ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

На правах рукописи

Васильев Иван Анатольевич

МЕТОДЫ И ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА ПОСТРОЕНИЯ СЕМАНТИЧЕСКИХ WEB-ПОРТАЛОВ

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

Диссертация
на соискание ученой степени
кандидата технических наук

Научный руководитель:
доктор технических наук, профессор,
заслуженный деятель науки и техники РФ Ямпольский В. З.

Томск – 2005

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Совершенствование существующих и разработка новых подходов к сбору, хранению, обработке и распространению информации является неотъемлемой частью процесса развития информационных технологий и информационных систем (ИС). Необходимость такого совершенствования во многом обусловлена непрерывным ростом количества электронных документов и их доступности, что на ряду со слабой структурированностью информационных фондов осложняет управление информацией и работу пользователей с ней. Существующие подходы к работе с информацией становятся не достаточно эффективными.

Для решения проблемы совершенствования доступа к растущему объему информации и информационным услугам, предоставляемым многочисленными источниками информации, специалистами была предложена концепция Web-порталов. Web-портал является программной системой, которая призвана обеспечить унифицированный доступ к информации, хранящейся во множестве разнородных информационных источников. Web-портал структурирует информацию и предоставляет средства для ее поиска.

Различные виды Web-порталов разрабатываются и внедряются в России и за рубежом. Перспективность данного подхода к интеграции и структуризации информации отмечается аналитиками и подтверждается пользователями. Огромное число пользователей сети Интернет обращаются к услугам различных поисковых Web-порталов, таких как «Yahoo!» (http://www.yahoo.com) или «Яндекс» (http://www.yandex.ru), а современные компании, такие как концерн Volkswagen, корейская вещательная корпорацию KOBACO или немецкая фармацевтическая корпорация Schering AG, внедряют [1] корпоративные Web-порталы, предлагаемые ведущими разработчиками данного класса программных продуктов.

Применение Web-порталов для интеграции источников информации и структуризации ее растущего объема поставило вопрос о повышении каче-

ства обработки информации в Web-порталах. Наиболее существенно проблема роста объема информации сказывается на качестве поиска в Web-порталах. Примером, демонстрирующим необходимость перехода на новый качественный уровень, является функционирование поисковых Web-порталов в сети Интернет. Обычно они предоставляют услуги двух типов: поиск по рубрикатору и полнотекстовый поиск. Если необходимая пользователю информация сосредоточена в какой-либо рубрике, то ему лучше воспользоваться возможностью просмотра этой рубрики, так как точность категоризации информации в рубрикаторе находится на очень высоком уровне. Это объясняется тем, что наполнение рубрикатора осуществляется вручную или полуавтоматически с участием модераторов Web-портала, которые учитывают смысл структурируемой информации. В свою очередь точность и полнота результатов полнотекстового поиска существенно ниже, чем у поиска по рубрикатору, так как информация обрабатывается без учета семантики информации. С ростом объема обрабатываемой информации возможность наполнения рубрикатора снижается – модераторы Web-портала не справляются с объемом информации. Если же пользователь обращается к полнотекстовому поиску, то проблема обработки большого объема информации возлагается на него самого – на поисковый запрос Web-портал выдает огромное количество результатов, среди которых пользователь должен дополнительно искать необходимую информацию. В настоящее время в Web-порталах информация обрабатывается на синтаксическом уровне, то есть без учета таких свойств естественного языка как синонимия, полисемия и омонимия. Это приводит к снижению качества обработки информации и в том числе к неудовлетворительным результатам поиска [2].

Для перехода на новый качественный уровень при обработке информации необходимо вести обработку на семантическом уровне, то есть учитывать ее смысл.

За последние несколько лет активное развитие получило направление в информационных технологиях, занимающееся проблемами учета семантики

в рамках информационных систем. Это направление исследует семантические технологии, позволяющие создавать новый класс ИС. Созданные на основе семантических технологий ИС отличаются от традиционных тем, что:

- ИС при обработке информации в некоторой фиксированной предметной области использует знания из этой предметной области;
- знания предметной области выражаются явно – в виде модели (частично или полностью);
- модель выражает смысл терминов (понятий) предметной области через связи между ними;
- модель отражает различные точки зрения на предметную область.

Рассматриваемые в данном диссертационном исследовании Web-порталы являются многопользовательскими ИС, которые предоставляют унифицированный доступ к различным информационным источникам и программным приложениям. Web-порталы, как правило, обрабатывают большой объем информации. С учетом этого применение в рамках Web-портала новых подходов и методов к обработке информации имеет высокую практическую значимость, а исследование подходов и разработка методов построения Web-портала на основе семантических технологий являются актуальными.

В настоящее время исследования в области развития и внедрения семантических и портальных технологий ведутся как в России, так и за рубежом. Тем не менее, необходимо, констатировать значительный разрыв по количеству исследований в этой области между отечественным и зарубежным научным сообществом.

В качестве основополагающих исследований отечественных авторов нужно выделить [2-6]. Ряд работ по использованию семантических технологий поддерживается Российским Фондом Фундаментальных Исследований также, в том числе «Исследование принципов семантического поиска текстовой информации на основе использования интеллектуальных и статистических методов» (03-01-00572, Харин Н. П., МАДИ, Москва), «Инструментальные программные средства семантического поиска текстовой информации,

использующие интеллектуальные и статистические методы» (04-07-90328, Михайловский О. В., РосНИИИТ и АП, Москва); осуществляется также поддержка проектов в области разработки порталов, например, «Технология разработки специализированных Интернет-порталов знаний по гуманитарным наукам» (04-01-00884, Загорулько Ю. А., ИСИ СО РАН, Новосибирск). К сожалению, результаты выполненных проектов недостаточно публикуются и с ними трудно ознакомиться в сети Интернет.

Более многочисленными и доступными в сети Интернет являются результаты исследований и внедрений семантических технологий в структуру Web-порталов, выполненных зарубежными учеными [7-18]. Среди них можно выделить такие крупные проекты как «OntoWeb: Ontology-based information exchange for knowledge management and electronic commerce» [16] или «ODESeW: Automatic generation of knowledge portals for intranets and extranets» [18].

В результате анализа выполненных исследований необходимо отметить их недостаточность в области использования семантических технологий для описания семантики *контента* объектов Web-порталов. В соответствии с [19] объект может быть рассмотрен в трех разных аспектах – *структура*, *контекст* и *контент*. В большинстве исследований семантические технологии применяются для описания контекста объекта, в то время как в Web-порталах значительный интерес представляет описание семантики объектов с точки зрения контента.

В рамках данного исследования анализируется отечественный и зарубежный опыт создания семантических Web-порталов и предлагается новый подход к использованию семантических технологий в Web-порталах.

Целью диссертационного исследования *является разработка методов использования семантических технологий в Web-порталах для реализации информационных процессов в них с учетом семантики контента объектов.*

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

- разработать архитектуру семантического ядра Web-портала;
- разработать методы семантического описания контента объектов Web-портала;
- разработать методы использования описаний объектов Web-портала для реализации его функций на семантическом уровне.

Объектом исследования являются технологии построения Web-порталов.

Предметом исследования являются подходы и методы использования семантических технологий в Web-порталах для реализации информационных процессов на семантическом уровне.

Методы исследования. В ходе диссертационного исследования были использованы модели и методы теории множеств, профессионально-логический анализ и обобщение, метод экспертных оценок, методы объектно-ориентированного проектирования и программирования.

Научная новизна результатов исследования заключается в следующем:

- разработан метод семантического описания объектов Web-портала с точки зрения контента, использующий предложенную автором структуру семантических метаданных;
- разработан метод вычисления семантической близости метаданных, основанный на известном методе определения сотипности;
- разработаны методы поиска, категоризации и формирования рекомендации объектов Web-портала с учетом семантики их контента, основанные на методе вычисления близости семантических метаданных;
- разработана архитектура семантического ядра Web-портала, реализующего функции описания семантики контента объектов, поиска, категоризации и предоставления рекомендаций.

Практическая значимость исследования заключается:

- в программной реализации разработанного автором семантического ядра Web-портала;
- в применении предложенных методов для разработки семантического Web-портала для современной IT-компании;
- в возможности использования созданного семантического ядра Web-портала в системах управления знаниями [20].

На защиту выносятся:

1. метод семантического описания объектов Web-портала;
2. метод вычисления семантической близости метаданных;
3. методы поиска, категоризации и формирования рекомендации объектов Web-портала;
4. архитектура семантического ядра Web-портала.

Апробация. Основные научные положения и отдельные результаты работы докладывались и обсуждались на следующих конференциях:

- Международная научно-практическая конференция студентов, аспирантов и молодых ученых «Современные техника и технологии 2003»;
- Международная научно-практическая конференция студентов, аспирантов и молодых ученых «Современные техника и технологии 2004»;
- Международная научно-практическая конференция «Современные средства и системы автоматизации 2004».

Предложенные подходы и методы были протестированы в процессе практической реализации Web-порталов. Результаты исследования использовались при разработке и реализации Web-портала для «Центра профессиональной подготовки специалистов нефтегазового дела» ТПУ и Web-портала системы управления знаниями компании «ЭлеСи».

Диссертационное исследование выполнялось в соответствии с проектом «Создание информационно-программной среды научно-образовательного комплекса Томска для работы со знаниями и объектами интеллектуальной собственности» (контракт № 2093 от 1.11.2002) в рамках Федеральной Целевой Программы «Интеграция науки и высшего образова-

ния России на 2002-2006 годы» и темой научно-исследовательской работы, проводимой по заданию Министерства образования Российской Федерации (регистрационный номер 1.38.99) «Исследование методов представления, структуризации и контекстного поиска явных и неявных знаний для построения систем управления знаниями».

Публикации. По теме диссертационного исследования опубликовано 9 печатных работ, в том числе одна в реферируемом издании [21]. Имеется свидетельство государственного координационного центра информационных технологий об отраслевой регистрации разработки «Web-портал для работы с явными и неявным знаниями организации» в Отраслевом фонде алгоритмов и программ (свидетельство №4608; авторы Тузовский А. Ф., Васильев И. А., Козлов С. В., Усов М. В.; дата выдачи 29.04.2005).

Личный вклад автора. Все результаты, составляющие основное содержание диссертации, получены автором самостоятельно. В опубликованных работах лично автором обоснованы варианты использования семантических технологий в информационных системах в общем [22] и в частности в Web-порталах [20, 23, 24], пояснены разработанные методы описания семантики объектов Web-портала и вычисления их семантической близости [21, 25], описано разработанное семантическое ядро Web-портала [21] и приведены варианты применения разработанных методов и алгоритмов в работе Web-порталов [21, 26, 27, 28].

Структура и объем диссертации. Диссертация состоит из введения, 4 глав, заключения, списка литературных источников из 117 наименований и 6 приложений. Содержит 56 рисунков и 36 таблиц.

В первой главе рассматривается понятие Web-портала и их классификация по различным критериям. На основании анализа описаний существующих Web-порталов определяются их отличительные особенности и предлагается обобщенная архитектура, включающая инфраструктуру и множество функциональных модулей. Для современных Web-порталов отмечается проблема увеличения объема обрабатываемой информации, снижающая каче-

ство информационных процессов. Обосновывается, что решение данной проблемы возможно путем использования семантических технологий. Анализируется текущее состояние развития семантических технологий и существующие проекты по их использованию в Web-порталах. На основании анализа отмечается доминирующая роль онтологических моделей для целей представления семантики информации и недостаточность исследований в области описания семантики объектов Web-портала с точки зрения контента.

Во второй главе анализируются и обобщаются существующие в литературе варианты использования онтологий в информационных системах, в общем. Предлагаются варианты использования онтологий для реализации информационных процессов в Web-портале с учетом семантики контента объектов. С целью создания семантического Web-портала предлагается структура семантического ядра портала, реализующего предложенные варианты использования онтологии и позволяющего обрабатывать информацию с учетом ее семантики. Семантическое ядро состоит из сервера онтологий и сервера семантических метаданных. Рассматривается структура и функции указанных серверов. Описывается разработанная структура семантических метаданных для представления семантики контента объектов Web-портала.

В третьей главе описываются разработанные автором методы использования онтологий, обеспечивающие реализацию информационных процессов в Web-портале с учетом семантики объектов. Обосновывается структура онтологий для обеспечения работы семантического ядра портала. Поясняется метод формирования семантических метаданных, позволяющий описывать семантику контента объектов. Приводится подробное описание разработанных методов вычисления семантической близости элементов онтологии и метаданных, позволяющих количественно оценить схожесть семантических описаний объектов Web-портала. Предлагаются варианты применения разработанных методов для реализации функций семантического поиска, категоризации и формирования рекомендаций.

В четвертой главе описывается программная реализация разработанного семантического ядра. Поясняются основные программные интерфейсы, классы и компоненты, включенные в реализацию сервера онтологий и сервера семантических метаданных. Излагается методика тестирования разработанных методов и полученные результаты тестирования. Описываются результаты внедрения разработанных методов, алгоритмов и соответствующего программного обеспечения при создании семантических Web-порталов различного уровня.

Автор выражает благодарность профессору Ямпольскому В. З. за внимание к работе, замечания и методическую помощь во многом способствовавшие улучшению качества окончательного варианта рукописи. Автор признателен доценту Тузовскому А. Ф. за ценные консультации и всестороннюю поддержку данного исследования.

Глава 1. Порталы и семантические технологии

1.1. Анализ существующих подходов к реализации портала

1.1.1. Понятие портала и классификация порталов

Web-порталы (далее, порталы) являются таким классом программных систем, для которого терминология и классификация еще окончательно не сложились. Можно встретить различные определения понятия «портал» [29-31] и различные по функциональным возможностям его программные реализации.

В рамках данного исследования под порталом, будем понимать Web-приложение, обладающее, по сравнению с Web-сайтом, расширенной функциональностью и реализующее идею централизованного доступа сообщества пользователей к необходимой информации и сервисам.

Если Web-сайт – это набор логически взаимосвязанных страниц, доступных через Web-браузер по протоколу HTTP, то портал – это Web-сайт, который имеет широкий набор функций.

Классифицировать порталы можно по различным критериям. В приведенной ниже классификации в качестве критериев выступают тематика портала, целевая аудитория, решаемые порталом задачи и используемые технологии (рис. 1.1).

С точки зрения тематики порталы можно разделить на горизонтальные и вертикальные.

1. Информационно-тематическое наполнение и функции горизонтального портала нацелены на широкий круг пользователей. В сети Интернет такие порталы называют мега-порталами (Yahoo!, Яндекс и т.п.), так как они предоставляют информацию (погода, новости и т.п.) и функции (поиск сайтов, отправка электронной почты и т.п.), полезные практически всем пользователям Сети.

2. Вертикальные порталы предоставляют полный перечень необходимой информации и функций для определенного и обычно узкого круга поль-

зователей. Примером индустриальных вертикальных порталов могут служить порталы по страхованию, автомобилестроению и т.п.

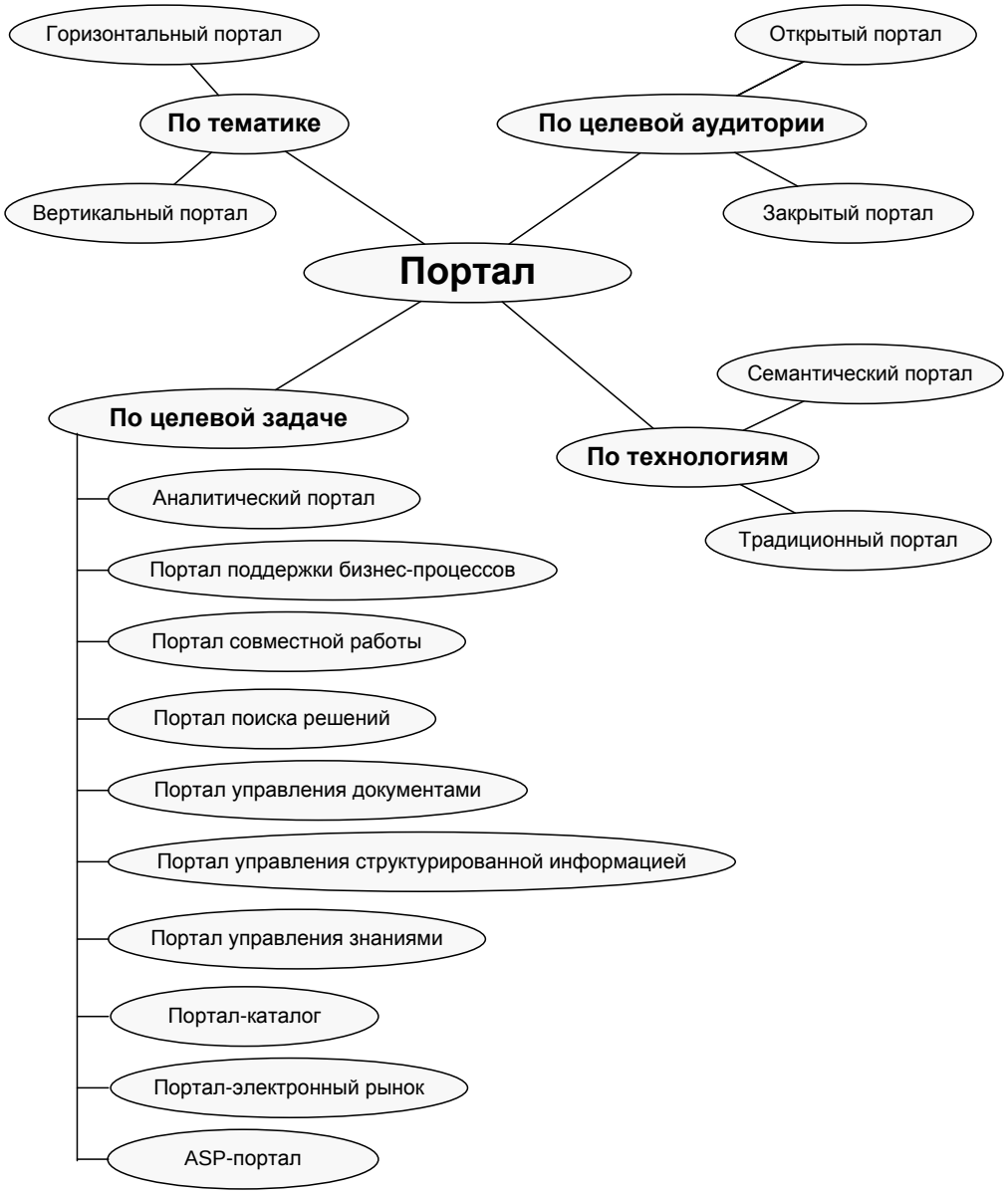


Рис. 1.1. Классификация порталов

Целевая аудитория портала может быть не ограничена и тогда портал является открытым. В противном случае портал является закрытым.

1. Открытые порталы доступны широкому сообществу пользователей. Чаще всего такие порталы размещаются в сети Интернет.

2. Закрытые порталы предоставляют доступ ограниченному кругу пользователей. Регистрация пользователя в таких порталах обычно проходит этап верификации, когда право регистрируемого на доступ к порталу под-

тверждается уполномоченными лицами. К этому виду обычно относятся порталы, размещенные в корпоративных сетях организаций. Они предназначены для сотрудников компании и известны под названием B2E-порталы.

С точки зрения целевой задачи портал может быть ориентирован на выполнение одной или нескольких задач. В качестве наиболее распространенных можно выделить несколько классов порталов, и каждый портал может быть отнесен к одному или более классам.

1. Аналитические порталы позволяют лицам, принимающим решения, получать и создавать отчеты.

2. Порталы поддержки бизнес-процессов реализуют специфические функции и поддерживают специфические процессы и приложения. Примером могут служить B2B-, B2E- или B2C-порталы.

3. Порталы совместной работы предоставляют пользователям виртуальные пространства для координации и выполнения совместной работы.

4. Порталы поиска решений предназначены для привлечения экспертов к решению проблем. Для этого в портале ведется учет пользователей и их компетенции, что позволяет выделять экспертов в конкретных областях знаний, находить их и пользоваться их опытом при решении проблем.

5. Порталы по управлению документами.

6. Порталы управления структурированной информацией.

7. Порталы управления знаниями. Призваны помочь компании эффективнее использовать имеющиеся у нее явные и неявные знания за счет управления знаниями на каждом этапе его жизненного цикла – на этапах выявления, создания, хранения, распространения и использования.

8. Порталы-каталоги. Систематизируют доступные информационные ресурсы и предоставляют возможность поиска необходимых ресурсов.

9. Порталы-электронные рынки. Они связывают продавцов и покупателей друг с другом, предоставляя специфическую информацию о рынках, товарах и услугах.

10. ASP-порталы (ASP, Application Service Provider). ASP-порталы, являясь собственностью какой-либо компании, предназначены для оказания услуг другим компаниям, то есть являются порталами типа B2B. Они предоставляют возможность компаниям-клиентам получать в аренду, как товары, так и услуги.

С точки зрения используемых технологий порталы могут быть разделены на традиционные и семантические.

1. В традиционных порталах информация обрабатывается без учета ее семантики.

2. Семантические порталы являются новым классом порталов, которые содержат модель знаний некоторой предметной области и используют ее для обработки информации с учетом семантики. Для реализации таких порталов помимо традиционных технологий используются активно развивающиеся семантические технологии.

1.1.2. Функции портала

На основании анализа описаний, опубликованных компаниями-производителями порталов (приложение 1), а также на основании анализа литературы [29, 30, 32-39] выявлены наиболее часто встречающиеся функции портала. Среди выявленных функций портала выделены три, которые позволяют отличить его от Web-сайта. Этими обязательными функциями являются:

- интеграция информационных источников;
- интеграция приложений;
- поиск по всем информационным источникам.

Все выявленные функции портала сгруппированы в функциональные модули (рис. 1.2). Три обязательные функции выделены в три соответствующих функциональных модуля. Следовательно, каждый портал содержит как

минимум три функциональных модуля – модуль интеграции данных, модуль интеграции приложений и сервисов и модуль индексирования и поиска.

Рис. 1.2. Инфраструктура и функциональные модули портала

1. *Модуль интеграции данных* предоставляет средства организации единого информационного пространства из множества разнородных источников информации. В его задачи входит организация унифицированного доступа к различным источникам информации – реляционным базам данных, хранилищам документов и электронной почты, каталогом пользователей и так далее.
2. *Модуль интеграции приложений и сервисов* решает задачу интеграции в портал различных программных приложений, а также предоставляет возможности расширения функциональности портала за счет добавления сервисов. Различие между приложениями и сервисами заключается в том, что приложения являются отдельно разработанными программными продуктами, а сервисы специально разрабатываются как часть портала.

3. *Модуль индексирования и поиска* позволяет находить требуемую пользователю информацию. Для этого модуль периодически сканирует все источники информации и индексирует их текстовое содержание. Созданные индексы используются для ускорения поиска информации по запросам.
4. *Модуль категоризации* позволяет структурировать все информационное пространство портала. Под структуризацией понимается разделение всего множества информации на связанные между собой подмножества. Каждое подмножество составляют связанные по смыслу информационные единицы – документы, описания объектов, таблицы баз данных и так далее. Связь между подмножествами может иметь различный характер. Например, структура может быть сетевой или иерархической. Помимо этого модуль предоставляет средства формирования необходимой структуры.
5. *Модуль профилирования* предоставляет возможности явной и неявной персонификации во время взаимодействия пользователя с порталом. Для этого каждый пользователь имеет в портале свой профиль – описание предпочтений и/или интересов. При явной персонификации непосредственно учитываются предпочтения пользователя. Примером явной персонификации является настройка внешнего вида страниц портала или выбор необходимой информации для отображения. Неявная персонификация основана на автоматическом анализе действий пользователя в портале и выявлении его предпочтений. Например, пользователю может быть предоставлена информация, которую он не искал, но которая соответствует его интересам.
6. *Модуль оповещения* использует push-технологии для информирования пользователей о различных событиях в портале. Примером событий могут служить публикация новой информации в портале по определенной тематике, появление новой версии документа, реги-

страция нового пользователя и т.п. В качестве средств оповещения могут выступать электронная почта, RSS-рассылки и т.п.

7. *Модуль представления* отвечает за формирование визуального представления требуемого пользовательского интерфейса портала. Обычно в модуле используется подход, предоставляющий возможность объединения нескольких источников данных («комбинирование содержания») в единое визуальное представление.
8. *Модуль безопасности* выполняет функции идентификации пользователей при подключении к порталу и формирования контекста безопасности при работе пользователя с функциями портала.

Реализация каждого функционального модуля зависит от используемой *инфраструктуры портала*, к которой относятся:

- сервер приложений;
- средства хранения данных и доступа к ним;
- средства программирования;
- средства визуального представления.

От инфраструктуры во многом зависит управляемость, расширяемость и работоспособность портала. Инфраструктура является основой для реализации всех остальных функций портала.

1.1.3. Архитектура портала

Подходы, технологии и стандарты, используемые как в рамках инфраструктуры портала, так и при реализации его функциональных модулей, являются общеупотребительными. Это позволяет описать обобщенную архитектуру портала, охватывающую потенциально возможную функциональность (рис. 1.3).

На уровне интерфейса пользователя используется тонкий клиент (Web-браузер), способный визуализировать представление информации, описанное на языке HTML. Для использования некоторых функциональных воз-

можностей портала пользователь может пользоваться и некоторыми другими клиентскими приложениями (например, клиент электронной почты, RSS-клиент и т.п.).

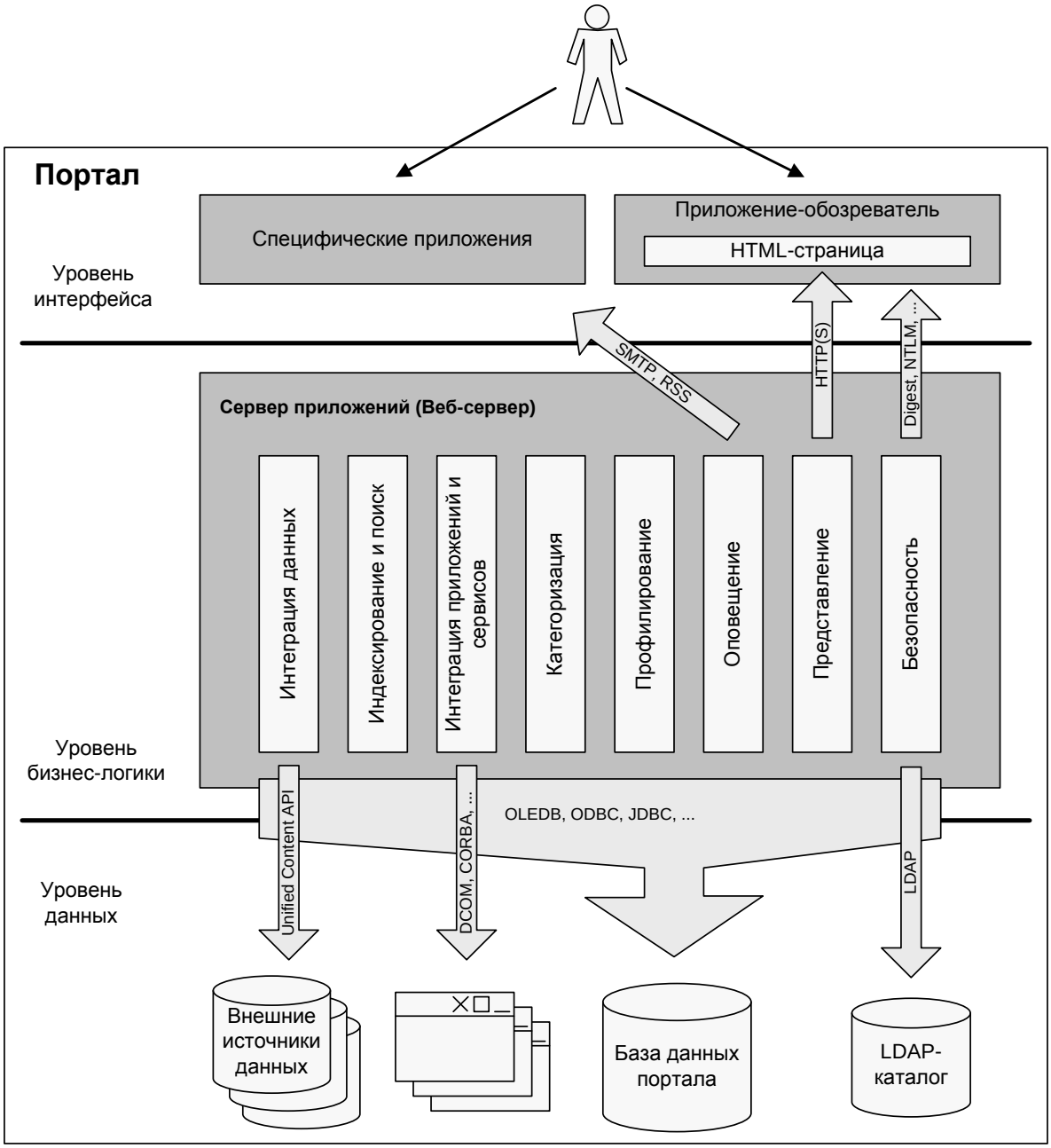


Рис. 1.3. Обобщенная архитектура портала

Для реализации *инфраструктуры портала* существует ряд широко используемых технологий и серверов приложений. К наиболее распространенным *серверам приложений* можно отнести такие программные продукты как Microsoft IIS, Apache HTTP Server, Oracle Application Server. В качестве *хра-*

нилищ данных используются реляционные базы данных, доступ к которым осуществляется с использованием технологий OLEDB, ODBC, JDBC и т.п. В качестве *технологий программирования* используются ASP, ASP.NET, PHP, JSP и прочие. *Визуальное представление* описывается на языке HTML, который интерпретируется Web-браузером пользователя.

При реализации *функциональных модулей* чаще всего используются следующие решения:

- Интеграция данных основывается на использовании программных посредников (mediator). Примером реализации данного подхода является программный интерфейс Unified Content API от компании IBM [40].
- Интеграция приложений и сервисов основывается на компонентных технологиях, таких как CORBA, DCOM, Web-сервисы, .NET Remoting.
- Для индексирования и поиска в корпусе полнотекстовых документов используется метод обратных индексов и различные статистические методы ранжирования результатов.
- Средства категоризации чаще всего основаны на кластерном анализе, использовании модели векторного пространства (vector space model) или статистических методах.
- В целях профилирования создаются описания пользователей с необходимым набором атрибутов для возможности их интерпретации.
- Оповещение пользователей выполняется по заранее определенным событиям с использованием таких push-технологий как рассылка электронной почты, RSS-рассылка и т.д.
- Формирование HTML-представления выполняется на основании объединения нескольких источников данных посредством ряда схожих программных технологий, таких как WebParts [41].

- В области обеспечения безопасности применяется правила распределения привилегий, разделение пользователей на группы и шифрование данных. Идентификация пользователей осуществляется по протоколам SSL, NTLM, Kerberos и т.п. Учетные записи пользователей могут храниться как в базе данных портала, так и в специализированных хранилищах, доступ к которым выполняется по протоколу LDAP.

Хотя подходы к разработке и реализации порталов можно считать достаточно проработанными с точки зрения методов и используемых технологий, существует объективная необходимость их развития. Эта необходимость обусловлена развитием телекоммуникационных технологий, делающих информацию более доступной, а также объективным ростом объема информации.

Портал является такой информационной системой (ИС), которая предоставляет сообществу пользователей унифицированный доступ информационному пространству, и поэтому проблема повышения качества информационных процессов при большом объеме информации в портале является особенно актуальной. Одним из подходов к решению данной проблемы является переход на семантический уровень при сборе, обработке, накоплении, хранении, поиске и распространении информации. Этот подход развивается в рамках направления «Семантические технологии».

1.2. Семантические технологии в порталах

Семантика наряду с синтаксисом и прагматикой является разделом семиотики – науки о знаках. Понятие знака тесно связано с понятиями денотата и сигнификата (рис. 1.4).

Внешний элемент (последовательность звуков, графических знаков и т.п.) – знак – связан, во-первых, с обозначаемым предметом, явлением дей-

ствительности – денотатом, и, во-вторых, с отражением этого предмета, явления в сознании человека – сигнификатом (концептом, понятием).

Рис. 1.4. Смысловой треугольник

В задачи *синтаксиса* входит изучение внутренних свойств знаковых систем, то есть отношений между знаками безотносительно к их интерпретации. В связи с неоднозначностью соответствия знака и сигнификата, приводящей к омонимии, полисемии и синонимии естественного языка, обработка информации на синтаксическом уровне снижает качество работы информационных систем.

В задачи *семантики* входит изучение отношений между знаком и денотатом, то есть рассмотрение интерпретаций знаков с учетом их неоднозначности. Поэтому качество обработки информации в ИС может быть увеличено за счет учета семантики информации (знаков).

Одним из наиболее перспективных направлений разрешения проблемы обработки информации на семантическом уровне являются семантические технологии.

В рамках семантических технологий разрабатываются подходы, стандарты и методы, которые обеспечивают возможность явного представления семантики информации. Явное представление семантики информации должно способствовать созданию программных систем, позволяющих обрабаты-

вать информацию на семантическом уровне. Сами по себе такие программные системы не «понимают» смысл обрабатываемой информации, но позволяют автоматизировать процессы поиска и структуризации информации на основании ее семантического описания.

Семантические технологии призваны стать основой для создания программных систем в различных областях. В зависимости от области применения и решаемой задачи подходы к представлению семантики и ее обработке могут варьироваться [42], но их объединяет наличие модели знаний, которая описывает семантику отдельных элементов информации и связи между ними.

Приложения, созданные на основе семантических технологий, отличаются от традиционных приложений тем, что в них:

- знания о предметной области представлены явно – в виде модели (частично или полностью);
- модель определяет соответствие между элементами информации (знаками) и понятиями (сигнификатами);
- модель описывает семантику через связи между понятиями;
- модель содержит знания о некоторых предметных областях;
- модель отражает различные точки зрения на предметные области.

На сегодняшний день в рамках семантических технологий наиболее активно исследуется и развивается онтологический подход к представлению знаний предметной области, на основании которого разрабатываются интеллектуальные информационные системы, и в том числе порталы. Онтология является моделью знаний, которая может использоваться для описания семантики объектов ИС.

1.2.1. Онтологический подход к представлению знаний

Онтологию предметной области можно считать логическим продолжением развития сетевых моделей представления знаний, таких как семантиче-

ские сети и системы фреймов. С начала 90-х годов онтологический подход стал активно развиваться в ряде прикладных направлений исследований.

1.2.1.1. Понятие онтологии

В 80-х годах термин «онтология» мигрировал из философии в область компьютерной науки, когда он был использован рядом исследовательских сообществ по Искусственному Интеллекту (ИИ) вначале в области инженерии знаний, в обработки естественных языков, а затем, в представлении знаний. В конце 90-х годов начались активные исследования возможности использования онтологии в таких областях, как интеграция информации, поиск информации в Интернет и управление знаниями. Позже онтологии стали рассматриваться в качестве ключевого элемента в концепции создания Semantic Web – нового этапа развития сети Word Wide Web [43].

Важно отметить, что задачи, которые решают исследователи в области ИИ с помощью онтологий, отличаются от задач в других областях компьютерной науки. Целью Искусственного Интеллекта является создание программно-аппаратной системы, имитирующей интеллектуальную деятельность человека. Подобная система должна быть способной заменить человека в какой-либо области деятельности. Однако онтология может быть использована и для менее глобальных целей – реализации на основании онтологии новых или совершенствования существующих функций программных систем, учитывающих семантику обрабатываемой информации. В данном исследовании онтология рассматривается со второй точки зрения.

Изначальное отсутствие четкого определения термина «онтология» и увеличение количества сфер применения этого понятия привели к еще большей терминологической разнородности. Ниже приводится некоторые из существующих определений.

«Онтология есть формальная спецификация групповой концептуализации» [8]. Под концептуализацией в ИИ понимается описание предметной области, определяющее множество объектов, существующих в описываемой

предметной области, и множество отношений между этими объектами. Недостатком многих существующих систем является отсутствие явного описания концептуализации, на которой основана система. Например, экспертная система содержит представленные в декларативной форме знания о решении проблем в какой-либо предметной области. Однако модификация заложенных в систему знаний или повторное их использование в других системах затруднительны в силу того, что концептуализация базы знаний и исходных предположений не выражена явно. В соответствие с приведенным определением онтология должна явно специфицировать концептуализацию какой-либо предметной области. Эта спецификация должна быть формальной, т.е. выраженной с использованием какого-либо формального языка. И, кроме того, концептуализация должна быть групповой, то есть отражать взгляды группы людей на предметную область, а не взгляд одного человека.

«Онтология – это логическая теория, которая ограничивает допустимые модели логического языка. Онтология в этом случае должна обеспечивать аксиомы, которые ограничивают значение нелогических символов (предикатов и функций) логического языка, используемых как «примитивы» для определенных целей представления. Цель онтологии – характеризовать концептуализацию, ограничивая возможные интерпретации нелогических символов логического языка для установления консенсуса о том, как описывать знания с использованием этого языка. Концептуализация рассматривается как множество неформальных правил, которые ограничивают структуру части действительности» [9].

Существует и множество других определений [7, 44]. Общим для всех существующих определений является понимание онтологии как модели представления знаний какой-либо предметной области в виде набора *понятий* этой предметной области и существующих между ними *отношений*. То есть онтология представляет модель предметной области в виде некоторой сетевой структуры, в которой семантика каждого понятия определяется через его отношения с другими понятиями. Причем во множестве отношений су-

ществует отношение типа «родитель-ребенок», упорядочивающее понятия предметной области в иерархию – таксономию понятий. К отношениям того типа относятся отношения «целое-часть» (part-of), «класс-подкласс» (is-a) и т.п. Таксономия не является математическим деревом, так как позволяет одному понятию-ребенку иметь несколько понятий-родителей. Именно таксономия является той составляющей, которая отличает онтологию от наиболее близких к ней сетевых моделей представления знаний – семантических сетей и систем фреймов.

Сетевые модели представляют знания в такой форме, что программы, использующие эти модели, чаще всего опираются только на наличие отношений между понятиями без учета типов этих отношений. Это связано с тем, что набор возможных типов отношений в сетевых моделях представления знаний не ограничен, инженер по знаниям может по необходимости добавлять новые типы отношений. Теоретически это означает, что программа, использующая сетевую модель, не может учитывать особенности различных типов отношений, то есть учитывать их семантику. В онтологии же создание сетевой структуры не является приоритетной задачей. Такой задачей является *создание таксономий по заранее известным отношениям*. Это позволяет создавать программы, которые бы учитывали семантику этих отношений.

Исходя из сказанного выше предлагается следующее формальное определение онтологии.

Определение 1.1. Онтология – это знаковая система

$$O = \langle C, R, L, P_C, P_{LC}, P_{LR} \rangle, \qquad (1.1)$$

в которой

$C = \{c_1, ..., c_n\}$ – конечное множество понятий в онтологии,

$R = \{r_1, ..., r_m\}$ – конечное множество бинарных отношений $r_i(c_x, c_y)$ между понятиями,

$L = \{l_1, ..., l_k\}$ – конечное множество лексических меток (словарь онтологии),

$P_C \subseteq C \times C, P_C \in R$ – антисимметричное, транзитивное, нерефлексивное бинарное отношение, являющееся отношением частичного порядка на множестве понятий C,

$P_{LC} \subseteq L \times C$ – бинарное отношение инцидентности между множествами L и C,

$P_{LR} \subseteq L \times R$ – бинарное отношение инцидентности между множествами L и R.

1.2.1.2. Классификация онтологий

В связи с разнообразием задач, решаемых с помощью онтологий, их можно дифференцировать по множеству признаков. На практике основными критериями классификации являются: назначение онтологии, выразительность онтологии, формальность онтологии. Далее рассматриваются некоторые из приведенных в литературе классификаций.

По назначению, согласно классификации в [45], онтологии подразделяют следующим образом (рис. 1.5):

- Онтологии верхнего уровня (top-level ontology). Содержат описания общих понятий, которые не связаны с конкретными предметными областями, то есть они применимы к любой из них. Такими понятиями могут быть «время», «пространство», «событие», «действие» и т.д.
- Онтологии предметных областей (domain ontology). Описывают терминологию в различных предметных областях.
- Онтологии задач (task ontology). Описывают конкретные процессы, характерные для различных предметных областей. Например, «банковская транзакция», «диагностика» и т.д.
- Онтологии приложения (application ontology). Онтология приложения объединяет в себе онтологию задач и онтологию предметной области для того, чтобы специализировать понятия из них для конкретного применения.

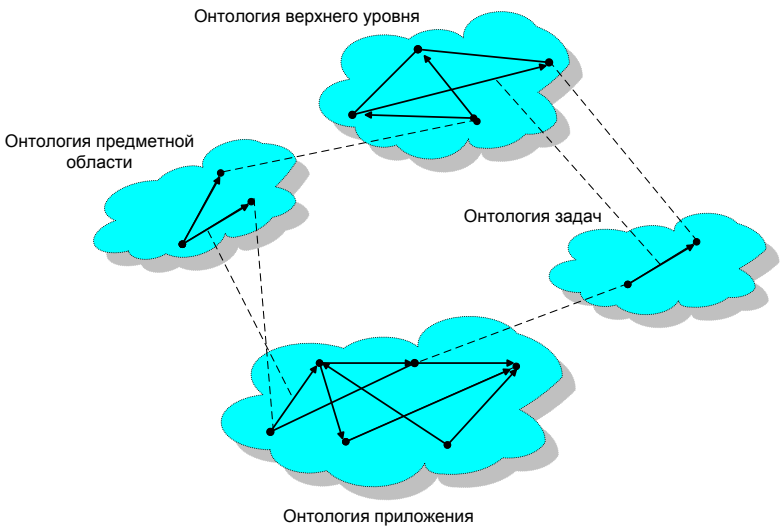


Рис. 1.5. Классификация онтологий по назначению

Различные авторы предлагают различные классификации онтологий по назначению [8, 10, 46], но все сходятся в одном – в необходимости создания онтологий верхнего уровня. С помощью них предполагается решить проблему определения соответствия различных онтологий между собой, а, следовательно, и проблему взаимодействия систем, использующих различные онтологии приложения. Онтологии приложения должны расширять онтологию верхнего уровня, уточнять ее в рамках конкретной предметной области. В идеале, должна существовать единственная онтология верхнего уровня, которая бы использовалась всеми инженерами по знаниям для создания онтологий приложений. Но в силу сложности централизации процесса создания онтологии верхнего уровня и существования различных взглядов на этот процесс, таких онтологий создано значительное количество. Например, онтология Джона Сова [47], реализующая философский подход к выявлению базовых категорий бытия, или онтология WordNet [48], отражающая лингвистический подход к анализу категорий окружающей действительности.

