные модели поиска.

Время поиска с применением онтологической модели сравнивалось с существующей подсистемой информационного поиска электронного архива НПО

«Марс». В данном эксперименте сравнивались временные показатели, затрачиваемые на поиск технических документов в рамках эксперимента с рабочим проектом. Сравнительные характеристики представлены в таблице 6.7.

Таблица 6.7. Сравнительные характеристики времени поиска документов в электронном архиве НПО «Марс»

Виды поиска	Существующая подсистема поиска в архиве	Онтологическая система поиска
Запросы, семантически явно определяющие предметную область	$<$ 1 мин.	$<$ 3 мин.
Запросы, слабо выражающие семантику предметной области	до 15 мин.	$<$ 5 мин.

На основании результатов выведем следующие заключения:

1. Онтологическая модель подсистемы информационной поддержки позволяет получать более качественные результаты, по сравнению с другими аналогичными системами, если пользователь использует запросы, которые семантически слабо выражают конкретную предметную область.
2. Онтологическая модель показывает худший результат в случае, если используются короткие запросы по причине того, что чем короче запрос, тем слабее идентифицируются концепты, в которых термины формируют текстовый вход.
3. Онтологическая модель показывает качественный результат, если пользователь использует запросы, содержащие более трех терминов. Чем больше терминов используется в запросе, тем лучше идентифицируются концепты из предметной онтологии.
4. Wiki-ресурсы описывают различные предметные области с разной степе-

нью детализации. Так, пользователи профиля «Программист», получали более качественные результаты поиска, чем пользователи профиля «Инженер» и «Проектировщик» по причине более детального описания соответствующей предметной области в электронной библиотеке.

5. Применение профилей пользователей электронного архива улучшает качество результатов проектных запросов к архиву ТД с применением онтологической модели по сравнению с поиском документов без учета информационных потребностей пользователей.
6. Вычислительные эксперименты показали, что концептуальная сеть проекта, сформированная автоматизированным способом посредством извлечения знаний из wiki-ресурсов, является предпочтительной в процессе информационной поддержки дополнительно к онтологии, разработанной экспертом предметной области.

6.4. Выводы по шестой главе

1. Онтологическая модель технического документа, учитывающая проектные этапы модели ЖЦ АС, является адекватной для решения задачи структуризации информационных ресурсов документальных электронных архивов. Сравнение с традиционной моделью технического документа показало, что кластерный анализ архива на основе предложенного онтологического подхода позволяет сократить время поиска документа похожей семантической группы (примерно на 13%), так как количество понятий онтологии ИПР значительно меньше количества терминов, извлеченных из ТД проектного архива. Одновременно онтологический подход к представлению ТД показывает лучшие результаты качества кластерного разбиения (примерно на 35%-40%) на крупных выборках документов и/или большом количестве кластеров.
2. Применение методов генетической оптимизации в процессе концептуаль-

ного индексирования ресурсов электронного архива позволяет решать задачу оптимизации состава понятий, включаемых в концептуальное представление структурных разделов ТД. Разработанные алгоритмы оптимизации являются сходимыми при условии использования предлагаемой целевой функции, структуры хромосом, операторов кроссинговера и мутации, а также рекомендуемых численных значений вероятностей кроссинговера и мутации.

3. Использование онтологической модели информационной поддержки в ИПР позволяет повысить качество выполняемых проектных запросов к электронному архиву ТД на основе отображения контекста проектной организации и реализуемого проекта на проектные запросы, учитывая накопленный опыт взаимодействия проектировщика с электронным архивом. Улучшение качества выполняемых проектных запрос в может достигать 35%-70% (F-мера, сочетающая характеристики точности и полноты).

Заключение

Основным итогом выполнения диссертационной работы является разработка новых научных основ процессов работы электронных архивов технической документации и принципиально новых средств взаимодействия «проектировщик – система» на основе онтологического подхода к анализу технической документации в проектировании автоматизированных систем с целью сокращения времени и повышения качества выполнения проектных запросов к электронным архивам технических документов.

К числу наиболее важных относятся следующие результаты.

1. Разработан онтологический подход, модели, методы и средства которого представляю собой теоретическую основу для анализа слабоструктурированных ресурсов проектной организации на начальных этапах проектирования сложных автоматизированных систем, позволяющие сократить время проектных процедур на 10-15% за счет использования нечетких логических формализмов при формировании проектных запросов к электронным архивам технических документов.
2. Предложена интегрированная модель системы онтологий интеллектуального проектного репозитория для решения задачи информационной поддержки автоматизированного проектирования, отличающаяся новой структурой и позволяющая выполнять информационное взаимодействие с проектными репозиториями на семантическом уровне.
3. Разработан метод концептуального индексирования слабоструктурированных информационных ресурсов, включающих в себя текстовые технические документы и проектные диаграммы, который позволил обосновать единый подход к интеллектуальному анализу проектной информации, на основе описания предметной области в виде онтологии.
4. На основе введенного понятия концептуального индекса разработаны новые методы интеллектуального анализа текстовых документов при авто-

матизированном проектировании, позволяющие улучшить качество формирования навигационной структуры технических документов проектного репозитория до 40% за счет использования формального онтологического описания контекста проектной организации при проектировании автоматизированных систем.

5. Разработан новый метод содержательной интерпретации кластеров технических документов и технических временных рядов на основе лингвистических шкал и приближенных множеств Павлака, позволяющий реализовывать объяснительную компоненту интеллектуального проектного репозитория на основе онтологии предметной области в терминах эксперта-проектировщика.
6. Разработаны и обоснованы нечеткая модель и методика оценки качества онтологии на основе свойств нечетких соответствий, позволяющие выполнять оперативный контроль процесса автоматизированного формирования онтологии и, как следствие, значительно сократить трудоемкость и время построения прикладной онтологии.
7. Разработаны методологические основы построения интеллектуальных онтологических систем информационной поддержки процесса проектирования автоматизированных систем, основанные на интеграции нечетко-логического, графо-аналитического и вероятностного подходов к анализу слабоструктурированной информации с целью интенсификации процессов интеллектуализации проектных репозиториев, что позволяет улучшить качество выполнения проектных запросов в среднем до 50%.
8. Показана эффективность применения онтологического подхода в решении ряда практических задач:
    - структурирование документальной информационной базы технических документов ФНПЦ АО «НПО «Марс» на основе созданной онтологии предметной области проектирования радиоэлектронных систем специального назначения.

- информационная поддержка проектировщика в процессе выполнения проектных запросов к электронному архиву ФНПЦ АО «НПО «Марс» во время реализации этапа опытно-конструкторских работ.

9. Разработанные алгоритмы и комплексы программ онтологического подхода к построению интеллектуальных проектных репозиториев являются новыми информационными ресурсами, обеспечивают взаимодействие проектировщика с электронным архивом технической документации на семантическом уровне, ориентированы на широкое использование конечными пользователями как в практических, так и в научных целях.

Необходимо отметить следующие ограничения диссертационного исследования. Ограничения методологических основ построения интеллектуальных онтологических систем информационной поддержки процесса проектирования автоматизированных систем связаны с их ориентацией на текстовые технические документы и проектные диаграммы формализованных нотаций (таких, как UML). Другим ограничением служит четкая ориентация процессов информационной поддержки на определенную предметную область, что связано с разработкой специализированных онтологий. Методы уточнения проектных запросов на основе онтологического подхода и с использованием концептуального индекса проектной организации показали свою неэффективность в случае, если такие запросы являются семантически определенными и четко выражают информационную потребность проектировщика.

Перспективы диссертационного исследования связаны с развитием методов онтологического подхода к анализу гетерогенных информационных ресурсов электронных архивов проектных организаций в направлении повышения точности и полноты выполнения проектных запросов, интеграции с нейросетевыми моделями и реализации интеллектуальных проектных репозиториев как аппаратно-программных комплексов.

Список сокращений и условных обозначений

АП	—	автоматизированное проектирование
АС	—	автоматизированная система
АРМ	—	автоматизированное рабочее место
БД	—	база данных
ВР	—	временной ряд
ЕИП	—	единое информационное пространство
ЕСКД	—	единая система конструкторской документации
ЕСПД	—	единая система программной документации
ЕСТД	—	единая система технологической документации
ЖЦ	—	жизненный цикл
ЗИП	—	задача информационного поиска
ЗНП	—	задача нечеткого поиска
ИБ	—	информационная база
ИО	—	информационное обеспечение
ИПР	—	интеллектуальный проектный репозиторий
ИПС	—	информационно-поисковые системы
ИР	—	информационный ресурс
ПО	—	предметная область
САПР	—	система автоматизированного проектирования
CALS	—	Continuous Acquisition and Life-cycle Support
DFD	—	Data Flow Diagram
HTML	—	HyperText Markup Language
IDEF	—	Icam DEFinition
OMG	—	Object Management Group
OWL	—	Web Ontology Language
RDF	—	Resource Description Framework
RDFS	—	Resource Description Framework Schema

SPARQL	—	SPARQL Protocol and RDF Query Language
UML	—	Unified Modeling Language
URI	—	Uniform Resource Identifier
W3C	—	World Wide Web Consortium
WWW	—	World Wide Web
XMI	—	XML Metadata Interchange
XML	—	eXtensible Markup Language

Список литературы

1. Батыршин, И.З. Нечеткие гибридные системы. Теория и практика / И.З. Батыршин, А.О. Недосекин, А.А. Стецко, В.Б. Тарасов, А.В. Язенин, Н.Г. Ярушкина; под ред Н.Г. Ярушкиной. – М.: ФИЗМАТЛИТ, 2007. – 208 с.

2. Андреев, А.М. Модели и методы автоматической классификации текстовых документов / А.М. Андреев, Д.В. Березкин, В.В. Сюзев, В.И. Шабанов // Вестн. МГТУ. Сер. Приборостроение. – М.: Изд-во МГТУ, 2003. – №3.

3. Баргесян, А.А. Анализ данных и процессов: учеб. пособие / А.А. Баргесян. – Санкт-Петербург : БХВ-Петербург, 2009.