Следующим критерием для классификации онтологий является их *выразительность*. Выразительность онтологии определяется степенью детальности описания вводимых в онтологии понятий. Чем больше ограничений на использование и больше отношений с другими понятиями содержит описа-

ние понятия, чем оно более детально. В [49] предлагается классифицировать онтологии по выразительности следующим образом (в порядке возрастания выразительности):

- Таксономия. Представляет собой множество понятий с заданными меду ними отношением «родитель-ребенок», которое упорядочивает понятия в иерархию – таксономию.
- Тезаурус. Выразительность онтологии возрастает, если к описанию понятий, организованных в виде иерархии (таксономии), добавить дополнительно взаимосвязи (отношения) с другими понятиями. Набор вводимых отношений зависит от решаемой задачи и предметной области. Например, если онтология будет использоваться для обработки естественного языка, то целесообразно ввести отношение синонимии, антонимии и т.п.
- Сеть понятий. В такой онтологии перечень возможных отношений не регламентирован, можно вводить сколько угодно отношений, и для этих отношений не регламентируется ни область определения (возможные субъекты отношений), ни область значений (возможные объекты отношений).
- Полная онтология. Для каждого понятия в такой онтологии дано определение в терминах других понятий. Для каждого отношения задана область определения и область значения и дополнительно могут быть заданы правила использования отношения. Полная онтология наиболее точно описывает требуемую предметную область и уменьшает количество непреднамеренных моделей при формализации онтологии с использованием логического языка.

Важно отметить, что приведенные классы выразительности онтологий никак не отражают форму или способ записи онтологии. Они указывают на то, *что* содержится в онтологии. В свою очередь *степень формальности онтологии* отражает то, *как* это содержание записывается. В [11] предложено

разделить онтологии по критерию формальности на следующие группы (по возрастанию формальности):

- Неформальные онтологии, выраженные на естественном языке.
- Полуформальные онтологии на упрощенном естественном языке. Они описываются на ограниченном по структуре и словарю естественном языке, что значительно уменьшает многозначность определений, свойственную естественному языку.
- Полуформальные онтологии на искусственном языке. Для описания онтологии используется искусственный, формально определенный язык, в результате чего онтология может использоваться в работе программных систем.
- Формальные онтологии. Описываются на формальном языке с явно определенным синтаксисом и семантикой и обладающим свойствами непротиворечивости и полноты. В качестве формальных способов записи онтологии используются логические языки – логика предикатов первого порядка и ее подмножества.

Такие характеристики онтологии как выразительность и формальность являются ключевыми для решения практических задач. Полные онтологии детально описывают предметную область задачи, то есть такого описания достаточно для решения прикладной задачи. А формальный способ фиксации выявленных знаний в виде онтологии (с использованием специальных языков) позволяет обрабатывать их с использованием компьютеров.

1.2.1.3. Языки описания онтологии

Как уже отмечалось, онтологии создаются, прежде всего, для того, чтобы зафиксировать знания в какой-либо предметной области и использовать их в различных приложениях. Чтобы приложение могло использовать онтологию, она должна быть описана на языке, понятном приложению.

В соответствии с классификацией онтологий по уровню формальности любая онтология, описанная на языке, отличном от естественного, является частично или полностью формальной. Таким образом, *язык описания онтологии* позволяет приложению использовать онтологию, так как ограничивает возможные интерпретации синтаксических конструкций, исключая многозначность записи, свойственную естественному языку.

В свою очередь классификация онтологий по степени выразительности отражает тот факт, что онтологии могут описывать предметную область с разной степенью детальности. Следовательно, для записи разных типов онтологий нужны языки с разными выразительными возможностями. Выразительность языка описания онтологии определяется лежащим в его основе *формальным способом представления знаний*. Например, для описания онтологии типа «тезаурус» достаточно свойств ориентированного графа. А для описания «полновесных онтологий» используются формальные способы представления знаний, основанные на логических исчислениях.

В мире разработано большое количество различных языков описания онтологий. Это такие языки как KL-ONE [50], KRYPTON [51], Loom [52], CLASSIC [53], Ontolingua [54], F-Logic [55], SHOE [56], RDF(S) [57], OWL [58] и прочие. Некоторые из них устарели и не используются на практике. Например, язык KL-ONE является прототипом всех языков, основанных на логическом формализме «дескриптивная логика», но в настоящее время уже не применяется, так как разработаны его усовершенствованные варианты. Есть языки, которые поддерживаются локальными научными группами, но не получают широкого распространения. Они используются для решения каких-либо частных прикладных задач или для проведения исследований. Примером такого языка является язык SHOE (Simple Html Ontology Extension). Но есть и языки, которые активно развиваются и используются широким кругом специалистов. Примером такого языка является OWL. Этот язык получает все более широкое применение в мире и в настоящее время

рекомендован к использованию (стандартизован) организацией World Wide Web Consortium.

Ниже приведен пример формального описания части онтологии.

Пример 1.1. Для предметной области «Компания» формализуем факт «сотрудники компании участвуют в проектах компании» с помощью дескриптивной логики класса *AL* (атрибутивный язык).

1. Проект ⊑ T
2. Сотрудник ⊑ T
3. Сотрудник ≡ ∃ Участвует_В_Проекте . Проект

Теперь запишем полученные выше высказывания дескриптивной логики с помощью XML-синтаксиса языка описания онтологий OWL DL (подкласс языка OWL для записи высказываний дескриптивной логики).

```
<?xml version="1.0"?>
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:owl="http://www.w3.org/2002/07/owl#"
   xmlns="http://kms.cctpu.edu.ru/ivanvasilyev/dissertationexample.owl#"
   xml:base="http://kms.cctpu.edu.ru/ivanvasilyev/dissertationexample.owl"
>
 <owl:Ontology rdf:about="Компания"/>

 <owl:Class rdf:ID="Проект"/>                                   } Запись высказывания 1

 <owl:Class rdf:ID=" Сотрудник">                                }
  <owl:equivalentClass>                                         }
   <owl:Restriction>                                            }
    <owl:onProperty>                                            }
     <owl:ObjectProperty rdf:ID="Участвует_В_Проекте"/>          } Объединенная запись высказываний 2 и 3
    </owl:onProperty>                                           }
    <owl:someValuesFrom rdf:resource="#Проект"/>                }
   </owl:Restriction>                                           }
  </owl:equivalentClass>                                        }
 </owl:Class>                                                   }

</rdf:RDF>
```

В приведенном примере показана связь языка описания онтологий с формальным способом представления знаний. Поэтому при выборе языка описания онтологий необходимо убедиться, что лежащий в его основе формальный способ представления знаний, обладает достаточной выразительностью для решения поставленных задач.

1.2.2. Семантические метаданные

Как уже было сказано, онтология является моделью знаний, в которой семантика понятий, заключающих знания предметной области, определяется через описание отношений между понятиями. В информационной системе, основанной на онтологии, семантика объектов должна описываться в терминах онтологии, и, следовательно, необходимы подходы к описанию семантики объектов ИС. Одним из подходов является создание семантических метаданных.

1.2.2.1. Понятие семантических метаданных

Понятие метаданных давно используется в области информационных систем. В качестве определения чаще всего используется следующее: «метаданные – это данные о данных». Метаданные призваны улучшить управляемость и доступность данных, так как предоставляют дополнительную информацию, которая нужна для выполнения этих задач.

Метаданные описывают объекты информационной системы. Метаданные обычно состоят из набора предопределенных элементов (свойств или атрибутов), которые описывают различные аспекты объекта. *Структура* метаданных может варьироваться, но всегда имеет следующие характеристики:

- конечный набор атрибутов;
- наличие названий у атрибутов;
- закрепленный смысл каждого атрибута (трактовка значения атрибута);
- возможность присвоить одному атрибуту несколько значений.

Структура метаданных, обладающая данными характеристиками, достаточна для описания свойств объекта, существенных с точки зрения решаемой задачи. Любой объект описания может быть рассмотрен в трех разных аспектах [19]:

1. Структура. Объект характеризуется как внутренней структурой, так и внешней структурой – отношениями с другими объектами в информационной системе. Чем больше объект структурирован – внутренне и внешне, – тем объект проще находить, проще им манипулировать и определять соответствие другим объектам.
2. Контекст. Контекст является внешним по отношению к объекту свойством и определяется тем, кто, зачем, когда и как создал этот объект. Контекст позволяет идентифицировать объект среди множества других объектов.
3. Контент. Объект создается в информационной системе для предоставления пользователям необходимой информации, и эта информация передается через информационное содержание объекта – контент.

Метаданные могут описывать один или более аспектов объекта из трех приведенных выше. В рамках семантических технологий наибольшее внимание уделяется исследованию таких метаданных, которые описывают *контекст* и *контент* объекта.

В данном диссертационном исследовании используется следующее определение семантических метаданных.

Определение 1.2. Семантические метаданные – метаданные, описывающие контекст и/или контент объекта в информационной системе с помощью понятий предметной области, определенных на некотором языке описания онтологии.

Пусть $Q = \{q_1,...,q_k\}$ – конечное множество объектов ИС. Тогда, используя определение 1.1 онтологии O (параграф 1.2.1.1), определим семантические метаданные для объекта $q_i \in Q$ как конечное множество $MD(q_i)$, содержащее упорядоченные пары (c_{ij}, k_{ij}).

$MD(q_i) = \{(c_{i1}, k_{i1}),...,(c_{in}, k_{in})\}$, где (1.2)

$c_{in} \in C$ – понятия из онтологии, относящиеся к объекту описания q_i,

$k_{in} \in (0;1]$ – коэффициент, обозначающий релевантность понятия c_{in} объекту q_i.

Семантические метаданные, как и классические метаданные, логически неразрывно связаны с описываемым информационным объектом. Кроме этого, семантические метаданные не могут быть созданы без существующего описания знаний предметной области на каком-либо языке представления знаний. Семантические метаданные позволяют:

1) устранить лексическую многозначность терминов, используемых для описания информационных объектов;
2) определять соответствие между различными информационными объектами, используя онтологию.

1.2.2.2. Структура и языки описания семантических метаданных

Развитие и стандартизация семантических метаданных является важным этапом на пути развития и распространения семантических технологий. В этом отношении можно выделить два основных направления исследований: исследования в области *структуры* семантических метаданных и исследования в области *языков описания* семантических метаданных.

В настоящее время исследования в области *структуры* семантических метаданных связаны с различными сферами применения семантических метаданных. То есть структура разрабатывается с учетом конкретных типов описываемых объектов.

Наиболее универсальной структурой семантических метаданных является стандарт Dublin Core [19]. С помощью данной структуры возможно описание разнотипных объектов. Все атрибуты, составляющие структуру метаданных Dublin Core (DC) можно разделить на три группы: семантические атрибуты, права собственности и системные атрибуты (табл. 1.1). Для каждого

атрибута задано его назначение, которое должно строго учитываться при использовании этой структуры.

Таблица 1.1. Группы атрибутов метаданных Dublin Core

Семантические атрибуты	**Права собственности**	**Системные атрибуты**
title	creator	date
subject	publisher	format
description	contributor	identifier
type	rights	language
source		
relation		
coverage		

В сочетании с контролируемым словарем (в качестве которого может выступать онтология) структура метаданных DC является достаточной для описания широкого спектра объектов. Однако иногда необходимо более подробное описание объекта, и в настоящее время разработано множество стандартов структуры – LOM [59], MARC [60], GILS [61], GELOS [62] и других – для конкретных областей применения семантических метаданных. Например, стандарт Learning Object Metadata (LOM) дополняет структуру DC атрибутами для детального описания объектов, хранящихся в обучающих информационных системах [19].

Как было сказано выше, семантические метаданные описывают контекст и/или контент объекта. В связи с этим необходимо констатировать, что все перечисленные структуры метаданных предназначены для описания контекста объекта и не предназначены для его описания с точки зрения контента. Описание контента наиболее важно для тех информационных систем, которые реализуют функции полнотекстовой обработки информации. Однако исследованиям семантических метаданных, описывающих контент, не уделяется должного внимания, что сдерживает практическую реализацию полнотекстовой обработки информации с учетом ее семантики. Явное упоминание о необходимости разработки структуры таких семантических метаданных содержится лишь в [63]. Ввиду важности и недостаточной проработанности этого вопроса в данной работе он исследуется подробно.

Исследование и разработка *языков описания* семантических метаданных ведутся относительно независимо от исследований в области их структуры. Это стало возможным благодаря тому, что требования к структуре метаданных регламентированы. На сегодняшний день наиболее распространенным и уже стандартизованным языком описания семантических метаданных является язык RDF [64]. Универсальность языка RDF, основанного на триплетах, позволяет описывать метаданные с любой структурой. Язык RDF удовлетворяет всем требованиям к структуре метаданных:

- *Структура метаданных должна иметь конечный набор атрибутов.* Язык RDF позволяет формировать собственный перечень атрибутов или импортировать его из других RDF-совместимых описаний.
- *Каждый атрибут должен иметь имя.* RDF основан на стандарте URI, что позволяет именовать атрибуты с сохранением уникальности имен.
- *Каждый атрибут должен предназначаться для описания определенного аспекта объекта.* С помощью языка RDF смысл атрибута не выражается. Смысл атрибута закрепляется вербально авторами структуры семантических метаданных, и он должен быть учтен при заполнении метаданных. Если атрибут является импортированным, то его смысл не должен оставаться неизменным. Если создается новый атрибут, то его смысл вербально регламентируется.
- *Каждый атрибут может иметь несколько значений.* Синтаксис языка RDF позволяет использовать атрибут в нескольких триплетах, что позволяет описывать несколько значений.

Использование языка RDF для описания семантических метаданных возможно, если модель предметной области описана на языке, основанном на XML-синтаксисе. Это ограничение на использование языка RDF вытекает из необходимости совпадения схем именования элементов, как в языке описа-

ния модели, так и в языке описания семантических метаданных. На практике наиболее часто используемый тандем «онтология – метаданные» реализуется в виде «OWL – RDF» или «RDF(S) – RDF».

1.3. Анализ существующих применений семантических технологий в порталах

На сегодняшний день выполнены или выполняются исследовательские и коммерческие проекты, в рамках которых разрабатываются порталы, использующие семантические технологии. Данные порталы получили название *семантических порталов* (СП). В таких порталах семантические технологии применяются для реализации различных функций. Далее рассматривается ряд наиболее широко опубликованных проектов внедрения семантических технологий в порталы. Некоторые дополнительные характеристики данных проектов приведены в приложении 2.

1. В проекте **OntoPortal** [17] разработан метод использования семантических технологий для тематических порталов в сфере образования. Портал такого рода должен содержать ссылки на ресурсы в сети Интернет, из которых обучаемый может почерпнуть информацию по определенной тематике. Для создания портала необходимо:

- описать онтологию требуемой предметной области;
- описать с использованием онтологии ресурсы в сети Интернет, которые содержат информацию по требуемой тематике.

На основании онтологии предметной области автоматически генерируются страницы портала. Сгенерированный портал содержит страницы для редактирования онтологии, страницы навигации по онтологии, т.е. для просмотра понятий и описаний ресурсов, а также для перехода между ними. При добавлении описаний ресурсов в онтологию повторной генерации страниц портала не требуется.

В данном проекте основной решаемой задачей является создание удобной для пользователя навигационной структуры портала. Плюсом предложенного в проекте подхода является то, что обучаемому вся необходимая информация предоставляется в одном месте и в удобном виде, и он не тратит время на поиск и структуризацию информации. Минусом является необходимость ручного поиска подходящих ресурсов в сети Интернет, а также использование собственного, не стандартизованного языка описания онтологии, основанного на XML.

2. В рамках проекта **Semantic Community Web Portal** [15] разработан подход к созданию портала для сообщества пользователей. Такой портал обеспечивает пользователю возможность размещать информацию в виде различных объектов (документов, сообщений и т.п.) и предоставлять другим пользователям доступ к ней. С помощью семантических технологий в портале реализуются основные функции портала – интеграция разнородных источников информации, структуризация информации и поиск. В качестве языка описания онтологии используется язык F-Logic, основанный на логическом формализме. Он позволяет описывать в онтологии экземпляры понятий. На основании описания онтологии автоматически генерируются страницы портала.

Плюсом предложенного подхода является комплексность решения основных задач портала с помощью семантических технологий. Кроме этого в ходе проекта была выполнена практическая реализация предлагаемого подхода, основным компонентом которой является программная среда по работе с онтологиями – KAON.

В качестве основного недостатка подхода необходимо отметить отсутствие возможности семантического описания объектов портала с точки зрения контента. В данном подходе под семантическими метаданными объекта понимается экземпляр понятия в совокупности с его отношениями с другими экземплярами предметной области. Это соответствует описанию объекта с

точки зрения контекста. Кроме этого недостатком является необходимость повторной генерации страниц портала при добавлении понятий в онтологию.

3. Последнего недостатка лишен подход к созданию семантического портала, разработанный в рамках проекта **ODESeW** [18]. Однако в отличие от предыдущего проекта в нем не рассматриваются вопросы интеграции информационных источников с помощью семантических технологий.

Для создания семантических порталов в проекте ODESeW реализована среда WebODE, которая предназначена для организации работы с онтологиями. Онтологии описываются на специально разработанном языке, но этот недостаток компенсируется возможностью импорта описаний онтологий на распространенных языках, таких как OWL, DAML+OIL, RDF(S). Под семантическими метаданными объекта также понимается экземпляр понятия в совокупности с его отношениями с другими экземплярами предметной области.

4. Наиболее успешное практическое внедрение получил семантический портал (http://museosuomi.cs.helsinki.fi), разработанный в рамках проекта **OntoViews** [65]. Семантические технологии используются в проекте для решения уже упоминавшихся задач интеграции, структуризации и поиска информации в порталах. Подход отличается используемыми семантическими технологиями: язык RDF(S) для описания онтологий, язык RDF для описания семантических метаданных, Prolog для выполнения логического вывода.

Семантические метаданные в виде экземпляров онтологии используются не только для стандартного поиска по шаблону, но и предложена новая концепция многоаспектного поиска. Кроме этого новым является использование семантических технологий для реализации функции формирования рекомендаций пользователю портала.

5. Задачей проекта **OntoWeaver** [66] является разработка методологии по созданию порталов, настраиваемых под потребности пользователей. В основе методологии лежит моделирование различных аспектов портала с использованием набора разработанных онтологий. Этап проектирования порта-

ла поддерживается разработанным в рамках проекта программным комплексом.

Онтологии (метамодели) описывают понятия и отношения, которые используются при спецификации аспектов портала (модели). Разделение процесса моделирования на ряд независимых процессов – моделирование структуры данных, моделирование структуры портала, моделирование представления портала, моделирование пользователя – позволяет достичь возможности настройки портала под потребности пользователя на этапе проектирования с возможностью последующей перенастройки.

В данном проекте семантические технологии используются на этапе проектирования портала. Вопросы описания семантики объектов портала затронуты не были.

6. Проект **SPortS** [67] посвящен проблеме интеграции Web-сервисов в порталы. В частности авторами рассматриваются следующие вопросы:

- использование порталом сторонних Web-сервисов, как источников информации;
- рекомендация пользователю сервисов на основании анализа действий пользователя в рамках портала;
- предоставление функциональность портала в виде Web-сервисов.

В рамках проекта разработан подход к автоматической генерации портала на основании декларативного описания портала и интегрируемых сервисов. Процесс создания портала состоит из двух этапов: наполнение онтологий и генерация портала. В целях обеспечения функциональности портала в рамках проекта разработано четыре онтологии:

- онтология предметной области предназначена для описания информации, доступной в рамках портала;
- онтология сервисов описывает на языке OWL-S те сервисы, которые будут интегрированы в портал;
- онтология портала предназначена для моделирования аспектов портала;

- метаонтология предоставляет примитивы для описания трех предыдущих онтологий.

Объектами описания в портале являются доступные Web-сервисы. В качестве описывающих их семантических метаданных выступают экземпляры понятий онтологии OWL-S. Подход к поиску необходимых сервисов аналогичен рассмотренным ранее и является поиском по шаблону.

7. Примером коммерческого проекта по созданию портала, использующего семантические технологии, является **Mondeca ITM** [68]. Созданный в рамках проекта семантический портал является порталом управления знаниями и расширяет функциональность традиционных систем управления содержанием на основе онтологии и тематической карты.

Центральным элементом портала является репозитарий, хранящий модели знаний и структуру информационного содержания. Онтология, описанная на языке OWL, моделирует понятия и отношения (тема, связь, роль), используемые при построении тематической карты. Такое описание создается для явного разделения элементов тематической карты на классы, так как в самой карте такой возможности нет [69]. Тематическая карта описывается на языке XTM и используется для описания документов и баз данных.

Значимым отличием от уже рассмотренных проектов является наличие подсистемы автоматического анализа текста для составления семантических метаданных.

Семантические технологии также используются для реализации функций просмотра и поиска. Просмотр основан на тематической карте и возможен как в текстовом, так и в графическом режиме. Инструмент составления поисковых запросов позволяет детально описать искомый объект с точки зрения его атрибутов (название, тип и т.п.) и контента. Он использует как онтологию, так и тематическую карту.

Рассмотренный перечень проектов не является полным в силу того, что исследования по данной тематике активно ведутся. Кроме указанных суще-

ствуют и другие проекты ([16, 70-73] и пр.), однако они схожи по решаемым задачам и полученным результатам с рассмотренными проектами.

В результате анализа существующих подходов к использованию семантических технологий в порталах необходимо констатировать следующее:

- исследования в области семантических технологий и их внедрения в порталы в настоящее время активно ведутся;
- подходы к использованию семантических технологий в порталах варьируются, что, с одной стороны говорит об их большом потенциале, а с другой стороны, об отсутствии единого взгляда на проблему;
- в области семантических технологий наработано множество различных программных инструментов, и стандартов.

В большинстве подходов для описания онтологии используется язык RDF(S), а для описания семантических метаданных – язык RDF. Онтология чаще всего используется для целей структуризации и поиска информации. Причем поиск реализуется на основании просмотра экземпляров онтологии с возможность фильтрации по шаблону. То есть под составлением семантических метаданных понимается описание контекста объектов портала (отношений с другими объектами), выражающееся в наполнении онтологии экземплярами и отношениями между экземплярами. Как следствие, результатом поиска является список экземпляров, а не список документов (и прочих объектов). Если же экземпляры имеют ссылки на документы, то семантические описания контента таких документов отсутствуют, а, следовательно, невозможен и семантический поиск по текстовому содержанию документов. Исключением является подход в проекте Mondeca ITM (описывается контент документов), но используемая в качестве модели знаний тематическая карта ограничивает выразительность описания объектов.

Выводы по главе

1. Используемые в традиционных порталах технологии не позволяют обрабатывать информацию с учетом ее семантики, что повышает трудоемкость поиска и категоризации информации, снижает качество обработки в условиях постоянного роста объема и темпов обновления информации.

2. Анализ существующих подходов к созданию традиционных порталов показывает, что их архитектура может быть описана в обобщенном виде. Архитектура современных порталов включает три уровня: уровень данных, уровень бизнес-логики и уровень интерфейса.

3. Использование семантических технологий в порталах позволит качественно улучшить их работу. В качестве модели представления знаний в семантических технологиях растущее применение находят онтологии предметных областей.

4. В результате анализа отечественных и зарубежных научных и коммерческих проектов выявлена недостаточность исследований в области использования семантических технологий для создания порталов нового поколения – семантических порталов. Поэтому целью диссертационного исследования является разработка методов, позволяющих использовать семантические технологии в порталах для реализации информационных процессов в них с учетом семантики контента объектов.

Глава 2. Исследование и разработка семантического ядра портала

2.1. Анализ вариантов использования онтологии

Семантические порталы (СП) развивают концепцию создания Semantic Web [43]. Они должны реализовывать информационные процессы – сбор, обработку, накопление, хранение, поиск и распространение – с учетом семантики информации. Предполагается, что состоящая из таких семантических узлов (сайтов и порталов) сеть Интернет, будет предоставлять пользователям информационные услуги на новом, семантическом уровне. В основе концепции Semantic Web лежит онтология как средство описания семантики информационных ресурсов.

В настоящее время существует ряд проблем в области использования онтологий, которые нужно учитывать при разработке семантических порталов:

Проблема 1. Невозможность *автоматического* определения соответствия двух произвольных онтологий. Эта проблема возникает при интеграции информации из различных источников. В качестве преодоления данного ограничения в [74] предлагается создавать онтологии с использованием общих онтологий верхнего уровня.

Проблема 2. Определение соответствия *противоречивых* онтологий, то есть такая ситуация, когда описание понятия из одной онтологии противоречит описанию понятия из другой онтологии. Решение этой проблемы возможно на основе использования «эпистемологического сдвига» [75], реализации которого для онтологического подхода пока не предложено [76].

Проблема 3. При использовании языка описания онтологий, основанного на логическом формализме, возникает проблема *производительности* программного обеспечения (ПО) для выполнения логического вывода на онтологии. Имеет место как минимум обратная полиномиальная зависимость между производительностью ПО и

количеством логических высказываний в онтологии. Необходимо искать компромисс между детальностью онтологии и производительностью ПО. В настоящее время такой компромисс чаще всего достигается экспериментальным путем.

В связи с указанными проблемами разработка семантических порталов ведется с соблюдением следующих ограничений [22]:

Ограничение 1 (позволяет обойти проблемы 1 и 2). Для описания информационных ресурсов содержащихся как в самом портале, так и во внешних источниках используется единая онтология (или набор онтологий), по которой достигнуто соглашение всех заинтересованных лиц – разработчиков и пользователей. То есть процесс создания и ведения онтологии контролируем, в отличие от сети Интернет в целом. Это позволяет в рамках одного портала стандартизировать процесс создания онтологий на уровне языков описания и форматов хранения. Даже при условии моделирования предметной области в виде множества объединяемых онтологий проблема терминологической дивергенции решается путем создания общего словаря терминов с их описанием и правилами использования. Для решения проблем, возникающих при необходимости изменения онтологии, применим подход версий, предложенный в [76]. При возникновении несовместимых версий необходимо совершить переход к новой версии с отказом от старых, несовместимых версий и произвести повторную верификацию информационных ресурсов.

Ограничение 2 (позволяет обойти проблему 3). Создаваемые для использования в портале онтологии неполно охватывают содержание информационных ресурсов. Такие онтологии не являются семантической копией обрабатываемой информации, а отражают лишь те аспекты, которые существенны для решения конкретных задач в рамках портала. Из-за этого размер онтологии, описываемой с использованием логических формализмов, может быть скорректи-

рован для достижения приемлемой производительности системы логического вывода.

Как показал анализ, выполненный в параграфе 1.3, в существующих подходах к созданию семантических порталов предлагаются различные варианты использования онтологии. Помимо этих подходов к созданию порталов существуют и отдельные исследования в области использования онтологий в рамках информационных систем (ИС). В связи с этим, в дополнение к анализу порталов, был проведен анализ вариантов использования онтологии в ИС в общем по обширным литературным источникам [63, 70, 74, 77-89]. Были выявлены и обобщены следующие варианты использования онтологии:

- **Проектирование компонентов ИС** [74]. В распоряжении проектировщика должна быть библиотека онтологий, содержащая онтологии предметных областей и онтологии задач [90]. При условии корректности онтологий их содержание может быть использовано при проектировании компонентов ИС. Необходимые части онтологий извлекаются из библиотеки и преобразуются в описание компонента. При таком подходе снижаются затраты на концептуальный анализ, всегда имеющий место при проектировании ИС. Результаты концептуального анализа фиксируются в онтологии, которая затем повторно используется. Таким образом, анализ проводится один раз, а его результаты используются многократно. Помимо проектирования компонентов ИС, онтология может быть использована в процессе реинжиниринга ИС [77]. Кроме этого, онтологии используются как метамодели, описывающие примитивы для моделирования различных аспектов ИС [66, 70]. На основании созданной модели генерируются компоненты ИС.
- **Проектирование схемы базы данных.** Как и в случае с компонентами ИС, онтология может быть использована для преобразования ее в схему базы данных с уменьшением затрат на концептуальный анализ и моделирование. В [78] рассматривается метод трансформации онтологии в схемы различных типов баз данных – реляционных, объектных,

дедуктивных. Вариант интеграции множества реляционных баз данных в единое хранилище данных (data warehouse) с использованием онтологии предложен в [79]. Единая концептуальная схема, необходимая для интеграции, строится путем установления соответствий между онтологией и схемами баз данных.

- **Проектирование пользовательского интерфейса ИС.** При наличии явно выраженной онтологии ИС можно использовать ее при проектировании экранных форм ИС. Пример такого подхода можно найти в [70].
- **Интеллектуальная интеграция информации.** Интеллектуальная интеграция информации из различных информационных источников основана на использовании технологии посредников (mediator). Для каждого информационного источника существует свой посредник, который предоставляет информацию о схеме данных в виде онтологии и способен трансформировать поисковые запросы к нужному формату. Это позволяет осуществлять интеграцию информации не на этапе проектирования, а во время функционирования ИС. Информационные источники могут быть как структурированными (базы данных) [80], так и слабо структурированными (документы, Web-страницы и т.п.) [81]. Также возможно описание разнородных информационных источников в терминах одной онтологии, что упрощает процесс интеграции информации [15, 16].
- **Обмен информацией между программными агентами.** В мультиагентной среде (http://www.fipa.org/) агенты могут быть использованы как носители информации в определенной предметной области. Один агент может находить других агентов, которые обладают нужной первому агенту информацией. Для этого каждый агент должен быть носителем информации двух типов – непосредственно информации из предметной области и информации о процессе обмена между агентами (так называемый «процесс объяснения»). В [82] рассматривается вари-

ант использования онтологии для описания обоих типов информации для агента.

- **Обмен информацией между ИС.** Две и более информационные системы, обрабатывающие семантически похожую информацию и обменивающиеся ею, могут использовать онтологию для автоматизации этого процесса. В [83] описаны условия применения онтологии и эффект от ее использования для указанной цели.
- **Описание объектов ИС**. Документы, звук, видео и другие объекты ИС могут быть описаны с помощью элементов онтологии. Получившееся описание является семантическими метаданными объекта, которые могут быть использованы для реализации различных функций ИС. Составление семантических метаданных является одной из основных задач в Semantic Web, и в этом направлении ведется большое число исследований. Предложены различные способы полуавтоматического описания объектов с использованием онтологии при наличии у них текстового содержания. В [84] предложен подход к описанию, позволяющий с помощью онтологии формировать списки простых предложений, близких по смыслу к содержанию текста. Пользователь должен устранять лексические неточности, выбирая из списка наиболее подходящие варианты. В [85] лексическая неоднозначность уменьшается за счет использования статистики повторяемости слов в сети Интернет (через Web-сервис поисковой системы Google). Можно утверждать, что на практике большинство ИС, использующих семантические технологии, реализуют тот или иной подход к полуавтоматическому описанию семантики объектов ИС.
- **Переформулирование поисковых запросов.** В поисковых системах, предоставляющих возможность поиска информации на основании набора слов, часто реализуется процедура переформулирования запроса. Цель этой процедуры – модификация исходного поискового запроса для улучшения показателей полноты (recall) и точности (precision) ре-

зультатов поиска с использованием поисковой системы. Использование онтологии в процессе переформулирования поискового запроса осуществляется путем модификации запроса на основании связей, существующих между понятиями в онтологии. Например, в [87] предложен метод расширения запроса (query expansion).

- **Семантический поиск.** Этот вариант использования онтологии предполагает, что все объекты, поиск которых возможен в рамках ИС, имеют семантические метаданные. Тогда поисковый запрос тоже должен быть представлен в виде элементов онтологии, и поиск объектов осуществляется на основании понятий с учетом отношений между понятиями. В [15-18] используется подход к семантическому поиску на основании просмотра онтологии. В [86] рассматривается многоаспектный поиск, который является разновидностью поиска через просмотр онтологии. Такой подход аналогичен поиску по ключевым словам (ограниченный словарь), но его возможности шире за счет использования таксономии. Поиск через просмотр обычно дополняется поиском по шаблону, формируемому на основании описания понятий в онтологии [15, 16, 18]. Другой подход к формированию запросов и процессу поиска представлен в [63, 67, 68]. Здесь запрос представляет собой произвольный набор понятий онтологии, а для поиска релевантных объектов используется процедура оценки соответствия объекта запросу.
- **Описание профилей пользователей ИС.** Описание профилей пользователей с использованием элементов онтологии во многом схоже с описанием других объектов ИС. Существует ряд вариантов описания профиля пользователя. В [67] профиль пользователя описывает краткосрочные информационные предпочтения пользователя при работе с порталом. В [88] профиль пользователя ИС описывает области интересов пользователя. И эта информация затем используется ИС для уточнения поисковых запросов и при выборе интересных для пользователя объектов (рекомендации к ознакомлению). В [89] профиль отражает

уровень знаний пользователя в определенной области знаний. На этой информация основывается процедура поиска экспертов по определенным вопросам и проблемам. Создание программных систем, использующих онтологию для поиска экспертов – ключевых источников неявных знаний – является одной из приоритетных задач в управлении знаниями [91].

- **Формирование списка объектов ИС, связанных с исходным объектом.** Сетевая структура онтологии может быть использована в ИС для навигации по объектам. Например, пользователь просматривает документ. У документа есть автор, который рассматривается как объект, связанный с документом. Для этого в онтологии должны существовать понятия «документ» и «автор» и они должны быть связаны некоторым отношением. В [86] данный вариант использования онтологии реализован применительно к области искусства.
- **Формирование списка объектов ИС, похожих на исходный объект.** Если объекты в ИС описаны семантическими метаданными, то онтология может рассматриваться как пространство, в котором возможна оценка близости двух различных семантических метаданных. Такой подход используется в [67] для рекомендации пользователю некоторого Web-сервиса на основании сравнения описания пользователя с описаниями доступных Web-сервисов.
- **Семантическое связывание.** Структура онтологии может быть использована для динамической генерации навигационного меню Web-приложений и дополнительных ссылок между страницами Web-приложений, что рассматривается в [17].

Подход к систематизации вариантов использования онтологии предложен в [74]. В соответствии с ним использование онтологии в ИС варьируется в зависимости от этапа жизненного цикла ИС, на котором применяется онтология, и от уровня ИС, на котором применяется онтология.

1. В основных этапах жизненного цикла ИС имеет место:

1.1. Использование онтологии на этапе проектирования ИС;

1.2. Использования онтологии в процессе функционирования ИС;

1.3. Использование онтологии в процессе развития ИС.

2. По уровням ИС имеет место:

2.1. Использование онтологии на уровне интерфейса пользователя;

2.2. Использование онтологии на уровне бизнес-логики;

2.3. Использование онтологии на уровне информационных ресурсов.

В таблице 2.1 отражено соотношение выявленных вариантов использования онтологии с данной систематизацией.

Таблица 2.1. Систематизация выявленных вариантов использования онтологии

По этапам ЖЦ ИС / **По уровням ИС**	**Проектирование**	**Функционирование**	**Развитие**
Интерфейс	• Проектирование пользовательского интерфейса ИС	• Семантическое связывание	
Бизнес-логика	• Проектирование компонентов ИС	• Обмен информацией между программными агентами • Обмен информацией между ИС • Переформулирование поисковых запросов • Семантический поиск • Формирование списка объектов ИС, связанных с исходным объектом • Формирование списка объектов ИС, похожих на исходный объект	• Реинжиниринг компонентов ИС
Информационные ресурсы	• Проектирование схемы базы данных	• Интеллектуальная интеграция информации • Описание объектов ИС • Описание профилей пользователей ИС	

Анализ показал, что понимание роли семантических технологий в ИС у большинства специалистов совпадает. Это совершенствование функций ИС по структуризации и предоставлению информации пользователям. Большинство вариантов использование онтологии нацелено на решение этих задач. Однако выбираемые исследователями подходы и методы варьируются.

Предлагаемые в данном диссертационном исследовании методы по работе с семантикой объектов объединены в *семантическое ядро портала* [21].

Определение 2.1. Семантическое ядро портала – это компонент или набор компонентов, которые реализуют функции, используемые порталом и позволяющие ему учитывать семантику обрабатываемой информации.

2.2. Место и функции семантического ядра портала

Обобщая мировой опыт создания семантических порталов, можно говорить об определенной иерархии шагов построения семантического портала [92]. Данная иерархия может быть графически представлена в виде пирамиды (рис. 2.1), уровни которой отражают шаги по созданию семантического портала. Реализация любого из представленных уровней возможна только после реализации нижележащих уровней.

Рис. 2.1. Шаги построения семантического портала

1. **Семантические технологии** являются основой для развития современных информационных систем, и в частности для создания семантических

порталов, использующих модель знаний предметной области для реализации интеллектуальных функций и предоставления разнообразных сервисов. В состав семантических технологий входят:

- формальные способы представления знаний;
- языки описания онтологий;
- языки описания семантических метаданных;
- инструментальные средства по работе с онтологиями и семантическими метаданными (создание, хранение, обработка);
- протоколы взаимодействия и обмена данными между программными системами, использующими семантические технологии.

2. **Инфраструктура семантического портала** расширена по сравнению с инфраструктурой традиционного портала (рис. 1.2) за счет использования следующих элементов:

- протоколов, языков и инструментальных средства, выбранных из множества доступных семантических технологий, и необходимых для решения поставленных задач;
- программных средств интеграции семантических технологий для удобства их использования в семантическом портале;
- методов и алгоритмов использования семантических технологий для реализации общесистемных функций семантического портала (описание объектов, поиск, категоризация и т.д.).

3. **Онтологии верхнего уровня** являются необходимым условием для обеспечения возможности интеграции создаваемого семантического портала в среду Semantic Web, то есть для взаимодействия с другими информационными системами, использующими семантические технологии.

4. **Онтологии приложения** [45] описывают знания о предметной области, в которой функционирует приложение, а также об объектах, которыми управляет приложение, и о процессах, автоматизируемых приложением. Если в портале используются онтологии верхнего уровня, то онтологии приложе-

ния должна дополнять и уточнять понятия, введенные в онтологиях верхнего уровня.

5. **Семантический портал** разрабатывается с использованием семантических и прикладных (программных, системных) технологий и реализует функции для решения поставленных задач.

Следует отметить, что исследования в области семантических технологий и создания онтологий верхнего уровня проводятся в большом числе стран и организаций и имеют значительные (в некоторой части даже стандартизованные) результаты. Результаты исследований в области инфраструктуры семантических порталов и онтологий приложения гораздо скромнее. Исследования в этих областях редки, слабо проанализированы и обобщены. Поэтому эти исследования являются актуальными и необходимыми.

Разработанное в данном диссертационном исследовании семантическое ядро портала (СЯП) можно отнести к уровню инфраструктуры семантического портала, потому что реализуемые в нем методы использования семантических технологий могут быть задействованы различными функциональными модулями портала для учета семантики обрабатываемой информации (рис 2.2). СЯП объединяет элементы семантических технологий (методы, протоколы, языки и инструментальные средства), расширяющие инфраструктуру портала (рис. 1.2) до инфраструктуры семантического портала.

При разработке СЯП учитывались два возможных сценария создания и внедрения семантического портала:

1. Семантический портал может разрабатываться на базе ранее внедренного традиционного портала. Тогда СЯП должно предоставлять возможность интеграции с существующим порталом с целью его развития до уровня работы с семантикой.
2. Семантический портал создается «с чистого листа». В этом случае требования к СЯП менее жесткие – оно должно решать поставленные задачи, а разработчики должны сами решать вопросы его интеграции.

Рис. 2.2. Инфраструктура и функциональные модули семантического портала

Рис. 2.3. Место семантического ядра в семантическом портале

Из приведенных сценариев внедрения следует, что СЯП не должно зависеть от платформы, на которой реализуется семантический портал. Поэтому семантическое ядро разрабатываться в виде набора необходимых серверов, доступ к которым осуществляется по открытым стандартным протоколам (рис. 2.3).

Как уже отмечалось, СЯП должно предоставить функции, позволяющие реализовать в портале информационные процессы, учитывающие семантику информации. В данном исследовании учет семантики информации не подразумевает детального семантического анализа текста. Переход на семантический уровень осуществляется за счет устранения синтаксической многозначности и учета связей между понятиями. То есть СЯП предоставляет следующие возможности:

1. Учет в процессе обработки информации наличия омонимии и полисемии в естественном языке. Это достигается за счет моделирования знаний предметной области в виде онтологии, которая содержит понятия, которые в свою очередь имеют множественные лексические представления. В результате появляется возможность выявления омонимов и многозначных слов в предметной области и возможность устранения неоднозначности текстового содержания информационного ресурса.
2. Учет в процессе обработки информации наличия эквивалентных лексических конструкций (синонимов) в естественном языке. Это достигается за счет закрепления за понятиями онтологии множественных лексических представлений – синонимов. В результате появляется возможность сравнения синтаксически различной, но семантически похожей информации.
3. Учет в процессе обработки информации иерархической природы понятий, выражающейся в виде отношения «понятие – более узкое понятие» («класс – подкласс»). Это достигается за счет использования таксономии понятий.

Эти возможности СЯП используются для структуризации и поиска информации в портале и предоставления ее пользователям. То есть в данном диссертационном исследовании развиваются существующие подходы к созданию семантических порталов, в которых семантические технологии используются на этапе функционирования портала. Разработанное семантическое ядро портала поддерживает следующие ранее выявленные варианты использования онтологии:

1. описание объектов портала;
2. семантический поиск;
3. формирование списка объектов, связанных с исходным объектом;
4. формирование списка объектов, похожих на исходный объект.

На перечисленных вариантах использования онтологии основываются разработанные методы обработки описаний объектов портала, позволяющие учитывать семантику контента. Предлагаемые методы подробно описаны в третьей главе. Для реализации этих методов СЯП предоставляет возможность:

- описывать предметные области в виде онтологии для выявления синонимов, омонимов, многозначных терминов и построения таксономии терминов;
- описывать объекты портала с использованием семантических метаданных разработанной автором структуры.

Функционально семантическое ядро портала разделено на два модуля:

- модуль по работе с онтологиями;
- модуль по работе с семантическими метаданными.

Функциональность семантического ядра по работе с онтологиями сосредоточена в **сервере онтологий**, а функциональность по работе с семантическими метаданными – в **сервере семантических метаданных** (рис. 2.4).

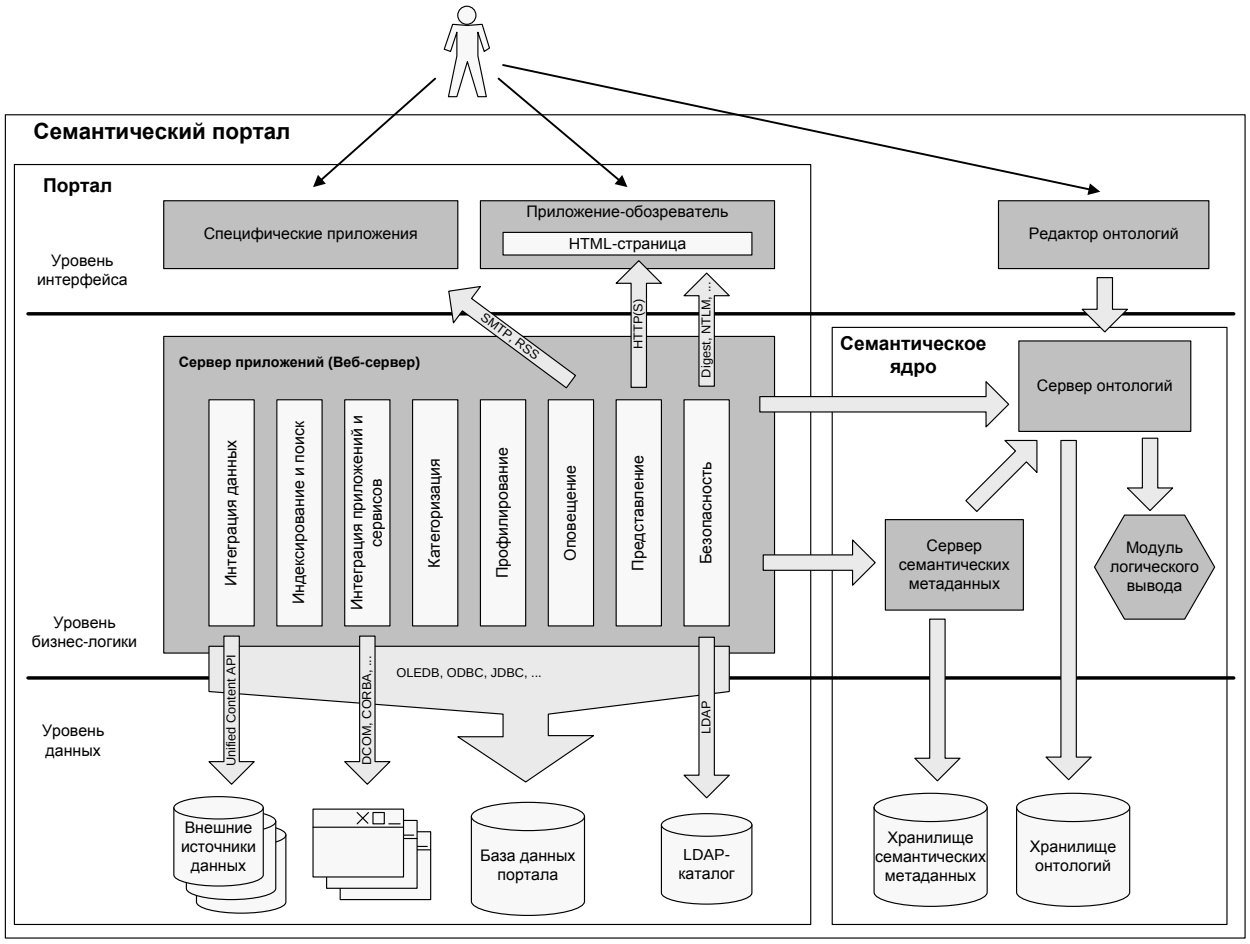


Рис. 2.4. Интеграция портала с семантическим ядром

2.3. Сервер онтологий

Сервер онтологий (СО) – это отдельно функционирующая программная система, хранящая множество онтологий и предоставляющая к ним доступ. Сервер онтологий используется другими программными системами, которым нужны его функции в процессе реализации жизненного цикла онтологии: создания, оценки, использования и последующей актуализации.

2.3.1. Выбор языка описания онтологии

Для описания модели знаний предметной области в виде онтологии необходимо выбрать язык описания онтологий. Как было показано в параграфе 1.2.1.3, существует множество таких языков. Для использования в сервере онтологий выбор осуществлялся между теми языками описания онтоло-

гий, в основе которых лежат логические формализмы. В таких языках онтология рассматривается как логическая теория и такой подход к онтологии имеет ряд преимуществ.

Во-первых, при таком подходе возможен анализ логического языка на достаточность его выразительных возможностей для формализации знаний в определенной предметной области. Одни логические языки являются более выразительными – позволяют формализовать сложные факты, другие – менее выразительные. Во-вторых, возможен анализ логического языка на разрешимость и вычислительную сложность. Этот фактор нужно рассматривать при программной реализации инструментов по работе с онтологией. Если для формализации предметной области в виде онтологии выбран неразрешимый логический язык, то невозможно создать надежный программный инструмент по обработке онтологии. Для разрешимых языков создание инструментария возможно, но для его функционирования могут потребоваться различные вычислительные ресурсы. В-третьих, на онтологии возможно применение формальных процедур логического вывода, позволяющих расширить множество формул онтологии. Логический вывод может быть использован для разных целей. Например, с его помощью можно проверить онтологию на непротиворечивость. В-четвертых, процедура формального логического вывода не зависит от предметной области, что позволит применять ее к онтологиям с любым содержанием.

В настоящее время существует три логических формализма, которые используются для целей описания онтологии:

- логика предикатов первого порядка (ЛПП) используется в языке Ontolingua [54];
- фреймовая логика (ФЛ) используется в языке F-Logic [55];
- дескриптивная логика (ДЛ) используется в языках DAML-ONT [93], OIL [94], DAML+OIL [95], OWL [58].

Для разработанного семантического ядра была выбрана дескриптивная логика [23, 24] в силу ее свойств, среди которых наиболее существенными являются (таблица 2.2):

- разрешимость;
- выразительная вариативность;
- автоматическая классификация понятий.

Таблица 2.2. Сравнение логических формализмов по избранным свойствам

Формализм / **Свойство**	**ЛПП**	**ФЛ**	**ДЛ**
Разрешимость	-	+	+
Выразительная вариативность	-	-	+
Автоматическая классификация понятий	-	-	+

Свойство *разрешимости* является важным потому, что при использовании формализма в рамках программной системы не должно возникать ситуаций, когда получение ответа от системы логического вывода невозможно, а, следовательно, невозможно и выполнение операций, основанных на логическом выводе. Разрешимость логического языка гарантирует получение ответа. Но в зависимости от вычислительной сложности логического языка на поиск ответа может быть потрачено различное количество времени.

Существует прямая зависимость между выразительной мощностью логического языка и его вычислительной ресурсоемкостью. Чем выразительнее язык, тем более точно можно описать предметную область с помощью этого языка, но и время, затраченное на логический вывод, будет значительным. Дескриптивная логика позволяет найти компромисс между выразительными потребностями и доступными вычислительными ресурсами. ДЛ представляет множество логических языков, обладающих различной выразительной мощностью. В зависимости от задачи можно выбрать язык с достаточной выразительностью и минимальной вычислительной ресурсоемкостью. В этом заключается свойство *выразительной вариативности* дескриптивной логики.

Свойство *автоматической классификации понятий* основано на логическом выводе и гарантирует, что для каждого понятия будет определено место в иерархии понятий (таксономии) исходя из описания понятия. Это свойство используется для построения таксономии понятий, на основе использования которой в данном диссертационном исследовании разработаны методы семантической обработки информации.

Из всех языков описания онтологии, использующих дескриптивную логику, был выбран OWL, так как он был специально разработан с учетом опыта использования других подобных языков и прошел процесс стандартизации в организации World Wide Web Consortium. Из трех подмножеств языка OWL, был выбран язык OWL DL, который позволяет наиболее полно использовать возможности дескриптивной логики, поддерживая ее выразительную разновидность – *SHIQ* [96].

2.3.2. Определение онтологии, основанной на дескриптивной логике

Дескриптивная логика – это семейство логических языков. Базовыми синтаксическими элементами языка могут выступать *атомарные понятия* (одноместные предикаты), *атомарные отношения* (двухместные предикаты) и *экземпляры* (константы). К ним могут применяться *конструкторы* языка для создания *комплексных понятий* и *комплексных отношений*.

Дескриптивным языком с минимальной выразительностью является *атрибутивный язык* (*AL*), синтаксис и семантика которого представлены в таблице 2.3. Семантика языка определяется с помощью интерпретации I, которая включает непустое множество Δ^I и функцию интерпретации, которая ставит в соответствие каждому атомарному понятию A множество $A^I \subseteq \Delta^I$ и каждому атомарному отношению R множество $R^I \subseteq \Delta^I \times \Delta^I$.

Таблица 2.3. Синтаксис и семантика атрибутивного языка

№	Название	Синтаксис	Семантика
1	Атомарные понятия	C, D → A	
2	Атомарное отношение	R	

3	Универсальное понятие	$\top$	Δ^I
4	Пустое понятие	$\bot$	$\varnothing$
5	Атомарное отрицание	$\neg A$	$\Delta^I \setminus A^I$
6	Пересечение	$C \sqcap D$	$C^I \cap D^I$
7	Ограничение на значение	$\forall R.C$	$\left\{a \in \Delta^I \mid \forall b.(a,b) \in R^I \rightarrow b \in C^I\right\}$
8	Ограниченная квантификация существования	$\exists R.\top$	$\left\{a \in \Delta^I \mid \exists b.(a,b) \in R^I\right\}$

Выразительность атрибутивного языка расширяется за счет введения дополнительных конструкторов с сохранением разрешимости получаемого языка. В таблице 2.4 приведены примеры расширения атрибутивного языка.

Таблица 2.4. Некоторые расширения атрибутивного языка

№	Название	Синтаксис	Семантика
1	Объединение	$C \sqcup D$	$C^I \cup D^I$
2	Полная квантификация существования	$\exists R.C$	$\left\{a \in \Delta^I \mid \exists b.(a,b) \in R^I \wedge b \in C^I\right\}$
3	Количественные ограничения на отношения	$\geq n\ R$ $\leq n\ R$	$\left\{a \in \Delta^I \left\| \{b \mid (a,b) \in R^I\}\right| \geq n\right\}$ $\left\{a \in \Delta^I \left\| \{b \mid (a,b) \in R^I\}\right| \leq n\right\}$
4	Произвольное отрицание	$\neg C$	$\Delta^I \setminus C^I$

База знаний, описанных с помощью дескриптивной логики, состоит из двух частей: *TBox* и *ABox*. *TBox* содержит описания понятий и отношений между понятиями. *ABox* содержит описания экземпляров понятий.

Процедура логического вывода применяется к *TBox* для:

- определения выполнимости описаний понятий (satisfiability);
- автоматической классификации понятий (subsumption).

Процедура логического вывода применяется к *ABox* для:

- определения выполнимости описаний экземпляров (consistency);
- заключения о том, относится ли экземпляр к понятию (instantiation).

С учетом свойств ДЛ, накладывающих ограничения на состав онтологии, в данном исследовании используется следующее определение онтологии, уточняющее определение 1.1.

Определение 2.2. Онтология, основанная на дескриптивной логике – это знаковая система

$$O_{DL} = \langle C, CD, R, A, I, V, R_I, A_I, L, P_C, P_R, P_A, P_{IC}, P_{LC}, P_{LR}, P_{LA}, P_{LI} \rangle, \quad (2.1)$$

в которой

$C = \{c_1,...,c_n\}$ – конечное множество понятий в онтологии,

$CD = \{cd_1,...,cd_t\}$ – множество стандартных типов данных, включающее два типа {string, integer},

$R = \{r_1,...,r_m\}$ – конечное множество бинарных отношений $r_i(c_x, c_y)$ между понятиями,

$A = \{a_1,...,a_w\}$ – конечное множество атрибутов, т.е. бинарных отношений $a_i(c_x, cd_y)$ между понятиями и стандартными типами данных,

$I = \{i_1,...,i_l\}$ – конечное множество экземпляров в онтологии,

$V = \{v_1,...,v_q\}$ – конечное множество конкретных значений стандартного типа,

$R_I = \{ri_1,...,ri_m\}$ – конечное множество конкретизированных отношений, т.е. бинарных отношений $ri_i(i_x, i_y)$ между экземплярами,

$A_I = \{ai_1,...,ai_w\}$ – конечное множество конкретизированных атрибутов, т.е. бинарных отношений $ai_i(i_x, v_y)$ между экземпляром и конкретными значениями,

$L = \{l_1,...,l_k\}$ – конечное множество лексических меток (словарь онтологии),

$P_C \subseteq C \times C, P_C \in R$ – антисимметричное, транзитивное, нерефлексивное бинарное отношение, являющееся отношением частичного порядка на множестве понятий C,

$P_R \subseteq R \times R$ – антисимметричное, транзитивное, нерефлексивное бинарное отношение, являющееся отношением частичного порядка на множестве отношений R,

$P_A \subseteq A \times A$ – антисимметричное, транзитивное, нерефлексивное бинарное отношение, являющееся отношением частичного порядка на множестве атрибутов A,

$P_{IC} \subseteq I \times C$ – бинарное отношение инцидентности между множествами I и C,

$P_{LC} \subseteq L \times C$ – бинарное отношение инцидентности между множествами L и C,

$P_{LR} \subseteq L \times R$ – бинарное отношение инцидентности между множествами L и R,

$P_{LA} \subseteq L \times A$ – бинарное отношение инцидентности между множествами L и A,

$P_{LI} \subseteq L \times I$ – бинарное отношение инцидентности между множествами L и I.

Данное определение онтологии O_{DL} используется в дальнейшем для описания предлагаемой структуры семантических метаданных и предлагаемых методов по работе с семантикой объектов портала.

2.3.3. Свойства языка OWL

Язык Ontology Web Language (OWL) стал результатом многолетних исследований и экспериментов в области языков описания онтологий. Сейчас он рассматривается [97] как основной язык для реализации концепции Semantic Web. Этот язык предназначен для использования в Интернет, поэтому его синтаксис должен быть основан на используемых в этой сети открытых стандартах. Зависимость языка OWL от других стандартов показана на (рис. 2.5).

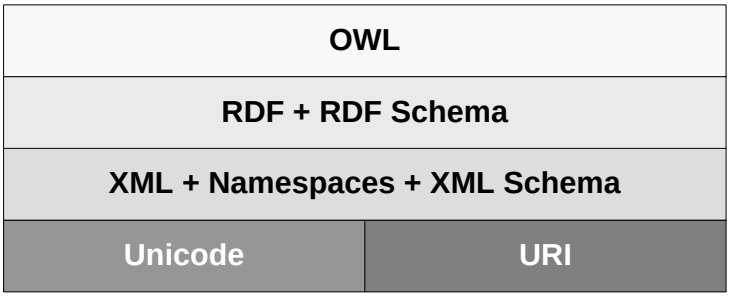


Рис. 2.5. Стандарты в основе языка OWL

Стандарт Unicode определяет способ кодирования символов из различных алфавитов для их передачи, обработки и отображения [98]. Он является основой для взаимодействия различных вычислительных устройств, позволяя им обмениваться информацией.

Стандарт URI (Uniform Resource Identifier) определяет способ уникальной идентификации и адресации документов, или в более общем случае, ресурсов в сети Интернет [99]. Другими словами, URI – это способ именования любых сущностей, который обеспечивает уникальность имени во всем мире и является удобным для представления в компьютере и удобным для передачи по сети.

Стандарт XML (eXtensible Markup Language) определяет синтаксис языка, позволяющего описывать любые данные в структурированном виде [100]. Так как структура XML-документа определяется вложенностью тегов, то документ созданный одной компьютерной программой, может быть обработан другой программой, если существует договоренность о структуре документа. То есть компьютер может обрабатывать структуру документа, но он не может определить семантику обрабатываемого документа. Структура XML-документа и именование тегов рассматриваются в дополнительных стандартах. Для задания договоренности о структуре XML-документа разработан стандарт XML Schema [101]. Для уникального именования тегов разработан стандарт XML Namespaces [102], основанный на стандарте URI. Язык XML широко используется как формат обмена данными. Также синтаксис языка XML используется для записи конструкций других языков, таких как RDF или OWL.

Стандарт RDF (Resource Description Framework) определяет язык, который предназначен для записи машиночитаемых метаданных об информации, находящейся на различных сайтах в Интернет [64]. Для описания информации используются триплеты «объект-атрибут-значение». Набор таких триплетов и составляет метаданные какого-либо ресурса в сети Интернет. Объект из одного триплета может выступать в качестве значения атрибута в другом триплете. Таким образом, метаданные графически могут быть представлены в виде ориентированного графа (рис. 2.6).

Фактически стандарт RDF состоит из двух частей: спецификации RDF-модели данных (ориентированные графы) и основанной на XML синтаксисе

записи этих моделей (RDF/XML). Описание модели данных является основной частью стандарта. Спецификация RDF/XML определяет набор имен тегов и атрибутов в пространстве имен «http://www.w3.org/1999/02/22-rdf-syntax-ns#», который позволяет описывать с помощью языка XML размеченные ориентированные графы.

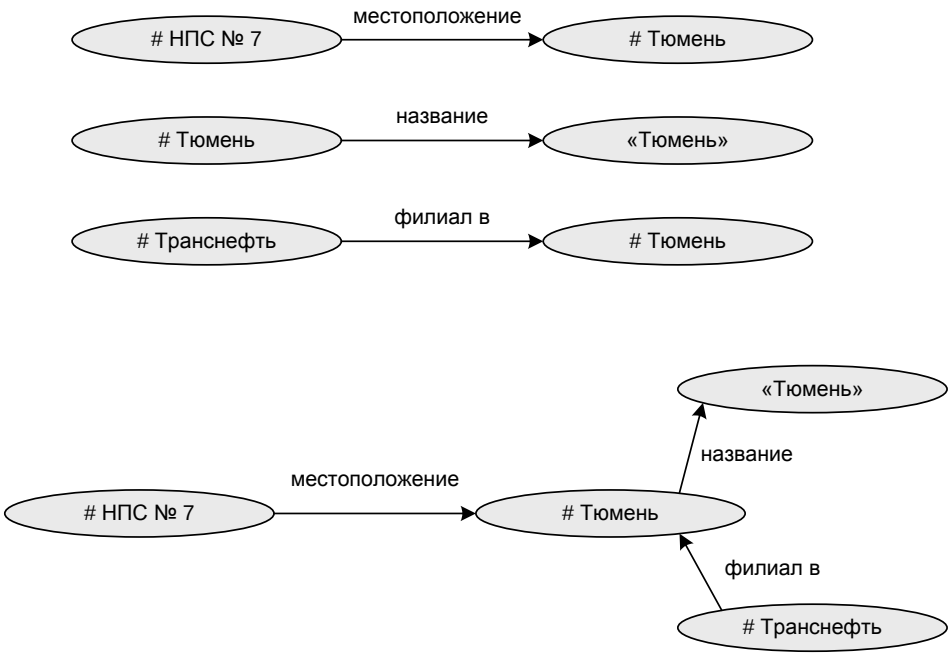


Рис. 2.6. Пример триплетов и RDF-графа на их основе

С помощью орграфа уже можно описать какую-либо предметную область и компьютер сможет ее обрабатывать. Но он не сможет сравнивать различные орграфы между собой, потому что в модели RDF все элементы – объекты, предикаты и значения – являются «ресурсами» и нет их разделения на типы, тогда как соотносить нужно однотипные элементы. Для решения этой проблемы разработан стандарт RDF Schema [57], который позволяет интерпретировать данные, описанные на языке RDF. Это осуществлено за счет предоставления возможности явного указания на то, какие ресурсы являются классами, а какие – отношениями. Кроме этого, существует возможность описывать иерархии классов и отношений (рис. 2.7). Обычно стандарты RDF

и RDF Schema (RDFS) используются совместно для записи метаданных какого-либо ресурса и обозначаются аббревиатурой RDF(S).

Выразительных возможностей языка RDF(S) достаточно, например, для описания онтологии типа «Таксономия» или «Тезаурус», поэтому этот язык считается также языком описания онтологий. Но его выразительности не достаточно для описания более сложных онтологий, что привело к разработке новых, специальных языков, таких как OWL.

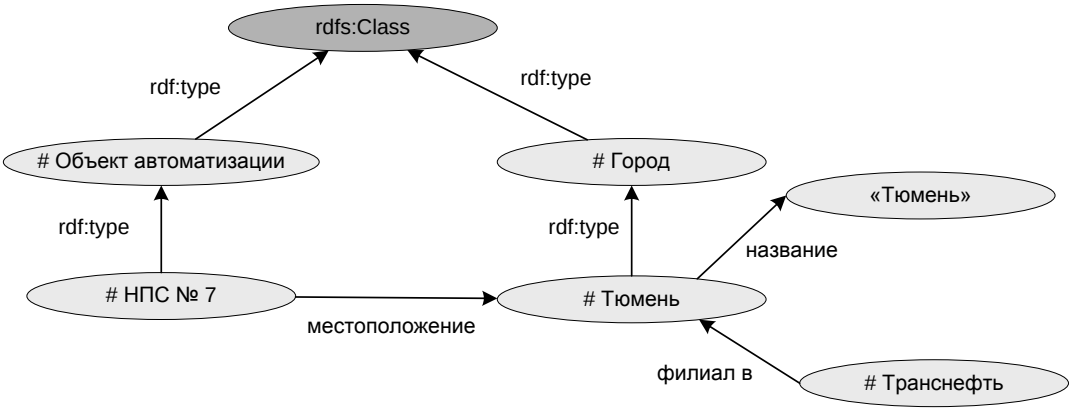


Рис. 2.7. Пример RDF(S)-графа

Язык OWL [58] превосходит RDF(S) по выразительности за счет наличия в нем дополнительных примитивов с заданной семантикой. Язык OWL основан на дескриптивной логике, но не ограничивается ей, поэтому язык OWL имеет несколько степеней выразительности. Он состоит из трех языков, которые приведены ниже по степени уменьшения выразительности:

- **OWL Full.** Полный язык называется OWL Full и использует все примитивы (базовые элементы) языка. Он также позволяет объединять эти примитивы произвольным способом с примитивами языка RDF(S). Преимущество OWL Full заключается в том, что он полностью совместим с RDF(S), как синтаксически, так и семантически: любой допустимый RDF(S)-документ также является допустимым документом языка OWL Full и любое верное логическое заключение для RDF(S) является также верным заклю-

чением для OWL Full. Недостатком языка OWL Full является отсутствие возможности преобразования высказываний языка в формулы дескриптивной логики, что не позволяет применять *разрешимые* процедуры логического вывода.

- **OWL DL**. Является подмножеством языка OWL Full, который ограничивает способ использования примитивов языков OWL и RDF(S), обеспечивая тем самым возможность использования дескриптивной логики. Ограничение заключается в запрещении применения примитивов друг к другу и таким образом, гарантирует, что язык соответствует хорошо изученной дескриптивной логике класса *SHIQ* [96]. Это позволяет обеспечить поддержку эффективного логического вывода. Недостатком является то, что в OWL DL отчасти теряется совместимость с RDF(S).
- **OWL Lite**. Является еще более ограниченным языком, чем OWL DL. Он содержит подмножество примитивов языка OWL DL. OWL Lite не содержит перечисляемые классы (enumerated classes), утверждения о непересекаемости (disjointness statements) и произвольные мощности (arbitrary cardinality). Преимущество этого языка, в том, что его проще изучить пользователю и проще реализовать разработчикам программных инструментов. Этот язык предназначен для решения таких задач, в которых необходимо построение таксономий и несложные ограничения.

Язык OWL получает все более широкое распространение в мире в силу своих свойств: градации выразительности, баланса выразительности и разрешимости, а также наличия программного инструментария поддержки работы с ним.

2.3.4. Функции и структура сервера онтологий

Для реализации в семантическом ядре выбранных вариантов использования онтологии сервер онтологий предоставляет следующие функции:

1. хранение онтологий;
2. предоставление онтологий;
3. логический вывод;
4. поиск в онтологии запрашиваемых понятий и отношений.

Функции СО могут быть сгруппированы по этапам жизненного цикла онтологии (таблица 2.5).

Таблица 2.5. Функции сервера онтологий на этапах жизненного цикла онтологии

Создание	**Оценка**	**Использование**	**Изменение**
• Логический вывод • Хранение онтологий • Предоставление онтологий	• Предоставление онтологий	• Предоставление онтологий • Логический вывод • Поиск понятий и отношений в онтологии	• Логический вывод • Хранение онтологий • Предоставление онтологий

Структура сервера онтологий приведена на рисунке 2.8.

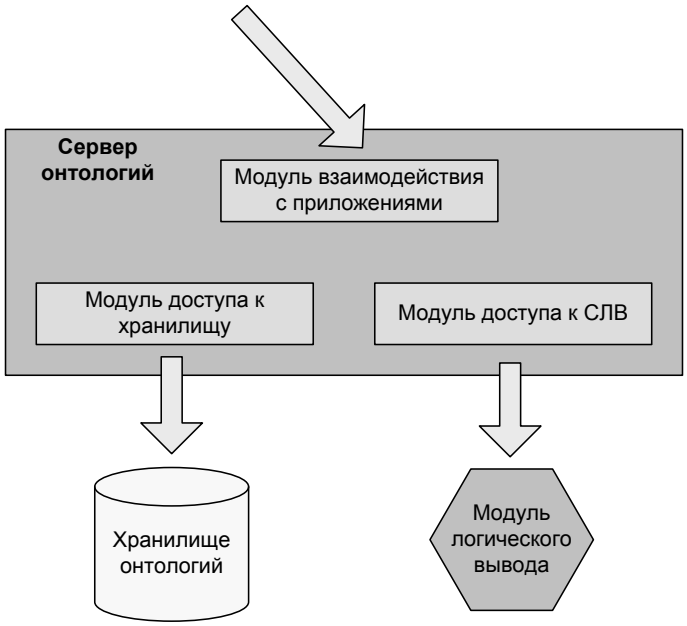


Рис. 2.8. Структура сервера онтологий

Модуль взаимодействия с приложениями предоставляет интерфейс доступа к функциям СО. Примером приложения, использующего сервер онтологии, является редактор онтологий.

Модуль доступа к хранилищу отвечает за взаимодействие с подсистемой хранения онтологий. Модуль реализует функции хранения и предоставления онтологий.

Модуль доступа к СЛВ предоставляет возможность использования формального логического вывода для функций обработки онтологии и выполнения специфических запросов. Этот модуль необходим в связи с тем, что онтология рассматривается как логическая теория и для работы с ней используется система логического вывода (СЛВ). Функции по обработке онтологии возлагаются на СЛВ, которая реализует алгоритмы логического вывода.

Таким образом, разработанный сервер онтологий, входящий в состав семантического ядра портала, хранит описания онтологий на языке OWL DL, предоставляет к ним доступ, проверяет их на правильность и обеспечивает выполнение специфических запросов к онтологии.

Логический вывод, использующий дескриптивную логику, позволяет решать следующие задачи:

- проверка комплексного понятия на непротиворечивость;
- проверка включения одного комплексного понятия в другое комплексное понятие, т.е. проверка наличия между понятиями отношения «класс-подкласс»;
- проверка двух комплексных понятий на пересечение;
- получение списка понятий, находящихся в таксономии на один уровень выше заданного комплексного понятия;
- получение списка понятий, находящихся в таксономии на один уровень ниже заданного комплексного понятия;
- получение списка понятий, находящихся в таксономии на любое количество уровней выше заданного комплексного понятия;

- получение списка понятий, находящихся в таксономии на любое количество уровней ниже заданного комплексного понятия;
- получение списка экземпляров, которые относятся к заданному комплексному понятию;
- получение списка понятий, к которым относится заданный экземпляр;
- проверка наличия отношения между заданным экземпляром I и комплексным понятием.

Указанные возможности СЛВ используются для проверки онтологии на непротиворечивость и выполнения запросов.

2.4. Сервер семантических метаданных

Сервер семантических метаданных (ССМ) – это отдельно функционирующая программная система, хранящая семантические метаданные, предоставляющая к ним доступ и обрабатывающая их.

2.4.1. Структура семантических метаданных

Используемая *структура* семантических метаданных для описания контента объектов портала идентична структуре, предложенной в [63].

Таблица. 2.6. Атрибуты семантических метаданных

Параметры / № Атрибута	Название	Описание	Возможные значения
1	Содержит знание о	Атрибут указывает на те элементы из онтологии, которые наиболее точно описывают семантику контента объекта	а) отдельные элементы онтологии б) триплеты элементов из онтологии

В структуре представлен один атрибут – *семантический атрибут*, – который описывает с помощью элементов онтологии контент объекта (табл. 2.6). Однако возможные значения данного атрибута отличаются от [63] и мо-

гут быть представлены не только с помощью единичных элементов онтологии, но и с помощью триплетов элементов (рис. 2.9).

Под единичным элементом понимается значение вида «субъект». Под триплетом понимается значение вида «субъект–предикат–объект». С учетом того, что в онтологии содержаться элементы различных типов (понятия, отношения и др.), то субъекты, предикаты и объекты в содержании семантического атрибута могут принимать различные значения.

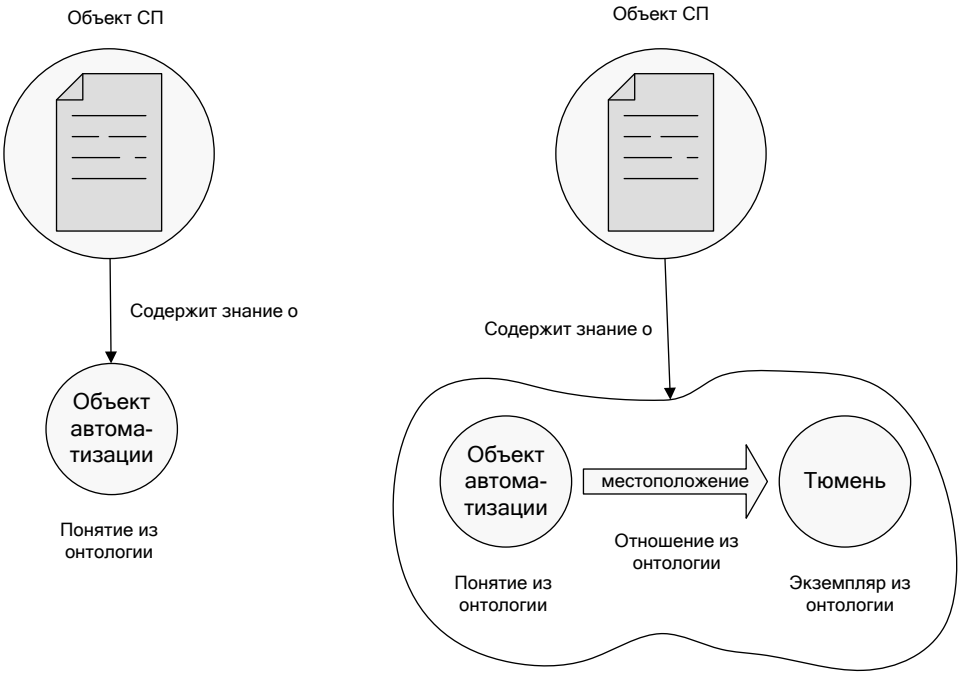


Рис. 2.9. Возможные значения семантического атрибута

В соответствии с определением 2.2 в онтологии содержатся:

- понятия (множество C);
- экземпляры понятий (множество I);
- конкретные значения (множество V);
- отношения (множество R);
- атрибуты (множество A).

Субъектом описания может выступать понятие или экземпляр. Предикатом может быть отношение или атрибут. Причем значением атрибута (объектом) может быть только конкретное значение, а значением отношения

(объектом) – понятие или экземпляр. Путем комбинирования всех возможных значений получаем следующий перечень вариантов значений семантического атрибута (таблица 2.7).

Таблица 2.7. Значения семантического атрибута

№	Возможные значения		
	Субъект	**Предикат**	**Объект**
1	c_x		
2	i_x		
3	c_x	r_y	c_z
4	c_x	r_y	i_z
5	i_x	r_y	c_z
6	i_x	r_y	i_z
7	c_x	a_y	v_z
8	i_x	a_y	v_z

Варианты 1 и 2 представляют собой единичные элементы из онтологии. Варианты 3-8 являются триплетами. Так как семантический атрибут может иметь множество значений и в структуре семантических метаданных является единственным, то понятия «содержание семантических метаданных» и «значение семантического атрибута» являются взаимозаменяемыми. Таким образом, содержание семантических метаданных представляет собой множество триплетов и множество единичных значений, которые вместе составляют элементы семантических метаданных, относящихся к объекту описания.

Онтология унифицировано описывает знания всей предметной области. Семантические метаданные унифицировано описывают те знания предметной области, которые содержаться в объекте портала. То есть семантические метаданные «связывают» объекты портала с частями предметной области. Исходя из того, что строительными блоками онтологии являются триплеты (семантика понятия определяется его отношениями с другими понятиями), то и использование триплетов в семантических метаданных позволяет наиболее точно отражать взаимосвязь объекта портала с частями предметной области.

Ясно, что с помощью триплетов невозможно выразить семантику контента объекта абсолютно точно. Триплеты не учитывают временных, модальных и прочих характеристик описываемой информации. Для этого нужны более сложные конструкции наподобие схем концептуального анализа Шенка [103]. Но у семантических метаданных другая функция. Они описывают семантику объекта с точки зрения контента, основываясь на онтологии. Следовательно, семантические метаданные не могут описать контент объекта точнее, чем это позволяет онтология. Чем точнее описана предметная область в онтологии, тем точнее можно описать контент объектов портала.

Далее приводится определение 2.3 для семантических метаданных $MD_{DL}(q_i)$, основанное на определении 2.2 онтологии O_{DL} и уточняющее определение 1.2 семантических метаданных $MD(q_i)$.

Определение 2.3. Семантическими метаданными с возможностью использования триплетов является структура вида

$$MD_{DL}(q_i) = MD_{SN}(q_i) \cup MD_{TR}(q_i). \quad (2.2)$$

Пусть $Q = \{q_1,...,q_k\}$ – конечное множество объектов семантического портала. Тогда семантические метаданные для объекта $q_i \in Q$ представляют собой объединение двух конечных множеств $MD_{SN}(q_i)$ и $MD_{TR}(q_i)$, содержащих упорядоченные пары (sn_{ij}, k_{ij}) и (tr_{if}, k_{if}) соответственно.

Тогда:

$$MD_{SN}(q_i) = \{(sn_{i1}, k_{i1}),...,(sn_{in}, k_{in})\} \quad (2.3)$$

$sn_{in} \in C \cup I$ – отдельный элемент онтологии (2.4)

$k_{in} \in (0;1]$ – коэффициент, обозначающий релевантность отдельного элемента sn_{in} объекту q_i

$$MD_{TR}(q_i) = \{(tr_{i1}, k_{i1}),...,(tr_{if}, k_{if})\} \quad (2.5)$$

$tr_{if} = \langle subj_{if}, pred_{if}, obj_{if} \rangle$ – триплет (2.6)

$subj_{if} \in C \cup I$ – субъект в триплете (2.7)

$pred_{if} \in R \cup A$ – предикат в триплете (2.8)

$obj_{if} \in C \cup I \cup V$ – объект в триплете (2.9)

$k_{if} \in (0;1]$ – коэффициент, обозначающий релевантность триплета tr_{if} объекту q_i.

Таким образом, семантические метаданные с возможностью использования триплетов представляются множеством упорядоченных пар $(smd_{i(n+f)}, k_{i(n+f)})$, где $smd_{i(n+f)}$ является отдельным элементом онтологии или триплетом.

$$MD_{DL}(q_i) = \{(smd_{i1}, k_{i1}),...,(smd_{i(n+f)}, k_{i(n+f)})\} \quad (2.10)$$

Данное определение семантических метаданных $MD_{DL}(q_i)$ используется в дальнейшем для описания методов обработки семантических метаданных.

2.4.2. Функции и структура сервера семантических метаданных

Для реализации выбранных вариантов использования онтологии сервер семантических метаданных, тесно взаимодействуя с сервером онтологий, предоставляет следующие функции:

1. формирование семантических метаданных;
2. хранение семантических метаданных;
3. предоставление семантических метаданных;
4. сравнение семантических метаданных.

Группировка функций ССМ по этапам жизненного цикла онтологии представлена в таблице 2.8.

Таблица 2.8. Функции сервера семантических метаданных в рамках жизненного цикла онтологии

Создание	Оценка	Использование	Изменение
		• Формирование семантических метаданных • Хранение семантических метаданных • Предоставление семантических метаданных • Сравнение семантических метаданных	

Структура ССМ может быть представлена аналогично структуре сервера онтологий (рис. 2.10).

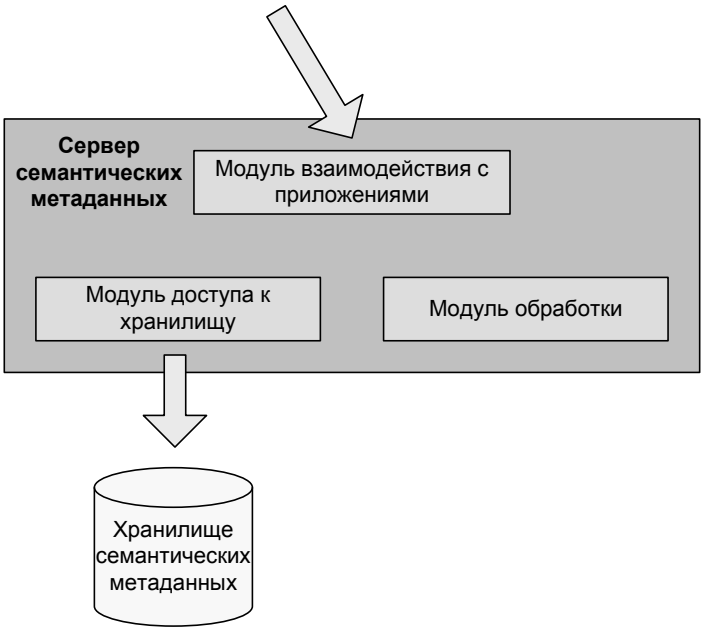


Рис. 2.10. Структура сервера семантических метаданных

Модуль взаимодействия с приложениями предоставляет интерфейс доступа к функциям ССМ.

Модуль доступа к хранилищу отвечает за взаимодействие с подсистемой хранения семантических метаданных. Модуль реализует функции хранения и предоставления семантических метаданных. Для записи семантических метаданных описанной выше структуры используется язык RDF [64]. Во-первых, он совместим с языком описания онтологий OWL DL. Во-вторых, его синтаксис позволяет делать высказывания относительно триплетов. Это свойство языка RDF называется «воплощением» (reification) и обеспечивает возможность связывания объектов с описывающими их триплетами.

Модуль обработки реализует функции формирования семантических метаданных с использованием онтологии и их семантического сравнения.

2.5. Использование семантического ядра портала

Созданные с помощью функций семантического ядра онтологии предметных областей и семантические метаданные объектов СП используются при обработке объектов. Как уже говорилось, семантическое ядро предоставляет функции:

- описания объектов портала;
- семантического поиска;
- формирования списка объектов, связанных с исходным объектом;
- формирования списка объектов, похожих на исходный объект.

Семантическое описание объекта портала является результатом *аннотирования* – процесса формирования семантических метаданных объекта. При аннотировании устанавливается релевантность элементов онтологии объекту портала. Семантические метаданные являются основой для реализации трех других функций семантического ядра.

Для выполнения *семантического поиска* объектов портала необходимо наличие семантических метаданных у объектов и представление поискового запроса пользователя в виде семантических метаданных – описания запроса. Поиск выполняется путем сравнения семантических метаданных с запросом. Объект считается релевантным запросу в том случае, когда в его семантических метаданных содержаться *все* элементы из запроса, или подклассы этих элементов. Таким образом, при поиске учитывается иерархия понятий предметной области.

Функция формирования списка объектов, связанных с исходным объектом в семантическом портале может иметь различное применение. В разработанном семантическом портале эта функция СЯ использовалась для выполнения категоризации – проверки соответствия объекта заданным категориям. Для выполнения категоризации объектов с учетом их семантики необходимо наличие семантических метаданных у объектов и у категорий, к которым нужно отнести объекты. Предполагается, что все множество возмож-

ных категорий будет иерархически упорядочено. Категоризация выполняется путем сравнения семантических метаданных объекта и категории. Объект считается относящимся к категории, если в его семантических метаданных содержаться *хотя бы некоторые* элементы из семантических метаданных категории, или подклассы этих элементов. Очевидно, что объект может быть отнесен к одной или более категории.

Функция формирования списка объектов, похожих на исходный объект также может иметь различное применение. В разработанном семантическом портале она использовалась для *формирования рекомендаций*. Формирование рекомендаций заключается в предоставлении пользователю множества объектов, семантически похожих на некоторый объект, фигурирующий в запросе. Выполнение поиска похожих объектов осуществляется на основании сравнения семантических метаданных объектов.

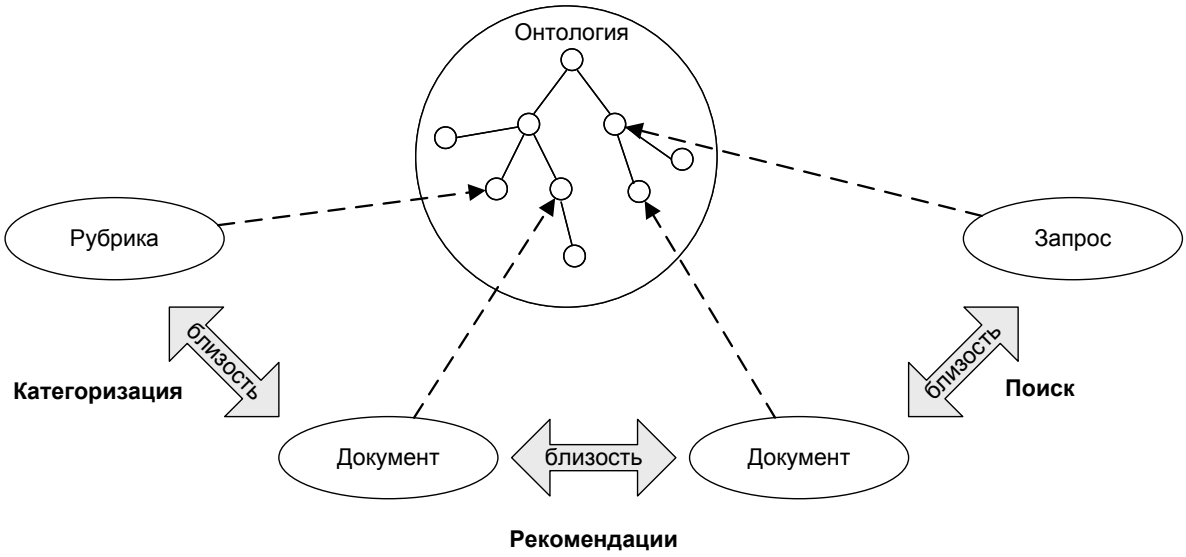


Рис. 2.11. Использование алгоритма оценки семантической близости

Для предоставления функций поиска, категоризации и формирования рекомендаций с учетом семантики объектов разработанное семантическое ядро реализует методы оценки близости семантических метаданных (рис. 2.11). *Метод вычисления близости семантических метаданных* учитывает особенности поиска, категоризации и формирования рекомендаций. Он осно-

ван на *методе вычисления семантической близости элементов онтологии*. Подробно указанные методы рассматриваются в третьей главе.

Выводы по главе

1. Анализ онтологического подхода к построению семантических порталов показывает, что для решения существующих проблем автоматического определения соответствия онтологий и выявления противоречий между онтологиями в порталах должен использоваться набор согласованных онтологий, а проблема производительности систем логического вывода решается путем комбинирования алгоритмического и логического подходов.