4. Басалин, П.Д. Модель представления знаний интеллектуальной САПР цифровой аппаратуры / П.Д. Басалин // Труды Всероссийской конференции «Интеллектуальные информационные системы». – Ч. 1. Воронеж, 2001. – С. 121–122.

5. Берштейн, Л.С. Нечеткие графы и гиперграфы / Л.С. Берштейн, А.В. Боженюк. – М.: Научный мир, 2005. – 256 с.

6. Берштейн, Л.С. Нечеткие модели для экспертных систем САПР / Л.С. Берштейн, А.В. Боженюк, Н.Г. Малышев. – М.: Энергоатомиздат, 1991.

7. Боровикова, О.И. Подход к представлению знаний в многоязычных информационных системах / О.И. Боровикова, Ю.А. Загорулько // Одиннадцатая национальная конференция по искусственному интеллекту КИИ-2008 с международным участием: Труды конференции. – Т.3. – М.: ЛЕНАНД, 2008. – С. 154–163.

8. Валькман, Ю.Р. Анализ понятия образ: отношения «образы-понятия» / Ю.Р. Валькман // Одиннадцатая национальная конференция по искус-

ственному интеллекту КИИ-2008 с международным участием: Труды конференции. – Т. 1. - М.: ЛЕНАНД, 2008. – С. 369–377.

9. Войшвилло, Е.К. Понятие как форма мышления: логико-гносеологический анализ / Е.К. Войшвилло. – Москва: Изд-во МГУ, 1989.

10. Володина, М.Н. Информационная природа термина / М.Н. Володина // Филологические науки. – 1996. –№1.

11. Воробьев, А.М. Создание единого информационного пространства предприятия / А.М. Воробьев, Д.К. Щеглов //Материалы семинара «Развитие информационной инфраструктуры Концерна». – М.: ОАО «Концерн ПВО «Алмаз-Антей», 2007. – С. 93–104.

12. Вязгин, В.А. Математические методы автоматизированного проектирования / В.А. Вязгин, В.В. Федоров. – М.: Высш. школа, 1989.

13. Гаврилова, Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский. – СПб.: Питер, 2000.

14. Гаврилова, Т.А. Извлечение и структурирование знаний для экспертных систем / Т.А. Гаврилова, К.Р. Червинская. – М.: Радио и связь, 1992.

15. Гасанов, Э.Э. Информационно-графовая модель данных с нечеткой логикой /Э.Э. Гасанов, А.А. Фещук // Труды 4 Международной конференции по математическому моделированию, Москва (27 июня – 4 июля 2000 г.). – Том 2. – Москва: Станкин, 2001.

16. Гарольд, Э. XML. Справочник : пер. с англ / Э. Гарольд, С. Минс. – СПб. : Символ-Плюс, 2002. – 576 с.

17. Справочник информационного работника / науч. ред. Р.С. Гиляревский, В.А. Минкина. – СПб.: Профессия, 2005. – 96 с.

18. ГОСТ-Р ИСО 15926 Промышленные автоматизированные системы и интеграция. Интеграция данных жизненного цикла для перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Часть 1. Обзор и основополагающие принципы. – М.: Стандартинформ, 2010. – 14 с.

19. РД 50-34.698-90 Методические указания. Информационная технология. Комплекс стандартов и руководящих документов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов. – М.: Стандартинформ, 1990. – 28 с.

20. ГОСТ 22487-77 Проектирование автоматизированное. Термины и определения. – М.: Стандартинформ, 1977. – 35 с.

21. ГОСТ 27.002-2015 Надежность в технике (ССНТ). Термины и определения. – М.: Стандартинформ, 2015. – 22 с.

22. ГОСТ 34.003-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения. – М.: Стандартинформ, 2009. – 15 с.

23. ГОСТ 7.73-96 Система стандартов по информации, библиотечному и издательскому делу. Поиск и распространение информации. Термины и определения. – Минск: ИПК Издательство стандартов, 1997. – 16 с.

24. Гиляревский, Р.С. Информационная потребность / Р.С. Гиляревский, Ю.А. Гриханов // Библиотечная энциклопедия. – М.: Пашков дом, 2007. – С. 419–420.

25. Гранд, М. Шаблоны проектирования в Java / М. Гранд; пер. с англ. С. Беликовой. – М.: Новое знание, 2004. – 559 с.: ил.

26. Гринёв, С.В. Терминоведение: итоги и перспективы / С.В. Гринёв // Терминоведение; под ред. Татаринова В.А., Кульпиной В.Г. – М.: Московский лицей, 1993.

27. Губин, М.В. Модели и методы представления текстового документа в системах информационного поиска [Электронный ресурс] / М.В. Губин. – Режим доступа: http://maxgubin.com/articles/thesis.pdf

28. Добров, Б.В. Лингвистическая онтология по естественным наукам и технологиям: основные принципы разработки и текущее состояние / Б.В. Добров, Н.В. Лукашевич // Десятая национальная конференция по искусственному интеллекту с международным участием (Обнинск, 25–28 сентября 2006 г.). – М.: Физматлит, 2006.

29. Дюбуа, Д. Теория возможностей. Приложения к представлению знаний в информатике / Д. Дюбуа, А. Прад: пер. с фр. – М.: Радио и связь, 1990.

30. Ермаков, А.Е. Синтаксический разбор в системах статистического анализа текста / А.Е. Ермаков, В.В. Плешко // Информационные технологии. – 2002. –№7.

31. Загоруйко, Н.Г. Прикладные методы анализа данных и знаний / Н.Г. Загоруйко. – Новосибирск: Изд-во Ин-та математики, 1999. – 270 с.

32. Загоруйко, Н.Г. Формирование базы лексических функций и других отношений для онтологии предметной области / Н.Г. Загоруйко, А.М. Налетов, А.А. Соколова, В.А. Чурикова // Труды международной конференции Диалог-2004. – М.: Наука, 2004. – С. 202–204.

33. Загорулько, Ю.А. Автоматизация сбора онтологической информации об интернет-ресурсах для портала научных знаний / Ю.А. Загорулько // Известия Томского политехнического университета. – 2008. –№5.

34. Загорулько, Ю.А. Семантический подход к анализу документов на основе онтологии предметной области [Электронный ресурс] / Ю.А. Загорулько, И.С. Кононенко, Е.А. Сидорова. – Режим доступа: http://www.dialog-21.ru/digests/dialog2006/materials/html/SidorovaE.htm

35. Заде, Л.А. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л.А. Заде. – М.: Мир, 1976. – 165 с.

36. Калиниченко, Л.А. Эффективная поддержка баз данных с онтологическими зависимостями: Реляционные языки вместо дескриптивных логик / Л.А. Калиниченко // Программирование. – 2012. – № 6.

37. Карпенко, А.П. Меры важности концептов в семантической сети онтологической базы знаний / А.П. Карпенко // Наука и образование: электронное научно-техническое издание. – М. : Московский государственный технический университет им. Н.Э. Баумана. – 2010. – №7.

38. Киселев, М. Метод кластеризации текстов, основанный на попарной близости термов, характеризующих тексты, и его сравнение с метрическими методами кластеризации / М. Киселев // Интернет-математика 2007 : сб. работ участников конкурса науч. проектов по информ. поиску; отв. ред. П. И. Браславский. – Екатеринбург : Изд-во Урал. ун-та, 2007. – С. 74–83.

39. Коваленко, В.В. Проектирование информационных систем : учебное пособие / В.В. Коваленко – М.: ФОРУМ, 2012. – 320 с.

40. Кофман, А. Введение в теорию нечетких множеств / А. Кофман– М.: Радио и связь, 1982.

41. Кренке, Д. Теория и практика построения баз данных / Д. Кренке. – 9-е изд. переработ. и доп. – СПб.: Питер, 2005. – 859 с.

42. Курейчик, В.В. Перспективные архитектуры генетического поиска / В.В. Курейчик // Программные продукты и системы. – 1998. –№ 3. – С. 47-48.

43. Леонтьева, Н.Н. К теории автоматического понимания естественных текстов: Семантические словари: состав, структура, методика создания / Н.Н. Леонтьева. – М.: Изд-во МГУ, 2001.

44. Малышев, Н.Г. Нечеткие модели для экспертных систем в САПР / Н.Г. Малышев, Л.С. Берштейн, А.В. Боженюк. – М.: Энергоатомиздат, 1991. – 136 с.

45. Маннинг, К. Введение в информационный поиск / К. Маннинг, П. Рагхаван, Х. Шютце: пер. с англ. – М. : ООО «И.Д. Вильямс», 2011.

46. Матвеев, Ю.Н. Основы теории систем и системного анализа: учебное пособие. Ч. 1 / Ю.Н. Матвеев. – Тверь: ТГТУ, 2007.

47. Мордвинов, В. А. Онтология моделирования и проектирования семантических информационных систем и порталов: справочное пособие / В.А. Мордвинов; на правах рукописи. – Москва, 2005.

48. Наместников, А.М. Построение проектного интеллектуального репозитария / А.М. Наместников, А.В. Чекина, Н.В. Корунова // Информатика и экономика: сборник научных трудов; отв. ред. Н.Г. Ярушкина. – Ульяновск: УлГТУ. – 2007. – С. 119–125.

49. Наместников, А.М. Интеллектуальный сетевой архив электронных информационных ресурсов / А.М. Наместников, А.В. Чекина, Н.В. Корунова // Программные продукты и системы. – 2007. – № 4. – С. 10–13.

50. Наместников, А.М. Организация интеллектуального хранилища на основе нечеткой кластеризации / А.М. Наместников, Н.Г. Ярушкина, А.Г. Селяев, Е.В. Суркова, А.А. Островский, Н.В. Корунова // XI научно-практиче-

ская конференция «Реинжиниринг бизнес-процессов на основе современных технологий. Системы управления знаниями» (РБП-СУЗ-2008): сборник научных трудов. – М. – 2008. – С. 332–335.