2. В качестве существенного шага в наделении современных порталов семантическими функциями является создание в его структуре новой важной подсистемы – семантического ядра.

3. Основными компонентами семантического ядра являются сервер онтологий и сервер семантических метаданных.

4. Поиск понятий, экземпляров и отношений, а также логический вывод, реализуемые сервером онтологий, основываются на использовании дескриптивной логики класса *SHIQ* и языка описания онтологий OWL DL.

5. Формирование, хранение, предоставление и сравнение семантических метаданных, реализуемые сервером семантических метаданных, основываются на предложенной структуре метаданных с использованием триплетов.

Глава 3. Разработка методов и алгоритмов для семантического ядра портала

3.1. Состав и структура онтологической модели для использования в семантическом портале

Процесс создания онтологической модели для ее использования в семантическом портале состоит из нескольких этапов (рис. 3.1).

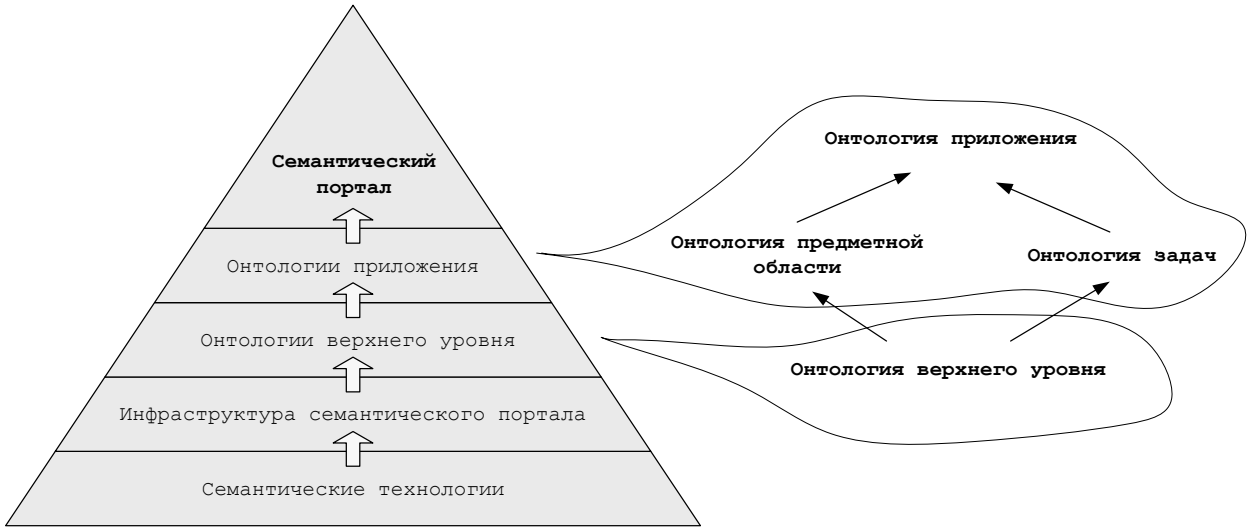


Рис. 3.1. Место процесса создания онтологической модели в процессе создания семантического портала

Сначала необходимо решить вопрос об использовании *онтологии верхнего уровня*. Данный тип онтологий предназначен для обеспечения возможности взаимодействия семантического портала с другими информационными системами – обмена информацией вместе с ее семантическим описанием. На данном этапе принимается решение об использовании или не использовании онтологии верхнего уровня. Если такую онтологию планируется использовать, то необходимо выбрать ее из ряда существующих онтологий верхнего уровня. В идеале должна существовать одна онтология верхнего уровня, которая будет использоваться во всех ИС. В связи с наличием различных точек зрения на процесс ее разработки и на ее состав и структуру, сегодня суще-

ствует ряд онтологий верхнего уровня. В качестве примера можно привести такие онтологии верхнего уровня как Sowa's KR Ontology [47], DOLCE [104], Generalized Upper Model [105], WordNet [48], Suggested Upper Merged Ontology [106] и др. Так как онтологии верхнего уровня создавались с использованием различных языков описания онтологий, то перед использованием выбранной онтологии верхнего уровня может потребоваться преобразование ее к языку, используемому в конкретном семантическом портале.

Онтология предметной области и *онтология задач* описывают соответственно сущности и процессы, которые имеют место в предметной области, и информация о которых обрабатывается семантическим порталом. Если используется онтология верхнего уровня, то описываемые сущности и процессы становятся подклассами сущностей и процессов, введенных в онтологии верхнего уровня, наследуя от них отношения и атрибуты. Если же онтология верхнего уровня не используется, то онтология предметной области и онтология задач разрабатываются целиком «с чистого листа».

Онтология приложения дополняет онтологию предметной области и онтологию задач, описывая те понятия, которые соответствуют объектам семантического портала. Например, в рамках данного диссертационного исследования в онтологию приложения вошли такие понятия как «документ», «ссылка», «рубрика», «область знаний» и «специалист».

Таким образом, онтологическая модель, используемая в семантическом портале, в общем случае состоит из четырех частей: онтологии верхнего уровня, онтологии предметной области, онтологии задач и онтологии приложения. Первые три части фокусируются на описании предметной области семантического портала (понятия и их взаимосвязи). Информация из этой предметной области обрабатывается порталом в виде объектов. Четвертая составная часть онтологической модели – онтология приложения – описывает отношения типов объектов с понятиями предметной области, то есть семантику типов объектов с точки зрения контекста.

Разделение используемой в семантическом портале онтологической модели на *описание предметной области* (онтологии верхнего уровня, онтологии предметной области, онтологии задач) и *описание типов объектов* (онтология приложения) является существенным. Описание предметной области в рамках одного СП может меняться. Это означает, что СП, функционирующий в одной предметной области, может использоваться и в другой предметной области при наличии онтологического описания этой предметной области. В свою очередь описание типов объектов портала является неизменным. СП создается для обработки объектов определенного типа (документы, ссылки, и т.п.) и полагается на наличие в онтологической модели соответствующих понятий.

В связи с указанным разделением состава онтологической модели накладывается ограничение на ее структуру. Структура должна отражать явное разграничение между *переменной* (описание предметной области) и *неизменной* (описание типов объектов) частями онтологии. Для этого в онтологическую модель, созданную в рамках данного диссертационного исследования, было введено дополнительное понятие «Объект портала», подклассами которого являются понятия, соответствующие типам объектов портала (рис. 3.2).

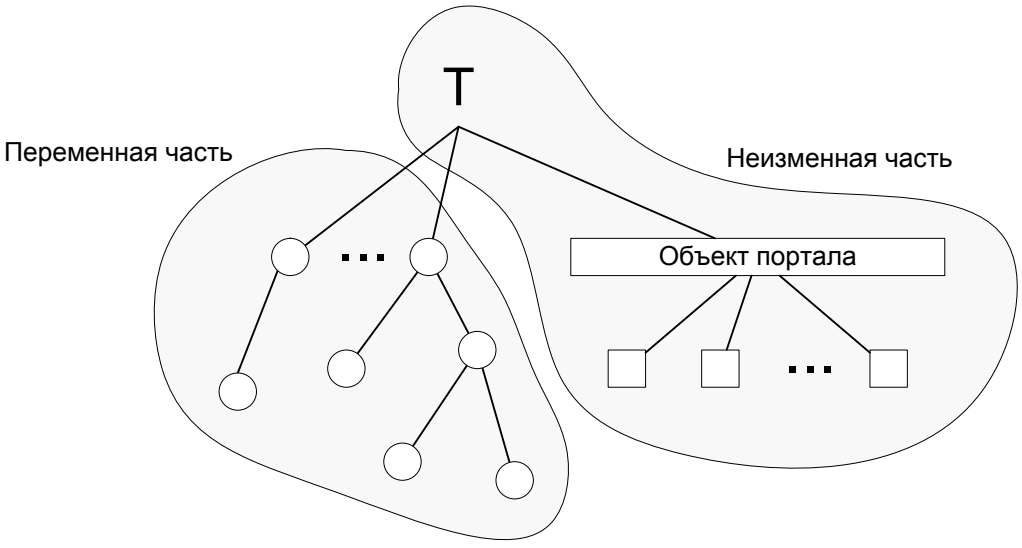


Рис. 3.2. Таксономия понятий в разработанной онтологической модели

Эта онтологическая модель была разработана совместно с сотрудниками группы КСУЗ (лаб. ОСУ, ИКЦ ТПУ) для части предметной области «Автоматизация технологических процессов». В нее вошло 7 понятий верхнего уровня. Общее количество понятий составило 578, количество отношений – 15, максимальная вложенность понятий – 12. Пример части онтологической модели приведен на рисунке 3.3.

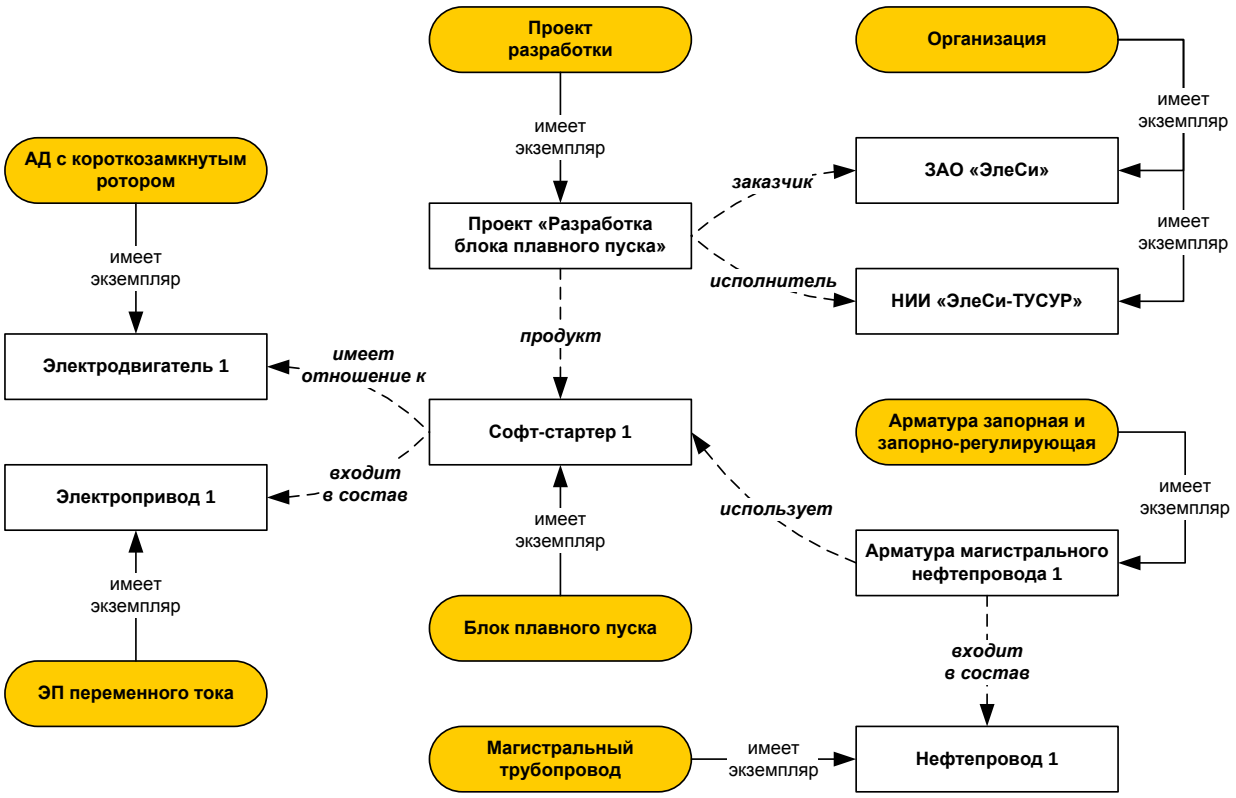


Рис. 3.3. Пример части онтологической модели

Онтологическая модель была описана на языке OWL DL, и количество строк составило 3210, а размер файла – 184 Кбайта. Описание неизменной части онтологической модели приведено в таблице 3.1.

Таблица 3.1. Элементы неизменной части онтологической модели

№	Тип	Название	Описание
1	Понятие	Объект портала	Объект, являющийся носителем информации, относящейся к предметной области, описанной в онтологической модели

2	Понятие	Документ	Файл, хранящийся в базе данных портала
3	Понятие	Ссылка	Указание на файл, хранящийся вне базы данных портала
4	Понятие	Рубрика	Раздел каталога документов и ссылок портала
5	Понятие	Специалист	Сотрудник компании, обладающий знаниями в предметной области, описанной в онтологической модели
6	Понятие	Область знаний	Совокупность знаний, являющаяся частью предметной области, описанной в онтологической модели
7	Отношение	Связанность	Принадлежность элемента онтологической модели к семантическим метаданным объекта
8	Атрибут	Номер объекта	Идентификация объекта портала

3.2. Метод формирования семантических метаданных

Семантические метаданные применяются для описания объектов семантического портала и используются в процедурах семантической обработки информации. Объекты могут либо иметь, либо не иметь текстовое описание. В зависимости от этого формирование семантических метаданных будет выполняться различными способами. В данном диссертационном исследовании разработан *метод формирования семантических метаданных*, который определяет *правила выбора предикатов и объектов из онтологии*, а также определяет *алгоритм поиска понятий и экземпляров в тексте*.

Формирование семантических метаданных объекта портала должен выполнять человек. Он должен в соответствии с сущностью предмета описания определять элементы семантических метаданных. Элементы представляют собой либо триплеты со структурой «субъект–предикат–объект», либо отдельные понятия или экземпляры из онтологии, которые будем называть «субъект» (параграф 2.4.1, таблица 2.7). Создавая элемент семантических метаданных, человек обязательно должен указать «субъект». После этого он может дополнительно указать «предикат» и «объект».

Если субъект указывается человеком таким образом, чтобы отражать сущность предмета описания, то на выбор предиката и объекта накладываются дополнительные ограничения, которые вытекают из правил формирования высказываний дескриптивной логики [96].

Определение 2.2 онтологии O_{DL} (параграф 2.3.2) дано с учетом свойств дескриптивной логики. На основании этого определения в рамках метода формирования семантических метаданных сформулированы *правила выбора предикатов и объектов из онтологии*.

Множество возможных предикатов в триплете ограничивается выбранным субъектом триплета. В таблице 3.2 приведены правила формирования множества M_{PRED} возможных предикатов в триплете на основании определения онтологии O_{DL}.

Таблица 3.2. Правила определения возможного значения предиката в триплете

Значение субъекта	**Правило**
Понятие c_x	$M_{PRED} = \{pr_i \in R \cup A \mid pr_i(c_x, c_y) \vee pr_i(c_x, cd_y)\}$
Экземпляр i_x	$C_{INST}(i_x) = \{c_i \in C \mid P_{IC}(i_x, c_i)\}$ $M_{PRED} = \{pr_i \in R \cup A \mid (pr_i(c_x, c_y) \vee pr_i(c_x, cd_y)) \wedge c_x \in C_{INST}(i_x)\}$

То есть, в качестве предиката человек может выбрать те отношения или атрибуты, которые в онтологии определены для субъекта – понятия или экземпляра.

После выбора предиката человек должен обязательно указать объект триплета. Множество возможных объектов зависит от выбранного предиката. Правила формирования множества M_{OBJ} возможных объектов в триплете на основании определения онтологии O_{DL} приведены в таблице 3.3.

Таблица 3.3. Правила определения возможного значения объекта в триплете

Значение предиката	**Правило**
Отношение r_x	$M_{OBJ} = \{obj_i \in C \cup I \mid r_x(c_x, obj_i) \vee (r_x(c_x, c_y) \wedge P_{IC}(obj_i, c_y))\}$
Атрибут a_x	$M_{OBJ} = \{obj_i \in cd_j \mid a_x(c_x, cd_j)\}$

То есть, возможные значения предиката определяются либо областью конкретных значений атрибута, либо областью значений отношения.

При соблюдении указанных правил человек формирует элементы семантических метаданных. Ограничений на количество элементов в семантических метаданных не накладывается.

Если семантические метаданные формируются на основании текстового описания объекта, то в дополнение к правилам выбора предикатов и объектов используется *алгоритмом поиска понятий и экземпляров в тексте*. Это позволяет частично автоматизировать процесс выбора субъекта из онтологии. С этой целью текстовое описание анализируется на наличие понятий и экземпляров, которые могут выступать в качестве субъектов в элементах семантических метаданных.

Задачей алгоритма является поиск лексических меток понятий и экземпляров из онтологии в текстовом описании объекта для формирования множества возможных субъектов в элементах семантических метаданных.

Обозначим через L_O текстовое описание объекта, которое можно представить как упорядоченное семейство слов, исключив из него знаки препинания. В онтологии O_{DL} в свою очередь задано множество L лексических меток элементов онтологии. Каждая лексическая метка $l_i \in L$ может быть представлена как упорядоченное семейство слов, если удалить из нее знаки препинания. Из множества L выделяем подмножество $L_S \subseteq L$, содержащее лексические метки понятий и экземпляров.

$$L_S = \left\{ l_i \in L \mid P_{LC}(l_i, c_j) \vee P_{LI}(l_i, i_k) \right\} \qquad (3.1)$$

Перед поиском лексических меток из множества L_S в текстовом описании L_O выполняется морфологический анализ [107] с целью определения *нормальной формы* слов, входящих в состав лексических меток $l_g \in L_S$ и в состав семейства L_O. В результате получаем множество нормализованных лексических меток L'_S и упорядоченное семейство нормализованных слов L'_O соответственно.

Обозначим количество слов в семействе L'_O через Len_O. Введем упорядоченное семейство слов W. Найденные в результате поиска понятия и экземпляры образуют соответственно множества M_C и M_I.

С учетом введенных обозначений алгоритм поиска понятий и экземпляров в тексте L_O можно представить следующим образом (рис. 3.4).

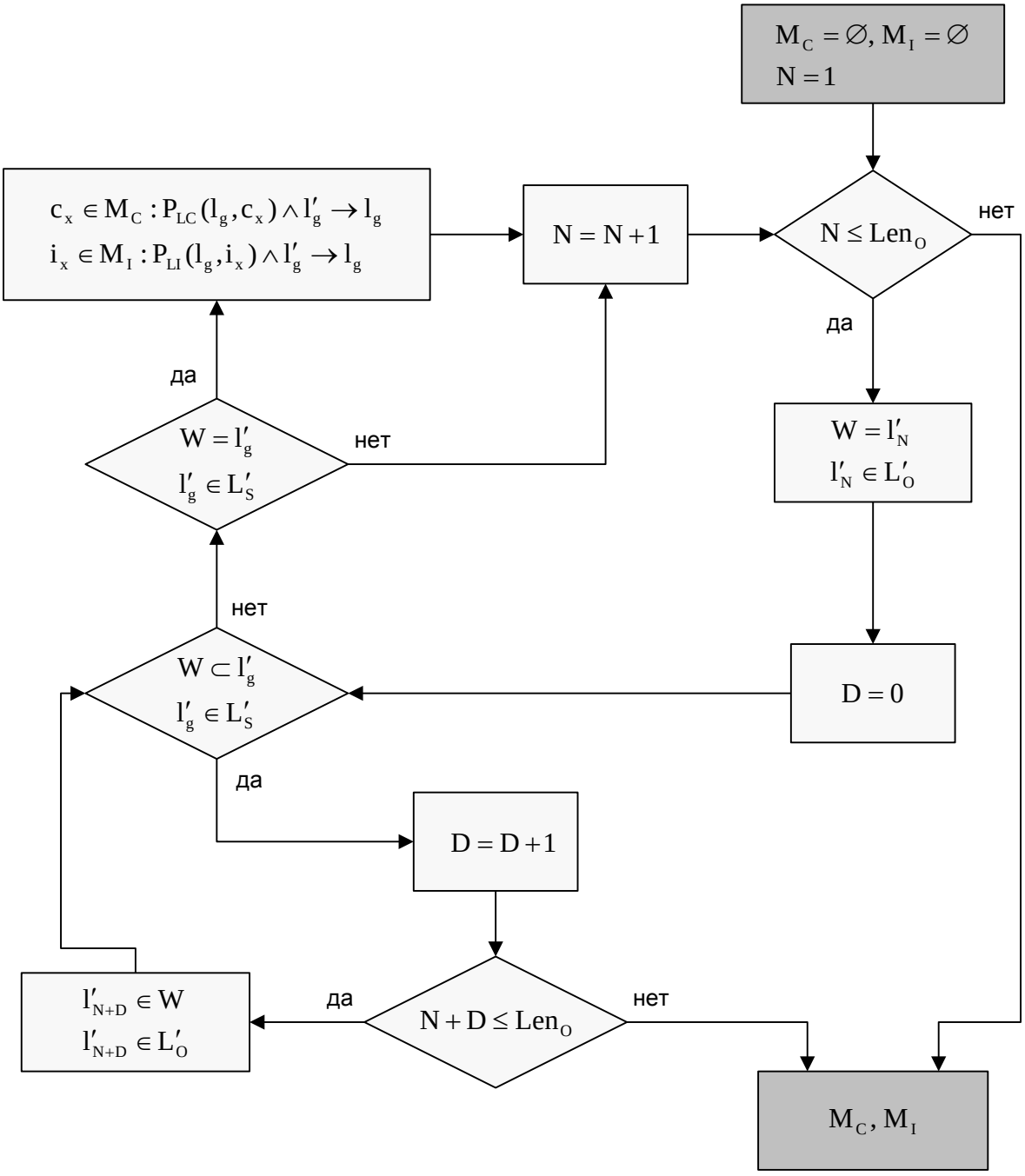


Рис. 3.4. Алгоритм поиска понятий и экземпляров в тексте

Алгоритм построен таким образом, что:

- при наличии во множестве лексических меток двух меток с одинаковым началом совпадения ищутся в тексте сначала для более длинной метки, а потом – для более короткой;

- не учитывается возможные синтаксические варианты расположения лексических меток в тексте;
- учитываются морфологические варианты лексических меток.

Результатом работы алгоритма являются множество понятий M_C и множество экземпляров M_I, которые могут иметь отношение к объекту, для которого формируются семантические метаданные.

Человек, формирующий семантические метаданные, должен отредактировать полученное множество понятий и экземпляров:

- удалить элементы, не отражающие сущность объекта описания;
- устранить многозначность, если множество содержит элементы с одинаковыми лексическими метками;
- дополнить множество понятиями и экземплярами, не найденными алгоритмом.

После этого элементы множества могут быть использованы для формирования триплетов в соответствии с описанными выше правилами выбора предикатов и объектов.

Во время функционирования семантического портала рассмотренный метод используется при формировании семантических метаданных для различных типов объектов. Например, в процессе семантического описания знаний человека не задействуется алгоритм поиска понятий и экземпляров в тексте, так как нет соответствующего текстового описания его знаний. В свою очередь для документа, например, семантические метаданные создаются на основании его текстового содержания, что позволяет задействовать алгоритм поиска понятий и экземпляров.

3.3. Метод вычисления семантической близости элементов онтологии

Метод вычисления семантической близости элементов онтологии основан на определении 2.2 онтологии O_{DL} (параграф 2.3.2). Он развивает подход к оценке близости объектов, основанный на использовании *сотипности*

(cotopy). Сотипность объектов – это оценка схожести положения сравниваемых объектов в некоторой иерархии [108].

Для оценки близости семантических метаданных в целом необходим метод оценки семантической близости следующих элементов онтологии [25]:

- понятия (множество C);
- экземпляры понятий (множество I);
- отношения (множество R);
- атрибуты (множество A);
- конкретные значения (множество V).

Не каждый указанный элемент онтологии может быть сравнен с любым другим элементом онтологии. Ниже приведена таблица допустимых сравнений.

Таблица 3.4. Допустимые сравнения между элементами онтологии

Кандидат / Эталон	**Понятие**	**Экземпляр**	**Отношение**	**Атрибут**	**Значение**
Понятие	+	+			
Экземпляр	+	+			
Отношение			+		
Атрибут				+	
Значение					+

Пара сравниваемых элементов рассматривается как упорядоченная в том смысле, что первый элемент пары является *эталоном*, с которым сравнивается второй элемент пары – *кандидат*. Из этого следует, что в общем случае показатель семантической близости упорядоченной пары элементов (O_1, O_2) может быть не равным показателю семантической близости упорядоченной пары элементов (O_2, O_1).

3.3.1. Вычисление семантической близости двух понятий

В онтологии O_{DL} на множестве понятий C задано отношение нестрогого частичного порядка P_C. $P_C(c_k, c_l)$ означает, что c_k предшествует c_l, или что

c_l следует за c_k. Причем P_C задано так, что среди элементов множества C существует *единственный минимальный* элемент $c_{top} \in C$.

Иерархия понятий с единственной вершиной (таксономия понятий), заданная отношением P_C, используется для определения семантической близости понятий.

Для каждого понятия $c_i \in C$ существует множество $C_{ANC}(c_i)$, являющееся подмножеством C и содержащее понятия, предшествующие понятию c_i, а также само понятие c_i.

$$C_{ANC}(c_i) = \{c_j \in C \mid P_C(c_j, c_i) \vee c_j = c_i\} \quad (3.2)$$

Для оценки семантической близости двух понятий вводятся два показателя, основанные на сравнении множеств $C_{ANC}(c_i)$:

а) семантическая близость двух понятий без учета наследования

$$SC_F(c_k, c_l) = \frac{|C_{ANC}(c_k) \cap C_{ANC}(c_l)|}{|C_{ANC}(c_k) \cup C_{ANC}(c_l)|} \quad (3.3)$$

$SC_F(c_k, c_l) \in (0;1]$, так как (3.4)

$c_{top} \in C_{ANC}(c_k) \cap C_{ANC}(c_l)$ и $c_{top} \in C_{ANC}(c_k) \cup C_{ANC}(c_l)$ при любых $c_k \in C$ и $c_l \in C$.

б) семантическая близость двух понятий с учетом наследования

$$SC_C(c_k, c_l) = k_{st} * SC_F(c_k, c_l) \quad (3.5)$$

$$k_{st} = \begin{cases} 1, \text{ если } c_k \in C_{ANC}(c_l) \\ 0, \text{ иначе} \end{cases} \quad (3.6)$$

$$SC_C(c_k, c_l) \in [0;1] \quad (3.7)$$

3.3.2. Вычисление семантической близости двух экземпляров

В онтологии O_{DL} для каждого экземпляра $i_x \in I$ существует:

- непустое множество $C_{INST}(i_x)$, включающее понятия, к которым относится экземпляр i_x.

$$C_{INST}(i_x) = \{c_j \in C \mid P_{IC}(i_x, c_j)\}, \; C_{INST}(i_x) \neq \varnothing \quad (3.8)$$

- множество $R_{INST}(i_x)$, включающие все конкретизированные отношения экземпляра i_x.

$$R_{INST}(i_x) = \{ri_j \in R_I \mid ri_j(i_x, i_y)\} \quad (3.9)$$

- множество $A_{INST}(i_x)$, включающие все конкретизированные атрибуты экземпляра i_x.

$$A_{INST}(i_x) = \{ai_j \in A_I \mid ai_j(i_x, i_y)\} \quad (3.10)$$

Семантической близости двух экземпляров i_x и i_y складывается из их реляционной близости и их типовой близости. Реляционная близость позволяет оценить схожесть двух экземпляров исходя из их отношений с другими экземплярами онтологии. Типовая близость основана на семантической близости понятий, к которым относятся экземпляры.

а) семантическая близость двух экземпляров без учета наследования

реляционная близость двух экземпляров

$$SI_L(i_x, i_y) = \frac{SI_{LR}(i_x, i_y) + SI_{LA}(i_x, i_y)}{2},\ SI_L(i_x, i_y) \in [0;1] \quad (3.11)$$

$$SI_{LR}(i_x, i_y) = \begin{cases} \dfrac{\left|R_{EQU}(i_x, i_y)\right|}{\left|R_{INST}(i_x) \cap R_{INST}(i_y)\right|}, \text{если } R_{INST}(i_x) \cap R_{INST}(i_y) \neq \varnothing \\ 0, \text{иначе} \end{cases} \quad (3.12)$$

$$R_{EQU}(i_x, i_y) = \{ri_i \in R_I \mid ri_i(i_x, i_z) \wedge ri_i(i_y, i_z)\} \quad (3.13)$$

$$SI_{LA}(i_x, i_y) = \begin{cases} \dfrac{\left|A_{EQU}(i_x, i_y)\right|}{\left|A_{INST}(i_x) \cap A_{INST}(i_y)\right|}, \text{если } A_{INST}(i_x) \cap A_{INST}(i_y) \neq \varnothing \\ 0, \text{иначе} \end{cases} \quad (3.14)$$

$$A_{EQU}(i_x, i_y) = \{ai_i \in A_I \mid ai_i(i_x, v_z) \wedge ai_i(i_y, v_z)\} \quad (3.15)$$

типовая близость двух экземпляров без учета наследования

$$SI_{CF}(i_x, i_y) = \frac{\sum_{c_i \in C_{INST}(i_x)}^{n} \max_{c_j \in C_{INST}(i_y)}^{m} (SC_F(c_i, c_j))}{n} \quad (3.16)$$

$$SI_{CF}(i_x, i_y) \in (0;1] \tag{3.17}$$

семантическая близость двух экземпляров без учета наследования

$$SI_F(i_x, i_y) = \frac{k_{IC} * SI_{CF}(i_x, i_y) + k_{IL} * SI_L(i_x, i_y)}{k_{IC} + k_{IL}} \tag{3.18}$$

С учетом того, что $k_{IC} \in (0;1]$ и $k_{IL} = 1 - k_{IC}$, получаем

$$SI_F(i_x, i_y) \in (0;1]. \tag{3.19}$$

б) семантическая близость двух экземпляров с учетом наследования

типовая близость двух экземпляров с учетом наследования

$$SI_{CC}(i_x, i_y) = \frac{\sum_{c_i \in C_{INST}(i_x)}^{n} \max_{c_j \in C_{INST}(i_y)}^{m} (SC_C(c_i, c_j))}{n} \tag{3.20}$$

$$SI_{CC}(i_x, i_y) \in [0;1] \tag{3.21}$$

семантическая близость двух экземпляров с учетом наследования

$$SI_C(i_x, i_y) = \frac{k_{IC} * SI_{CC}(i_x, i_y) + k_{IL} * SI_L(i_x, i_y)}{k_{IC} + k_{IL}} \tag{3.22}$$

С учетом того, что $k_{IC} \in (0;1]$ и $k_{IL} = 1 - k_{IC}$, получаем

$$SI_C(i_x, i_y) \in [0;1]. \tag{3.23}$$

Коэффициенты k_{IC} и k_{IL} позволяют настраивать процедуру вычисления семантической близости двух экземпляров в зависимости от содержания *ABox* онтологии O_{DL}. Если экземпляры описаны в онтологии в основном с помощью связей с другими экземплярами или конкретными значениями, то необходимо установить соотношение $k_{IC} < k_{IL}$. В противном случае необходимо установить соотношение $k_{IC} \geq k_{IL}$.

3.3.3. Вычисление семантической близости понятия экземпляру

Сравнение двух разнотипных элементов онтологии возможно лишь с некоторым допущением, которое выражается коэффициентом. При сравнении экземпляра и понятия используется показатель семантической близости понятий.

а) семантическая близость понятия экземпляру без учета наследования

Используется коэффициент $d_{ICF} \in (0;1]$.

$$SIC_F(i_x, c_y) = d_{ICF} * \max_{c_j \in C_{INST}(i_x)} \left(SC_F(c_j, c_y)\right) \quad (3.24)$$

$$SIC_F(i_x, c_y) \in (0; d_{ICF}] \quad (3.25)$$

б) семантическая близость понятия экземпляру с учетом наследования

Используется коэффициент $d_{ICC} \in (0;1]$.

$$SIC_C(i_x, c_y) = d_{ICC} * \max_{c_j \in C_{INST}(i_x)} \left(SC_C(c_j, c_y)\right) \quad (3.26)$$

$$SIC_C(i_x, c_y) \in [0; d_{ICC}] \quad (3.27)$$

3.3.4. Вычисление семантической близости экземпляра понятию

Для сравнении понятия и экземпляра также используется уточняющий коэффициент и показатель семантической близости понятий.

а) семантическая близость экземпляра понятию без учета наследования

Используется коэффициент $d_{CIF} \in (0;1]$.

$$SCI_F(c_y, i_x) = d_{CIF} * \max_{c_j \in C_{INST}(i_x)} \left(SC_F(c_y, c_j)\right) \quad (3.28)$$

$$SCI_F(c_y, i_x) \in (0; d_{CI}] \quad (3.29)$$

б) семантическая близость экземпляра понятию с учетом наследования

Используется коэффициент $d_{CIC} \in (0;1]$.

$$SCI_C(c_y, i_x) = d_{CIC} * \max_{c_j \in C_{INST}(i_x)} \left(SC_C(c_y, c_j)\right) \quad (3.30)$$

$$SCI_C(c_y, i_x) \in [0; d_{CIC}] \quad (3.31)$$

3.3.5. Вычисление семантической близости двух отношений

На множестве отношений R задано отношение нестрогого частичного порядка P_R. $P_R(r_k, r_l)$ означает, что r_k предшествует r_l, или что r_l следует за r_k. Для каждого отношения $r_i \in R$ существует множество $R_{ANC}(r_i)$, являющееся подмножеством R и содержащее отношение, предшествующие r_i, а также само отношение r_i.

$$R_{ANC}(r_i) = \{r_j \in R \mid P_R(r_j, r_i) \vee r_j = r_i\} \quad (3.32)$$

Отношение P_R задает иерархию с множеством минимальных элементов, которая используется для определения семантической близости отношений.

а) семантическая близость двух отношений без учета наследования

$$SR_F(r_k, r_l) = \frac{|R_{ANC}(r_k) \cap R_{ANC}(r_l)|}{|R_{ANC}(r_k) \cup R_{ANC}(r_l)|} \quad (3.33)$$

$$SR_F(r_k, r_l) \in [0;1] \quad (3.34)$$

б) семантическая близость двух отношений с учетом наследования

$$SR_C(r_k, r_l) = k_{SR} * SR_F(r_k, r_l) \quad (3.35)$$

$$k_{SR} = \begin{cases} 1, \text{ если } r_k \in R_{ANC}(r_l) \\ 0, \text{ иначе} \end{cases} \quad (3.36)$$

$$SR_C(r_k, r_l) \in [0;1] \quad (3.37)$$

3.3.6. Вычисление семантической близости двух атрибутов

На множестве атрибутов A задано отношение нестрогого частичного порядка P_A. $P_A(a_k, a_l)$ означает, что a_k предшествует a_l, или что a_l следует за a_k. Для каждого атрибута $a_i \in A$ существует множество $A_{ANC}(a_i)$, являющееся

подмножеством A и содержащее атрибуты, предшествующие a_i, а также сам атрибут a_i.

$$A_{ANC}(a_i) = \{a_j \in A \mid P_A(a_j, a_i) \vee a_j = a_i\} \quad (3.38)$$

Отношение P_A задает иерархию с множеством минимальных элементов, которая используется для определения семантической близости атрибутов.

а) семантическая близость двух атрибутов без учета наследования

$$SA_F(a_k, a_l) = \frac{|A_{ANC}(a_k) \cap A_{ANC}(a_l)|}{|A_{ANC}(a_k) \cup A_{ANC}(a_l)|} \quad (3.39)$$

$$SA_F(a_k, a_l) \in [0;1] \quad (3.40)$$

б) семантическая близость двух атрибутов с учетом наследования

$$SA_C(a_k, a_l) = k_{SA} * SA_F(a_k, a_l) \quad (3.41)$$

$$k_{SA} = \begin{cases} 1, \text{ если } a_k \in A_{ANC}(a_l) \\ 0, \text{ иначе} \end{cases} \quad (3.42)$$

$$SA_C(a_k, a_l) \in [0;1] \quad (3.43)$$

3.3.7. Вычисление близости конкретных значений

В онтологии O_{DL} конкретными значениями являются строковые литералы и числа. Их сравнение не относится к области семантического сравнения, но необходимо для расчета близости семантических метаданных.

Обозначим показатель близости двух конкретных значений как $CV(v_k, v_l) \in [0;1]$.

Существует ряд признанных алгоритмов определения $CV(v_k, v_l)$ для строковых литералов: метод Левенштейна [109], метод Q-грамм [110], алгоритм Soundex [111], алгоритм MetaPhone [112] и т.д. Любой из перечисленных алгоритмов может быть использован также и для сравнения чисел, рассматриваемых в качестве строковых литералов.

3.4. Метод вычисления близости семантических метаданных

Разработанный метод вычисления близости семантических метаданных основан на определении 2.2 онтологии O_{DL} (параграф 2.3.2), определении 2.3 семантических метаданных MD_{DL} (параграф 2.4.1) и использует описанный выше метод вычисления семантической близости элементов онтологии.

При вычислении близости пары семантических метаданных $MD_{DL}(q_i)$ и $MD_{DL}(q_j)$ учитывается характер отношений как между элементами двух метаданных, так и между метаданными целиком. Всего было рассмотрено четыре возможных способа сравнения семантических метаданных (табл. 3.5).

Таблица 3.5. Показатели близости семантических метаданных

Сравнение элементов / **Сравнение метаданных**	**без учета наследования**	**с учетом наследования**
с пересечением метаданных	$SM_{FO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right)$	$SM_{CO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right)$
с перекрытием метаданных	$SM_{FS}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right)$	$SM_{CS}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right)$

Для описания методов расчета указанных показателей введены понятия «пересекающихся» и «перекрывающихся» семантических метаданных.

Определение 3.1. Семантические метаданные $MD_{DL}(q_i)$ и $MD_{DL}(q_j)$ являются *пересекающимися*, если *хотя бы для одного* элемента из $MD_{DL}(q_i)$ существует близкий элемент (показатель близости больше нуля) из $MD_{DL}(q_j)$.

Для пересекающихся семантических метаданных расчет близости без учета и с учетом наследования выполняется следующим образом.

$$SM_{FO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right) = \frac{\sum_{smd_{ix} \in MD_{DL}(q_i)}^{n} \max_{smd_{jy} \in MD_{DL}(q_j)}^{m} (k_{ix} * k_{jy} * SE_F(smd_{ix}, smd_{jy}))}{n} \quad (3.44)$$

$$SM_{CO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right) = \frac{\sum_{smd_{ix} \in MD_{DL}(q_i)}^{n} \max_{smd_{jy} \in MD_{DL}(q_j)}^{m} (k_{ix} * k_{jy} * SE_C(smd_{ix}, smd_{jy}))}{n} \quad (3.45)$$

Определение 3.2. Семантические метаданные $MD_{DL}(q_i)$ и $MD_{DL}(q_j)$ являются *перекрывающимися*, если *для каждого* элемента из $MD_{DL}(q_i)$ существует близкий элемент (показатель близости больше нуля) из $MD_{DL}(q_j)$.

Для перекрывающихся семантических метаданных расчет близости без учета и с учетом наследования выполняется следующим образом.

$$SM_{FS}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right) = \begin{cases} SM_{FO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right), \text{ если } \prod \max(SE_F(smd_{ix}, smd_{jy})) > 0 \\ 0, \text{ иначе} \end{cases}$$

(3.46)

$$SM_{CS}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right) = \begin{cases} SM_{CO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right), \text{ если } \prod \max(SE_C(smd_{ix}, smd_{jy})) > 0 \\ 0, \text{ иначе} \end{cases}$$

(3.47)

Определение 2.3 семантических метаданных MD_{DL} (параграф 2.4.1) допускает в качестве их элементов, как триплеты, так и отдельные элементы онтологии. Следовательно, при сравнении элементов семантических метаданных необходима возможность сравнения триплетов и отдельных элементов онтологии. Предлагается считать важность показателей семантической близости отдельных элементов онтологии одинаковой.

На основании сказанного показатели близости элементов метаданных без учета наследования $SE_F(smd_{ix}, smd_{jy})$ и с учетом наследования

$SE_C(smd_{ix}, smd_{jy})$, используемые в формулах 3.44 – 3.47, предлагается вычислять по следующей схеме.

Если эталонный элемент сравниваемых метаданных является триплетом, то в знаменателе результата будет 3.

$$\text{Например, } SE_F\left((c_i, r_j, i_k), (i_x, r_y, i_z)\right) = \frac{SCI_F(c_i, i_x) + SR_F(r_j, r_y) + SI_F(i_k, i_z)}{3} \qquad (3.48)$$

Если эталонный элемент сравниваемых метаданных является отдельным элементом онтологии, то в знаменателе результата будет 1.

$$\text{Например, } SE_F\left((i_i), (c_x)\right) = \frac{SIC_F(i_i, c_x)}{1} = SIC_F(i_i, c_x) \qquad (3.49)$$

Если в элементе метаданных, являющимся кандидатом, нет частей для сравнения с эталонными частями, то в результат подставляется 0.

$$\text{Например, } SE_F\left((c_i, r_j, i_k), (i_x)\right) = \frac{SCI_F(c_i, i_x) + 0 + 0}{3} = \frac{SCI_F(c_i, i_x)}{3} \qquad (3.50)$$

Если в эталонном элементе метаданных меньше частей, чем в элементе-кандидате, то подставляем в результат 0.

$$\text{Например, } SE_F\left((c_i), (i_x, r_y, i_z)\right) = \frac{SCI_F(c_i, i_x) + 0 + 0}{1} = SCI_F(c_i, i_x) \qquad (3.51)$$

Для сравнения элементов семантических метаданных с учетом наследования используются показатели близости элементов онтологии также с учетом наследования.

$$\text{Соответственно, } SE_C\left((i_i), (c_x)\right) = SIC_C(i_i, c_x) \qquad (3.52)$$

А если элементы семантических метаданных необходимо сравнить без учета наследования, то и показатели близости элементов онтологии также используются без учета наследования.

Соответственно, $SE_F((i_i),(c_x)) = SIC_F(i_i, c_x)$ (3.53)

Полный перечень операций сравнения элементов семантических метаданных приведен в приложении 3.

В результате анализа областей значений у показателей семантической близости элементов онтологии было установлено, что

$$SM_{FS}(MD_{DL}(q_i), MD_{DL}(q_j)) = SM_{FO}(MD_{DL}(q_i), MD_{DL}(q_j)) \quad (3.54)$$

Это равенство вытекает из выполнимости условия $\prod \max(SE_F(smd_{ix}, smd_{jy})) > 0$ при любых значениях smd_{ix} и smd_{jy} в силу того, что $SC_F(c_k, c_l) \in (0;1]$, $SI_F(i_x, i_y) \in (0;1]$, $SIC_F(i_x, c_y) \in (0; d_{IC}]$ и $SCI_F(c_y, i_x) \in (0; d_{CI}]$.

Также были определены области значений показателей близости семантических метаданных.

$$SM_{FO}(MD_{DL}(q_i), MD_{DL}(q_j)) \in (0;1] \quad (3.55)$$

$$SM_{CO}(MD_{DL}(q_i), MD_{DL}(q_j)) \in [0;1] \quad (3.56)$$

$$SM_{CS}(MD_{DL}(q_i), MD_{DL}(q_j)) \in [0;1] \quad (3.57)$$

Таким образом, функциональность для сравнения семантических метаданных, предоставляемая семантическим ядром, может применяться тремя разными способами:

1. для сравнения пересекающихся семантических метаданных без учета наследования;

2. для сравнения пересекающихся семантических метаданных с учетом наследования;
3. для сравнения перекрывающихся семантических метаданных с учетом наследования.

При этом следует отметить, что указанные способы оценки семантической близости могут быть применены для решения различных задач. Например, в данном диссертационном исследовании первый способ применялся для реализации функции формирования рекомендаций, второй – для реализации функции категоризации, а третий – функции семантического поиска.

3.5. Метод фильтрации множества кандидатов

Метод вычисления близости семантических метаданных позволяет количественно оценить схожесть между двумя объектами. На практике оценка близости обычно выполняется между *объектом-эталоном* и множеством *объектов-кандидатов*. Например, поисковый запрос (эталон) сравнивается с описаниями документов (множество кандидатов), хранящихся в портале, в результате чего формируется множество релевантных запросу документов. То есть метод вычисления близости семантических метаданных можно рассматривать как средство *ранжирования* объектов-кандидатов на основании объекта-эталона. После ранжирования те кандидаты, семантическая близость которых эталону меньше некоторого порогового значения, исключаются из результирующего множества объектов.

Очевидно, что чем больше множество кандидатов, тем дольше выполняется ранжирование. Поэтому уменьшение множества кандидатов за счет исключения из него объектов с заведомо низким показателем семантической близости способно увеличить вычислительную эффективность операции семантического сравнения. С этой целью разработан *метод фильтрации множества кандидатов*.

Метод фильтрации основан на использовании системы логического вывода для дескриптивной логики (ДЛ). Он позволяет отфильтровать из

множества кандидатов те объекты, семантическая близость которых объекту-эталону *равна нулю*. Следовательно, он применим только при вычислении близости семантических метаданных *с учетом наследования*, потому что только у этих показателей значение может быть равно нулю.

Метод фильтрации определяет:

- способ представления семантических метаданных объектов СП в онтологии;
- способ формирования запроса к системе логического вывода на основании семантических метаданных объекта-эталона.

Способ представления семантических метаданных в онтологии заключается в использовании специального отношения для связывания экземпляров, представляющих объекты СП, с их семантическими метаданными. Указанное отношение относится к *неизменной* (параграф 3.1) части онтологии и введено потому, что язык OWL DL в отличие от языка RDF не обладает свойством воплощения (reification). Таким образом, в онтологии, основанной на ДЛ, невозможно отразить используемую структуру семантических метаданных, то есть невозможно связать триплет с объектом с помощью отношения. Поэтому в онтологии семантические метаданные представляются лишь частично, но этого достаточно для фильтрации множества объектов-кандидатов.

Определение 3.3. В онтологии *фиктивным* является тот экземпляр некоторого понятия, который не соответствует никакой сущности из предметной области, а предназначен для представления этого понятия в виде экземпляра. Такие экземпляры нужны для представления семантических метаданных MD_{DL} в онтологии O_{DL}.

Далее описан способ представления семантических метаданных в онтологии.

Пусть $Q = \{q_1,...,q_k\}$ – конечное множество объектов СП. В онтологии O_{DL} определено множество понятий $C_E \subset C$, обозначающих типы объектов СП. Определено отношение $ri_{sm} \in R_I$ для связывания объектов СП с их семантическими метаданными. Для каждого объекта СП q_i в онтологии создается экземпляр io_j соответствующего понятия. Множество всех таких экземпляров обозначим через $I_O = \{io_1,...,io_k\}$, $I_O \subset I$. Для каждого понятия $c_g \in C$ в онтологии создается *фиктивный экземпляр* is_g. Множество всех фиктивных экземпляров в онтологии обозначим через $I_S = \{is_1,...,is_n\}$, $I_S \subset I$.

Для каждого элемента smd_i из семантических метаданных $MD_{DL}(q_j)$ объекта q_j в онтологии задается отношение между экземплярами в соответствии со следующими правилами (табл. 3.6), в которых экземпляр io_j соответствует объекту q_j, для которого в онтологию вносятся семантические метаданные.

Таблица 3.6. Правила представления элементов семантических метаданных объекта в онтологии

№	Тип элемента семантических метаданных	Отношение между экземплярами в онтологии
1	(c_x)	$ri_{sm}(io_j, is_x)$, где is_x является фиктивным экземпляром для понятия c_x
2	(i_x)	$ri_{sm}(io_j, i_x)$
3	(c_x, r_y, c_z)	$ri_{sm}(io_j, is_x)$, где is_x является фиктивным экземпляром для понятия c_x $ri_{sm}(io_j, is_z)$, где is_z является фиктивным экземпляром для понятия c_z
4	(c_x, r_y, i_z)	$ri_{sm}(io_j, is_x)$, где is_x является фиктивным экземпляром для понятия c_x $ri_{sm}(io_j, i_z)$
5	(i_x, r_y, c_z)	$ri_{sm}(io_j, i_x)$ $ri_{sm}(io_j, is_z)$,

		где is_z является фиктивным экземпляром для понятия c_z
6	(i_x, r_y, i_z)	$ri_{sm}(io_j, i_x)$ $ri_{sm}(io_j, i_z)$
7	(c_x, a_y, v_z)	$ri_{sm}(io_j, is_x)$, где is_x является фиктивным экземпляром для понятия c_x
8	(i_x, a_y, v_z)	$ri_{sm}(io_j, i_x)$

Модифицированная указанным способом онтология содержит семантические метаданные всех объектов СП. Онтология в таком виде загружается в систему логического вывода для дескриптивной логики. После этого возможно формирование *запросов к СЛВ*, которые бы позволяли выполнять фильтрацию множества объектов-кандидатов.

Предлагаемый *способ формирования запроса* позволяет составить на основании семантических метаданных объекта-эталона *комплексное понятие* в терминах дескриптивной логики, для которого с помощью СЛВ определяется множество экземпляров, удовлетворяющих этому комплексному понятию. Найденные таким способом экземпляры соответствуют тем объектам СП, семантическая близость которых объекту-эталону *больше нуля*.

Процедура составления комплексного понятия варьируется в зависимости от используемого показателя близости семантических метаданных.

Для показателя SM_{CO} комплексное понятие c_{qco} формируется следующим образом.

Пусть заданы семантические метаданные $MD_{DL}(q_{etalon})$ для объекта-эталона q_{etalon}. В онтологии определено отношение $r_{sm} \in R$, соответствующее отношению ri_{sm}. Для каждого экземпляра понятия $i_x \in I$ существует непустое множество $C_{INST}(i_x)$, включающее понятия, к которым относится экземпляр i_x.

$$C_{INST}(i_x) = \{c_j \in C \mid P_{IC}(i_x, c_j)\},\ C_{INST}(i_x) \neq \varnothing \qquad (3.58)$$

Тогда на основании множества элементов $smd_i \in MD_{DL}(q_{etalon})$ семантических метаданных объекта-эталона формируется множество *промежуточных*

комплексных понятий C_{IMC} в терминах дескриптивной логики по следующим правилам (табл. 3.7).

Таблица 3.7. Правила преобразования элементов семантических метаданных в понятия дескриптивной логики

№	Тип элемента семантических метаданных	Комплексное понятие
1	(c_x)	$\exists r_{sm}.c_x$
2	(i_x)	$\mathbf{т}_{c_y \in C_{INST}(i_x)} \exists r_{sm}.c_y$
3	(c_x, r_y, c_z)	$(\exists r_{sm}.c_x)\ \mathbf{т}\ (\exists r_{sm}.c_z)$
4	(c_x, r_y, i_z)	$(\exists r_{sm}.c_x)\ \mathbf{т}\ (\mathbf{т}_{c_y \in C_{INST}(i_z)} \exists r_{sm}.c_y)$
5	(i_x, r_y, c_z)	$(\mathbf{т}_{c_y \in C_{INST}(i_x)} \exists r_{sm}.c_y)\ \mathbf{т}\ (\exists r_{sm}.c_z)$
6	(i_x, r_y, i_z)	$(\mathbf{т}_{c_y \in C_{INST}(i_x)} \exists r_{sm}.c_y)\ \mathbf{т}\ (\mathbf{т}_{c_y \in C_{INST}(i_z)} \exists r_{sm}.c_y)$
7	(c_x, a_y, v_z)	$\exists r_{sm}.c_x$
8	(i_x, a_y, v_z)	$\mathbf{т}_{c_y \in C_{INST}(i_x)} \exists r_{sm}.c_y$

Каждый элемент семантических метаданных преобразуется в комплексное понятие, которое включается в множество C_{IMC}.

Элементы полученного множества C_{IMC} входят в состав требуемого комплексного понятия c_{qco} в виде комплексного понятия c_{co}. Понятие c_{co} является *объединением* понятий из множества C_{IMC} (выражение 3.59).

$$c_{co} = \mathbf{т}_{c_{im} \in C_{IMC}} c_{im} \qquad (3.59)$$

$$c_{qco} = c_{entity}\ \mathbf{у}\ c_{co}, \text{ где} \qquad (3.60)$$

c_{entity} – комплексное понятие, описывающее требуемый тип объекта, и формируемое на основании понятий из неизменной части онтологии.

Для показателя SM_{CS} описание комплексного понятия c_{qcs} формируется аналогично предыдущему алгоритму за тем лишь исключением, что элементы множества промежуточных понятий C_{IMC} используются для формиро-

вания комплексного понятия c_{cs} с помощью операции *пересечения* (выражение 3.61).

$$c_{cs} = \mathop{\mathbf{y}}_{c_{im} \in C_{IMC}} c_{im} \tag{3.61}$$

$$c_{qcs} = c_{entity}\ \mathbf{y}\ c_{cs} \tag{3.62}$$

3.6. Применение методов вычисления семантической близости и фильтрации множества кандидатов

Метод вычисления близости семантических метаданных применяется в комплексе с методом вычисления семантической близости элементов онтологии и методом фильтрации для реализации в СП функций семантического поиска, категоризации и формирования рекомендаций. Общие шаги по использованию указанных методов приведены на рисунке 3.5.

Рис. 3.5. Использование метода вычисления близости семантических метаданных

Семантический поиск:

1. Объектом-эталоном при семантическом поиске является поисковый запрос, представленный в виде семантических метаданных.
2. Процедура формирования множества объектов-кандидатов для выполнения среди них семантического поиска заключается в выборе пользователем тех понятий из неизменной части онтологии, которым соответствуют требуемые типы объектов. Если, например, необходимо найти документы и ссылки, то указываются соответствующие понятия. В результате выбора определяется значение

комплексного понятия c_{entity}, используемое в процессе фильтрации множества кандидатов.

3. К множеству объектов-кандидатов относятся все объекты выбранных типов.
4. Для фильтрации множества объектов кандидатов формируется комплексное понятие c_{cs} и выполняется обращение к СЛВ. Результаты обращения обрабатываются с целью установления соответствия между найденными экземплярами и объектами СП.
5. В процессе обработки результатов из множества кандидатов удаляются те объекты, семантические метаданные которых не содержат всех понятий и экземпляров, присутствующих в семантических метаданных поискового запроса.
6. Сравнение поискового запроса с семантическими метаданными объектов-кандидатов осуществляется с использованием показателя $SM_{CS}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right)$. Каждому объекту-кандидату присваивается значение релевантности в диапазоне [0;1].
7. Все объекты-кандидаты упорядочиваются по уменьшению показателя релевантности.

В результате семантического поиска пользователю предлагается список найденных по запросу объектов, упорядоченных по релевантности и дополнительно сгруппированных по типам объектов.

Категоризация:

1. В качестве объекта-эталона выступает рубрика каталога.
2. Из базы данных извлекаются семантические метаданные всех объектов СП (за исключением рубрик), для которых нужно проверить соответствие рубрике. Значение понятия c_{entity} устанавливается равным объединению соответствующих понятий-кандидатов.

3. Множеством объектов-кандидатов является множество объектов СП за исключением самих рубрик. Семантическое сравнение рубрики с рубрикой при категоризации не имеет смысла, но имеет практическое значение при проверке правильности расположения рубрики-кандидата в иерархии рубрик.
4. Для фильтрации множества объектов кандидатов формируется комплексное понятие C_{co} и выполняется обращение к СЛВ. Результаты обращения обрабатываются с целью установления соответствия между найденными экземплярами и объектами СП.
5. В процессе обработки результатов из множества кандидатов удаляются те объекты, семантические метаданные которых не содержат хотя бы одного понятия или экземпляра, присутствующего в семантических метаданных рубрики.
6. Сравнение семантических метаданных рубрики с семантическими метаданными объектов-кандидатов осуществляется с использованием показателя $SM_{CO}\left(MD_{DL}(q_i), MD_{DL}(q_j)\right)$. Каждому объекту-кандидату присваивается значение релевантности в диапазоне [0;1].
7. Во множестве объектов-кандидатов остаются лишь те объекты, которые имеют показатель релевантности больше нуля, то есть относятся к рубрике.

В результате категоризации в базу данных СП заносится перечень объектов, относящихся к рубрике, для последующего использования.

Нужно отметить, что наряду с описанным процессом категоризации, называющимся «полной категоризацией», в СП также применяется частичная категоризация. Она применяется для соотнесения одного объекта-кандидата с множеством рубрик. При частичной категоризации фильтрация не используется, так как множество кандидатов содержит один заранее известный объект. Каждая рубрика поочередно сравнивается с кандидатом для установления того, к каким рубрикам относится объект.

Формирование рекомендаций:

1. Объектом-эталоном является некоторый документ, для которого нужно найти другие документы, семантически близкие к нему по текстовому содержанию.
2. Из базы данных СП извлекаются семантические метаданные, как внутренних документов, так и внешних документов (ссылок, имеющих семантические метаданные).
3. Во множество объектов-кандидатов входят все документы за исключением объекта-эталона.
4. Фильтрация множества кандидатов не осуществляется, так как семантическая близость между двумя документами при использовании алгоритма $SM_{FO}(MD_{DL}(q_i), MD_{DL}(q_j))$ не может быть равной нулю.
5. Множество кандидатов остается прежним.
6. Семантическое сравнение семантических метаданных эталона с семантическими метаданными объектов-кандидатов осуществляется с использованием показателя $SM_{FO}(MD_{DL}(q_i), MD_{DL}(q_j))$. Каждому объекту-кандидату присваивается значение близости в диапазоне (0;1].
7. Множество кандидатов упорядочивается по уменьшению показателя близости эталону.

В результате из полученного упорядоченного множества выбираются и рекомендуются пользователю первые N документов, показатель близости которых не ниже *порогового значения для рекомендации*. Пороговое значение для рекомендации предполагается устанавливать экспериментальным путем.

Выводы по главе

1. Онтологическая модель, используемая семантическим ядром портала, должна иметь структуру, в которой выделена неизменная часть – онтология приложения, и переменная часть, которая включает онтологии верхнего уровня, предметных областей и задач.

2. Предложенные методы вычисления близости элементов онтологии основываются на использовании сотипности (cotopy) элементов онтологии, которая определяется положением сравниваемых элементов в заданной иерархии.

3. Для выполнения процедур поиска, категоризации и предоставления рекомендаций необходимы разные методы вычисления близости семантических метаданных, учитывающие специфику каждой из процедур.

4. Производительность процедур поиска, категоризации и предоставления рекомендаций, основанных на методе вычисления близости семантических метаданных, может быть увеличена за счет применения метода фильтрации множества кандидатов.

5. Предложенный метод представления семантических метаданных в системе логического вывода способен находить метаданные с показателем близости равным нулю. Это позволяет применять систему логического вывода для фильтрации множества кандидатов в процедурах поиска и категоризации.

Глава 4. Проектирование, программная реализация и апробация семантического ядра портала

Семантическое ядро портала (СЯ) обеспечивает работу с онтологиями предметных областей и семантическими метаданными, реализуя, в том числе, предложенные методы по их обработке. Предложенная структура СЯ приведена на рисунке 4.1 и включает сервер онтологий (СО) и сервер семантических метаданных (ССМ).

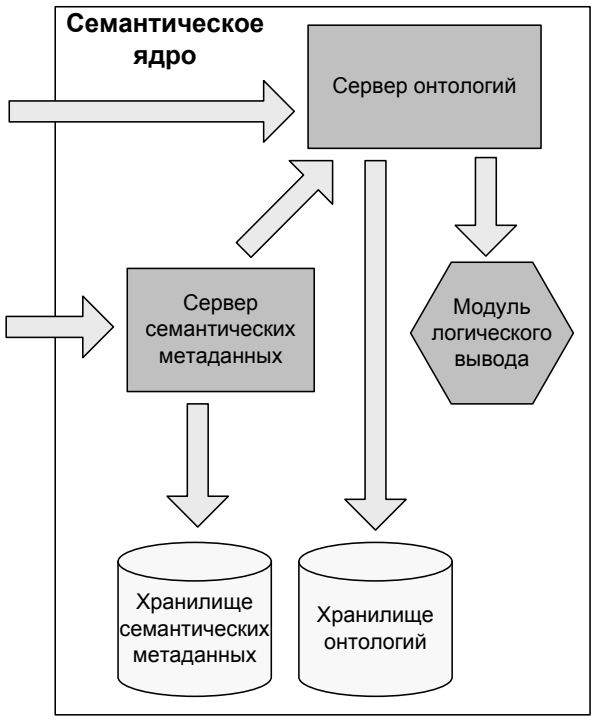


Рис. 4.1. Структура семантического ядра портала

4.1. Проектирование и программная реализация семантического ядра портала

Хотя оба сервера были спроектированы как две отдельно функционирующие программные системы, между ними существует логическая связь. Во-первых, семантические метаданные, описанные на языке RDF, содержат элементы онтологии, описанной на языке OWL DL. Эта зависимость семантических метаданных от онтологии выражается в использовании URI-имен

из онтологии. Для реализации этой логической связи был разработаны интерфейсы *IResource* и *ILiteral*, которые используется в обоих серверах. Классы-сущности сервера онтологий реализуют эти интерфейсы, что позволяет серверу семантических метаданных ссылаться на эти сущности при хранении метаданных (приложение 4, рис. 1, 2). Во-вторых, ССМ использует СО при вычислении близости семантических метаданных, поэтому объекты-сущности СО импортируются в ССМ.

В остальном сервер онтологий и сервер семантических метаданных были спроектированы как два независимых приложения.

4.1.1. Проектирование и программная реализация сервера онтологий

Сервер онтологий реализует *объектную модель онтологии*, описанной на языке OWL DL, и позволяет (рис. 4.2):

- получать доступ к функциям сервера онтологий с использованием технологии .NET Remoting [113];
- хранить файлы с описаниями онтологий в файловой системе;
- выполнять запросы к онтологии, использующие логический вывод.

Указанная структура СО реализована в виде набора классов, перечисленных на следующей UML-диаграмме (рис. 4.3).

Объектная модель онтологии содержит *классы-сущности*, представляющие элементы онтологии, *класс для трансляции OWL-данных* во внутреннее представление сервера и *управляющий класс*, реализующий программный интерфейс доступа к функциям сервера онтологий.

В качестве *классов-сущностей* были выделены: атомарное понятие, экземпляр, отношение, атрибут, целочисленное значение и строковое значение. Для первых четырех типов сущностей в онтологии заданы лексические метки, а для двух оставшихся типов лексическими метками являются их значения,

представленные в текстовом виде (приложение 4, рис. 3). Лексические метки составляют словарь онтологии.

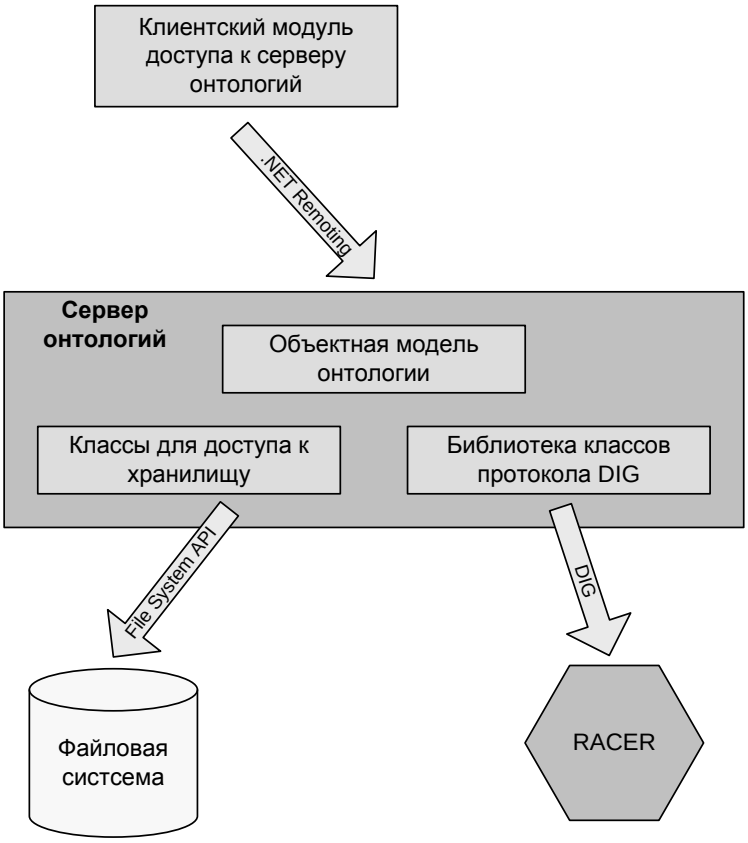


Рис. 4.2. Модули, составляющие программную реализацию СО

Класс *«Транслятор OWL-данных»* предназначен для трансляции описания онтологии на языке OWL DL в систему взаимосвязанных объектов соответствующих классов-сущностей. Результатом трансляции является наборы выявленных понятий, экземпляров, атрибутов, отношений, строковых и целочисленных значений. Сгенерированное внутреннее объектное представление онтологии используется для ускорения доступа к элементам онтологии и выполнения запросов.

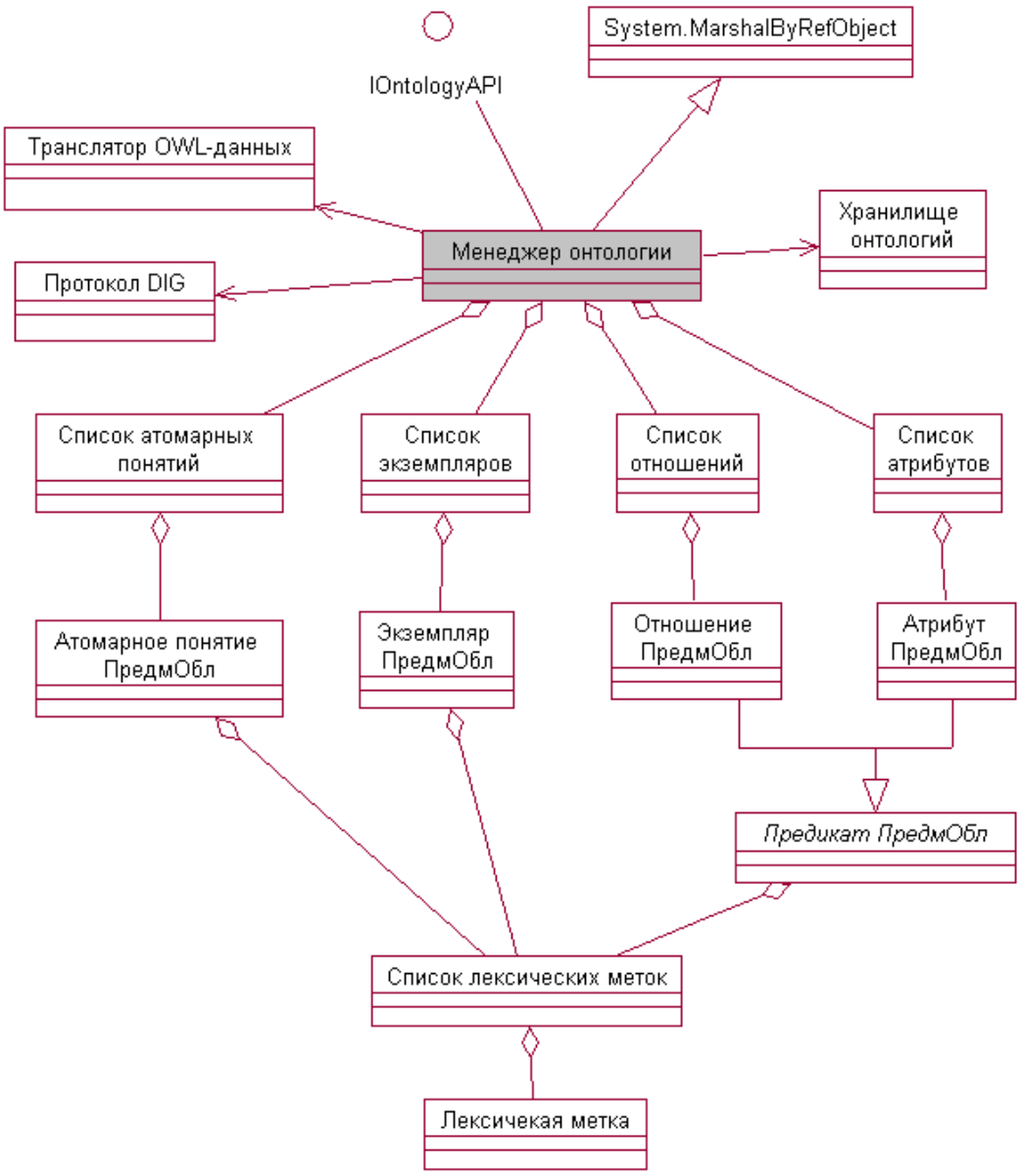


Рис. 4.3. UML-диаграмма классов, реализуемых СО

Управляющий класс *«Менеджер онтологии»* реализует программный интерфейс IOntologyAPI, определяющий методы и свойства сервера онтологий, которые представлены в следующей таблице.

Таблица 4.1. Методы и свойства интерфейса IOntologyAPI

№	Тип	Название	Комментарий
Настройки сервера онтологий			
1	свойство	Журнал обращений к серверу	Свойство определяет, ведется ли журнал обращений
2	свойство	Адрес системы логического вывода	Адрес доступа к СЛВ
3	свойство	Атрибут идентификации объектов	Применяется к экземплярам, ссылающимся на объекты семантиче-

			ского портала
4	свойство	Отношение для семантических метаданных	Используется при представлении семантических метаданных в онтологии
Работа с хранилищем онтологий			
5	свойство	Идентификатор активной онтологии	
6	свойство	Идентификатор хранилища онтологий	
7	метод	Список онтологий в хранилище	
8	метод	Активизировать онтологию	Онтология загружается из хранилища в оперативную память для использования. Сервер может одновременно работать только с одной онтологией.
9	свойство	Онтология активизирована	Свойство позволяет проверить активизирована ли необходимая онтология
10	метод	Сохранить онтологию	Сохраняет активную онтологию в хранилище
11	свойство	Автосохранение	Свойство определяет, сохранять ли онтологии при ее изменении
Работа с содержанием активной онтологии			
12	метод	Добавить временное значение атрибута	Временные значения не хранятся в онтологии, а добавляются только в СЛВ для целей представления семантических метаданных
13	метод	Добавить временное значение отношения	
14	метод	Добавить экземпляр	
15	метод	Добавить постоянное значение атрибута	Постоянные значения хранятся в онтологии
16	метод	Добавить постоянное значение отношения	
17	свойство	Список атомарных понятий	
18	метод	Получить атомарное понятие	
19	свойство	Список экземпляров	
20	метод	Получить экземпляр	
21	свойство	Список атрибутов	
22	метод	Получить атрибут	
23	свойство	Список отношений	
24	метод	Получить отношение	
25	метод	Получить фиктивный экземпляр понятия	Для целей представления семантических метаданных в онтологии вводятся фиктивные экземпляры понятий
26	метод	Получить атрибуты для атомарного понятия	
27	метод	Получить отношения для атомарного понятия	
28	метод	Получить область значений атри-	

		бута	
29	метод	Получить область значений отношения	
Запросы к онтологии			
30	метод	Понятия пересекаются	Использует СЛВ
31	метод	Родители понятия	Использует СЛВ
32	метод	Дети понятия	Использует СЛВ
33	метод	Предки понятия	Использует СЛВ
34	метод	Потомки понятия	Использует СЛВ
35	метод	Предки отношения	Не использует СЛВ
36	метод	Потомки отношения	Не использует СЛВ
37	метод	Экземпляры понятия	Использует СЛВ
38	метод	Понятия экземпляра	Использует СЛВ
39	метод	Понятие содержит экземпляр	Использует СЛВ
40	метод	Значения атрибута	Не использует СЛВ
41	метод	Значения отношения	Не использует СЛВ

Для организации доступа к функциям сервера онтологий была выбрана технология .NET Remoting. Эта технология позволяет обращаться к методам и свойствам уделенных объектов. Технология может использовать для передачи данных протоколы HTTP или TCP. Использование протокола TCP обеспечивает высокую скорость взаимодействия с объектом, но не гарантирует работоспособность при прохождении сетевого трафика через брандмауэры. Использование протокола HTTP обеспечивает меньшее быстродействие, но позволяет работать через брандмауэры. Для реализации сервера онтологий был выбран протокол TCP, так как предполагается использование сервера онтологий в одном сетевом сегменте клиентскими приложениями.

Для использования технологии .NET Remoting управляющий класс «Менеджер онтологии» помимо реализации интерфейса IOntologyAPI наследует функциональность класса System.MarshalByRefObject, входящего в состав инфраструктуры .NET (приложение 4, рис. 4). Принцип доступа клиентских приложений к объекту управляющего класса «Менеджер онтологии» представлен на рисунке 4.4.

Клиентские приложения для доступа к серверу онтологий должны использовать *«клиентский модуль доступа»*, который реализует интерфейс IOntologyAPI и отвечает за передачу вызовов по сети с использованием технологии .NET Remoting. Управляемое (managed) клиентское приложение

(разработанное для выполнения в среде .NET) непосредственно использует клиентский модуль.

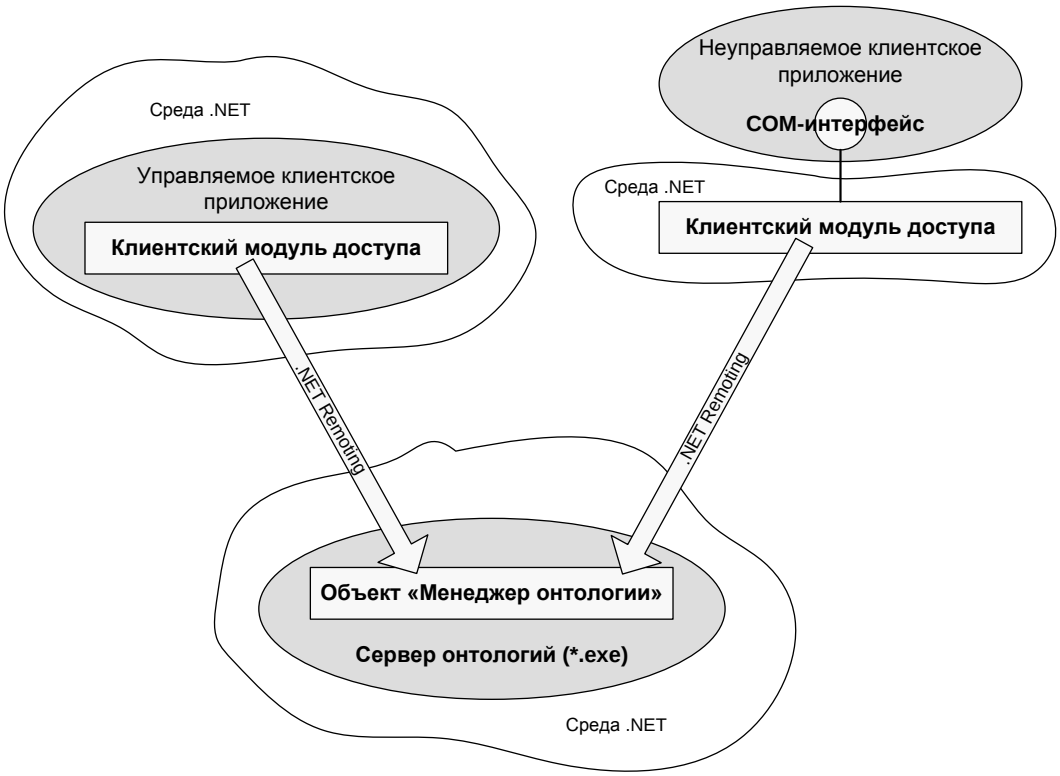


Рис. 4.4. Организация доступа клиентских приложений к СО

Например, сервер семантических метаданных, использующий сервер онтологий, является управляемым приложением. В качестве управляемого клиентского приложения реализовано и приложение для настройки и запуска СО (рис. 4.5).

Для неуправляемых (unmanaged) приложений разработан *COM-интерфейс* (рис. 4.4), позволяющий воспользоваться функциональностью клиентского модуля. Этот интерфейс может использоваться, например, при разработке семантического портала на базе ASP- или PHP-технологий, так как они поддерживают COM-технологию.

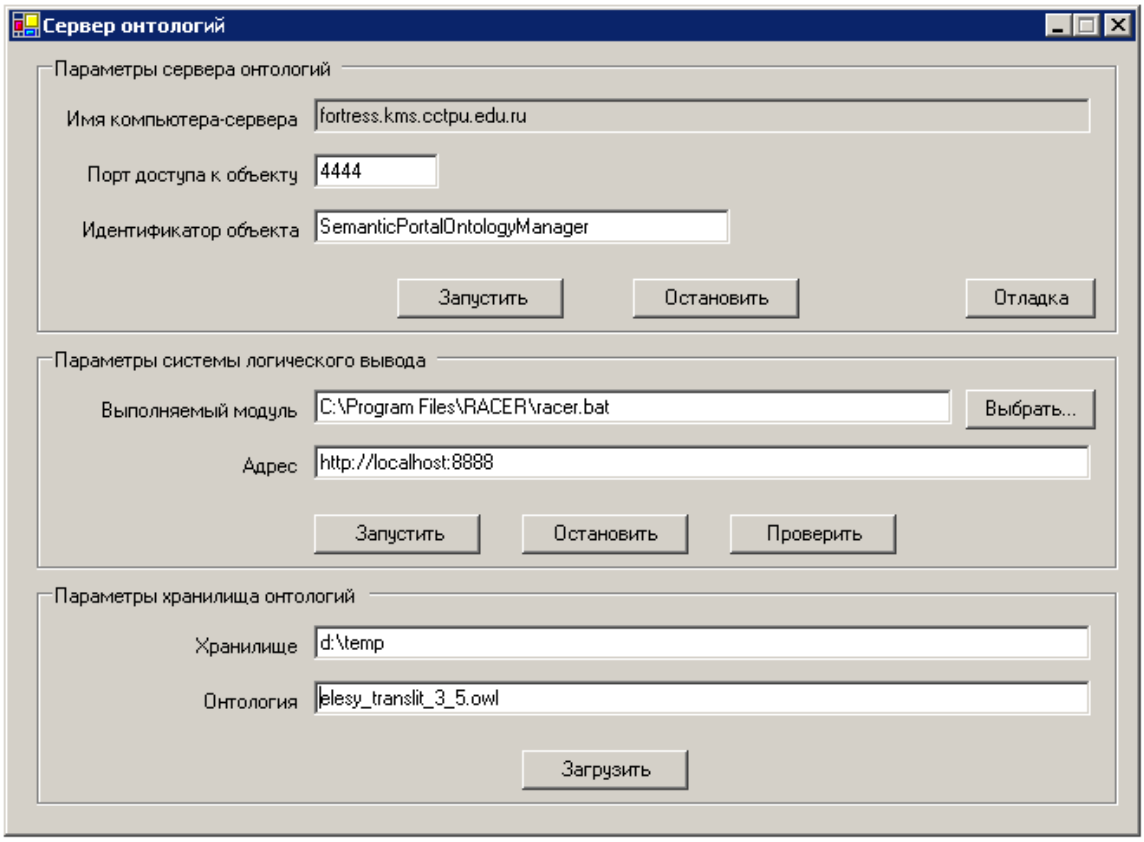


Рис. 4.5. Главная форма приложения для запуска и настройки СО

В качестве подсистемы хранения онтологий, описанных на языке OWL DL, была выбрана файловая система. Такой подход достаточно прост для реализации и одновременно остается возможность использования существующих редакторов онтологий. Все редакторы онтологий поддерживают работу с файловой системой. Для работы с файловой системой менеджер онтологии использует класс *«Хранилище онтологий»* (рис. 4.6).

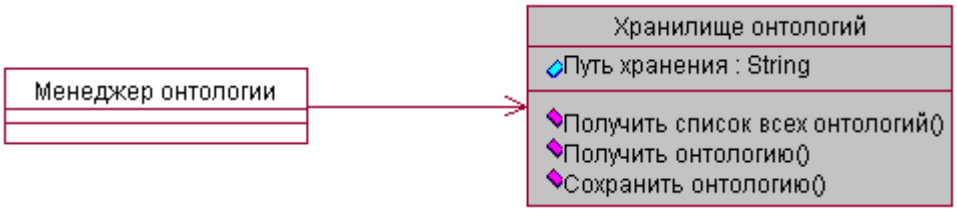


Рис. 4.6. UML-диаграмма класса для управления хранением онтологий в файловой системе

В качестве *системы логического вывода* (СЛВ) для дескриптивной логики была выбрана свободно распространяемая система RACER версии

1.7.24 [114]. RACER реализует логический вывод для дескриптивной логики класса $ALCQHI_{R+}(D^-)$, которая расширяет атрибутивный язык (*AL*) такими возможностями как произвольное отрицание (*C*), транзитивные отношения ($_{R+}$), инверсные отношения (*I*), иерархия отношений (*H*), количественные ограничения на отношения (*Q*) и некоторые конкретные домены (D^-). В качестве конкретных доменов поддерживаются строки и числа. Дескриптивная логика класса $ALCQHI_{R+}(D^-)$ по выразительности является подклассом *SHIQ*, не поддерживая лишь перечисляемые типы (nominals). На сегодняшний день система RACER реализует наиболее выразительную дескриптивную логику с использованием высокопроизводительного алгоритма (tableau-based algorithm) логического вывода, который используется для обработки онтологий, описанных на языке OWL DL.

Если при описании онтологий на языке OWL DL учитывать ограничения дескриптивной логики $ALCQHI_{R+}(D^-)$ (не использовать перечисляемые типы), то созданные описания онтологий на языке OWL DL могут быть автоматически транслированы в формулы дескриптивной логики класса $ALCQHI_{R+}(D^-)$ и переданы системе логического вывода RACER.

Система RACER поддерживает два протокола взаимодействия: JRacer, основанный на протоколе TCP, и DIG, основанный на протоколе HTTP. Для взаимодействия с СЛВ RACER был выбран протокол DIG, так как он был специально разработан для взаимодействия с различными СЛВ, основанными на ДЛ. Использование этого протокола устраняет зависимость от RACER и при появлении новых СЛВ позволяет использовать их.

В соответствии со спецификацией протокола DIG версии 1.1 [115] была спроектирована и разработана *библиотека классов протокола DIG*. Протокол предусматривает три типа операций:

- управление базой знаний (создание и очищение);
- наполнение базы знаний логическими утверждениями;
- запросы к базе знаний.

В приложении 4 приведены диаграммы классов, используемых при наполнении базы знаний (рис. 5, 6) и при выполнении запросов (рис. 7). В логических утверждениях используются базовые понятия протокола DIG (приложение 4, рис. 5).

Созданная библиотека классов используется сервером онтологий для взаимодействия с СЛВ RACER.

Ниже приводится пояснение функционирования сервера онтологий на примере вызова у класса «Менеджер онтологий» метода «Активизировать онтологию» (рис. 4.7).

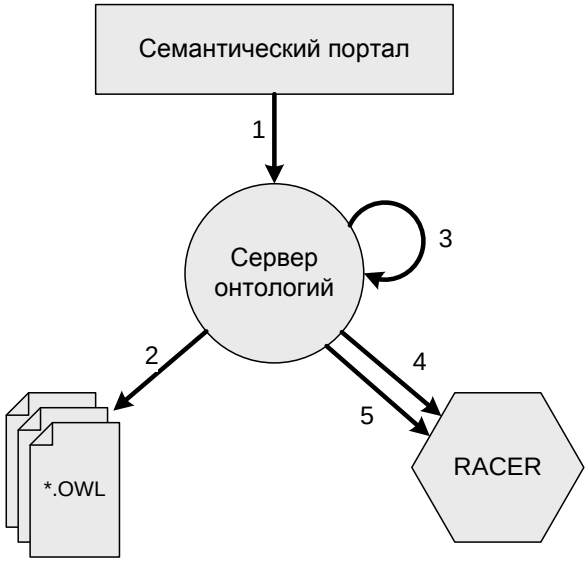


Рис. 4.7. Процесс активизации онтологии сервером

1. Клиентское приложение с помощью *клиентского модуля доступа*, использующего технологию .NET Remoting, вызывает *метод сервера онтологий «Активизировать онтологию»*. При вызове этого метода серверу онтологий передается *идентификатор требуемой онтологии*. Вызов метода обрабатывает *объект класса «Менеджер онтологии»*.
2. Для менеджера онтологии уже задан *идентификатор хранилища онтологий*. Менеджер онтологий через *объект класса «Хранилище онтологий»* обращается к этому *хранилищу* и получает по

идентификатору онтологии OWL-поток, содержащий описание требуемой онтологии.

3. Полученный OWL-поток менеджер онтологии транслирует во *внутреннюю объектную модель* с помощью *объекта класса «Транслятор OWL-данных»*. Полученная объектная модель состоит из объектов соответствующих *классов-сущностей*.
4. Для выполнения логического вывода онтология должна быть передана в СЛВ. С этой целью менеджер онтологии транслирует внутреннюю объектную модель онтологии в *логические высказывания протокола DIG* и передает их системе RACER.
5. Дополнительно менеджер онтологии для каждого понятия, присутствующего в онтологии, передает системе RACER фиктивный экземпляр. При передаче также используется протокол DIG.

На этом активизация требуемой онтологии завершается. Требуемая онтология представлена в сервере онтологий двумя моделями – внутренней объектной моделью и набором логических высказываний в СЛВ. Если клиентское приложение обращается к элементам онтологии, то логический вывод не требуется и менеджер онтологии использует только объектную модель. Если же необходимо выполнить запросы к онтологии, то менеджер онтологии задействует СЛВ.

4.1.2. Проектирование и программная реализация сервера семантических метаданных

Сервер семантических метаданных реализует *объектную модель семантических метаданных* предложенной структуры (параграф 2.4.1) и позволяет (рис. 4.8):

- получать доступ к функциям сервера семантических метаданных с использованием технологии .NET Remoting;

- хранить семантические метаданные различных объектов в реляционной базе данных Microsoft SQL Server 2000;
- вычислять близость семантических метаданных с использованием предложенного метода (параграф 3.4).