51. Наместников, А.М. Анализ возможности применения технологии Семантический WEB в интеллектуальных хранилищах данных / А.М. Наместников // AIS'08, CAD-2008. «Интеллектуальные системы»: сборник научных трудов. – Т.2. Интеллектуальные САПР. – М. : Физматлит. – 2008. – с. 190–195.

52. Наместников, А.М. Перспективы применения технологии Семантический WEB в интеллектуальных хранилищах данных / А.М. Наместников // Известия Самарского научного центра Российской академии наук. Специальный выпуск: Четверть века изысканий и экспериментов по созданию уникальных технологий и материалов для авиаракетостроения УНТЦ-ФГУП ВИАМ. – Т.1. Самара: Издательство Самарского научного центра РАН. – 2008. – С. 235–239.

53. Наместников, А.М. Интеллектуальный проектный репозитарий / А.М. Наместников, Н.Г. Ярушкина, Н.В. Корунова, А.А. Островский, Ю.А. Радионова, А.Г. Селяев, А.В. Чекина // Одиннадцатая национальная конференция по искусственному интеллекту КИИ-2008 с международным участием: труды конференции. – Т.3. – М.: ЛЕНАНД. – 2008. – С. 345–352.

54. Наместников, А.М. Возможности мониторинга динамики развития проекта в интеллектуальном проектном репозитарии / А.М. Наместников, А.В. Чекина // Одиннадцатая национальная конференция по искусственному интеллекту КИИ-2008 с международным участием: труды конференции. – Т.3. – М.: ЛЕНАНД. – 2008. – С. 99–106.

55. Наместников, А.М. Интеллектуальные проектные репозитории: монография / А.М. Наместников. – Ульяновск: УлГТУ, 2009. – С. 110.

56. Наместников, А.М. Концептуальная индексация проектных документов / А.М. Наместников, А.А. Филиппов // Автоматизация процессов управления. – 2010. – №2(20). – С. 34-39.

57. Наместников, А.М. Хранилище проектных документов / А.М. Наместников, А.А. Филиппов // Тезисы докладов 43-й научно-технической конференции УлГТУ «Вузовская наука в современных условиях» (26-31 января 2009 года). – Ульяновск : УлГТУ. – 2009. – С. 114-115.

58. Наместников, А.М. XML репозиторий проектных документов / А.М. Наместников, А.А. Филиппов // Всероссийская конференция с элементами научной школы для молодежи «Проведение научных исследований в области обработки, хранения, передачи и защиты информации», 1-5 декабря 2009 г. Россия: сборник научных трудов. В 4 т. – Ульяновск : УлГТУ, 2009. –Т. 4. – С. 254-256.

59. Наместников, А.М. Концептуальная индексация проектных документов / А.М. Наместников, А.А. Филиппов // Интеллектуальный анализ временных рядов: сборник научных трудов семинара с международным участием «Интеллектуальный анализ временных рядов» по результатам НИР, поддержанной ФЦП, проект № 02.740.11.5021, г. Ульяновск, 15 июня 2010 г. – Ульяновск : УлГТУ. – 2010. – С. 69-77.

60. Наместников, А.М. Нечеткая кластеризация концептуальных индексов проектных документов / А.М. Наместников, А.А. Филиппов // Интегрированные модели и мягкие вычисления в искусственном интеллекте: сборник научных трудов 6-й Международной научно-технической конференции (Коломна, 16-19 мая 2011 г.). В 2-х томах. – М.: Физматлит. – 2011. – Т.2. – С. 958-968.

61. Наместников, А.М. Реализация системы кластеризации концептуальных

индексов проектных документов / А.М. Наместников, А.А. Филиппов // Автоматизация процессов управления. – 2011. – №3(25). – С. 46-50.

62. Наместников, А.М. Разработка инструментария для интеллектуального анализа технической документации / А.М. Наместников, Р.А. Субхангулов, А.А. Филиппов // Известия Самарского научного центра Российской академии наук. – 2011. – № 4. – Т.13. – С. 984-990.

63. Наместников, А.М. Метод онтологической кластеризации документов в интеллектуальном проектном репозитории / А.М. Наместников, А.А. Филиппов // Гибридные и синергетические интеллектуальные системы: теория и практика : материалы 1-го международного симпозиума; под ред. проф. А.В. Колесникова. – Калининград : Изд-во БФУ им. И. Канта. – 2012. – С. 205–213.

64. Наместников, А.М. Система кластеризации и полнотекстового поиска проектных документов на основе прикладной онтологии / А.М. Наместников, Р.А. Субхангулов, А.А. Филиппов // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012 (16-20 октября 2012 г., г. Белгород, Россия): труды конференции. – Белгород: Изд-во БГТУ. – 2012. – Т.2. – С. 104-111.

65. Наместников, А.М. Метод генетической оптимизации онтологических представлений проектных документов в задаче индексирования / А.М. Наместников, А.А. Филиппов // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012 (16-20 октября 2012 г., г. Белгород, Россия): труды конференции. – Белгород: Изд-во БГТУ. – 2012. – Т.4. – С. 84-91.

66. Наместников, А.М. Онтологически-ориентированная система кластеризации и полнотекстового поиска проектных документов / А.М. Наместников,

Р.А. Субхангулов, А.А. Филиппов // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2013): материалы III Междунар. научн.техн. конф. (Минск, 21-23 февраля 2013г.) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск: БГУИР. – 2013. – С. 219-224.

67. Наместников, А.М. Применение нечетких моделей в задачах кластеризации и информационного поиска текстовых проектных документов / А.М. Наместников, Р.А. Субхангулов, А.А. Филиппов // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов VII-й Международной научно-практической конференции (Коломна, 20-22 мая 2013 г.). В 3-х томах. – М.: Физматлит. – 2013. – Т3. – С. 1278-1289.

68. Наместников, А.М. Концептуальное индексирование и кластеризация архива проектной документации на основе онтологии / А.М. Наместников // Наукоемкие технологии. – М.: Радиотехника. – 2013. – №5. – Т.14. – С. 73-78.

69. Наместников, А.М. Применение тезаурусов и онтологий в интеллектуальных архивах проектной документации / А.М. Наместников, Н.Г. Ярушкина // Наукоемкие технологии. – М.: Радиотехника. – 2013. – №5. – Т.14. – С. 79-86.

70. Наместников, А.М. Онтологически-ориентированная модель классификаций текстовых документов / А.М. Наместников, Р.А. Субхангулов // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2014): материалы IV Междунар. научн.техн. конф. (г. Минск, 20-22 февраля 2014 г.) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск: БГУИР. – 2014. – С. 385-389.

71. Наместников, А.М. Интеграция реляционных данных на основе онтологического подхода / А.М. Наместников, А.О. Колесов // Четырнадцатая

национальная конференция по искусственному интеллекту с международным участием КИИ-2014 (24-27 сентября 2014 г., г. Казань, Россия): труды конференции. – Казань. – 2014. – Т.3. – С. 146-154.

72. Наместников, А.М. Формирование информационных запросов к электронному архиву на основе концептуального индекса / А.М. Наместников, Р.А. Субхангулов // Радиотехника. – М.: Радиотехника. – №7. – 2014. – С. 126-129.

73. Наместников, А.М. Формирование навигационной структуры электронного архива технической документации на основе онтологии / А.М. Наместников, А.А. Филиппов // Радиотехника. – М.: Радиотехника. – 2014. – №11. – С. 108-117.

74. Наместников, А.М. Онтологический подход к формированию проектных запросов интеллектуального агента / А.М. Наместников, Р.А. Субхангулов // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2015): материалы V Междунар. научн.техн. конф. (г. Минск, 19-21 февраля 2015 г.) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск: БГУИР. – 2015. – С. 407-412.

75. Наместников, А.М. Онтологическая модель контекстного поиска электронных документов в архиве проектной организации / А.М. Наместников, Р.А. Субхангулов // Радиотехника. – М.: Радиотехника. – 2015. – №6. – С. 73-78.

76. Наместников, А.М. Метауровень информационного обеспечения САПР: от теории к практике: монография / А. М. Наместников.– Ульяновск: УлГТУ, 2015. – 176 с.

77. Наместников, А.М. Интеграция нечетко-гранулярных и онтологических методов в задаче анализа временных рядов / А.М. Наместников, Н.Г.

Ярушкина, Т.В. Афанасьева, Г.Ю. Гуськов // Автоматизация процессов управления. – 2015. – №2(40). – С. 72-79.

78. Наместников, А.М. Онтологический подход к формированию контекстных запросов в электронном архиве технических документов / А.М. Наместников // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2016): материалы VI Междунар. научн.техн. конф. (г. Минск, 18-20 февраля 2016 г.) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск: БГУИР. – 2016. – С. 415-420.

79. Namestnikov, A. An Ontology-Based Model of Technical Documentation Fuzzy Structuring / A. Namestnikov, A. Filippov, V. Avvakumova // Proceedings of the 2nd International Workshop on Soft Computing Applications and Knowledge Discovery (SCAKD 2016) co-located with the 13th International Conference on Concept Lattices and Their Applications (CLA 2016), Moscow, Russia, July 18, 2016, pp. 63-74.

80. Наместников, А.М. Разработка многоагентной системы извлечения знаний из гетерогенных источников / А.М. Наместников, Г.Ю. Гуськов, В.С. Мошкин, А.А. Филиппов, Н.Г. Ярушкина // Радиотехника. – М.: Радиотехника. – 2016. – №9. – С. 57-63.

81. Наместников, А.М. Онтологический подход к структурированию знаний проектной организации / А.М. Наместников // Радиотехника. – М.: Радиотехника. – 2016. – №9. – С. 77-83.

82. Наместников, А.М. Способ уточнения контекстных запросов к архиву технических документов на основе онтологии / А.М. Наместников // Пятнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2016 (3-7 октября 2016 г., г. Смоленск, Россия): труды конференции. – 2016. – Т.2. – С. 98-105.

83. Наместников, А.М. Программная система преобразования UML-диаграмм в онтологии на языке OWL / А.М. Наместников, Г.Ю. Гуськов // Пятнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2016 (3-7 октября 2016 г., г. Смоленск, Россия): труды конференции. – 2016. – Т.3. – С. 270-278.