Рис. 4.8. Модули, составляющие программную реализацию ССМ

Указанная структура ССМ реализована в виде набора классов, перечисленных на следующей UML-диаграмме (рис. 4.9).

Объектная модель семантических метаданных включает *классы-сущности для элементов языка RDF*, сам *класс семантических метаданных*, *классы семантических выражений*, входящих в состав семантических метаданных, и *управляющий класс*.

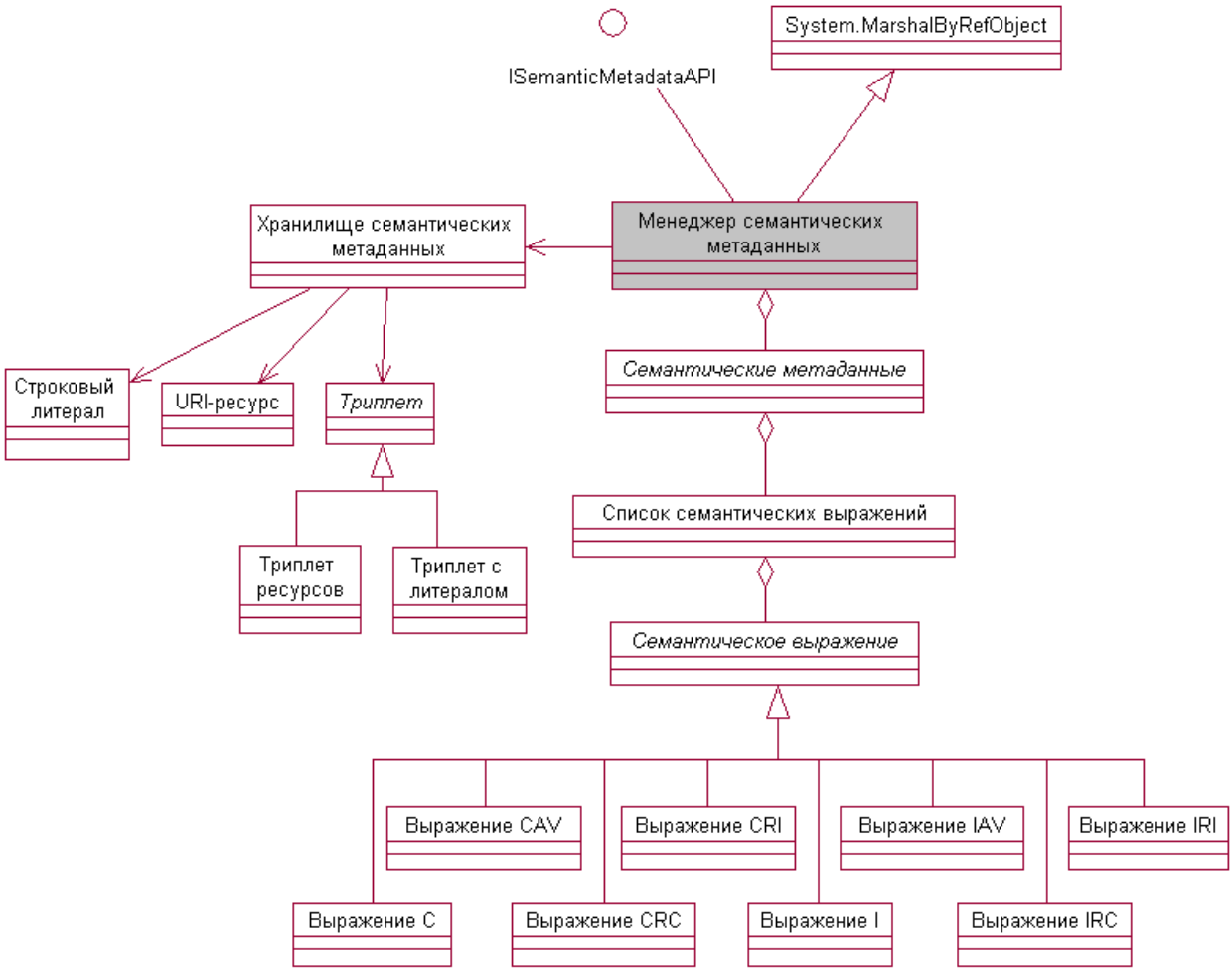


Рис. 4.9. UML-диаграмма классов, реализуемых CCM

В соответствии со спецификацией [64] основными понятиями языка RDF являются *ресурс*, *литерал* и *триплет*. Триплет может быть двух типов: «ресурс – ресурс – ресурс» и «ресурс – ресурс – литерал». Для целей обмена данными между хранилищем семантических метаданных и управляющим классом были разработаны соответствующие *классы-сущности элементов языка RDF* (приложение 4, рис. 8).

Семантические метаданные представлены классом, объекты которого хранят множество семантических выражений, а также предоставляют вспомогательные методы. Количество семантических выражений в метаданных неограниченно. Для каждого из восьми типов семантических выражений разработан соответствующий класс (приложение 4, рис. 9).

Управляющий класс *«Менеджер семантических метаданных»* реализует программный интерфейс ISemanticMetadataAPI, который определяет методы и свойства сервера семантических метаданных, представленные в следующей таблице.

Таблица 4.2. Методы и свойства интерфейса ISemanticMetadataAPI

№	Тип	Название	Комментарий
Настройки сервера семантических метаданных			
1	свойство	Хранилище метаданных	Ссылка на объект класса хранилища метаданных
2	свойство	Менеджер онтологии	Ссылка на объект класса менеджера онтологии
Работа с хранилищем семантических метаданных			
3	метод	Добавить метаданные	
4	метод	Обновить метаданные	
5	метод	Удалить метаданные	
6	метод	Получить метаданные	
Работа с RDF			
7	метод	Построить выражение	Преобразует RDF-данные в объект семантического выражения соответствующего типа
8	метод	Получить RDF-представление	Предоставляет содержание хранилища семантических метаданных в виде RDF-данных
Реализация метода вычисления близости семантических метаданных			
9	метод	Сравнить пересекающиеся метаданные без учета наследования	Вычисляет показатель SM_{FO}
10	метод	Сравнить пересекающиеся метаданные с учетом наследования	Вычисляет показатель SM_{CO}
11	метод	Сравнить перекрывающиеся метаданные с учетом наследования	Вычисляет показатель SM_{CS}

Сервер семантических метаданных предоставляет доступ к реализуемым свойствам и методам посредством технологии .NET Remoting. Для этого реализующий его класс – менеджер семантических метаданных – наследуется от системного класса System.MarshalByRefObject (приложение 4, рис. 10). Сервер запускается в отдельном процессе, для доступа к которому, аналогично серверу онтологий, реализован *клиентский модуль доступа* и *COM-интерфейс*. Клиентский модуль доступа используется управляемыми приложениями среды .NET, такими как приложение для настройки и запуска ССМ (рис. 4.10). COM-интерфейс используется неуправляемыми клиентами.

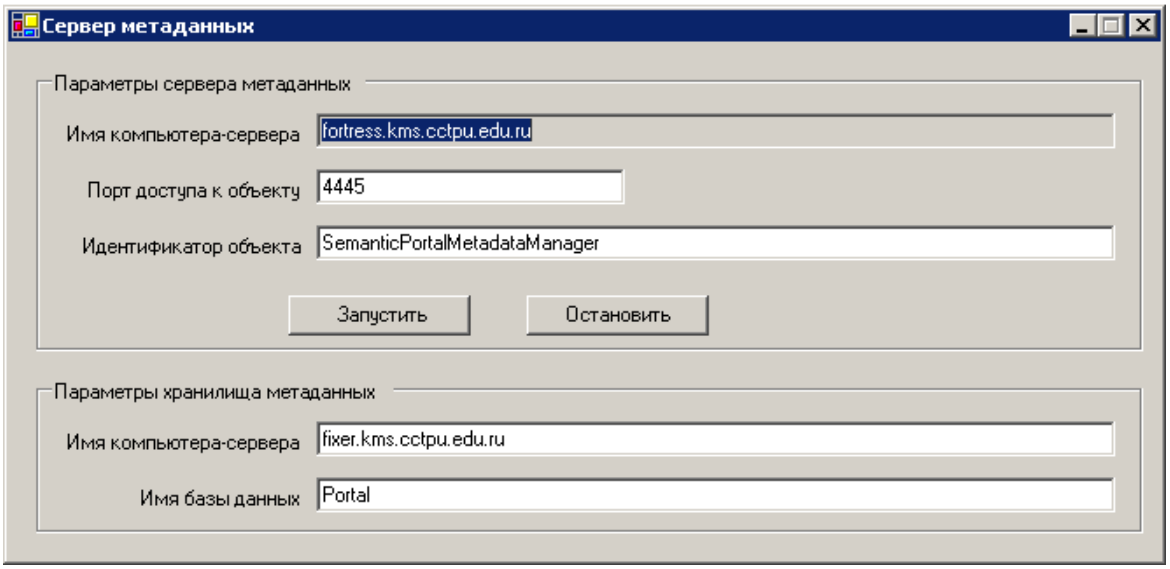


Рис. 4.10. Главная форма приложения для запуска и настройки ССМ

Для хранения семантических метаданных разработано *специальное хранилище на базе РСУБД Microsoft SQL Server 2000*. Схема хранения представлена на рисунке 4.11. Она проектировалась с учетом требования минимизации избыточности хранения. В ней пространства имен, литералы, ресурсы и оба типа триплетов не дублируются при хранении, даже если относятся к двум или более семантическим метаданным.

Данное хранилище RDF-данных отличается от существующих решений (например, RDFGateway, Sesame). Во-первых, оно не предоставляет возможность выполнения запросов. Это обусловлено тем, что знания описываются на языке OWL DL, более выразительном, чем RDF, и поэтому функции запросов и логического вывода реализованы в сервере онтологий. Во-вторых, хранилище разработано с возможностью группировки RDF-данных. В существующих решениях все RDF-высказывания, относящиеся к одной онтологии, хранятся совместно. В разработанном хранилище есть возможность группировки RDF-высказываний в семантические метаданные отдельных объектов описания. Такой подход позволяет отделить общие знания от знаний об отдельных объектах описания. Это в свою очередь позволяет реализовать функцию семантического поиска объектов, а не элементов онтологии.

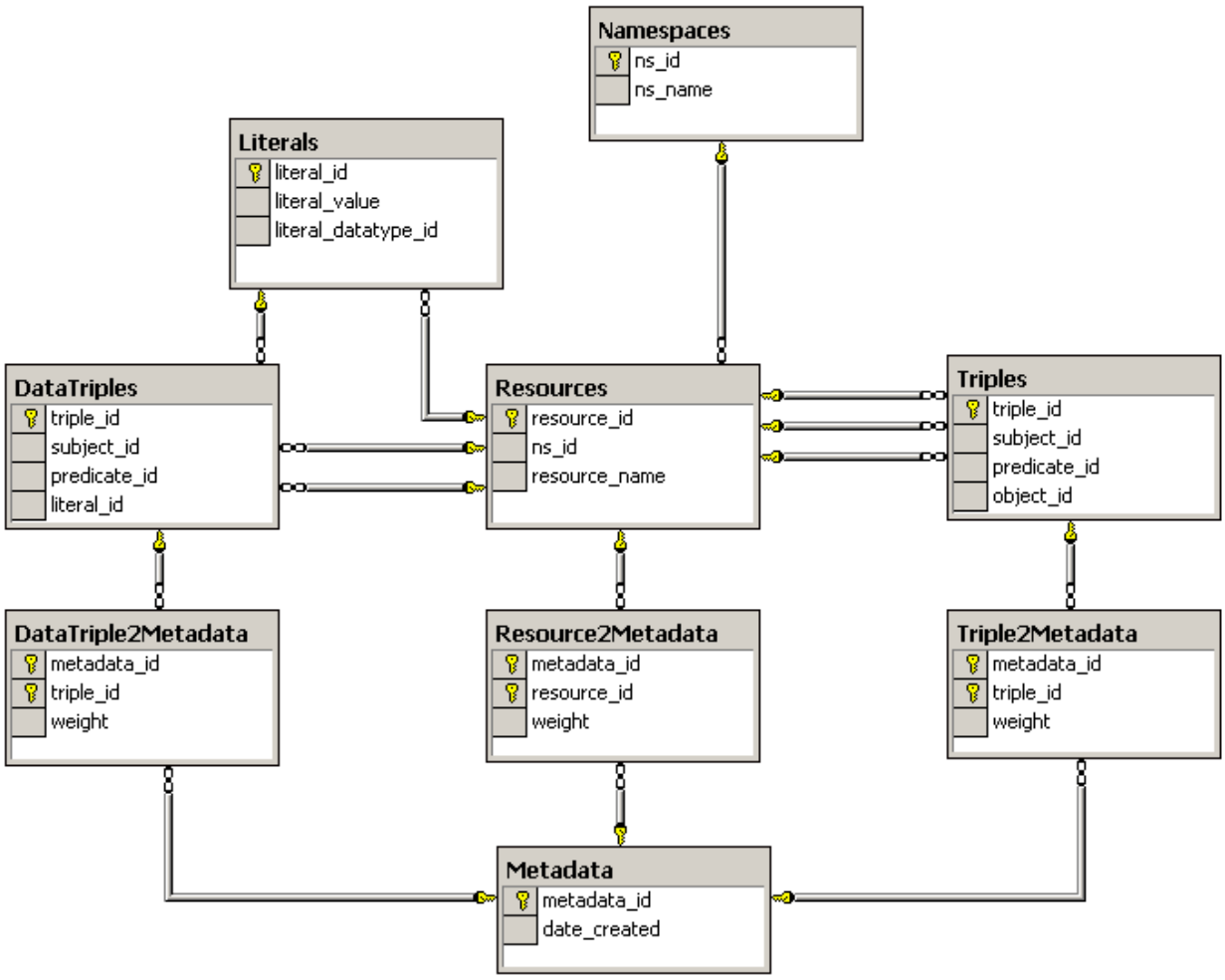


Рис. 4.11. Схема данных хранилища семантических метаданных

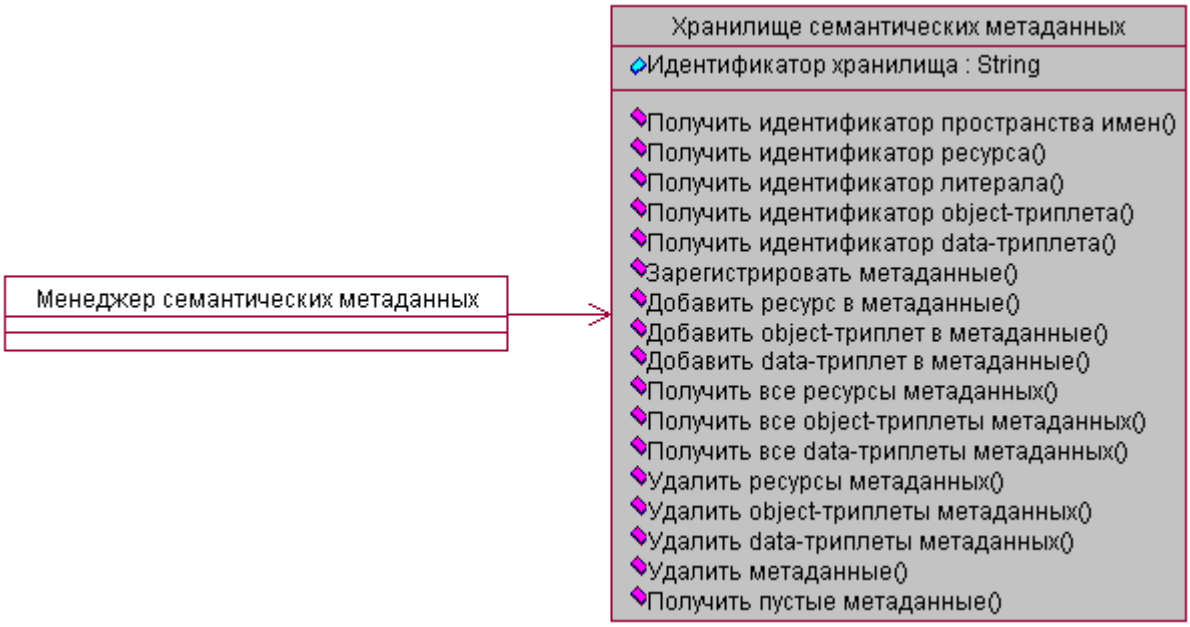


Рис. 4.12. UML-диаграмма класса для управления хранилищем семантических метаданных

ССМ получает доступ к хранилищу посредствам объекта класса *«Хранилище семантических метаданных»*, который предоставляет методы по добавлению, изменению, получению и удалению RDF-данных (рис. 4.12).

Реализация предложенного метода вычисления близости семантических метаданных выполнена в виде трех функций класса *«Менеджер семантических метаданных»*, каждая из которых позволяет вычислить соответствующий показатель близости (выражения 3.55, 3.56, 3.57). Для расчета этих показателей используется сервер онтологий, предоставляющий функции логического вывода и запросов к онтологии. В частности онтология используется для расчета семантической близости элементов онтологии, являющихся частью семантических метаданных.

4.1.3. Вспомогательные функции

Программная реализация вспомогательных алгоритмов для семантического ядра выполнена в виде *«Сервисного класса»* (рис. 4.13), который подключается к клиентскому приложению, использующему семантическое ядро.

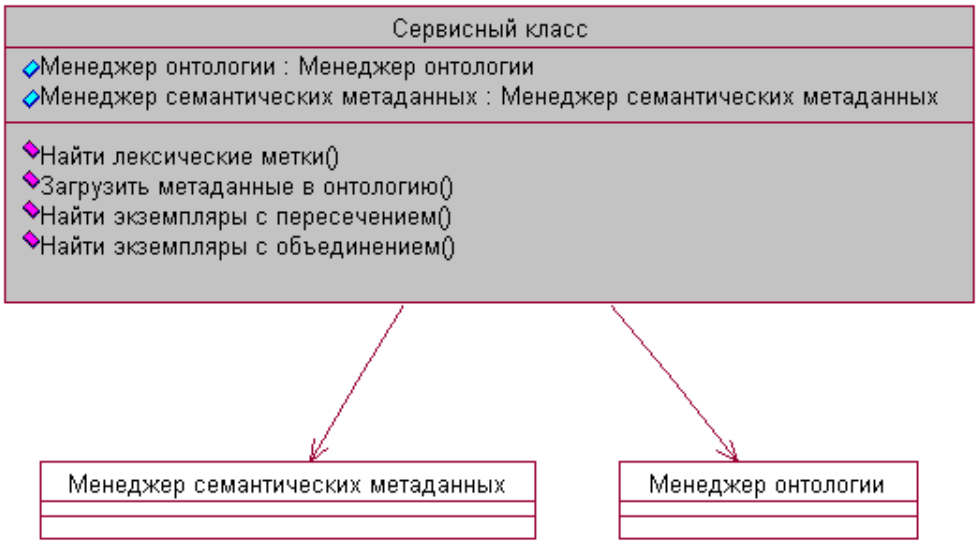


Рис. 4.13. UML-диаграмма сервисного класса

К вспомогательным функциям отнесены:

1. Поиск лексических меток из онтологии в произвольном тексте. Данная функция является программной реализацией одноименного алгоритма, определенного в методе формирования семантических метаданных (параграф 3.2).
2. Загрузка метаданных в онтологию. Данная функция реализует предложенный способ представления семантических метаданных в онтологии для реализации функции фильтрации.
3. Фильтрация множества кандидатов перед расчетом показателя SM_{CO} (выражение 3.60).
4. Фильтрация множества кандидатов перед расчетом показателя SM_{CS} (выражение 3.62).

4.1.4. Степень программной реализации семантического ядра портала

Разработанное и спроектированное семантическое ядро портала полностью реализовано программно на платформе Microsoft .NET с использованием языка программирования C#. В общей сложности для программной реализации было разработано с использованием языка моделирования UML [116] 177 классов и интерфейсов, а объем кода составил более 16 тысяч строк (таблица 4.3). Вклад автора диссертационного исследования в разработку и проектирование составляет 100%, а в программную реализацию – 88%.

Таблица 4.3. Группы классов, составляющих программную реализацию разработанного семантического ядра портала

№	Группа классов	Количество классов и интерфейсов	Объем кода, строк
Сервер онтологий			
1	Объектная модель онтологии	43	6206
2	Доступ к хранилищу онтологий	3	128
3	Объектная модель протокола DIG	99	4238
4	Приложение сервера онтологий	2	768
5	Клиентский модуль доступа к серверу онтологий	1	111
Сервер семантических метаданных			
6	Объектная модель семантических метаданных	24	3283

7	Доступ к хранилищу семантических метаданных	1	721
8	Приложение сервера семантических метаданных	2	441
9	Клиентский модуль доступа к серверу семантических метаданных	1	86
Прочее			
10	Вспомогательные функции	1	973
	ВСЕГО	**177**	**16955**

4.2. Тестирование семантического ядра портала

Разработанное семантическое ядро портала имеет четыре функции:

1. аннотирование объектов;
2. семантический поиск;
3. формирование списка объектов, связанных с исходным объектом;
4. формирование списка объектов, похожих на исходный объект.

Функция аннотирования упрощает процесс формирования семантических метаданных для объектов портала. Семантический поиск позволяет искать объекты портала с учетом их семантики. Формирование списка объектов, связанных с исходным объектом, используется для категоризации объектов портала. В свою очередь формирование списка объектов, похожих на исходный объект, используется для предоставления рекомендаций пользователям портала.

Тестирование указанных функций семантического ядра выполнялось с использованием онтологии, созданной для части предметной области «Автоматизация технологических процессов» [117]. Общее количество понятий в онтологии составило 578, количество отношений – 15, максимальная вложенность понятий – 12, количество лексических меток на русском языке для каждого элемента онтологии – от 1 до 9.

4.2.1. Тестирование функции аннотирования объектов

Функция формирования семантических метаданных (аннотирования) является основополагающей в информационных системах, учитывающих се-

мантику информации при реализации информационных процессов. В разработанном семантическом ядре остальные функции основываются на обработке семантических метаданных объектов портала.

Однако автоматический переход от синтаксиса к семантике является нетривиальной задачей, и разработанный метод аннотирования не исключает участия человека при составлении семантических метаданных объектов портала. Поэтому основной целью тестирования была проверка простоты использования методов и программных средств составления семантических метаданных.

Для этого в онтологию было помещено 1227 экземпляров различных понятий предметной области. Из них 112 экземпляров были выявлены в результате анализа документов, относящихся к выбранной предметной области. Эти экземпляры содержали от 1 до 3 лексических меток. Остальные 1115 экземпляров были автоматически сгенерированны специально созданной программой. Для таких экземпляров использовались лексические метки родительских понятий.

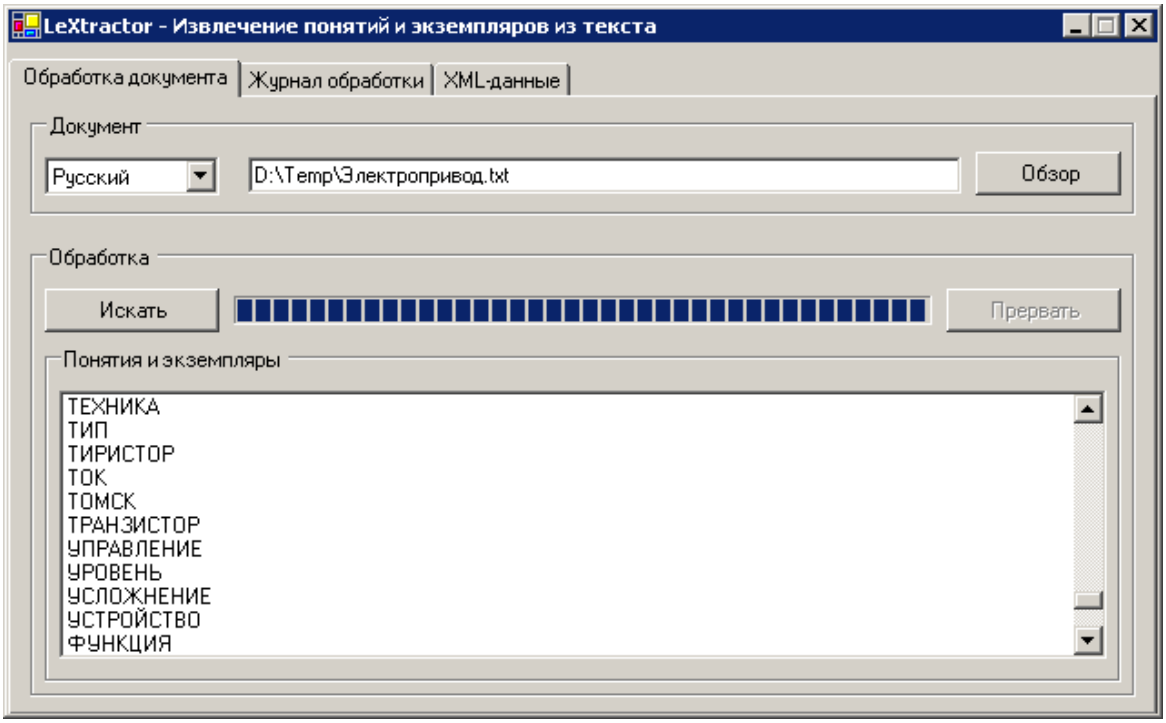


Рис. 4.14. Результат поиска понятий и экземпляров из онтологии в тексте документа

Наполненная экземплярами онтология использовалась для аннотирования 27 рубрик каталога и 160 документов, относящихся к выбранной предметной области. Аннотирование выполнялось в соответствии с методом формирования семантических метаданных (параграф 3.2).

Для 16 документов при формировании семантических метаданных использовался алгоритм поиска понятий и экземпляров в тексте документа (рис. 4.14).

Найденные в документах понятия и экземпляры использовались для формирования триплетов семантических метаданных документов (рис. 4.15).

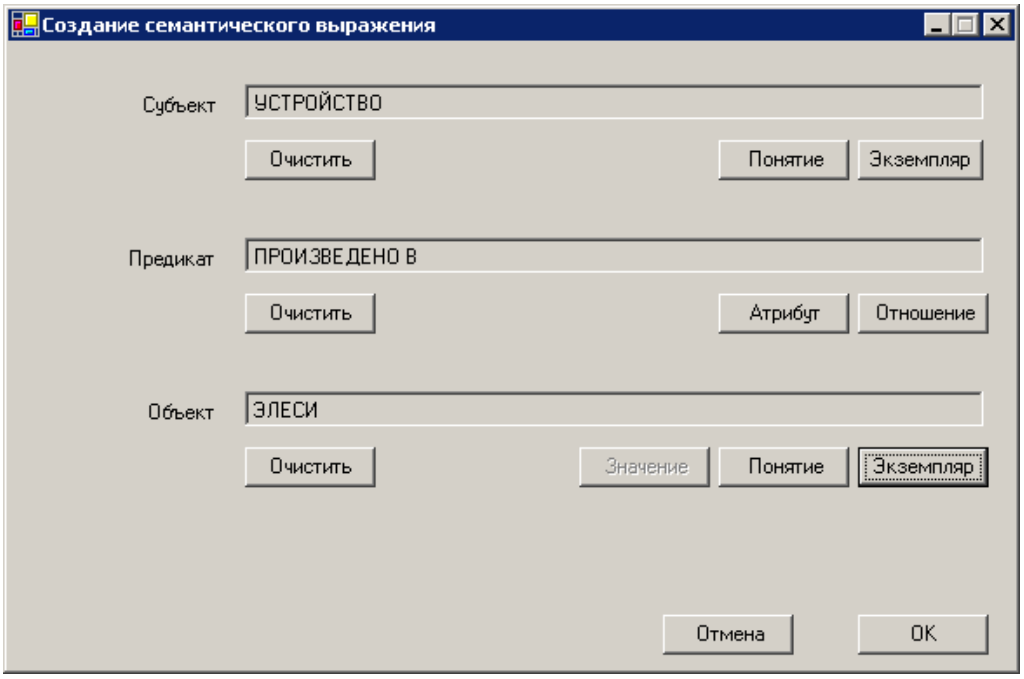


Рис. 4.15. Экранная форма для создания элемента семантических метаданных

Хотя алгоритм поиска понятий и экземпляров в тексте документов ускоряет процесс формирования семантических метаданных, но, тем не менее, процесс формирования семантических метаданных является трудоемким и требует хорошего знания структуры и состава используемой онтологии. В связи с этим возникла необходимость развития средств визуального представления онтологии и текста документа при составлении семантических метаданных.

Аннотации для остальных 144 документов были сгенерированны специально созданной программой в соответствии с методом формирования семантических метаданных без использования алгоритма поиска понятий и экземпляров в тексте. Для каждого документа генерировалось от 5 до 10 элементов семантических метаданных. Созданные таким образом семантические метаданные не отражают текстовое содержание документов, но позволяют протестировать программную реализацию метода формирования семантических метаданных, а также сформировать множество объектов портала, достаточное для тестирования остальных функций семантического ядра.

Для проверки корректности программной реализации правил формирования триплетов (параграф 3.2), сгенерированные аннотации, содержащие 1059 триплетов и отдельных элементов, были загружены в систему логического вывода вместе с целевой онтологией. В СЛВ был выполнен логический вывод для поиска противоречий элементов семантических метаданных ограничениям онтологии. Поиск дал отрицательный результат, что можно считать показателем корректной работы протестированных программных алгоритмов.

Тестовая иерархия из 27 рубрик (приложение 5) также была проаннотирована с использованием метода формирования семантических метаданных.

При составлении семантических метаданных необходимо указывать значение коэффициента релевантности элементов метаданных объекту описания (определение 2.3). В случае с документами и рубриками значение этого коэффициента устанавливалось равным 1. Для документов это оправдано потому, что их семантические метаданные отражают наличие понятий и экземпляров в тексте документов. Для рубрик это оправдано потому, что они группируют объекты портала и, следовательно, их семантические метаданные описывают максимально возможную семантику группируемых объектов. Каждый отдельный объект, относящийся к рубрике, содержит в своих семантических метаданных соответствующие коэффициенты. Например, у доку-

мента все коэффициенты равны 1, а у специалиста эти коэффициенты могут отличаться от 1, так как в каких-то частях предметной области он обладает меньшими знаниями, а в каких-то – большими.

4.2.2. Тестирование функции семантического поиска

Тестирование функции семантического поиска выполнялось с точки зрения качества поиска. Параметрами, по которым обычно оценивается качество работы информационно-поисковой системы (ИПС), являются *точность* и *полнота* поиска. Как показывает практика, для существующих в сети Интернет полнотекстовых ИПС данные показатели находятся на уровне 50% [3]. Это связано с тем, что реальная релевантность, как правило, ниже формальной релевантности, а удовлетворенность, как правило, ниже реальной релевантности (рис. 4.16).

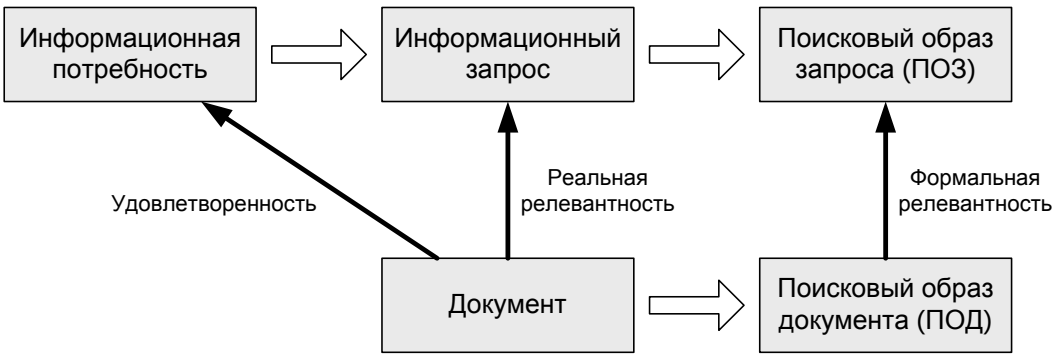


Рис. 4.16. Схема оценки качества работы ИПС

Отмеченное снижение релевантности на пути от формальной оценки до оценки человеком связано с преобразованиями, которые претерпевает информационная потребность человека до попадания в ИПС. В результате этих преобразований в изначальную информационную потребность вносятся искажения:

- Человек может не точно выразить свою информационную потребность.

- Кроме этого искажения вносятся во время формирования ПОЗ. Чаще всего это выражается в утрате контекста поиска. Под утратой контекста надо понимать использование слов из информационного запроса не в том смысле, который заложен в них человеком. Основанная причина утраты контекста – полисемия и омонимия естественного языка.
- ПОД также может неточно отражать текстовое содержание документа. Основная причина – утрата контекста.
- Метод вычисления формальной релевантности также вносит погрешность.

Метод вычисления близости семантических метаданных, на котором основывается семантический поиск, разрабатывался с целью уменьшения искажений при формировании ПОЗ и ПОД и при расчете формальной релевантности (таблица 4.4).

Таблица 4.4. Сравнение различных видов информационно-поисковых систем по параметрам, влияющим на качество поиска

Вид ИПС / **Факторы**	**Полнотекстовая ИПС**	**ИПС со словарем**	**ИПС со словарем и синонимами**	**ИПС с онтологией**
Учет синонимии	-	-	+	+
Учет омонимии	-	частично	частично	+
Учет полисемии	-	частично	частично	+
Учет широты понятия	-	-	-	+
Плюсы	любая предметная область, простота формирования ПОД и ПОЗ	увеличение точности поиска	увеличение полноты поиска	увеличение полноты и точности поиска
Минусы	недостаточная точность и полнота поиска	ограниченная предметная область	ограниченная предметная область	ограниченная предметная область

Специфика полнотекстовых ИПС в том, что основной причиной снижения точности и полноты являются искажения при формировании ПОЗ и ПОД (они формируются автоматически), а методы расчета формальной релевантности имеют высокий показатель качества. В ИПС с ограниченным сло-

варем понятий (тезаурус или онтология) повышается качество формирования ПОЗ и ПОД (за счет участия человека). Поэтому *при сохранении качества расчета формальной релевантности в этих ИПС показатели полноты и точности поиска могут быть улучшены*.

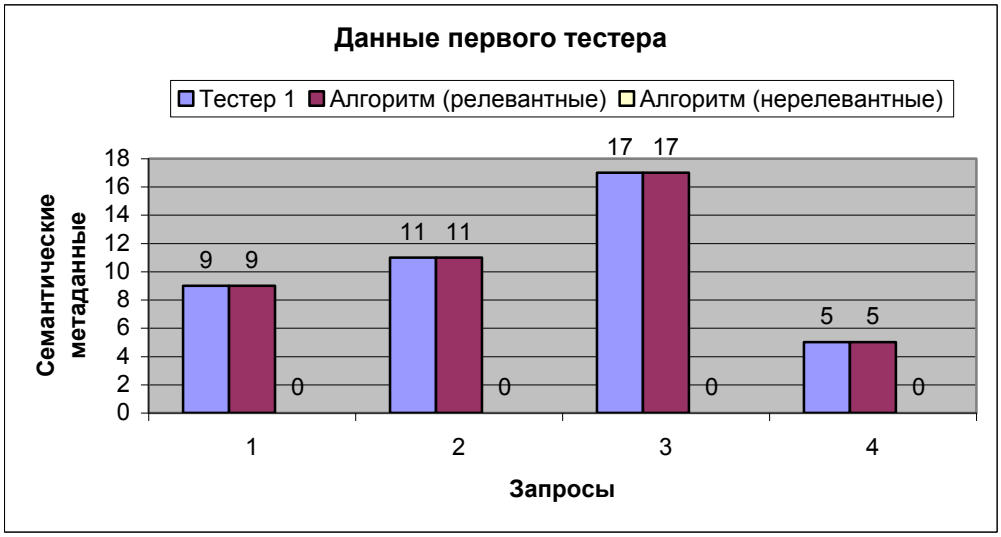
Для проверки качества расчета *формальной релевантности* трем потенциальным пользователям семантического портала (далее, тестеры) было предложено вручную оценить формальную релевантность 4-х запросов относительно семантических метаданных всех 160-ти документов. Всем тестерам были предложены одинаковые запросы, которые были подобраны таким образом, что алгоритм семантического поиска возвращал непустое множество документов.

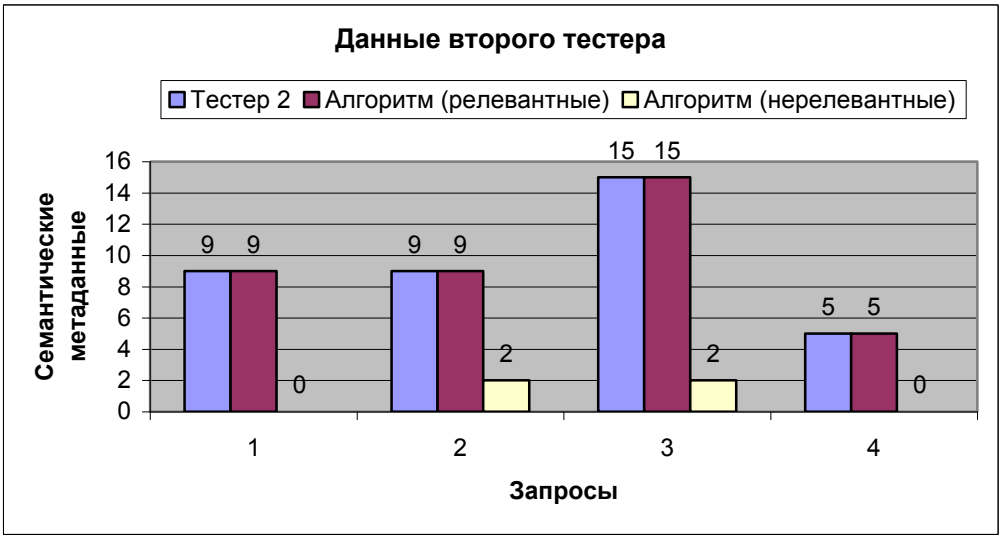
Пример 4.1. Запрос представляет собой набор триплетов и одиночных элементов. Ниже приведена таблица элементов 4-го запроса.

Таблица 4.5. Элементы семантических метаданных, описывающих 4-ый запрос

Составляющая элемента / Элемент	**Субъект**	**Предикат**	**Объект**
1	Понятие «ДОКУМЕНТ»	Отношение «АВТОР»	Понятие «СОТРУДНИК»
2	Понятие «ЭЛЕКТРОПРИВОД»	Отношение «ИСПОЛЬЗУЕТ»	Понятие «ИНВЕРТОР»
3	Понятие «ПЕРЕДАТОЧНОЕ ЧИСЛО»		

В задачи тестера входил отбор семантических метаданных документов, подходящих под запрос (рис. 4.17), а также проставление оценок релевантности по шкале [0;1] для каждого элемента из запроса каждому элементу семантических метаданных документов.





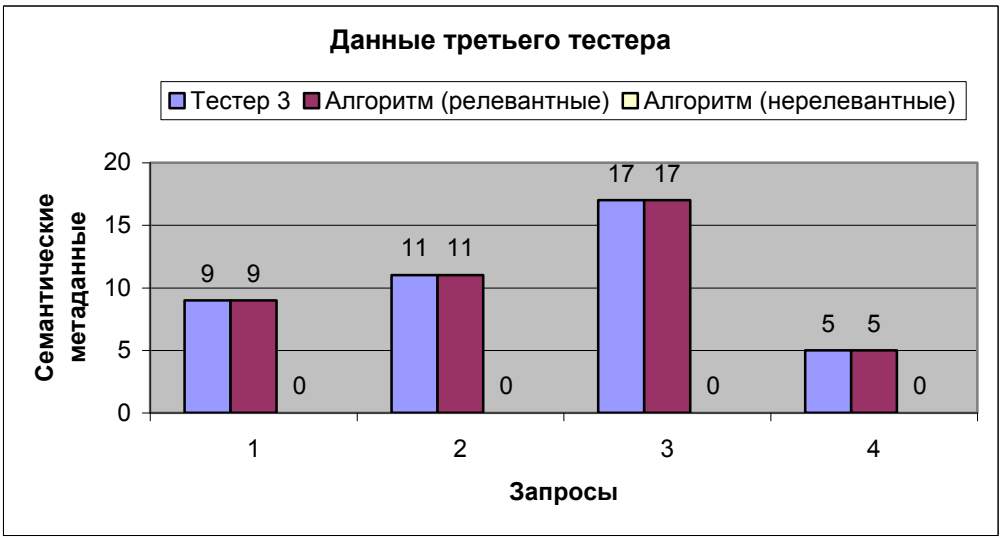


Рис. 4.17. Сравнение результатов работы тестеров с результатами работы алгоритма

На основании полученных данных было выполнено сравнение результатов работы тестеров с результатами работы алгоритмов семантического поиска (параграф 3.6). Для этого по аналогии с показателями полноты и точности рассчитывались показатели *формальной полноты* и *формальной точности*.

Формальная полнота поиска $R_F = \frac{a_{md}}{a_{md} + b_{md}}$, где (4.1)

a_{md} – количество релевантных семантических метаданных, выданных алгоритмом

b_{md} – количество релевантных семантических метаданных, не выданных алгоритмом

Таблица 4.6. Показатель формальной полноты алгоритмов семантического поиска

Тестер / Запрос	Тестер 1	Тестер 2	Тестер 3
1	1	1	1
2	1	1	1
3	1	1	1
4	1	1	1

Формальная точность поиска $P_F = \frac{a_{md}}{a_{md} + c_{md}}$, где (4.2)

a_{md} – количество релевантных семантических метаданных, выданных алгоритмом

c_{md} – количество нерелевантных семантических метаданных, выданных алгоритмом

Таблица 4.7. Показатель формальной точности алгоритмов семантического поиска

Тестер / Запрос	Тестер 1	Тестер 2	Тестер 3
1	1	1	1
2	1	0,818	1
3	1	0,882	1
4	1	1	1

Алгоритмы семантического поиска показали высокие результаты по формальной полноте и точности поиска. Однако их нельзя сравнивать с показателями полноты и точности полнотекстовых ИПС в силу того, что *формальные* показатели не учитывают *удовлетворенность* пользователей результатами поиска. В данном тестировании удовлетворенность не была оценена потому, что семантические метаданные документов были сгенерированы произвольным образом и не отражали текстового содержания документов. Тем не менее, при достижении высокой степени соответствия семантических метаданных текстовому содержанию документов можно ожидать высоких показателей полноты и точности поиска с помощью предложенных алгоритмов.

В результате тестирования зафиксировано снижение формальной точности поиска для запросов 2 и 3 при оценке вторым тестером (таблица 4.7). Анализ причин этого снижения выявил, что пользователи могут по-разному интерпретировать включение понятия из онтологии в поисковый запрос. Понятие в запросе может использоваться как для ссылки на термин из предметной области, так и для целенаправленного объединения нескольких понятий в группу. Например, в запросе указано понятие «Контроллер». Такой запрос можно интерпретировать как:

1. Найти документы о том, что такое контроллер и как он устроен, исключая документы с описаниями конкретных моделей контроллеров.
2. Найти документы обо всех конкретных моделях контроллеров, описанных в базе знаний.
3. Найти все документы, которые касаются контроллеров в общем и которые о частных моделях контроллеров.