84. Наместников, А.М. Система управления программными проектами на основе онтологического подхода / А.М. Наместников, Г.Ю. Гуськов // Автоматизация процессов управления. – 2016. – №3(45). – С. 88-94.

85. Нариньяни, А.С. Кентавр по имени ТЕОН: Тезаурус+Онтология / А.С. Нариньяни // Труды Международной конференции ДИАЛОГ-2001. – М. – 2001. – Т.1. – С.184-188.

86. Нгуен, Б.Н. Модель информационного поиска на основе семантических метаописаний / Б.Н. Нгуен, А.Ф. Тузовский // Управление большими системами. – М.: ИПУ РАН. – 2013. – С.51–92.

87. Норенков, И.П. Основы автоматизированного проектирования: учеб. для вузов / И.П. Норенков. – 4-е изд., перераб. и доп. – М.: Изд-во МГТУ им. Н. Э. Баумана, 2009.

88. Острейковский, В.А. Теория систем / В.А. Острейковский. – М.: Высш. шк., 1997.

89. Поспелов, Д.А. Логико-лингвистические модели в системах управления / Д.А. Поспелов. – М.: Энергоатомиздат, 1981.

90. Поспелов, Д.А. Моделирование рассуждений / Д.А. Поспелов. – М.: Радио и связь, 1989.

91. Рассел, С. Искусственный интеллект. Современный подход / С. Рассел, П. Норвиг. – М.: Вильямс, 2006. – 1408 с.

92. РД 50-680-88 Методические указания. Автоматизированные системы. Основные положения - Руководящий документ по стандартизации. – М. – 1989.

93. Рубашкин, В.Ш. Представление и анализ смысла в интеллектуальных информационных системах / В.Ш. Рубашкин. – М.: Наука, 1989.

94. Самбук, А. Управление документацией в проектах разработки ПО / А. Самбук // Открытые системы. – 2006. – №7. – С. 54-58.

95. Семенов, С.В. Анализ системных основ электронных документов / С.В. Семенов // Программные продукты и системы. – 2007. –№2. – С. 60-61.

96. Сидорова, Е.А. Подход к разработке лингвистических онтологий / Е.А. Сидорова // Одиннадцатая национальная конференция по искусственному интеллекту КИИ-2008 с международным участием: труды конференции. – М.: ЛЕНАНД. – 2008. – Т.3. – С. 181-189.

97. Силин, В. Tamino. Информационный сервер для электронного бизнеса [Электронный ресурс] / В. Силин. – Режим доступа: http://citforum.ru/internet/articles/xml_tamino.shtml.

98. Скурихин, А.Н. Генетические алгоритмы / А.Н. Скурихин // Новости искусственного интеллекта. – 1995. – №4. – С. 6-17.

99. Соколов, А.В. Философия информации: профессионально-мировоззренческое пособие / А.В. Соколов. – СПб.: СПбГУКИ. – 2010. – С. 246-277.

100. Соловьев, В.Д. Онтологии и тезаурусы: учебное пособие / В.Д. Соловьев, Б.В. Добров, В.В. Иванов, Н.В. Лукашевич. – Казань, Москва. – 2006.

101. Солтон, Дж. Динамические библиотечно-информационные системы / Дж. Солтон. – М.: Мир, 1978.

102. Соснин, П.И. Логика понятий / П.И. Соснин. – Саратов: Изд-во Саратовского уни-верситета, 1986.

103. Соснин, П.И. Создание и использование автоматизированной базы опыта проектной организации / П.И. Соснин, В.А. Маклаев. – Ульяновск: УлГТУ, 2012. – 362 с.

104. Суперанская, А.В. Общая терминология: Вопросы теории / А.В. Суперанская, Н.В. Подольская, Н.В. Василева. – М.: Наука, 1989.

105. Титов, Ю.А. САПР технологических процессов / Ю.А. Титов. – Ульяновск, 2009.

106. Тэрано, Т. Прикладные нечеткие системы / Т. Тэрано, К. Асаи, М. Сугэно. – М.: Мир, 1993. – 368 с.

107. Уэно, Х. Представление и использование знаний / Х. Уэно, Т. Кояма, Т. Окамото и др.: пер. с япон. – М.: Мир, 1989.

108. Филиппов, А.А. Концептуальный индексатор проектных документов / А.А. Филиппов // Тезисы докладов 45-й научно-технической конференции УлГТУ «Вузовская наука в современных условиях» (24-29 января 2011 года). – Ульяновск: УлГТУ, 2011. – С. 181.

109. Филиппов, А.А. Онтологически-ориентированное индексирование проектных документов. XML-сервер Tamino как ядро интеллектуального проектного репозитория / А.А. Филиппов // Вузовская наука в современных условиях : сборник материалов 46-й научно-технической конференции (23-28 января 2013 года). В 3 ч. – Ульяновск: УлГТУ. – 2012. – Ч.2. – С. 154-157.

110. Филиппов, А.А. Индексирование и кластеризация проектных документов на основе графовой модели онтологии / А.А. Филиппов // Информатика,

моделирование, автоматизация проектирования: сборник научных трудов; под. ред. Н. Н. Войта. – Ульяновск: УлГТУ. – 2011. – С. 367-372.

111. Филиппов, А.А. Реализация онтологически-ориентированных подсистем индексирования и кластеризации проектных документов / А.А. Филиппов // Информатика, моделирование, автоматизация проектирования: сборник научных трудов; под ред. Н. Н. Войта. – Ульяновск: УлГТУ. – 2012. – С. 389-397.

112. Холзнер, С. XML. Энциклопедия / С. Холзнер. – 2-е изд. – СПб.: Питер, 2004. – 1001 с.

113. Шапиро, Д.И. Принятие решений в системах организационного управления: использование расплывчатых категорий / Д.И. Шапиро. – М.: Энергоатомиздат, 1983.

114. Шильников, П.С. Компьютерная поддержка построения онтологий / П.С. Шильников // Программные продукты и системы. – 2006. – №2. – С. 50–52.

115. Ярушкина, Н.Г. Основы теории нечетких и гибридных систем: учеб. пособие / Н.Г. Ярушкина. – М.: Финансы и статистика, 2004. – 320 с.

116. Ярушкина, Н.Г. Интеллектуальный анализ временных рядов: учебное пособие / Н.Г. Ярушкина, Т.В. Афанасьева, И.Г. Перфильева. – М.: ИД «ФОРУМ»: ИНФРА-М, 2012. – 160 с.

117. Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 2003.

118. Baeza-Yates R., Ribeiro-Neto B. Modern Information Rertieval. ACM Press, New York, 1999.

119. D. Bahle, H. E. Williams, and J. Zobel. E?cient phrase querying with an auxiliary index. In K. Jarvelin, M. Beaulieu, R. Baeza-Yates, and S. H. Myaeng, editors, Proceedings of the ACM-SIGIR Conference on Research and Development in Information Retrieval, P. 215-221, Tampere, Finland, August 2002.

120. E. J. Barkmeyer, A. B. Feeney, P. Denno, D. W. Flater, D. E. Libes, M. P. Steves, and E. K. Wallace, "Concepts for automating systems integration," National Institute of Standards and Technology (NIST), Gaithersburg, MD, Tech. Rep. NISTIR 6928, February 2003.

121. David Beckett. The Design and Implementation of the Redland RDF Application Framework. In Proceedings of Semantic Web Workshop of the 10th International World Wide Web Conference, Hong-Kong, China, May 2001.

122. Berge C. Hypergraphs: combinatorics of finite sets. – Elsevier Science Publishers B.V., 1989.

123. Berry, M.W. Survey of Text Mining, Springer – 2003.

124. Bertails A., Arenas M., Prud'hommeaux E., Sequeda J., Editors. A Direct Mapping of Relational Data to RDF – http://www.w3.org/TR/rdb-direct-mapping/

125. S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein, OWL Web Ontology Language Reference, http://www.w3.org/TR/owl-ref, 10 February 2004, w3C Recommendation.

126. P. Bonatti and A. Tettamanzi. Some complexity results on fuzzy description logics. In A. Petrosino V. Di Ges'u, F. Masulli, editor, WILF 2003 International Workshop on Fuzzy Logic and Applications, LNCS 2955, Berlin, 2004. Springer Verlag.

127. Booch, G., Rumbaugh, J. and Jacobson, I. (1997). The Unified Modeling Language user guide: Addison-Wesley.

128. D. Brickley and R.V. Guha. Resource Description Framework (RDF) Schema Specification 1.0. Candidate recommendation, World Wide Web Consortium, March 2000. See http://www.w3.org/TR/2000/CR-rdf-schema-20000327.

129. Buccella A., Cechich A. and Brisaboa N.R., Ontology-Based Data Integration Methods: A Framework for Comparison, Revista Colombiana de Computacion, 2005

130. Stefan Buettcher, Charles L. A. Clarke, Gordon V. Cormack, Information Retrieval: Implementing and Evaluating Search Engines, The MIT Press, 2010.

131. C. Carpineto and G. Romano. GALOIS: An ordertheoretic approach to conceptual clustering. In Machine Learning, Proc. ICML 1993, pages 33–40. Morgan Kaufmann Publishers, 1993.

132. Castells, P., Fernandez, M., Vallet, D., Mylonas, P., Avrithis, Y.: Self-Tuning Personalized Information Retrieval in an Ontology-Based Framework. 1st IFIP International Workshop on Web Semantics (SWWS 2005). LNCS Vol. 3532 (2005) 455-470.

133. Abdur Chowdhury and M. Catherine McCabe. Improving information retrieval systems using part of speech tagging. Technical Report TR 1998-48, 1998.

134. M. Ciocoiu, D. Nau, and M. Gruninger, «Ontologies for integrating engineering applications», ASME Journal of Computing and Information Science in Engineering, vol. 1, no. 1, pp. 12–22, March 2001.

135. Bruce Croft, Donald Metzler, Trevor Strohman, Search Engines: Information Retrieval in Practice, Addison Wesley; 1 edition, 2009.

136. James A. Danowski. Wordij: A word-pair approach to information retrieval. In TREC, P. 131-136, 1992.

137. Das S., Sundara S., Cyganiak R., Editors. R2RML: RDB to RDF Mapping Language – http://www.w3.org/TR/r2rml/

138. D. Dubois, H. Prade/ Fuzzy sets in approximate reasoning, Part 1: Inference with possibility distributions //Fuzzy Sets and Systems, №100 (1999), pp. 73-132.

139. D. Dutta and J. P. Wolowicz, «An Introduction to Product Lifecycle Management (PLM),» in Proceedings of the 12th ISPE International Conference on Concurrent Engineering: Research and Applications, Fort WorthDallas, TX, USA, July 25-29 2005.

140. Fayyad U. and Piatetsky-Shapiro G., «From Data Mining to Knowledge Discovery: An Overview», Advances in knowledge Discovery and Data Mining, Fayyad U., Piatetsky-Shapiro G.

141. Ronen Feldman, James Sanger, The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data, Cambridge University Press, 2007.

142. S. Gauch, J. Chaffee, and A. Pretschner. Ontology-based personalized search and browsing. Web Intelli-gence and Agent Systems, vol. 1, no. 3-4, 2003.

143. A. Ghafour, P. Ghodous, B. Shariat, and E. Perna, An Ontology-based Approach for Procedural CAD Models Data Exchange. In Proceeding of the 2006 Conference on Leading the Web in Concurrent Engineering: Next Generation Concurrent Engineering P. Ghodous, R. Dieng-Kuntz, and G. Loureiro, Eds. Frontiers in Artificial Intelligence and Applications, vol. 143. IOS Press, Amsterdam, The Netherlands, 251-259, 2006.

144. J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarran. Indexing with wordnet synsets can improve text retrieval. In Proceedings ACLCOLING Workshop on Usage of WordNet for Natural Language Processing, 1998.

145. Gruninger, M. and Fox, M.S. (1995). Methodology for the Design and Evaluation of Ontologies. In: Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal.

146. B. Grosof, I. Horrocks, R. Volz, and S. Decker. Description logic programs: Combining logic programs with description logics. In Proc. of WWW 2003, Budapest, Hungary, May 2003, pages 48–57. ACM, 2003.

147. Gruber, T. R. 1992. ONTOLINGUA: A Mechanism to Support Portable Ontologies, KSL-91-66, Knowledge Systems Laboratory, Stanford University.

148. Gruber T.R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. KSL-93-04, Knowledge Systems Laboratory, Stanford University, 1993.

149. Gruninger, M., and Fox, M. S. 1995. Methodology for the Design and Evaluation of Ontologies. Paper presented at the Workshop on Basic Ontological Issues in Knowledge Sharing, 19–20 August, Montreal, Quebec, Canada.

150. N. Guarino, C. Masolo, and G. Vetere: Ontoseek: Content-based Access to the Web, IEEE Intelligent Systems, Vol. 14, No. 3, pp. (www.loacnr.itPapersOntoSeek.pdf)

151. H. Haav and T. Lubi. A survey of concept-based information retrieval tools on the web. In 5th East-EuropeanConference, ADBIS 2001, Vilnius, Lithuania, September 2001, pp. 29-41.

152. Donna Harman. What we have learned, and not learned, from trec. In Proc. of the BCS IRSG'2000, P. 2-20.

153. Patrick Hayes. RDF Model Theory. Working draft, World Wide Web Consortium, September 2001. See http://www.w3.org/TR/rdf-mt/.

154. Horng, Y.-J, Chen, S.-M. and Lee, C.-H. (2001) Automatically constructing multi-relationship fuzzy concept in fuzzy information retrieval systems, IEEE International Fuzzy Systems Conference, pp. 606-609.

155. Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From SHIQ and RDF to OWL: The making of a web ontology language. Journal of Web Semantics, 1(1):7–26, 2003.

156. Hug, C., Front, A., Rieu, D., Henderson-Sellers, B. A method to build information systems engineering process metamodels. Journal of Systems and Software, Volume 82, Issue 10, October 2009, Pages 1730-1742

157. IEEE Recommended Practice for Architectural Description of Software-Intensive Systems. Institute of Electrical and Electronics Engineers, Sept. 2000. IEEE Std 1471-2000.

158. Jackson P., Mouliner I. Natural language processing for online applications: text retrieval, extraction, and categorization. John Benjamins Publishing Company. Amsterdam / Philadelphia. 2002.

159. Kanti Mardia et al. (1979). Multivariate Analysis. Academic Press.

160. Hideki Kozima. Text segmentation based on similarity between words. In Meeting of the Association for Computational Linguistics, P. 286-288, 1993.

161. Robert Krovetz and W. Bruce Croft. Lexical ambiguity and information retrieval. Information Systems, 10(2):115-141, 1992.

162. Carsten Lutz. Reasoning with concrete domains. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, pages 90–95. Morgan Kaufmann Publishers Inc., 1999.

163. C. Lutz. Description logics with concrete domains—a survey. In Advances in Modal Logics Volume 4. King's College Publications, 2003.

164. Matthias H., Gerald R., Harald R. A Comparison of RDB-to-RDF Mapping Languages. In: Proceedings of the 7th International Conference on Semantic Systems (I-Semantics), Graz, Austria, 07 September 2011 – 09 September 2011.

165. R. S. Michalski and R. Stepp. Learning from observation: Conceptual clustering. In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, editors, Machine Learning, An Artificial Intelligence Approach, volume II, pages 331–363, Palo Alto, 1983. TIOGA Publishing Co.

166. Mizoguchi, R. Vanwelkenhuysen, J.; Ikeda, M. Task Ontology for Reuse of Problem Solving Knowledge. Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing. IOS Press. 1995. 46-59.

167. Monderson J.N., Nair P.S. Fuzzy graphs and fuzzy hypergraphs. – Heidelberg; New-York: Physica-Verl., 2000

168. Christof Monz. Computational semantics and information retrieval. In Proceedings of the 2nd Workshop on Inference in Computational Seman tics (ICoS-2), P. 1-5, 2000.

169. A. Odd and G. Vasilakis, Building an Ontology of CAD Model Information. Geometric Modeling, Numerical Simulation, and Optimization Norway: SINTEF, Pages 11-41, 2007.

170. Noy N.F., McGuinness D.L. Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical

Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.

171. Ogawa, Y., Morita, T. and Kobayashi, K. (1991) A fuzzy document retrieval system using the keyword connection matrix and a learning method. Fuzzy Sets and Systems, 39: 163-179.

172. L. Patil, D. Dutta, and R. Sriram, "Ontology-based exchange of product data semantics," IEEE Transactions on Automation, Science and Engineering, Accepted for publication.

173. Pawlak, Z. Rough sets, International Journal of Computer and Information Sciences, 11, 341-356, 1982.

174. Pawlak, Z. Rough Sets, Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Dordrecht, 1991.

175. Pawlak Z. Rough Sets: Present State and Future Prospects// Intelligent Automation and Soft Computing. 1996. V. 2.

176. Pereira, R., Ricarte, I., Gomide, F. Relational Ontology in Information Retrieval Systems. In: Fuzzy Databases and Data Mining, Proc. IFSA2005, Tsinghua University Press, 2005, 509-514.

177. Jay M. Ponte and W. Bruce Croft. Text segmentation by topic. In European Conference on Digital Libraries, P. 113-125, 1997.

178. Salton, G., Automatic Text Processing. Addison-Wesley Publishing Company, Inc., Reading, MA, 1989.

179. G. Salton, J. Allan, and C. Buckley. Approaches to Passage Retrieval in Full Text Information Systems. In Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, P. 49-58, 1993.

180. Serrano-Guerrero, J., Olivas, J., Mata, J., Garces, P. Physical and Semantic Relations to Build Ontologies for Representing Documents. In: Fuzzy Databases and Data Mining, Proc. IFSA2005, Tsinghua University Press, 2005, 503-507.

181. R. Schevers and H. Drogemuller, Converting the industry foundation classes to the web ontology language. In CSIRO Manuf. and VIC; Inf. Technol., Highett, editors, Semantics, Knowledge and Grid, 2005. SKG '05. First International Conference on, Beijing, Pages 73-83, 2005.

182. A. Sieg, B. Mobasher, and R. Burke. Web search personalization with ontological user pro?les. In ACMSixteenth Conference on Information and KnowledgeManagement, CIKM 2007, Lisbon, Portugal, November 2007.

183. E. Sirin, B. Parsia, B. Cuenca Grau, A. Kalyanpur, and Y. Katz. Pellet: A Practical OWL-DL Reasoner. Technical report, University of Maryland Institute for Advanced Computer Studies (UMIACS), 2005. http:mindswap.orgpapersPelletDemo.pdf.

184. Alan F. Smeaton, Ruairi O'Donnell, and Fergus Kelledy. Indexing structures derived from syntax in TREC-3: System description. P. 100-110, 1994.

185. Fei Song and W. Bruce Croft. A general language model for information retrieval (poster abstract). Research and Development in Information Retrieval, P. 279-280, 1999.

186. Ashok Srivastava, Text Mining: Classification, Clustering, and Applications, Chapman and Hall/CRC, 2009.

187. Stojanovic L. at al. The role of ontologies in autonomic computing systems/ IBM Systems Journal Vol. 43, №3, 2004, pp. 598-616.

188. Umberto Straccia. A fuzzy description logic. In Proc. of the 15th Nat. Conf. on Artificial Intelligence (AAAI-98), pages 594–599, Madison, USA, 1998.

189. Umberto Straccia. A framework for the retrieval of multimedia objects based on four-valued fuzzy description logics. In F. Crestani and Gabriella Pasi, editors, Soft Computing in Information Retrieval: Techniques and Applications, pages 332–357. Physica Verlag (Springer Verlag), Heidelberg, Germany, 2000.