Разработанные алгоритмы семантического поиска (параграф 3.6) учитывают только третий вариант. В связи с этим возникает необходимость дальнейшей доработки алгоритмов семантического поиска и связанных с ним методов вычисления близости семантических метаданных. Эти задачи выходят за рамки данной работы, но являются актуальными на перспективу.

Результаты работы тестеров также использовались для определения значений коэффициентов, необходимых при сравнении элементов онтологии во время семантического поиска (выражения 3.26, 3.30). Тестеры указывали сходство между элементами запроса и элементами семантических метаданных документов. На основании этих оценок для каждого типа пар сравниваемых элементов было получено среднее значение соответствующих коэффициентов (таблица 4.8).

Таблица 4.8. Коэффициенты для семантического поиска

№	Тип пары элементов	Коэффициент	Значение
1	экземпляр – понятие	$d_{ICC} \in (0;1]$	0,21
2	понятие – экземпляр	$d_{CIC} \in (0;1]$	0,96

Коэффициент d_{ICC} указывает на то, что тестеры в некоторых ситуациях сочли релевантными те семантические метаданные документов, в которых не упоминался искомый экземпляр. В свою очередь значение коэффициента d_{CIC} подтверждает наличие указанной выше проблемы интерпретации понятий в запросе. Указанные коэффициенты опосредованно используются при расчете показателя SM_{CS} (выражение 3.47), который позволяет ранжировать найденные объекты портала.

4.2.3. Тестирование функции категоризации

Функция категоризации основывается на тех же показателях семантической близости элементов онтологии, что и функция семантического поиска. Поэтому тестирование функции категоризации выполнялось с целью уточнения значений коэффициентов, полученных в результате тестирования семантического поиска.

Для этого трем тестерам было предложено 20 из 160 проаннотированных документов. На основании семантических метаданных документа тестер должен был выбрать рубрики (приложение 5), семантические метаданные которых соответствуют документу. Ограничением было лишь то, что документ

нельзя относить к двум или более *подчиненным* рубрикам. Например, документ может относиться к рубрикам 1 и 2.3, но не может относиться к рубрикам 1 и 1.4, так как последние являются *подчиненными*, и необходимо выбрать одну из подчиненных рубрик, которая больше соответствует семантическим метаданным документа.

В результате ручной категоризации для всех 20-ти документов было выбрано 49 различных рубрик. Причем по 43-м рубрикам мнения тестеров совпали, а по 6-и – нет. Получившееся несовпадение объясняется ранее выявленной неоднозначностью трактовки понятий в семантических метаданных.

Автоматическая категоризация выполнялась также с учетом указанного выше ограничения. Решение об отнесении документа к рубрике принималось на основании значения показателя близости SM_{CO} (выражение 3.45) между семантическими метаданными рубрики и документа. Выполнение автоматической категоризации с использованием ранее заданных коэффициентов показало полное совпадение результатов по указанным 43 рубрикам. По остальным 6 рубрикам результаты автоматической категоризации соответствовали результатам тех тестеров, которые интерпретировали понятия в метаданных так, как того требуют методы семантического сравнения элементов онтологии.

В результате тестирования коэффициенты остались неизменными, а также была подтверждена необходимость учета возможности различной интерпретации понятий в семантических метаданных.

4.2.4. Тестирование функции выработки рекомендации

Функция рекомендации объектов, похожих на исходный, тестировалась с целью выбора коэффициентов для семантического сравнения элементов онтологии, с целью установления порогового значения для рекомендации и с целью проверки качества отбора документов для рекомендации.

В целях тестирования трем тестерам было предложено 3 произвольных документа из 160-ти проаннотированных. Семантические метаданные каждого из трех предложенных документов сравнивались тестером с семантическими метаданными оставшихся 159-ти документов. При этом тестер указывал близость элементов семантических метаданных в диапазоне (0;1], а также указывал, считает ли он семантические метаданные схожими (да/нет).

На основании полученных оценок близости элементов семантических метаданных были рассчитаны коэффициенты, используемые в методе вычисления семантической близости элементов онтологии (выражения 3.24, 3.28). Значения коэффициентов, вычисленные методом усреднения, приведены в таблице 4.9.

Таблица 4.9. Коэффициенты для категоризации

№	Тип пары элементов	Коэффициент	Значение
1	экземпляр – понятие	$d_{ICF} \in (0;1]$	0,07
2	понятие – экземпляр	$d_{CIF} \in (0;1]$	0,49

Указанные значения коэффициентов были использованы для автоматического расчета схожести документов, выбранных тестерами, с остальными документами (рис. 4.18). Показателем схожести являлся показатель SM_{FO} (выражение 3.44).

Данные, полученные в результате автоматического расчета схожести, использовались для определения *порогового значения для рекомендации*. От этого значения зависит *качество* рекомендации, которое оценивалось аналогично поиску – через показатели формальной полноты и формальной точности. Чем больше пороговое значение для рекомендации, тем выше точность и меньше полнота. И наоборот, чем ниже пороговое значение для рекомендации, тем меньше точность и выше полнота.

Было принято решение обеспечить максимальное значение показателя *полноты* рекомендации. Поэтому при определении порогового значения для рекомендации вычислялось *минимальное значение показателя схожести документов*.

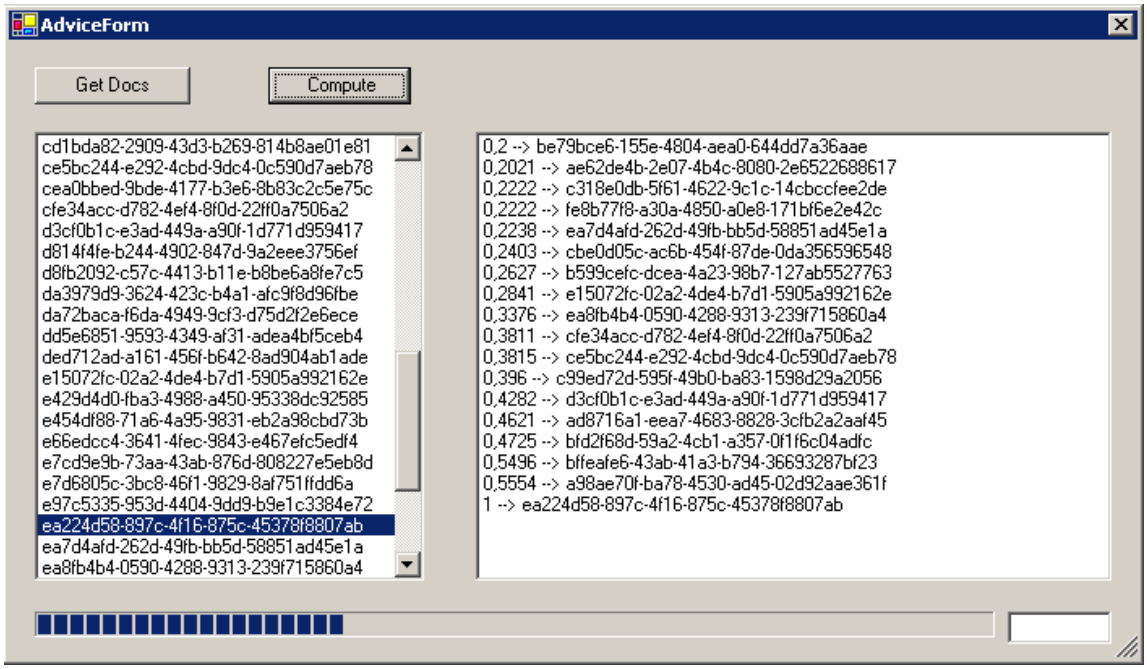


Рис. 4.18. Процесс автоматического расчета показателя схожести документов

Пусть $D = \{d_1,...,d_n\}$ – множество документов, а $MD_{DL} = \{MD_{DL}(d_1),...,MD_{DL}(d_n)\}$ – множество семантических метаданных этих документов. $MD_T(MD_{DL}(d_x))$ – множество семантических метаданных из множества MD_{DL}, которые тестер посчитал схожими с семантическими метаданными документа d_x. $MinSM(MD_{DL}(d_x))$ – минимальное значение показателя SM_{FO} при сравнении семантических метаданных документа d_x с семантическими метаданными из множества $MD_T(MD_{DL}(d_x))$.

В результате для каждого из девяти документов, обработанных тестерами, были получены минимальные значения показателя схожести $MinSM(MD_{DL}(d_x))$ (табл. 4.10).

Таблица 4.10. Значения показателя схожести документов

№	Тестер	Документ тестера	Количество схожих документов	Минимальное значение показателя схожести
1	1	1	5	0,4652
2	1	2	2	0,5052
3	1	3	1	0,4642
4	2	1	0	-
5	2	2	2	**0,4469**

6	2	3	3	0,492
7	3	1	7	0,5
8	3	2	2	0,5072
9	3	3	3	0,5389

В качестве порогового значения для рекомендации было выбрано значение 0.4469, являющееся *минимальным в тестовой выборке*. При данном значении были определены формальная полнота и формальная точность (рис. 19, 20, 21) для восьми тестовых документов (для одного тестового документа второй тестер не указал схожих документов).

Формальная полнота рекомендации $R_F = \frac{a_{md}}{a_{md} + b_{md}}$, где (4.3)

a_{md} – количество выданных алгоритмом семантических метаданных, которые тестер указал как схожие

b_{md} – количество не выданных алгоритмом семантических метаданных, которые тестер указал как схожие

Формальная точность рекомендации $P_F = \frac{a_{md}}{a_{md} + c_{md}}$, где (4.4)

a_{md} – количество выданных алгоритмом семантических метаданных, которые тестер указал как схожие

c_{md} – количество выданных алгоритмом семантических метаданных, которые тестер не указал как схожие

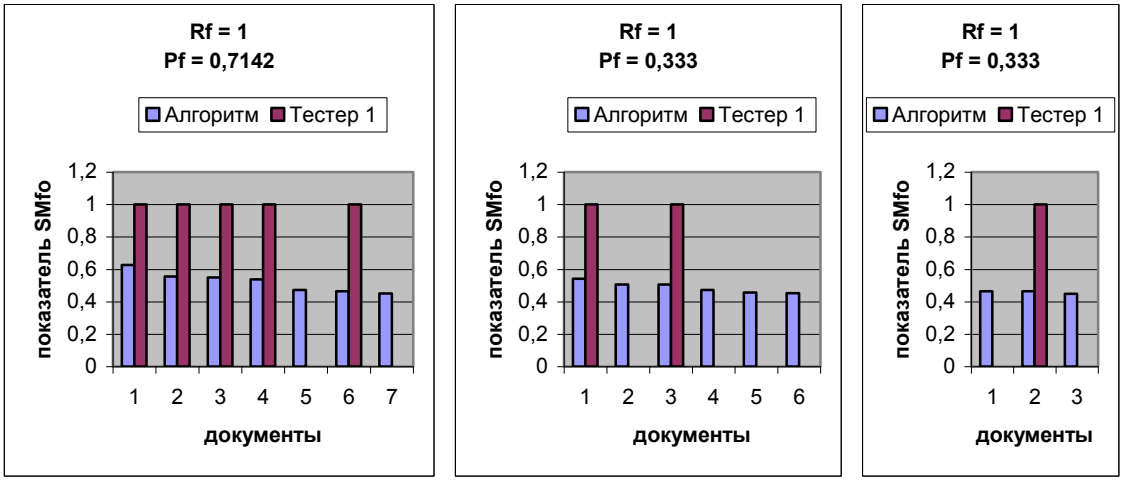


Рис. 4.19. Характеристики полноты и точности рекомендации по данным первого тестера

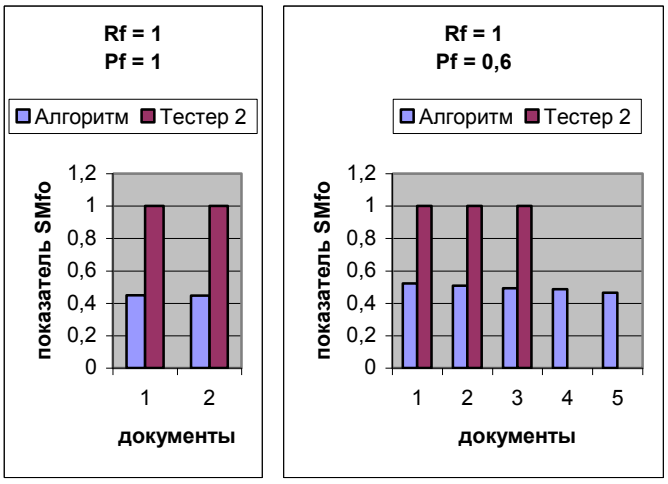


Рис. 4.20. Характеристики полноты и точности рекомендации по данным второго тестера

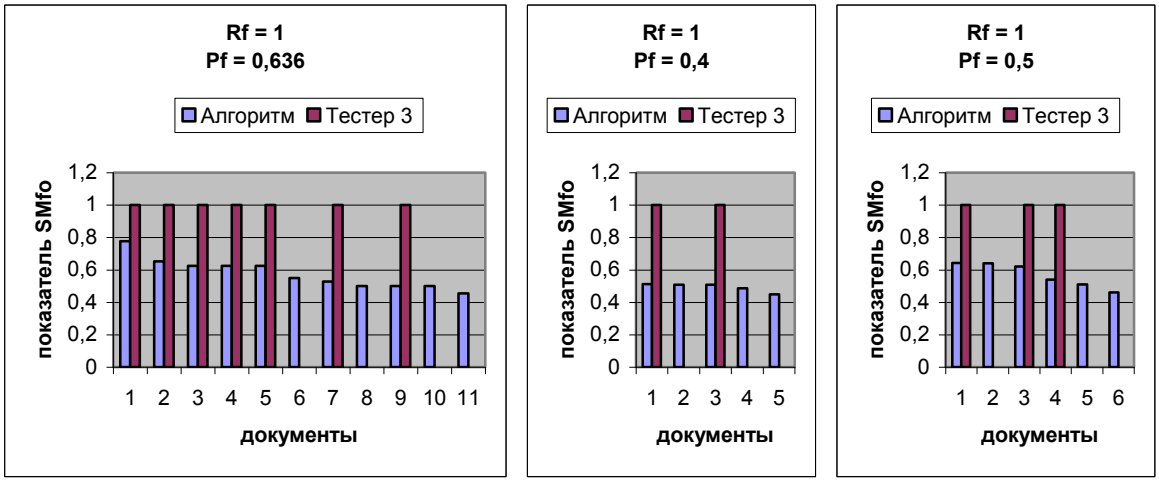


Рис. 4.21. Характеристики полноты и точности рекомендации по данным третьего тестера

Для тестовой выборки из 9-и документов при пороговом значении для рекомендации равным 0.4469 формальная полнота рекомендации равна 1, а формальная точность варьируется от 0.333 до 1 и в среднем составляет 0.5646.

4.3. Применение семантического ядра в порталах

Функции семантического ядра, основанные на разработанных методах и алгоритмах, применяются в двух Web-приложениях – в портале «Petroleum

Engineers Virtual Network», разработанном автором для «Центра профессиональной переподготовки специалистов нефтегазового дела» ТПУ, и в семантическом портале «Корпоративная система управления знаниями», разработанном автором для компании «ЭлеСи».

4.3.1. Портал «Petroleum Engineers Virtual Network»

Портал представляет собой программную систему управления явными и неявными знаниями для коллектива специалистов в области разработки нефтяных месторождений [26, 27, 28].

Рис. 4.22. Стартовая страница портала «Petroleum Engineers Virtual Network»

Система предоставляет следующие функции, обеспечивающие управление неявными знаниями:

- доступ к информации о специалистах в системе;
- поиск специалистов владеющих знаниями по искомой проблеме;
- взаимодействие специалистов посредством электронной почты;
- взаимодействие специалистов посредством online-общения;
- взаимодействие специалистов посредством дискуссий;
- автоматическое сохранение и доступ других сотрудников к результатам взаимодействия специалистов;
- ранжирование специалистов, в зависимости от их активности в процессе распространения знаний;
- курирование экспертами процесса обмена знаниями.

Система предоставляет следующие функции, обеспечивающие управление явными знаниями:

- хранение документов и ссылок на другие источники информации;
- добавление документов и ссылок на другие источники информации;
- простой и расширенный поиск по имеющимся документам;
- наличие рубрикатора хранимых знаний;
- ранжирование документов, в зависимости от частоты обращения к ним;
- получение пользователями новостей по интересующей их тематике.

Для описания явных и неявных знаний в системе используется словарь предметной области – тезаурус. Тезаурус содержит термины на русском и английском языках. Между терминами могут быть заданы отношения «перевод», «синоним», «близкий». Термины словаря используются для описания

профилей деятельности специалистов, а также для описания документов и ссылок (рис. 4.23).

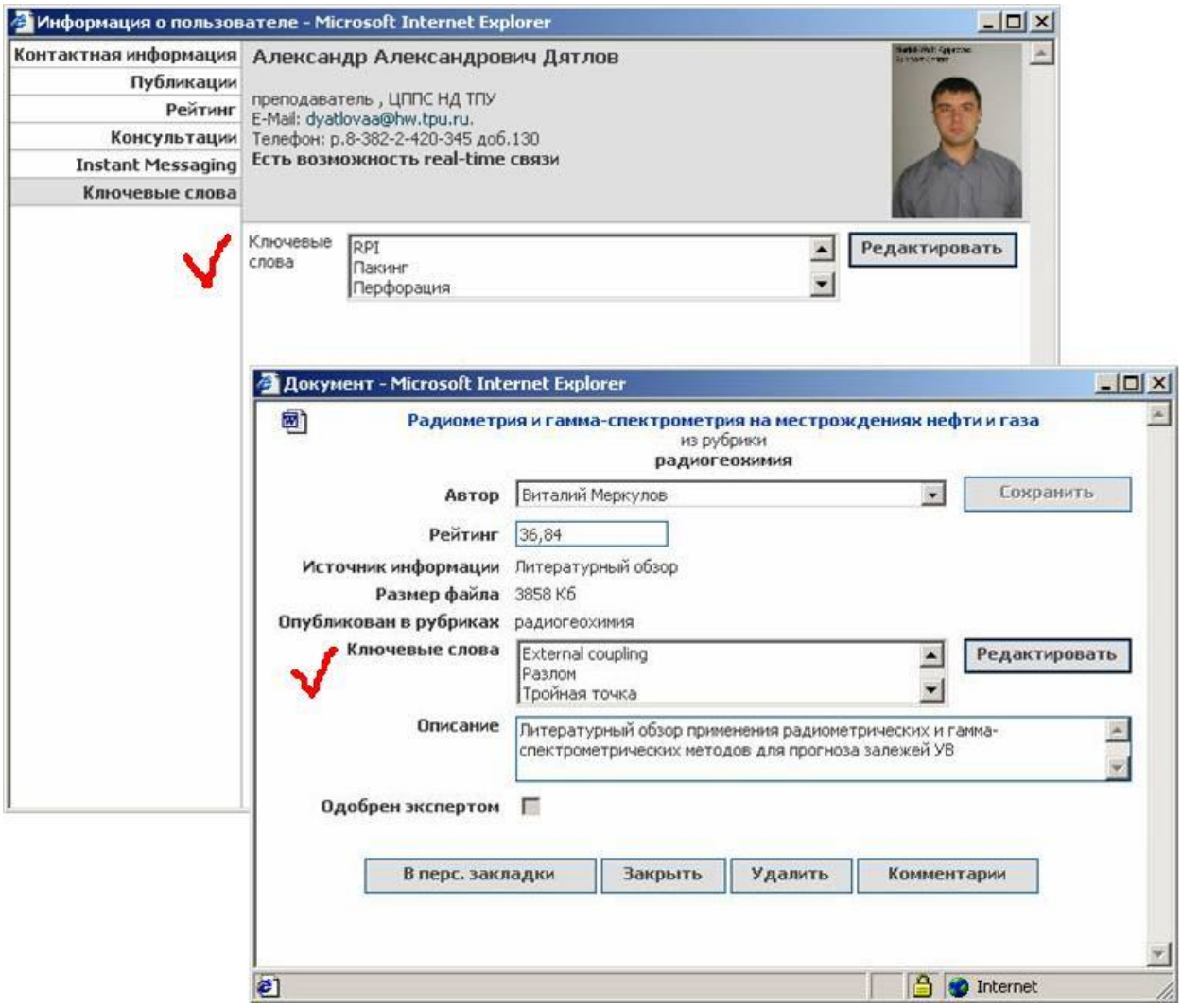


Рис. 4.23. Пример описания документа и специалиста с помощью терминов из тезауруса

В рамках данного портала был опробован алгоритм, вошедший в программную реализацию разработанного семантического ядра. А именно, при автоматическом составлении описания документа использовался модифицированный алгоритм поиска лексических меток в тексте документа (параграф 3.2). В данной реализации результатом работы алгоритма являются не понятия и экземпляры из онтологии, а термины из тезауруса.

Данный портал был внедрен (приложение 6) в Центре профессиональной переподготовки специалистов нефтегазового дела ТПУ. Программная

реализация портала была зарегистрирована (приложение 6) в Отраслевом фонде алгоритмов и программ.

4.3.2. Портал «Корпоративная система управления знаниями»

Корпоративная система управления знаниями разрабатывается в настоящее время для компании «ЭлеСи». В качестве программно-технической части системы разработан и внедрен (приложение 6) семантический портал, функциональность которого основывается на использовании разработанного семантического ядра.

В качестве наиболее существенных функциональных подсистем портала реализуются:

- подсистема электронной библиотеки;
- подсистема профилей компетенции ведущих сотрудников компании;
- подсистема поиска;
- подсистема для работы экспертов компании.

Разработанная для системы управления знаниями онтология части предметной области «Автоматизация технологических процессов» помещена в хранилище онтологий. Каждая из перечисленных подсистем основывается на разработанной онтологии.

Подсистема электронной библиотеки предназначена для централизации доступа к информационным хранилищам компании. Для этого подсистема предоставляет средства формирования структуры рубрик каталога и описания хранящихся документов и ссылок. Рубрики каталога, документы и ссылки описываются семантическими метаданными. Это позволяет выполнять категоризацию информационных объектов с использованием функции семантического ядра. Причем под документами понимаются файлы, хранимые внутри портала, а ссылки указывают на прочие источники информации в компании, как электронные, так и на твердых носителях.

Подсистема профилей компетенции предназначена для формирования структуры областей знаний компании и описания компетенции сотрудников. Области знаний и сотрудники описываются семантическими метаданными, что позволяет использовать семантическое ядро для группировки специалистов по областям знаний.

В портале на семантическом уровне описаны такие объекты как рубрики каталога, документы, ссылки, области знаний, специалисты и эксперты. *Подсистема поиска* использует семантическое ядро для реализации семантического поиска всех указанных типов объектов.

За определенными областями знаний в компании закреплены эксперты. *Среда для работы экспертов* сосредотачивает в одном месте средства, необходимые экспертам для распространения знаний по определенному направлению. Среди этих средств персональная электронная библиотека, дневник новостей (блог), персональная область для дискуссий и размещения ответов на часто задаваемые вопросы.

Портал реализуется с использованием технологии ASP.NET. Для интеграции семантического ядра в портал разработан базовый класс для Web-форм и базовый класс для объектов портала.

Базовый класс для Web-форм расширяет системный класс System.Web.UI.Page для того, чтобы каждая серверная страница портала имела доступ к функциям семантического ядра, сосредоточенным в сервере онтологий и сервере семантических метаданных (рис. 4.24).

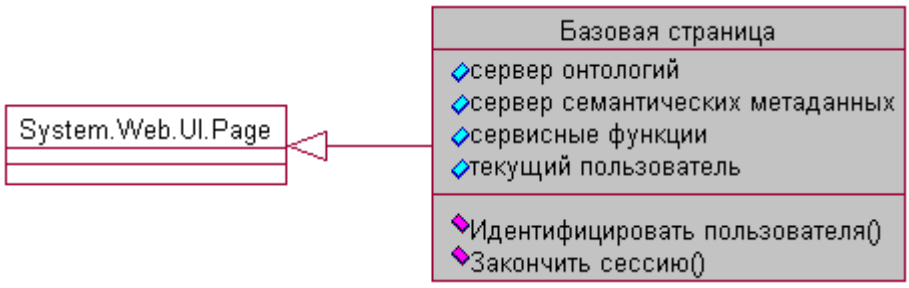


Рис. 4.24. UML-диаграмма базового класса для серверных страниц портала

Базовый класс для объектов портала определяет программный интерфейс доступа к семантическим описаниям соответствующих объектов портала (рис. 4.25).

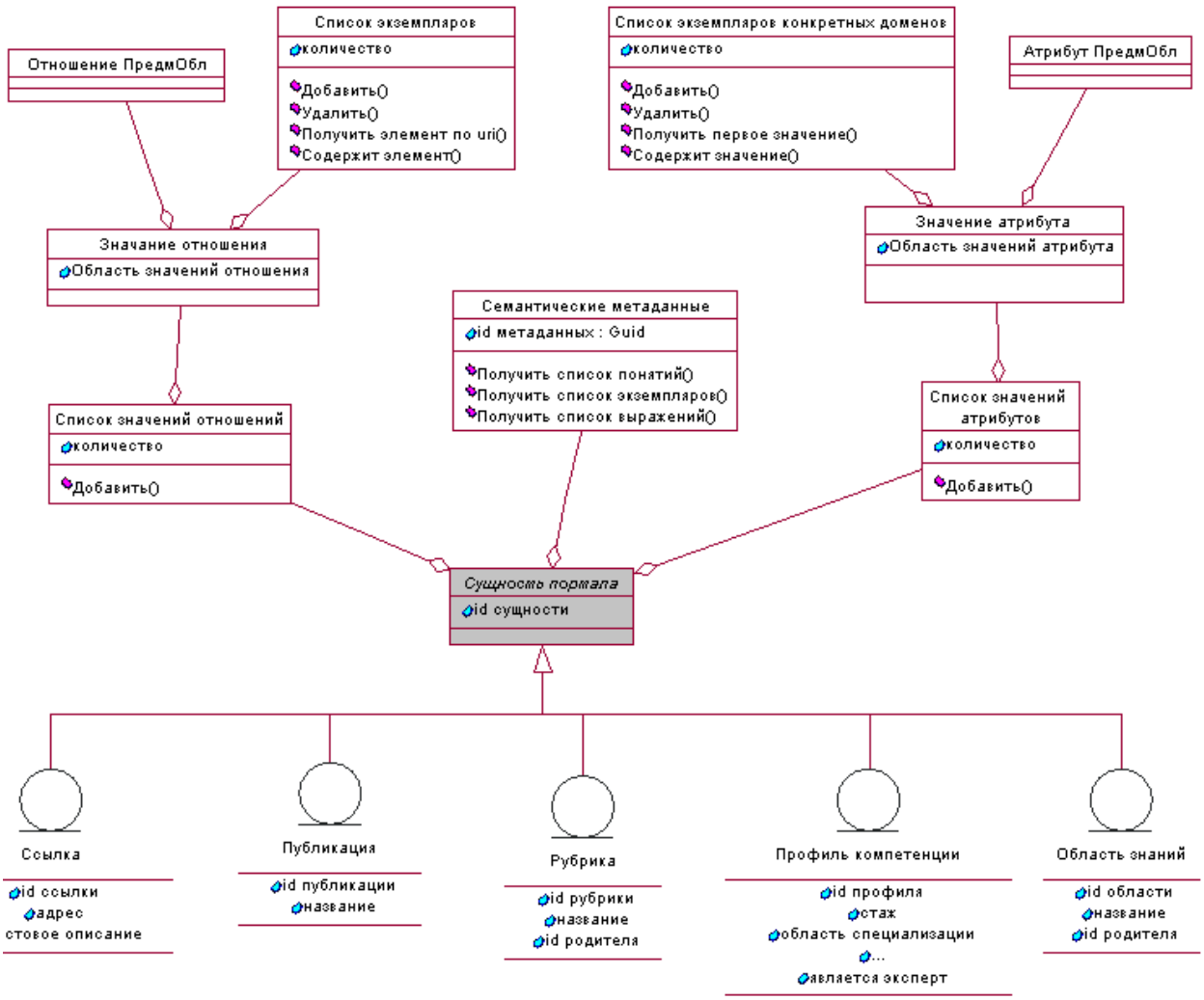


Рис. 4.25. UML-диаграмма базового класса для объектов портала

Семантические описания таких объектов состоят из описания сущности объекта в виде семантических метаданных (метаданные контента) и описания связей объекта с другими объектами (метаданные контекста). Поисковый образ объекта формируется путем объединения двух составляющих.

Выводы по главе

1. Разработанные компоненты семантического ядра реализуют предложенные методы создания и обработки семантических метаданных с использованием технологии .NET Remoting, что позволяет обеспечить независимость семантического ядра от остальных элементов инфраструктуры портала.

2. Реализуемые семантическим ядром методы тестировались с применением метода экспертных оценок. В ходе компьютерного и экспертного тестирования были зафиксированы высокие показатели качества выполнения алгоритмических процедур поиска, категоризации и предоставления рекомендаций.

3. Результаты исследований современного уровня развития семантических технологий, разработка функциональной структуры и программная реализация компонентов семантического ядра способствуют практическому решению задачи создания семантического портала.

ЗАКЛЮЧЕНИЕ

Диссертация посвящена решению научно-технической задачи разработки методов и инструментальных средств для создания семантических Web-порталов. Разработанное по результатам исследований семантическое ядро портала, реализующее предложенные методы формирования и обработки семантических метаданных объектов портала, может служить основой для создания семантических порталов в различных предметных областях.

В ходе диссертационного исследования получены следующие основные результаты:

1. Выполнен анализ существующих подходов к разработке семантических порталов. Выявлена доминирующая роль онтологического подхода к созданию семантических порталов. Показано, что с помощью онтологий может решаться широкий круг задач повышения качества работы информационных систем.

2. Проведен анализ и обобщение возможных вариантов использования онтологий в информационных системах. Для реализации информационных процессов в портале с учетом семантики объектов предложены варианты использования онтологии.

3. Разработан состав и структура семантического ядра портала. Ядро состоит из сервера онтологий и сервера семантических метаданных. Функциональность семантического ядра портала основывается на логическом формализме представления знаний – дескриптивной логике. В соответствии с указанным формализмом выбраны языки записи онтологии и семантических метаданных для использования в семантическом ядре портала. Обоснована структура онтологии, обеспечивающая работу семантического ядра портала.

4. Разработан метод формирования семантических метаданных для создания описаний объектов портала. Разработаны методы вычисления семантической близости элементов онтологии и метаданных, формализующие ис-

пользование понятия сотипности. Указанные методы применены в функциях семантического поиска, категоризации и формирования рекомендаций.

5. Выполнена программная реализация разработанного семантического ядра портала, составившая в общей сложности более 16 тысяч строк кода. Осуществлено тестирование программного кода на сгенерированном множестве семантических метаданных.

6. Разработанные структуры, методы и алгоритмы построения семантических Web-порталов, а также соответствующее программное обеспечение, внедрены в двух организациях (ЗАО «ЭлеСи», Центр профессиональной переподготовки специалистов нефтегазового дела ТПУ) при создании для них семантических порталов различного уровня.

СПИСОК ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ

1. IBM case studies for WebSphere software [Электронный ресурс]. – Режим доступа: http://www-306.ibm.com/software/success/cssdb.nsf/customerVW?OpenView&Start=1&Count=1000&ExpandView&RestrictToCategory=wssoftware

2. Поляков В. Н. Интеллектуальная поисковая машина. Концептуальный проект // Труды Казанской школы по компьютерной и когнитивной лингвистике TEL-2000 17–20 октября 2000 г. – Казань: изд-во «Сэлэт», 2000. – Выпуск 5. – С. 87-119.

3. Поляков В. Н., Бодров Д. А. Проблемы создания эффективных поисковых машин // Обработка текста и когнитивные технологии. Сборник научных статей. – 2002. – Выпуск 7. – С. 48-71.

4. Нариньяни А. С. Кентавр по имени ТЕОН: Тезаурус + Онтология // Труды международного семинара «Диалог'2001» по компьютерной лингвистике и ее приложениям. – 2001. – Том 1. – С. 184-188.

5. Россеева О. И., Загорулько Ю. А. Организация эффективного поиска на основе онтологий // Труды международного семинара «Диалог'2001» по компьютерной лингвистике и ее приложениям. – 2001. – Том 2. – С. 333-342.

6. Боровикова О. И., Загорулько Ю. А. Организация порталов знаний на основе онтологий // Труды международного семинара «Диалог'2002» по компьютерной лингвистике и интеллектуальным технологиям. – 2002. – Том 2. – С. 76-82.

7. Mizoguchi R. A step towards ontological engineering // Proc. of the 12th National Conference on AI of JSAI. – 1998. – P. 24-31.

8. Borst W. N. Construction of engineering ontologies for knowledge sharing and reuse. PhD Thesis. University of Tweenty, Enschede, Netherlands. Centre for Telematica and Information Technology. – 1997. – 243 p.

9. Guarino N. Understanding, building and using ontologies // International Journal of Human-Computer Studies, February/March 1997. – Volume 46. – Issue 2-3. – P. 293-310.

10. Takeda H. Ontologies [Электронный ресурс]: презентация. – Режим доступа: http://www-kasm.nii.ac.jp/~takeda/lectures/soken/ontologies-for-lecture04.pdf

11. Uschold M., Gruninger M. Ontologies: principles, methods and applications // Knowledge Engineering Review. – June 1996. – Volume 11(2). – P. 93-113.

12. Gruber T. R. Towards principles for the design of ontologies used for knowledge sharing // International Journal of Human-Computer Studies. – 1995. – Volume 43. – P. 907-928.

13. Studer R., Benjamins V. R., Fensel D. Knowledge engineering: principles and methods // Proc. of the conference on data and knowledge engineering. – 1998. – Volume 25. –Issue 1-2. – P. 161-197.

14. Gomez-Pérez A. Evaluation of ontologies // International journal of intelligent systems. – 2001. – Volume 16(3). – P. 391-409.

15. Staab S., Angele J., Decker S. et al. Semantic community web portals // Proc. of the 9th international World Wide Web conference. – Amsterdam: Elsevier Science, 2000. – P. 473-491.

16. Spyns P., Oberle D., Volz R. et al. OntoWeb – a Semantic Web community portal // Proc. of 4th international conference on practical aspects of knowledge management. – 2002. – P. 189-200.

17. Woukeu A., Wills G., Conole G. et al. Ontological hypermedia in education: A framework for building web-based educational portals // Proc. of world

conference on educational multimedia, hypermedia and telecommunications. – 2003. – P. 349-357.

18. Corcho O., Gómez-Pérez A., López-Cima A. et al. ODESeW: Automatic generation of knowledge portals for intranets and extranets // Proc. of the 2nd international Semantic Web conference. – 2003. – Volume 2870. – P. 802-817.

19. Suárez-Figueroa M. C., García-Castro R., Gómez-Pérez A. et al. Identification of standards on metadata for ontologies. KnowledgeWeb Deliverable D1.3.2 [Электронный ресурс]. – 2005. – Режим доступа: http://knowledgeweb.semanticweb.org/semanticportal/servlet/download?ontology=Documentation+Ontology&concept=Deliverable&instanceSet=kweb&instance=D1.3.2%3A+Identification+of+standards+on+metadata+for+ontologies&attribute=On-line+PDF+Version&value=D1.3.2-vFinal.pdf

20. Тузовский А. Ф., Васильев И. А. Структура системы управления знаниями // Труды международного симпозиума «Информационные и системные технологии в индустрии, образовании и науке. – Караганда: Издательство КарГТУ, 2003. – С. 286-288.

21. Тузовский А. Ф., Васильев И. А., Усов М. В. Программная реализация основных компонент информационно-программного обеспечения системы управления знаниями // Известия ТПУ. – 2004. – Том 307. – №7. – С. 116-122.

22. Васильев И. А., Усов М. В. Применение онтологического подхода в информационных системах // Труды X-ой международной научно-практической конференции студентов, аспирантов и молодых ученых «Современные техника и технологии 2004». – Томск: Изд-во ТПУ, 2004. – Том 2. – С. 123-124.

23. Усов М. В., Васильев И. А. Роль дескриптивной логики в порталах управления знаниями // Труды X-ой международной научно-практической конференции студентов, аспирантов и молодых ученых

«Современные техника и технологии 2004». – Томск: Изд-во ТПУ, 2004. – Том 2. – С. 212-213.

24. Васильев И. А. Выбор средства представления знаний для их использования в работе информационного портала организации // Материалы XLII Международной научной студенческой конференции «Студент и научно-технический прогресс»: Информационные технологии. – Новосибирск: Изд-во НГУ, 2004. – С. 56-60.

25. Васильев И. А. Оценка семантической близости объектов с использованием дескриптивной логики // Материалы 5-ой научно-практической конференции «Современные средства и системы автоматизации». – Томск: Изд-во ТУСУР, 2004. – С. 160-163.

26. Васильев И. А. Организация коллективной работы пользователей с документами в сети Internet // Материалы XL Международной научной студенческой конференции «Студент и научно-технический прогресс». – Новосибирск: Изд-во НГУ, 2002. – С. 31-32.

27. Васильев И. А., Бубнов Д. В., Козлов С. В. Использование онтологии предметной области для поддержки работы сложных технических систем // Труды IX-ой международной научно-практической конференции студентов, аспирантов и молодых ученых «Современные техника и технологии». – Томск: Изд-во ТПУ, 2003. – Том 2. – С. 144-145.

28. Васильев И. А., Бубнов Д. В., Козлов С. В. Использование сети Интернет для активизации взаимодействия специалистов предметной области // Труды Всероссийской научно-практической конференции «Технологии ИНТЕРНЕТ – на службу обществу» – Саратов: Копипринтер, 2003. – С. 164-166.

29. Корпоративные информационные порталы [Электронный ресурс]. – Режим доступа: http://www.e-commerce.ru/biz_tech/implementation/management/corp_portals.html

30. Черняк Л. Корпоративный портал // Компьютерная неделя. – 1999. – № 31(205). – С. 30-35.

31. Технологии IBM для электронного бизнеса / Итоговый отчет за 2003 г. – М.: IBM, 2003. – 232 с.

32. Phifer G., Valdes R., Gootzit D. et el. Magic quadrant for horizontal portal products [Электронный ресурс]. – 2004. – Режим доступа: http://www.g2r.com/DisplayDocument?doc_cd=120327

33. SungKook Han. Commercial portal products. Semantic Web community portal project. DERI Research Report 2003-12-31 [Электронный ресурс]. – 2003. – Режим доступа: http://sw-portal.deri.org/papers/deliverables/CommercialPortal.pdf

34. Shilakes C., Tylman J. Enterprise Information Portals. – Merrill Lynch Inc., 1998.

35. Firestone J. M. Enterprise information portals and knowledge management. – Oxford: Butterworth-Heinemann, 2003. – 422 p.

36. Eckerson W. Business portals: Drivers, definitions and rules. – The data warehousing institute. – 1999.

37. White C. The enterprise information portal marketplace. Decision processing brief DP-99-01. – Database Associates International Inc. – 1999.

38. Murray G. The Portal is the desktop. – Intraspect Inc. – 1999.

39. Mercy J. A better understanding of the enterprise information portal market [Электронный ресурс]. – Режим доступа: http://intranetjournal.com/articles/200110/eip_10_03_01a.html

40. WebSphere portal server from IBM [Электронный ресурс]. – Режим доступа: http://mithras.itworld.com/download/bloorwebportal.pdf

41. A developer's introduction to web parts [Электронный ресурс]. – Режим доступа: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odc_sp2003_ta/html/sharepoint_northwindwebparts.asp

42. TopQuadrant Technology briefing. Semantic technology [Электронный ресурс]. – 2004. – Режим доступа: http://www.topquadrant.com/documents/TQ04_Semantic_Technology_Briefing.PDF

43. Ding Y., Fensel D., Klein M., Omelayenko B. The Semantic Web: Yet another hip? // Proc. of conference on data and knowledge engineering. – 2002. – Volume 41(3). – P. 205-227.

44. Wielinga B. J., Schreiber A. T. Reusable and sharable knowledge bases: a European perspective // Proc. of international conference on building and sharing of very large-scaled knowledge bases. – 1993. – P. 103-115.

45. Guarino N. Semantic matching: formal ontological distinctions for information organization, extraction and integration // In Pazienza M.T., Information extraction: a multidisciplinary approach to an emerging information technology. – NY: Springer. – 1997. – P. 139-170.

46. Gomez-Pérez A., Fernandez-Lopez M., Corcho O. Ontological engineering with examples from the areas of knowledge management, e-Commerce and the Semantic Web. – NY: Springer, 2004. – 410 p.

47. Sowa J. F. Knowledge representation: logical, philosophical and computational foundations. – CA: Brooks Cole Publishing Co, 2000. – 512 p.

48. Fellbaum C. WordNet: An electronic lexical database (language, speech and communication). – The MIT Press, 1998. – 423 p.

49. Towards a methodology for ontology-driven conceptual modeling. Ontological analysis of taxonomic relationships [Электронный ресурс]. – Режим доступа: http://lisi.insa-lyon.fr/~jpierson/lisi-seminaires/2000-2001/download/guarino-051000.pdf

50. Brachman R. J., Schmolze J. G. An overview of the KL-ONE knowledge representation system // Cognitive Science. – 1985. – Volume 9. – №2. – P. 171-216.

51. Brachman R. J., Fikes R. E., Levesque H. J. KRYPTON: A functional approach to knowledge representation // IEEE COMPUTER. – 1983. – Volume 16(10). – P. 67-73.

52. MacGregor R., Bates R. The Loom knowledge representation language // Proc. of the knowledge-based systems workshop. – 1987. – P. 17-29.

53. Borgida A., Brachman R. J., McGuinness D. L., Resnick L. A. CLASSIC: A structural data model for objects // Proc. of the ACM SIGMOD international conference on management of data. – 1989. – P. 59-67.

54. Fikes R., Farquhar A., Rice J. Tools for assembling modular ontologies in Ontolingua // Proc. of the 14th national conference on Artificial Intelligence. – 1997. – P. 436-441.

55. Kifer M., Lausen G. F-Logic: A higher-order language for reasoning about objects, inheritance, and scheme // Proc. of the ACM SIGMOD international conference on management of data. – 1989. – P. 134-146.