190. Studer R.,Benjamins R., Fensel D. Knowledge Engineering: Principles and Methods // Data and Knowledge Engineering, 25(1-2), 1998. p. 161-197.

191. Stumme G., Hotho F., Berendt B. Semantic Web Mining. State of the art and future directions/ Web Semantics: Science, Services and Agents on the World Wide Web, №4, 2006, pp. 124-143.

192. S. Szykman, R. Sriram, and W. Regli, "The role of knowledge in nextgeneration product development systems," ASME Journal of Computing and Information Science in Engineering, vol. 1, no. 3, pp. 3–11, March 2001.

193. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans Syst, Cybern. SMC-15(1) (1985) 116-132

194. Uschold M., Gruninger M. Ontologies: Principles, Methods and Applications. In Knowledge Engineering Review 11(2), 1996, pp. 93–155.

195. Vallet, D., Fernandez, M., Castells, P.: An Ontology-Based Information Retrieval Model. 2nd European Semantic Web Conference (ESWC 2005). LNCS Vol. 3532 (2005) 455-470

196. Ellen M. Voorhees. Natural language processing and information retrieval. In Information Extraction: Towards Scalable, Adaptable Systems, P. 32-48, 1999.

197. Wang, L.X., Mendel, J.M.: Generating fuzzy rules from numerical data with applications. IEEE Trans. Systems, Man, Cybern. 22(6) (1992) 1414-1427

198. Olaf Wolkenhauer. Data Engineering: Fuzzy Mathematics in Systems Theory and Data Analysis. John Wiley & Sons, 2001.

199. L. A. Zadeh. Fuzzy sets. Information and Control, 8(3):338–353, 1965.

200. Tom Heath, Christian Bizer. Linked Data: Evolving the Web into a Global Data Space. URL:http://www.linkeddatabook.com/editions/1.0

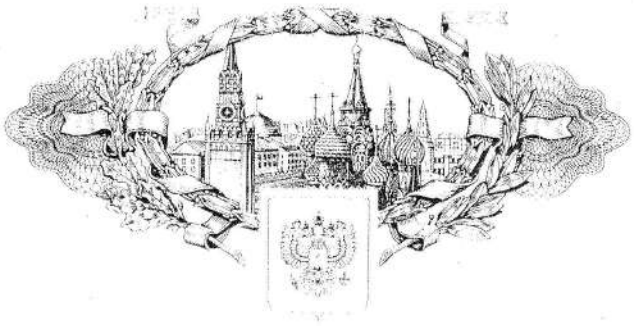
201. Tim Berners-Lee. Linked Data. URL:http://www.w3.org/DesignIssues/LinkedData.html

202. Resource Description Framework (RDF). URL:http://www.w3.org/RDF

Приложение 1.

Свидетельства о регистрации программ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012617586

Онтологически-ориентированный индексатор проектных документов

Правообладатель(ли): ***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Ульяновский государственный технический университет» (RU)***

Автор(ы): ***Наместников Алексей Михайлович, Филиппов Алексей Александрович (RU)***

Заявка № 2012615323
Дата поступления 26 июня 2012 г.
Зарегистрировано в Реестре программ для ЭВМ
22 августа 2012 г.

Руководитель Федеральной службы по интеллектуальной собственности

Б.П. Симонов

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012617587

Предметно-ориентированный редактор онтологий

Правообладатель(ли): ***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Ульяновский государственный технический университет» (RU)***

Автор(ы): ***Наместников Алексей Михайлович, Субхангулов Руслан Айратович (RU)***

Заявка № **2012615324**

Дата поступления **26 июня 2012 г.**

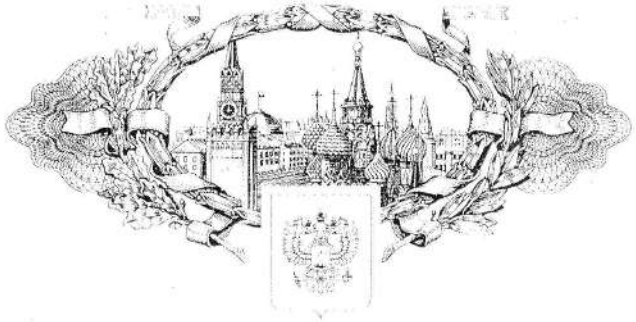
Зарегистрировано в Реестре программ для ЭВМ

22 августа 2012 г.

Руководитель Федеральной службы по интеллектуальной собственности

Б.П. Симонов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012617589

Онтологически-ориентированный кластеризатор проектных документов

Правообладатель(ли): ***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Ульяновский государственный технический университет» (RU)***

Автор(ы): ***Наместников Алексей Михайлович, Филиппов Алексей Александрович (RU)***

Заявка № 2012615326

Дата поступления 26 июня 2012 г.

Зарегистрировано в Реестре программ для ЭВМ

22 августа 2012 г.

Руководитель Федеральной службы по интеллектуальной собственности

Б.П. Симонов

Приложение 2

Фрагмент схемы онтологии предметной области

```
<rdf:RDF
 <!-- Life Circle -->
   <rdfs:Class rdf:ID="Stage"/>
   <rdfs:Class rdf:ID="StageConcept"/>
   <rdf:Property rdf:ID="PartOfStage">
     <rdfs:domain rdf:resource="#Stage" />
     <rdfs:range rdf:resource="#Stage" />
   </rdf:Property>
   <rdf:Property rdf:ID="ConnectToConcept">
     <rdfs:domain rdf:resource="#StageConcept" />
     <rdfs:range rdf:resource="#Concept" />
   </rdf:Property>
   <rdf:Property rdf:ID="ConnectToStage">
     <rdfs:domain rdf:resource="#StageConcept" />
     <rdfs:range rdf:resource="#Stage" />
   </rdf:Property>
<!-- Life Circle End -->
<!-- Concepts -->
   <rdfs:Class rdf:ID="Concept"/>
   <rdfs:Class rdf:ID="Term"/>
   <rdfs:Class rdf:ID="ConceptTerm"/>
   <rdfs:Class rdf:ID="ConceptInstance"/>
   <rdf:Property rdf:ID="PartOf">
     <rdfs:domain rdf:resource="#Concept" />
     <rdfs:range rdf:resource="#Concept" />
   </rdf:Property>
   <rdf:Property rdf:ID="SubclassOf">
     <rdfs:domain rdf:resource="#Concept" />
     <rdfs:range rdf:resource="#Concept" />
   </rdf:Property>
   <rdf:Property rdf:ID="ConnectToCTConcept">
     <rdfs:domain rdf:resource="#ConceptTerm" />
     <rdfs:range rdf:resource="#Concept" />
   </rdf:Property>
   <rdf:Property rdf:ID="ConnectToCTTerm">
     <rdfs:domain rdf:resource="#ConceptTerm" />
```

```
    <rdfs:range rdf:resource="#Term" />
  </rdf:Property>
  <rdf:Property rdf:ID="ConnectToCTFreq">
    <rdfs:domain rdf:resource="#ConceptTerm" />
    <rdfs:range rdf:resource=
      "http://www.w3.org/2001/XMLSchema#float" />
  </rdf:Property>
  <rdf:Property rdf:ID="ConnectToCIConcept">
    <rdfs:domain rdf:resource="#ConceptInstance" />
    <rdfs:range rdf:resource="#Concept" />
  </rdf:Property>
  <rdf:Property rdf:ID="ConnectToCIInstance">
    <rdfs:domain rdf:resource="#ConceptInstance" />
    <rdfs:range rdf:resource="#Instance" />
  </rdf:Property>
<!-- Concepts End -->
<!-- Instances -->
  <rdfs:Class rdf:ID="Instance"/>
  <rdfs:Class rdf:ID="InstanceTerm"/>
  <rdf:Property rdf:ID="ConnectToInstance">
    <rdfs:domain rdf:resource="#InstanceTerm" />
    <rdfs:range rdf:resource="#Instance" />
  </rdf:Property>
  <rdf:Property rdf:ID="ConnectToTerm">
    <rdfs:domain rdf:resource="#InstanceTerm" />
    <rdfs:range rdf:resource="#Term" />
  </rdf:Property>
  <rdf:Property rdf:ID="ConnectToFreq">
    <rdfs:domain rdf:resource="#InstanceTerm" />
    <rdfs:range rdf:resource=
      "http://www.w3.org/2001/XMLSchema#float" />
  </rdf:Property>
<!-- Instances End-->
</rdf:RDF>
```

Фрагмент онтологии предметной области

```
<rdf:RDF
<!-- Life Circle Ontology -->
  <Stage rdf:ID="Разработка концепцииАС "/>
  <Stage rdf:ID="Изучениеобъекта_">
    <PartOfStage rdf:resource="Разработка концепцииАС "/>
  </Stage>
  ...
  <Stage rdf:ID="Послегарантийноеобслуживание_">
    <PartOfStage rdf:resource="Стадия сопровожденияАС "/>
  </Stage>
<!-- Life Circle Ontology End -->
<!-- Domain Ontology -->
  <Concept rdf:ID="Серия стандартов 34"/>
  <Concept rdf:ID="Общетехнические термины">
    <PartOf rdf:resource="Серия стандартов 34"/>
  </Concept>
  ...
  <Concept rdf:ID="Оперативная информацияАС ">
    <SubclassOf rdf:resource="Выходная информацияАС "/>
  </Concept>
  <Concept rdf:ID="Серия стандартов 19"/>
  <Concept rdf:ID="Общие понятия">
    <PartOf rdf:resource="Серия стандартов 19"/>
  </Concept>
  ...
  <Concept rdf:ID="Индексирование адреса">
    <PartOf rdf:resource="Индексный регистр"/>
  </Concept>
<!-- Domain Ontology End -->
<!-- Project Ontology -->
  <Instance rdf:ID="Прибор"/>
  ...
  <Instance rdf:ID="Диагностик"/>
  <Instance rdf:ID="Микроконтроллер"/>
  <Term rdf:ID="документ"/>
  <Term rdf:ID="предпусков"/>
  ...
  <Term rdf:ID="сформир"/>
  <Term rdf:ID="многоканальн"/>
```

```
  <ConceptInstance rdf:ID="CInst1">
    <ConnectToCIConcept rdf:resource="Программно техническийкомплексАС  "/>
    <ConnectToCIInstance rdf:resource="Прибор"/>
  </ConceptInstance>
  ...
  <ConceptInstance rdf:ID="CInst160">
    <ConnectToCIConcept rdf:resource="Язык ассемблера"/>
    <ConnectToCIInstance rdf:resource="Микроконтроллер"/>
  </ConceptInstance>
  <InstanceTerm rdf:ID="CIndex1">
    <ConnectToInstance rdf:resource="Прибор"/>
    <ConnectToTerm rdf:resource="документ"/>
    <ConnectToFreq rdf:datatype="float">
      0,0483945306899893
    </ConnectToFreq>
  </InstanceTerm>
  ...
  <InstanceTerm rdf:ID="CIndex40267">
    <ConnectToInstance rdf:resource="Микроконтроллер"/>
    <ConnectToTerm rdf:resource="помощ"/>
    <ConnectToFreq rdf:datatype="float">
      0,0151719176135846
    </ConnectToFreq>
  </InstanceTerm>
<!-- Project Ontology End -->
</rdf:RDF>
```

Приложение 3.

Аннотированный отчет

АННОТИРОВАННЫЙ ОТЧЕТ
по годовому этапу научно-исследовательской работы № 1167 в рамках базовой части государственного задания в сфере научной деятельности по Заданию № 2014/232 за 2015 год

1. Тема: Разработка нового подхода к интеллектуальному анализу слабоструктурированных информационных ресурсов

2. Номер государственной регистрации: 115.02.10.10.109

3. Руководитель: Ярушкина Надежда Глебовна

4. Организация-исполнитель: Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Ульяновский государственный технический университет»

5. Телефон руководителя: 778079

6. Электронная почта руководителя: jng@ulstu.ru

7. Интернет-адрес (URL): www.ulstu.ru

8. Сроки проведения:
— начало: 01.02.2015
— окончание: 31.12.2015

9. Наименование годового этапа: Исследование моделей и методов интеллектуального анализа текстовых информационных ресурсов на основе онтологии

10. Плановое финансирование (рублей):
— проведения годового этапа: 1 000 752,00 руб.

11. Фактическое финансирование (рублей):
— проведения годового этапа: 942 123,06 руб.

12. Коды темы по ГРНТИ: 20.23.25 28.17.19

13. Приоритетное направление: Информационно-телекоммуникационные системы

14. Критическая технология: Технологии обработки, хранения, передачи и защиты информации

15. Полученные научные и (или) научно технические результаты: 1. Предложена новая структурно-функциональная модель онтологии текстовых информационных ресурсов, отличающаяся многоуровневой структурой и позволяющая выполнять запросы с учетом текущего контекста принятия решений. 2. Разработана онтологическая модель профиля пользователя информационно-поисковой системы, которая позволяет специфицировать опыт взаимодействия специалиста с архивом текстовых документов на концептуальном уровне. 3. Разработан алгоритм формирования контекстно-ориентированных запросов к электронному архиву текстовых документов на основе байесовского классификатора с учетом моделируемых информационных потребностей.

16. Полученная научная и (или) научно-техническая продукция: Разработана программная система информационной поддержки проектировщика, которая применяется в процессе проектирования автоматизированных систем при анализе содержимого электронного архива текстовой документации и позволяет достичь улучшенных технико-экономических показателей

объектов проектирования за счет сокращения времени выполнения опытно-конструкторских работ.

17. Ключевые слова и словосочетания, характеризующие результаты (продукцию): прикладная онтология, контекстно-ориентированный запрос, информационная потребность, байесовский классификатор, нечеткий граф

18. Наличие аналога для сопоставления результатов (продукции): Аналогами являются программные средства автоматизированного анализа технической документации и проектных решений: TDMS 4.0, электронный архив технической документации на базе ЭЛАР САПЕРИОН, AS-Archive, АСУ PartY'97 («Лоция Софт»), Bentley ProjectWise («НЕОЛАНТ»).

19. Преимущества полученных результатов (продукции) по сравнению с результатами аналогичных отечественных или зарубежных НИР:

а) по новизне: результаты являются новыми

б) по широте применения: в рамках организации или предприятия

в) в области получения новых знаний: в области применения новых знаний (для прикладного научного исследования)

20. Степень готовности полученных результатов к практическому использованию (для прикладного научного исследования и экспериментальной разработки): выполнен прототип (установки, методики, системы, программы и т.д.)

21. Предполагаемое использование результатов и продукции: Полученные научные и практические результаты предполагается использовать в крупных организациях, которые имеют большие электронные архивы текстовой документации. Возможность формирования к ним контексто-ориентированных запросов пользователей позволяет извлекать из слабоструктурированных информационных источников сохраненный ранее опыт. Разработанные модели и алгоритмы формирования моделей профилей пользователей позволяют в значительной степени учитывать индивидуальные информационные потребности.

22. Форма представления результатов: Результаты НИР представлены в виде: 1) научно-технического отчета; 2) монографии: Наместников А.М. Метауровень информационного обеспечения САПР: от теории к практике/ А.М. Наместников. – Ульяновск : УлГТУ, 2015. – 175 с. 3) статей в российских изданиях: 4) статей в зарубежных изданиях: 5) диссертации: Субхангулов Р.А. Онтологическая информационная поддержка проектирования в электронных архивах технической документации : диссертация на соискание ученой степени канд. техн. наук. Ульян. гос. техн. университет, Ульяновск, 2015.

23. Использование результатов в учебном процессе: продукция для обеспечения учебного процесса

24. Предполагаемое развитие исследований: Исследование предполагается продолжать в направлении построения математических моделей и алгоритмов автоматизации построения онтологий из внешних Интернет-ресурсов. Указанное направление включает решение задачи интеллектуального анализа проектов программных систем, состоящие из набора формализованных диаграмм (UML и ER-диаграмм в нотации IDEF1X) в рамках IT-организаций и на уровне глобальных репозиториев (например, такого как Github, основанного на системе контроля версий Git). Новый онтологический подход к интеллектуальному анализу проектных диаграмм информационных систем позволит использовать такие глобальные репозитории для поиска близких по решаемым задачам проектов и прототипов программных систем с минимальным участием высококвалифицированных специалистов-экспертов.

25. Количество сотрудников, принимавших участие в выполнении работы и указанных в научно-технических отчетах в качестве исполнителей приведено в приложении №1

26. Библиографический список публикаций, отражающих результаты научно-исследовательской работы приведен в приложении №2

Ректор Федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Ульяновский государственный технический университет»

(подпись) А.П. Пинков

М.П.

Руководитель проекта

(подпись) Н. Г. Ярушкина

Приложение 4.

Акт об использовании результатов диссертации

УТВЕРЖДАЮ
Генеральный директор,
председатель НТС ФНПЦ
АО «НПО «Марс», к.т.н.
_____ В.А. Маклаев
«12» 04 2017 г.

АКТ

об использовании результатов докторской диссертации Наместникова А.М. "Интеллектуальные репозитории технической документации в проектировании автоматизированных систем"

Научно-техническая комиссия в составе:

председателя комиссии: первый заместитель генерального директора по науке – начальник КНИО-2, к.т.н. Павлыгин Э.Д.,

членов комиссии: главный научный сотрудник, д.т.н. Токмаков Г.П.,
заместитель главного инженера по качеству и инженерно-техническому обеспечению - начальник управления 5, к.т.н. Емельянов А.А.
начальник отдела ИАСУП, к.т.н. Перцев А.А.
начальник отдела технической документации Ефремов А.Е.,
ведущий инженер-программист, к.т.н. Радионова Ю.А.,

настоящим актом подтверждает использование для анализа технических документов электронного архива ФНПЦ АО «НПО «Марс» следующих научных и практических результатов диссертационной работы А.М. Наместникова "Интеллектуальные репозитории технической документации в

проектировании автоматизированных систем":

- метод концептуального индексирования слабоструктурированных информационных ресурсов электронного архива проектной организации;
- метод формирования навигационной структуры текстовых технических документов электронного архива;
- нечеткая модель и методика оценки качества онтологических ресурсов проектной организации;
- комплекс программ, составляющий интеллектуальный проектный репозиторий и реализующий информационную поддержку проектировщика при формировании контекстно-ориентированных запросов к электронному архиву и построении навигационной структуры архива текстовых технических документов.

Комплекс программ онтологической информационной поддержки как подсистема электронного архива предприятия использован при проектировании автоматизированных систем.

Эффективность использования научно-технических результатов подтверждена экспериментальными исследованиями, целью которых являлось определение количественной оценки качества выполнения проектных поисковых запросов к электронному архиву в сравнении с традиционными методами поиска электронных технических документов на основе набора ключевых слов.

Для реализации информационной поддержки проектирования автоматизированных систем в электронном архиве на ФНПЦ АО «НПО «Марс» была разработана прикладная онтология, содержащая в своем составе около 500 понятий и более 15000 уникальных терминов. Точность выполнения проектных поисковых запросов к электронному архиву с использованием онтологических моделей примерно на 30% лучше по сравнению с системами Яндекс.Персональный поиск и Архивариус 3000. Среднее время поиска технического документа в одном сеансе работы с электронным архивом сократилось примерно на 50% (с 14 минут до 6-7 минут).

Результаты получены в ходе выполнения и внедрения х/д НИР № 230/2005 «Интеллектуальный сетевой архив электронных информационных ресурсов» и в рамках второго этапа «Разработка программной системы интеллектуального анализа текстовой информации» НИР «Система интеллектуального поиска и анализа в Интернет-СМИ и социальных сетях», выполняемых Ульяновским государственным техническим университетом по заказу ФНПЦ АО «НПО «Марс».

Председатель комиссии:
Первый заместитель генерального директора
по науке – начальник КНИО-2, к.т.н Э.Д. Павлыгин

Члены комиссии:
Главный научный сотрудник, д.т.н. Г.П. Токмаков

Заместитель главного инженера
по качеству и инженерно-техническому
обеспечению - начальник
управления 5, к.т.н. А.А. Емельянов

Начальник отдела ИАСУП, к.т.н. А.А. Перцев

Начальник отдела технической
документации А.Е. Ефремов

Ведущий инженер-программист, к.т.н. Радионова Ю.А.

Приложение 5.

Акт сдачи-приемки

Акт № 96/17

сдачи-приемки этапа 2

«Разработка ПС интеллектуального анализа текстовой информации» (шифр – «Терьер»)

г. Ульяновск «28» 09 2017 г.

Основание: договор от 01.02.2017 г. № 72/17-УлГТУ, техническое задание на выполнение НИР от «01» февраля 2017 г.

Мы, нижеподписавшиеся, представитель Исполнителя первый проректор – проректор по научной работе Ярушкина Н.Г., с одной стороны, и представитель Заказчика генеральный директор ФНПЦ АО «НПО «Марс» В.А. Маклаев с другой стороны составили настоящий акт о том, что в период с 28.08.2017 по 31.08.2017 проведена приемка работ по этапу 2 «Разработка ПС интеллектуального анализа текстовой информации» , выполненных в соответствии с договором от 01.02.2017 № 72/17-УлГТУ между УлГТУ и ФНПЦ АО «НПО «Марс».

В результате рассмотрения выполненных работ по разработке ПС интеллектуального анализа текстовой информации

УСТАНОВЛЕНО:

1. Этап 2 «Разработка ПС интеллектуального анализа текстовой информации» выполнен в полном объеме и соответствует техническому заданию от «01» февраля 2017 г. на выполнение НИР;
2. Этап 2 «Разработка ПС интеллектуального анализа текстовой информации» считать законченным и принятым.

Обнаруженные недостатки: нет.

3. Рекомендации: в рамках этапа 3 доработать следующее:
   - Расширить модель онтологических представлений знаний по теме «Ключевые руководители (лица СМИ) г. Ульяновска», такие как губернатор, его заместители, а также руководители гос. учреждений и крупных компаний, в количестве не более 200 человек.
   - Для закладок «Пользователи», «Группы», «Электронные СМИ» добавить фильтры для возможности выбора по параметрам;
   - Оптимизировать алгоритмы поисковых подсистем;

Твердая фиксированная цена этапа 2 по договору от 01.02.2017 № 72/17-УлГТУ составляет 280 000 (двести восемьдесят тысяч) рублей 00 копеек, НДС не облагается.

Аванс не перечислялся.

4. Следует к перечислению 280 000 (двести восемьдесят тысяч) рублей 00 копеек, НДС не облагается.

Исполнитель	Заказчик
Первый проректор - проректор по НР ФГБОУ ВО УлГТУ	Генеральный директор ФНПЦ АО «НПО «Марс»
_____ Н.Г. Ярушкина	_____ В.А.Маклаев
М.П. «____» _____ 2017	М.П. «28» 09 2017

Приложение 6.

Программы повышения квалификации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет» (УлГТУ)
Адрес: 432027, Ульяновская область,
г. Ульяновск, ул. Северный Венец, д.32
ИНН 7325000052 КПП 732501001
Банковские реквизиты:
УФК по Ульяновской области,
Банк: Отделение Ульяновск,
г. Ульяновск Код ТОФК 6800
л/с 21686Х85090
р/с 40501810073082000001
БИК 047308001, ОКАТО 73401000000
ОКПО 02069378, ОКОПФ 72
ОКВЭД 80.30
КБК 00000000000000000180
ОКТМО 73701000

Министерство образования и науки Российской Федерации
Адрес: 125009, г. Москва, ул. Тверская, д. 11, стр. 4
ИНН 7710539135 КПП 771001001
Межрегиональное операционное УФК
л/с 03951000740
р/с 40105810700000001901
в Операционном Департаменте Банка России, г. Москва 701
БИК 044501002
ОКТМО 45382000
ОКАТО 45286585000
ОКПО 00083380
ОГРН 1047796287440 (дата присвоения 23.04.2004)
ОКВЭД 84.11.11 ОКОГУ 1322500
ОКФС 12 ОКОПФ 75104

АКТ № 1
сдачи-приемки выполненных работ

по соглашению между Министерством образования и науки Российской Федерации и федеральным государственным бюджетным образовательным учреждением высшего образования «Ульяновский государственный технический университет» об условиях предоставления и использования субсидии на реализацию ведомственной целевой программы «Повышение квалификации инженерно-технических кадров на 2015–2016 годы» от 5 мая 2016 г. № 06.Z14.21.0042

составлен «14» марта 2017 г.

Предмет Соглашения: Предоставление Минобрнауки России субсидии из федерального бюджета Получателю с целью реализации дополнительных профессиональных программ повышения квалификации и стажировок инженерно-технических кадров, реализуемых на базе российских образовательных организаций с участием предприятий, исследовательских и инжиниринговых центров на территории России и за рубежом.

Мы, нижеподписавшиеся,
представитель Получателя исполняющий обязанности ректора федерального государственного бюджетного образовательного учреждения высшего образования «Ульяновский государственный технический университет» Пинков Александр Петрович, действующий на основании приказа Минобрнауки России от 29 декабря 2014 г. № 12-07-03/167 и Устава, с одной стороны, и представитель Минобрнауки России временно исполняющий обязанности директора Департамента

государственной политики в сфере подготовки рабочих кадров и ДПО Черноскутова Инна Анатольевна, действующий на основании доверенности от 22 февраля 2017 г. №ОВ-165/06, с другой стороны,

составили настоящий акт о том, что

работы, выполненные по соглашению между Министерством образования и науки Российской Федерации и федеральным государственным бюджетным образовательным учреждением высшего образования «Ульяновский государственный технический университет» об условиях предоставления и использования субсидии на реализацию ведомственной целевой программы «Повышение квалификации инженерно-технических кадров на 2015–2016 годы» от 5 мая 2016 г. № 06.Z14.21.0042, удовлетворяют условиям Соглашения, отчетная документация в надлежащем порядке оформлена.

Краткое описание выполненных работ:

1. Программа «Перспективы внедрения в автомобилестроении современных методов обеспечения качества

Обучено, чел.	Завершили стажировки на территории России, чел.	Завершили стажировки за рубежом, чел.
15	3	1

2. Программа «Стандарты авиационного приборостроения и процедуры сертификации»

Обучено, чел.	Завершили стажировки на территории России, чел.	Завершили стажировки за рубежом, чел.
17	3	2

3. Программа «Управление проектами разработки программного обеспечения»

Обучено, чел.	Завершили стажировки на территории России, чел.	Завершили стажировки за рубежом, чел.
15	3	1

4. Программа «Программная инженерия»

Обучено, чел.	Завершили стажировки на территории России, чел.	Завершили стажировки за рубежом, чел.
15	3	1

5. Программа «Основы построения систем передачи и обработки радиолокационной информации»

Обучено, чел.	Завершили стажировки на территории России, чел.	Завершили стажировки за рубежом, чел.
16	4	2

6. Программа «Бережливое производство»

Обучено, чел.	Завершили стажировки на территории России, чел.	Завершили стажировки за рубежом, чел.
15	3	1

Размер субсидии, предоставляемой из федерального бюджета Получателю в соответствии с условиями Соглашения, составляет 1 410 600 рублей 00 коп. (Один миллион четыреста десять тысяч шестьсот рублей 00 коп.).

Размер субсидии, перечисленной из федерального бюджета Получателю, составил 1 410 600 рублей 00 коп. (Один миллион четыреста десять тысяч шестьсот рублей 00 коп.).

Сумма субсидии, использованная на финансирование мероприятий, предусмотренных условиями Соглашения, составила 1 410 600 рублей 00 коп. (Один миллион четыреста десять тысяч шестьсот рублей 00 коп.).

Остаток неиспользованной суммы субсидии составляет 0 рублей 00 коп. (Ноль рублей 00 коп.).

Работу сдал:

От Получателя

Исполняющий обязанности ректора федерального государственного бюджетного образовательного учреждения высшего образования «Ульяновский государственный технический университет»

_____ А.П. Пинков

М.П.

Работу принял:

От Минобрнауки России

Временно исполняющий обязанности директора Департамента государственной политики в сфере подготовки рабочих кадров и ДПО Минобрнауки России

_____ И.А. Черноскутова

М.П.

1

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

СОГЛАСОВАНО

Зам. генерального директора
по управлению персоналом – начальник
управления ГФНПЦ АО «НПО «Марс»
наименование должности руководителя
(Милушкин С.А.)
«____» _____ 2016 г.

УТВЕРЖДАЮ

ФГБОУ ВПО «Ульяновский
государственный технический университет»
Первый проректор - проректор по ДиДО
(Афанасьев А.Н.)
«____» _____ 2016 г.

Дополнительная профессиональная программа повышения квалификации

«Программная инженерия»
(наименование программы)

Приоритетное направление развития науки, технологий и техники в Российской Федерации

Целевая категория участников программы:
специалисты с высшим образованием

Приоритетное направление модернизации и
технологического развития экономики России

Перспективные виды вооружения, военной и специальной техники

1

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

СОГЛАСОВАНО	**УТВЕРЖДАЮ**
Зам. генерального директора по управлению персоналом – начальник управления 1 ФНПЦ АО «НПО «Марс» *наименование должности руководителя* _____ (Милушкин С.А.) «____» _____ 2016 г.	ФГБОУ ВПО «Ульяновский государственный технический университет» Первый проректор , проректор по ДиДО _____ (Афанасьев А.Н.) «____» _____ 2016 г.

Дополнительная профессиональная программа повышения квалификации

«Управление проектами разработки программного обеспечения»
(наименование программы)

Приоритетное направление развития науки, технологий и техники в Российской Федерации

Целевая категория участников программы:
специалисты с высшим образованием

Приоритетное направление модернизации и
технологического развития экономики России

Перспективные виды вооружения, военной и специальной техники