56. Heflin J., Hendler J., Luke S. SHOE: A knowledge representation language for Internet applications. Technical Report [Электронный ресурс]. – 1999. – Режим доступа: http://www.cs.umd.edu/projects/plus/SHOE/pubs/techrpt99.pdf

57. RDF Vocabulary Description Language 1.0: RDF Schema [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/rdf-schema

58. OWL Web Ontology Language. Overview [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/owl-features/

59. The Learning Object Metadata standard [Электронный ресурс]. – Режим доступа: http://ieeeltsc.org/wg12LOM/lomDescription

60. MARC standards [Электронный ресурс]. – Режим доступа: http://www.loc.gov/marc

61. Application profile for the government information locator service (GILS) [Электронный ресурс]. – Режим доступа: http://www.gils.net/prof_v2.html

62. Standard element set for GELOS records [Электронный ресурс]. – Режим доступа: http://www.iszp.sk/katalog/gelos.html

63. Vallet D., Fernandez M., Castells P. An ontology-based information retrieval model // Proc. of the 2nd European Semantic Web conference. – NY: Springer, 2005. – P. 455-470.

64. Lassila O., Swick R.R. Resource Description Framework (RDF) Model and Syntax Specification [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

65. Makelä E., Hyvönen E., Saarela S., Viljanen K. OntoView – A tool for creating Semantic Web portals // Proc. of the 3rd international Semantic Web conference. – 2004. – P. 797-811.

66. Lei Y.-G., Motta E., Domingue J. OntoWeaver: an ontology-based approach to the design of data-intensive Web sites // Journal of Web Engineering. – 2005. – Volume 4. – №3. – P. 244-262.

67. Chenxi L., Lei Z., Jian Z., Ying Y., Yong Y. SPortS: Semantic + Portal + Service // Proc. of the ECAI 2004 Workshop on Application of Semantic Web Technologies to Web Communitites. – 2004. – Volume 107. – P. 161-173.

68. Mondeca ITM White Paper [Электронный ресурс]. – Режим доступа: http://www.mondeca.com/itm-wp-introduction-en.pdf

69. Vatant B. Ontology-driven topic maps [Электронный ресурс]. – 2004. – Режим доступа: http://www.idealliance.org/europe/04/call/xmlpapers/03-03-03.91/.03-03-03.html

70. Jin Y., Decker S., Widerhold G. OntoWebber: Model-driven ontology-based Web site management // Proc. of the 1st international Semantic Web working symposium. – 2001. – P. 529-547.

71. Zhdanova A. V., Henke J., Bachlechner D. et al. SW-Portal Prototype: Semantic DERI Use Case [Электронный ресурс]. – Режим доступа: http://www.deri.at/research/projects/sw-portal/papers/deliverables/d15.pdf

72. Agarwal S., Fankhauser P., Gonzalez-Ollala J. Semantic methods and tools for information portals // Proc. of the GI Jahrestagung conference. – Volume 1. – 2003. – P. 116-131.

73. Reynolds D., Shabajee P., Cayzer S., Steer D. Semantic portals demonstrator – lessons learnt. SWAD-Europe deliverable 12.1.7 [Электронный ресурс]. – Режим доступа: http://www.w3.org/2001/sw/Europe/reports/demo_2_report/

74. Guarino, N. Formal ontology and information systems // Proc. of the 1st international conference on formal ontology in information systems. – 1998. – P. 3-15.

75. Nebel B. Belief revision and default reasoning: syntax-based approaches // Proc. of the 2nd international conference on principles of knowledge representation and reasoning. – 1991. – P. 417-428.

76. Heflin J. Towards the Semantic Web: Knowledge representation in a dynamic, distributed environment. PhD thesis. University of Maryland, College Park, USA. – 2001. – 146 p.

77. Partridge C. Business objects: Re-engineering for reuse. – Oxford: Butterworth-Heinemann, 1996. – 453 p.

78. Ceri S., Fraternali P. Designing database applications with objects and rules: The IDEA methodology. – Addison Wesley, 1997. – 593 p.

79. Bergamaschi S., Castano S., De Capitani di Vimercati S., Montanari S., Vincini M. An intelligent approach to information integration. / In Guarino N. Formal Ontology in Information Systems. – IOS Press. – 1998.

80. Wiederhold G. Intelligent integration of information. – Boston: Kluwer Academic Publishers, 1996. – 216 p.

81. Snoussi H., Magnin L., Nie J.-Y. Toward an ontology-based Web data extraction // Proc. of the 15th conference of the Canadian society for computational studies of intelligence. – 2002. – P. 26-33.

82. Su X., Matskin M., Rao J. Implementing explanation ontology for agent system // Proc. of international conference on Web intelligence. – 2003. – P. 330-336.

83. Hartmann J., Sure Y. An infrastructure for scalable, reliable semantic portals // IEEE Intelligent Systems. – 2004. – Volume 19. – Issue 3. – P. 58-65.

84. Blythe J., Gil Y. Incremental formalization of document annotations through ontology-based paraphrasing // Proc. of the 13th international conference on World Wide Web. – 2004. – P. 455-461.

85. Cimiano P., Handschuh S., Staab S. Towards the self-annotating Web // Proc. of the 13th international conference on WWW. – 2004. – P. 462-471.

86. Hyvönen E., Saarela S., Viljanen K. Application of ontology techniques to view-based semantic search and browsing // Proc. of the 1st European Semantic Web symposium. – NY: Springer, 2004. – P. 92-106.

87. Khan L. R. Ontology-based information selection. PhD thesis. Faculty of the graduate school University of Southern California, California, USA. – 2000. – 129 p.

88. Park J., Lee D. An adaptive agent-based framework for knowledge management and sharing [Электронный ресурс]. – 2001. – Режим доступа: http://misrc.umn.edu/workingpapers/fullPapers/2001/0128_040101.pdf

89. Clark P., Thompson J., Holmback H., Duncan L. Exploiting a thesaurus-based semantic net for knowledge-based search // Proc. of the 12th conference on innovative applications of AI. – 2000. – P. 988-995.

90. Van Heijst G., Schreiber A. T., Wielinga B. J. Using explicit ontologies in KBS development // International journal of human and computer studies. – 1997. – Volume 46. – P. 183-292.

91. Тузовский А. Ф., Чириков С. В., Ямпольский В. З. Системы управления знаниями. Методы и технологии. – Томск: Изд-во НТЛ, 2005. – 260 с.

92. Semantic Portals – Requirements Specification. SWAD-Europe deliverable 12.1.5 [Электронный ресурс]. – Режим доступа: http://www.w3.org/2001/sw/Europe/reports/requirements_demo_2/

93. McGuinness D. L., Fikes R., Stein L. A., Hendler J. DAML-ONT: An ontology language for the Semantic Web / In Fensel D., Hendler J., Lieberman H., Wahlster W. Spinning the Semantic Web: Bringing the World Wide Web to its full potential. – Massachusetts: MIT Press, 2003. – 479 p.

94. Horrocks I., Fensel D., Broekstra J. et al. The Ontology Inference Layer OIL [Электронный ресурс]. – Режим доступа: http://www.cs.vu.nl/~dieter/oil/Tr/oil.pdf

95. McGuinness D. L., Fikes R., Hendler J., Stein L. A. DAML+OIL: An ontology language for the Semantic Web // IEEE Intelligent Systems. – 2002. – Volume 17. – №5. – P. 72-80.

96. The Description Logic handbook: theory, implementation, applications / ed. Baader F. – Cambridge: Cambridge University Press, 2003. – 564 p.

97. Bruijn J. Using ontologies: Enabling knowledge sharing and reuse on the Semantic Web. DERI Technical Report DERI-2003-10-29 [Электронный ресурс]. – 2003. – Режим доступа: http://homepage.uibk.ac.at/~c703239/publications/DERI-TR-2003-10-29.pdf

98. Unicode Standard [Электронный ресурс]. – Режим доступа: http://www.unicode.org/unicode/standard/standard.html

99. RFC 1630. Universal Resource Identifiers in WWW [Электронный ресурс]. – Режим доступа: http://www.ietf.org/rfc/rfc1630.txt?number=1630

100. Спенсер П. XML. Проектирование и реализация. – М.: Лори, 2001. – 510 с.

101. XML Schema Part 1: Structures [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/structures.html

102. Namespaces in XML [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/REC-xml-names

103. Шенк Р. Обработка концептуальной информации. – М.: Энергия, 1980. – 360 с.

104. Masolo C., Borgo S., Gangemi A. et al. Ontology Library. WonderWeb Deliverable D18 [Электронный ресурс]. – Режим доступа: http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf

105. Bateman J. A., Henschel R., Rinaldi F. The Generalized Upper Model 2.0 [Электронный ресурс]. – Режим доступа: http://www.fb10.uni-bremen.de/anglistik/langpro/webspace/jb/gum/gum-2.pdf

106. Niles I., Pease A. Origins of the Standard Upper Merged Ontology: A proposal for the IEEE Standard Upper Ontology [Электронный ресурс]. – 2001. – Режим доступа: http://projects.teknowledge.com/IJCAI01/Niles.ps

107. Белоногов Г. Г., Зеленков Ю. Г. Алгоритм морфологического анализа русских слов // Вопросы информационной теории и практики. – 1985. – № 53. – С. 62-93.

108. Mädche A., Staab S., Stojanovic N. et al. SEAL – A framework for developing SEmantic portALs // Proc. of the 18th British national conference on databases. – Oxford: Springer, 2001. – P. 1-22.

109. Levenshtein I. V. Binary codes capable of correcting deletions, insertions, and reversals // Cybernetics and Control Theory. – 1966. – Volume 10(8). – P. 707-710.

110. Ukkonen E. Approximate string matching with q-grams and maximal matches // Theoretical Computer Science. – 1992. – Volume 92(1). – P. 191-211.

111. Кнут Д. Искусство программирования. – М.: Вильямс, 2000. – Том 3. – 703 с.

112. Каньковски П. «Как ваша фамилия?» или русский MetaPhone // Программист. – 2002. – №8. – С. 36-39.

113. Маклин С., Нафтел Дж., Уильямс К. Microsoft .NET Remoting / пер. с англ. – М.: Издательско-торговый дом «Русская редакция», 2003. – 384 с.

114. Haarslev V., Möller R. RACER: A core inference engine for the Semantic Web // Proc. of the 2nd international workshop on evaluation of ontology-based tools. – 2003. – P. 27-36.

115. The DIG Description Logic Interface: DIG/1.1 [Электронный ресурс]. – Режим доступа: http://dl-web.man.ac.uk/dig/2003/02/interface.pdf

116. Боггс У., Боггс М. UML и Rational Rose. – М.: Лори, 2000. – 582 с.

117. Отчет по договору «Разработка проекта и базовых элементов системы управления знаниями компании ЭлеСи». Этап: Построение онтологии на конкретном примере предметной области компании «ЭлеСи». х/д № 8-22/04 от 30 ноября 2004 г.

ПРИЛОЖЕНИЯ

Приложение 1. Краткая характеристика порталов уровня предприятия

Таблица 5.1. Краткая характеристика порталов уровня предприятия, являющихся лидерами данного сектора рынка по состоянию на 2004 год

Хар-ки / ИПП	Поддерживаемые функции	Идентификация пользователей	Среда разработки	Возможность интеграции	Поддержка стандартов
BEA WebLogic Portal 8.1	поиск, управление содержанием, взаимодействие пользователей, доступ с мобильных устройств, электронная коммерция, маркетинговая компания	LDAP, SSPI	BEA WebLogic Workshop, Borland JBuilder	Microsoft Exchange, Lotus Notes, Oracle, SQL Server, Informix, Sybase, DB2 и пр.	J2EE, JSR 168, Struts, WSRP, XML, XMLBeans
IBM WebSphere Portal 5.02	поиск, взаимодействие пользователей, управление документами, редактирование документов, персонификация, делегирование административных прав, трансформация данных, несколько языков	LDAP, собственный реестр пользователей, внешняя аутентификация и авторизация	IBM Portal Toolkit	JDBC, Domino, PeopleSoft, внешние HTML-данные	HTML, HTTP, J2EE, JSR 168, SOAP, WML, WSRP
Microsoft SharePoint Portal Server 2003	публикация документов, обсуждение документов, автоматизация документооборота, поиск, проектные зоны, персонификация, взаимодействие пользователей	Внутренний (для приложения) и внешний (для компонентов) SSO, Active Directory	Microsoft FrontPage 2003, Visual Studio .Net 2003	Microsoft Office и около 300 адаптеров импорта данных	.NET, WebDAV, XML
OracleAS Portal 10g	поиск, автоматизация документооборота, управление документами, взаимодействие пользователей	SSO, LDAP, интеграция с внешними системами аутентификации	Oracle JDeveloper	HTML, Oracle AQ, Oracle, SQL Server, DB2, Sybase, Informix	HTML, JSR 168, .NET, SQL, WebDAV, WSRP, XML
Plumtree Enterprise Web Suite	поиск, взаимодействие пользователей, управление содержанием, категоризация, беспроводной доступ	LDAP, Oblix, Netegrity	Любая Java- или .NET-среда	SAP, PeopleSoft, Sibel, Microsoft Exchange, Lotus, Cognos, Documentum,	HTTP, HTTPS, J2EE, JSR 168, .NET, SOAP, WSDL, WSRP, XML
Sun Java System	персонификация, категоризация, по-	SSO, Liberty, SAMI, Active	Java System Studio, Java	Lotus Notes, XML, RSS,	HTML, iCal, IMAP, J2EE,

Portal Server 6.2	иск, управление содержанием, взаимодействие пользователей	Directory, LDAP, RADIUS, SafeWord, CryptonCard, JavaCard, Smart Card	System Portlet Builder, Java System Mobile, Application Builder	FatWire, Microsoft Exchange	JavaServlets, JCA, JSP, JSP 168, Liberty, RSS, SAMI, SOAP, UDDI, WSDL, WSRP, XML
Vignette Application Portal 7.0	управление содержанием, автоматизация документооборота, интеграция приложений, поиск	Встроенный реестр пользователей, IBM Directory Server, Active Directory, Sun ONE Directory Server	Vignette Application Builder	Около 70 адаптеров данных	J2EE, JSR 168, .NET, WSRP, XML

Приложение 2. Характеристики проектов по использованию семантических технологий в порталах

Таблица 5.2. Некоторые характеристики проектов по использованию семантических технологий в порталах

Характеристика проекта	**Описание**
Проект «OntoPortal»	
Разработчик	Университет Саутгемптона, Великобритания
Научный/Коммерческий	Научный
Дата начала проекта	2000
Дата окончания проекта	
Решаемая задача	Разработка методики создания портала для обучения по определенной тематике
Варианты использования онтологии	формирование семантических метаданных, навигация по онтологии, семантическое связывание
Язык описания онтологии	XML
Логический вывод	
Язык описания семантических метаданных	XML
Внедрение	Несколько прототипов по разным тематикам
Проект «Semantic Community Web Portal»	
Разработчик	Университет Карлсруэ, Германия
Научный/Коммерческий	Научный
Дата начала проекта	2000
Дата окончания проекта	
Решаемая задача	Разработка подхода к созданию портала для сообщества пользователей
Варианты использования онтологии в портале	формирование семантических метаданных, интеграция разнородных информационных источников, поиск по шаблону, навигация по онтологии.
Язык описания онтологии	F-Logic
Логический вывод	OntoBroker (поиск понятий и экземпляров)
Язык описания семантических метаданных	XML, RDF, HTML-A
Внедрение	среда KAON, прототип портала KA2 Portal, портал KAON Portal
Проект «ODESeW»	
Разработчик	Политехнический университет Мадрида, Испания
Научный/Коммерческий	Научный
Дата начала проекта	сентябрь 2002
Дата окончания проекта	февраль 2005
Решаемая задача	Разработка подхода к созданию семантического портала
Варианты использования онтологии в портале	формирование семантических метаданных, навигация по онтологии, поиск по шаблону
Язык описания онтологии	XML совместимый с OKBC (есть возможность импорта OWL, DAML+OIL, RDF(S), XCARIN).
Логический вывод	Prolog (верификация онтологии, поиск понятий и экземпляров)
Язык описания семантиче-	RDF (для экспорта описаний экземпляров)

ских метаданных	
Внедрение	Портал проекта Esperonto
Проект «OntoViews»	
Разработчик	Университет Хельсинки, Финляндия
Научный/Коммерческий	Научный
Дата начала проекта	2002
Дата окончания проекта	2004
Решаемая задача	Разработка базовых элементов семантического портала
Варианты использования онтологии в портале	формирование семантических метаданных, многоаспектный поиск, семантическое связывание
Язык описания онтологии	RDF(S)
Логический вывод	Prolog (поиск экземпляров, семантическое связывание)
Язык описания семантических метаданных	RDF
Внедрение	Портал для музеев Финляндии
Проект «OntoWeaver»	
Разработчик	Открытый университет, Великобритания
Научный/Коммерческий	Научный
Дата начала проекта	2002
Дата окончания проекта	
Решаемая задача	Разработка методологии моделирования и реализации портала, настраиваемого под потребности пользователя
Варианты использования онтологии в портале	моделирование структуры портала, моделирование внешнего вида страниц портала, моделирование пользователей, формирование семантических метаданных
Язык описания онтологии	RDF(S)
Логический вывод	Jess
Язык описания семантических метаданных	RDF
Внедрение	KMi Web Portal
Проект «SPortS»	
Разработчик	Университет Шанхая, Китай
Научный/Коммерческий	Научный
Дата начала проекта	2004
Дата окончания проекта	
Решаемая задача	Интеграция Web-сервисов в портал
Варианты использования онтологии в портале	семантическое сравнение (декомпозиция запросов), рекомендация Web-сервисов, формирование семантических метаданных
Язык описания онтологии	OWL DL, OWL-S
Логический вывод	Любой DL-модуль
Язык описания семантических метаданных	OWL-S
Внедрение	Прототип портала Apex Lab Portal
Проект «Mondeca ITM»	
Разработчик	Компания Mondeca SA, Франция
Научный/Коммерческий	Коммерческий
Дата начала проекта	
Дата окончания проекта	
Решаемая задача	Разработка портала для управления знаниями с использованием семантических технологий

Варианты использования онтологии в портале	формирование семантических метаданных (в сочетании с тематической картой), семантический поиск
Язык описания онтологии	OWL + XTM
Логический вывод	для тематической карты
Язык описания семантических метаданных	XTM, RDF
Внедрение	Mondeca ITM Portal

Приложение 3. Вычисление близости элементов семантических метаданных без учета наследования и с учетом наследования.

Таблица 5.3. Сравнение элементов метаданных без учета наследования. Часть 1

Эталон \ Кандидат	(c_x)	(i_x)
(c_i)	$SC_F(c_i, c_x)$	$SCI_F(c_i, i_x)$
(i_i)	$SIC_F(i_i, c_x)$	$SI_F(i_i, i_x)$
(c_i, r_j, c_k)	$\frac{SC_F(c_i, c_x)}{3}$	$\frac{SCI_F(c_i, i_x)}{3}$
(c_i, r_j, i_k)	$\frac{SC_F(c_i, c_x)}{3}$	$\frac{SCI_F(c_i, i_x)}{3}$
(i_i, r_j, c_k)	$\frac{SIC_F(i_i, c_x)}{3}$	$\frac{SI_F(i_i, i_x)}{3}$
(i_i, r_j, i_k)	$\frac{SIC_F(i_i, c_x)}{3}$	$\frac{SI_F(i_i, i_x)}{3}$
(c_i, a_j, v_k)	$\frac{SC_F(c_i, c_x)}{3}$	$\frac{SCI_F(c_i, i_x)}{3}$
(i_i, a_j, v_k)	$\frac{SIC_F(i_i, c_x)}{3}$	$\frac{SI_F(i_i, i_x)}{3}$

Таблица 5.4. Сравнение элементов метаданных без учета наследования. Часть 2

Эталон \ Кандидат	(c_x, r_y, c_z)	(c_x, r_y, i_z)
(c_i)	$SC_F(c_i, c_x)$	$SC_F(c_i, c_x)$
(i_i)	$SIC_F(i_i, c_x)$	$SIC_F(i_i, c_x)$
(c_i, r_j, c_k)	$\frac{SC_F(c_i, c_x) + SR_F(r_j, r_y) + SC_F(c_k, c_z)}{3}$	$\frac{SC_F(c_i, c_x) + SR_F(r_j, r_y) + SCI_F(c_k, i_z)}{3}$
(c_i, r_j, i_k)	$\frac{SC_F(c_i, c_x) + SR_F(r_j, r_y) + SIC_F(i_k, c_z)}{3}$	$\frac{SC_F(c_i, c_x) + SR_F(r_j, r_y) + SI_F(i_k, i_z)}{3}$
(i_i, r_j, c_k)	$\frac{SIC_F(i_i, c_x) + SR_F(r_j, r_y) + SC_F(c_k, c_z)}{3}$	$\frac{SIC_F(i_i, c_x) + SR_F(r_j, r_y) + SCI_F(c_k, i_z)}{3}$
(i_i, r_j, i_k)	$\frac{SIC_F(i_i, c_x) + SR_F(r_j, r_y) + SIC_F(i_k, c_z)}{3}$	$\frac{SIC_F(i_i, c_x) + SR_F(r_j, r_y) + SI_F(i_k, i_z)}{3}$
(c_i, a_j, v_k)	$\frac{SC_F(c_i, c_x)}{3}$	$\frac{SC_F(c_i, c_x)}{3}$
(i_i, a_j, v_k)	$\frac{SIC_F(i_i, c_x)}{3}$	$\frac{SIC_F(i_i, c_x)}{3}$

Таблица 5.5. Сравнение элементов метаданных без учета наследования. Часть 3

Эталон \ Кандидат	(i_x, r_y, c_z)	(i_x, r_y, i_z)

(c_i)	$SCI_F(c_i, i_x)$	$SCI_F(c_i, i_x)$
(i_i)	$SI_F(i_i, i_x)$	$SI_F(i_i, i_x)$
(c_i, r_j, c_k)	$\frac{SCI_F(c_i, i_x) + SR_F(r_j, r_y) + SC_F(c_k, c_z)}{3}$	$\frac{SCI_F(c_i, i_x) + SR_F(r_j, r_y) + SCI_F(c_k, i_z)}{3}$
(c_i, r_j, i_k)	$\frac{SCI_F(c_i, i_x) + SR_F(r_j, r_y) + SIC_F(i_k, c_z)}{3}$	$\frac{SCI_F(c_i, i_x) + SR_F(r_j, r_y) + SI_F(i_k, i_z)}{3}$
(i_i, r_j, c_k)	$\frac{SI_F(i_i, i_x) + SR_F(r_j, r_y) + SC_F(c_k, c_z)}{3}$	$\frac{SI_F(i_i, i_x) + SR_F(r_j, r_y) + SCI_F(c_k, i_z)}{3}$
(i_i, r_j, i_k)	$\frac{SI_F(i_i, i_x) + SR_F(r_j, r_y) + SIC_F(i_k, c_z)}{3}$	$\frac{SI_F(i_i, i_x) + SR_F(r_j, r_y) + SI_F(i_k, i_z)}{3}$
(c_i, a_j, v_k)	$\frac{SCI_F(c_i, i_x)}{3}$	$\frac{SCI_F(c_i, i_x)}{3}$
(i_i, a_j, v_k)	$\frac{SI_F(i_i, i_x)}{3}$	$\frac{SI_F(i_i, i_x)}{3}$

Таблица 5.6. Сравнение элементов метаданных без учета наследования. Часть 4

Кандидат / Эталон	(c_x, a_y, v_z)	(i_x, a_y, v_z)
(c_i)	$SC_F(c_i, c_x)$	$SCI_F(c_i, i_x)$
(i_i)	$SIC_F(i_i, c_x)$	$SI_F(i_i, i_x)$
(c_i, r_j, c_k)	$\frac{SC_F(c_i, c_x)}{3}$	$\frac{SCI_F(c_i, i_x)}{3}$
(c_i, r_j, i_k)	$\frac{SC_F(c_i, c_x)}{3}$	$\frac{SCI_F(c_i, i_x)}{3}$
(i_i, r_j, c_k)	$\frac{SIC_F(i_i, c_x)}{3}$	$\frac{SI_F(i_i, i_x)}{3}$
(i_i, r_j, i_k)	$\frac{SIC_F(i_i, c_x)}{3}$	$\frac{SI_F(i_i, i_x)}{3}$
(c_i, a_j, v_k)	$\frac{SC_F(c_i, c_x) + SA_F(a_j, a_y) + CV(v_k, v_z)}{3}$	$\frac{SCI_F(c_i, i_x) + SA_F(a_j, a_y) + CV(v_k, v_z)}{3}$
(i_i, a_j, v_k)	$\frac{SIC_F(i_i, c_x) + SA_F(a_j, a_y) + CV(v_k, v_z)}{3}$	$\frac{SI_F(i_i, i_x) + SA_F(a_j, a_y) + CV(v_k, v_z)}{3}$

Таблица 5.7. Сравнение элементов метаданных с учетом наследования. Часть 1

Кандидат / Эталон	(c_x)	(i_x)
(c_i)	$SC_C(c_i, c_x)$	$SCI_C(c_i, i_x)$
(i_i)	$SIC_C(i_i, c_x)$	$SI_C(i_i, i_x)$
(c_i, r_j, c_k)	$\frac{SC_C(c_i, c_x)}{3}$	$\frac{SCI_C(c_i, i_x)}{3}$
(c_i, r_j, i_k)	$\frac{SC_C(c_i, c_x)}{3}$	$\frac{SCI_C(c_i, i_x)}{3}$

(i_i, r_j, c_k)	$\frac{SIC_C(i_i, c_x)}{3}$	$\frac{SI_C(i_i, i_x)}{3}$
(i_i, r_j, i_k)	$\frac{SIC_C(i_i, c_x)}{3}$	$\frac{SI_C(i_i, i_x)}{3}$
(c_i, a_j, v_k)	$\frac{SC_C(c_i, c_x)}{3}$	$\frac{SCI_C(c_i, i_x)}{3}$
(i_i, a_j, v_k)	$\frac{SIC_C(i_i, c_x)}{3}$	$\frac{SI_C(i_i, i_x)}{3}$

Таблица 5.8. Сравнение элементов метаданных с учетом наследования. Часть 2

Кандидат / Эталон	(c_x, r_y, c_z)	(c_x, r_y, i_z)
(c_i)	$SC_C(c_i, c_x)$	$SC_C(c_i, c_x)$
(i_i)	$SIC_C(i_i, c_x)$	$SIC_C(i_i, c_x)$
(c_i, r_j, c_k)	$\frac{SC_C(c_i, c_x) + SR_C(r_j, r_y) + SC_C(c_k, c_z)}{3}$	$\frac{SC_C(c_i, c_x) + SR_C(r_j, r_y) + SCI_C(c_k, i_z)}{3}$
(c_i, r_j, i_k)	$\frac{SC_C(c_i, c_x) + SR_C(r_j, r_y) + SIC_C(i_k, c_z)}{3}$	$\frac{SC_C(c_i, c_x) + SR_C(r_j, r_y) + SI_C(i_k, i_z)}{3}$
(i_i, r_j, c_k)	$\frac{SIC_C(i_i, c_x) + SR_C(r_j, r_y) + SC_C(c_k, c_z)}{3}$	$\frac{SIC_C(i_i, c_x) + SR_C(r_j, r_y) + SCI_C(c_k, i_z)}{3}$
(i_i, r_j, i_k)	$\frac{SIC_C(i_i, c_x) + SR_C(r_j, r_y) + SIC_C(i_k, c_z)}{3}$	$\frac{SIC_C(i_i, c_x) + SR_C(r_j, r_y) + SI_C(i_k, i_z)}{3}$
(c_i, a_j, v_k)	$\frac{SC_C(c_i, c_x)}{3}$	$\frac{SC_C(c_i, c_x)}{3}$
(i_i, a_j, v_k)	$\frac{SIC_C(i_i, c_x)}{3}$	$\frac{SIC_C(i_i, c_x)}{3}$

Таблица 5.9. Сравнение элементов метаданных с учетом наследования. Часть 3

Кандидат / Эталон	(i_x, r_y, c_z)	(i_x, r_y, i_z)
(c_i)	$SCI_C(c_i, i_x)$	$SCI_C(c_i, i_x)$
(i_i)	$SI_C(i_i, i_x)$	$SI_C(i_i, i_x)$
(c_i, r_j, c_k)	$\frac{SCI_C(c_i, i_x) + SR_C(r_j, r_y) + SC_C(c_k, c_z)}{3}$	$\frac{SCI_C(c_i, i_x) + SR_C(r_j, r_y) + SCI_C(c_k, i_z)}{3}$
(c_i, r_j, i_k)	$\frac{SCI_C(c_i, i_x) + SR_C(r_j, r_y) + SIC_C(i_k, c_z)}{3}$	$\frac{SCI_C(c_i, i_x) + SR_C(r_j, r_y) + SI_C(i_k, i_z)}{3}$
(i_i, r_j, c_k)	$\frac{SI_C(i_i, i_x) + SR_C(r_j, r_y) + SC_C(c_k, c_z)}{3}$	$\frac{SI_C(i_i, i_x) + SR_C(r_j, r_y) + SCI_C(c_k, i_z)}{3}$
(i_i, r_j, i_k)	$\frac{SI_C(i_i, i_x) + SR_C(r_j, r_y) + SIC_C(i_k, c_z)}{3}$	$\frac{SI_C(i_i, i_x) + SR_C(r_j, r_y) + SI_C(i_k, i_z)}{3}$
(c_i, a_j, v_k)	$\frac{SCI_C(c_i, i_x)}{3}$	$\frac{SCI_C(c_i, i_x)}{3}$

(i_i, a_j, v_k)	$\frac{SI_C(i_i, i_x)}{3}$	$\frac{SI_C(i_i, i_x)}{3}$

Таблица 5.10. Сравнение элементов метаданных с учетом наследования. Часть 4

Кандидат / Эталон	(c_x, a_y, v_z)	(i_x, a_y, v_z)
(c_i)	$SC_C(c_i, c_x)$	$SCI_C(c_i, i_x)$
(i_i)	$SIC_C(i_i, c_x)$	$SI_C(i_i, i_x)$
(c_i, r_j, c_k)	$\frac{SC_C(c_i, c_x)}{3}$	$\frac{SCI_C(c_i, i_x)}{3}$
(c_i, r_j, i_k)	$\frac{SC_C(c_i, c_x)}{3}$	$\frac{SCI_C(c_i, i_x)}{3}$
(i_i, r_j, c_k)	$\frac{SIC_C(i_i, c_x)}{3}$	$\frac{SI_C(i_i, i_x)}{3}$
(i_i, r_j, i_k)	$\frac{SIC_C(i_i, c_x)}{3}$	$\frac{SI_C(i_i, i_x)}{3}$
(c_i, a_j, v_k)	$\frac{SC_C(c_i, c_x) + SA_C(a_j, a_y) + CV(v_k, v_z)}{3}$	$\frac{SCI_C(c_i, i_x) + SA_C(a_j, a_y) + CV(v_k, v_z)}{3}$
(i_i, a_j, v_k)	$\frac{SIC_C(i_i, c_x) + SA_C(a_j, a_y) + CV(v_k, v_z)}{3}$	$\frac{SI_C(i_i, i_x) + SA_C(a_j, a_y) + CV(v_k, v_z)}{3}$

Приложение 4. UML-диаграммы проектирования семантического ядра портала.

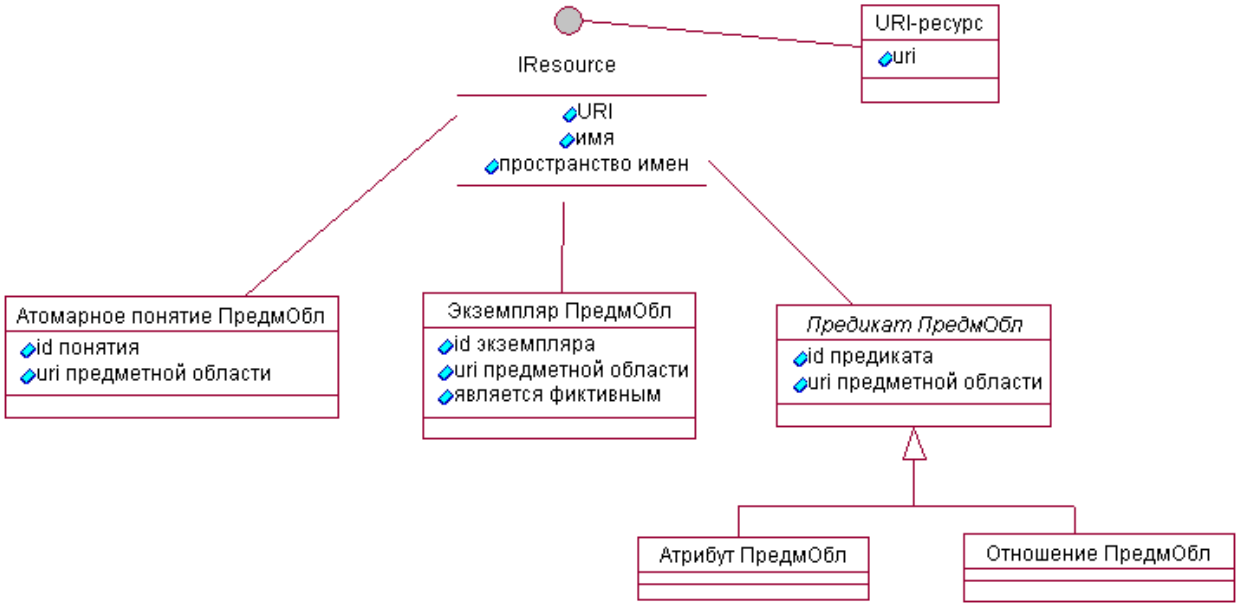


Рис. 5.1. Диаграмма классов, реализующих интерфейс IResource

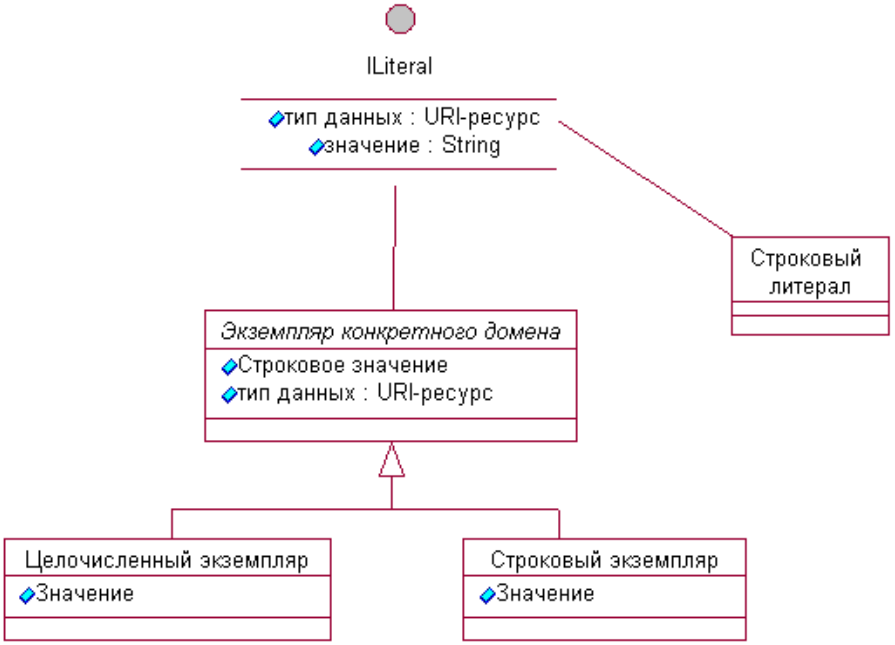


Рис. 5.2. Диаграмма классов, реализующих интерфейс ILiteral

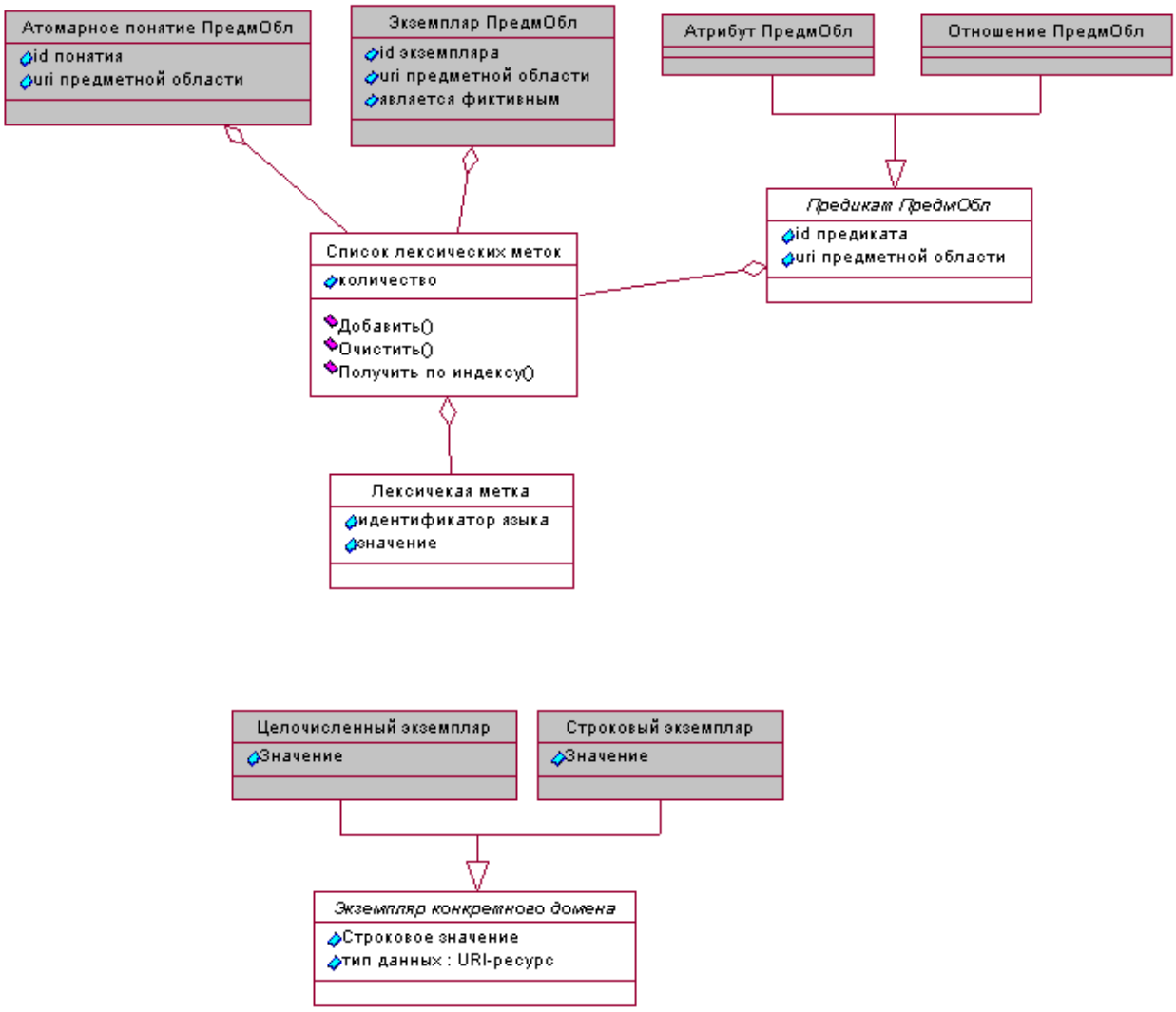


Рис. 5.3. Диаграмма классов-сущностей из объектной модели онтологии

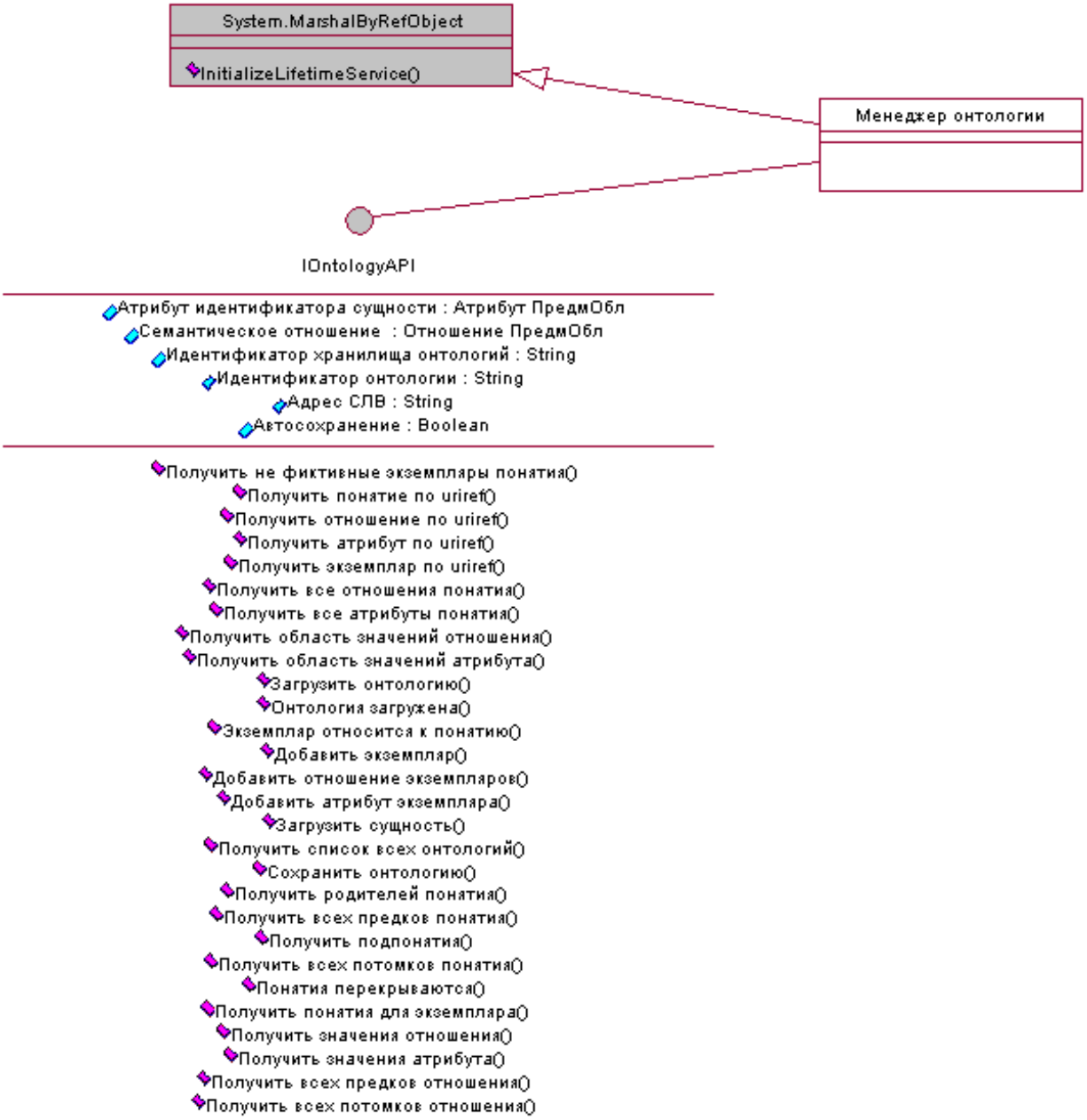


Рис. 5.4. Диаграмма наследования для класса «Менеджер онтологии»

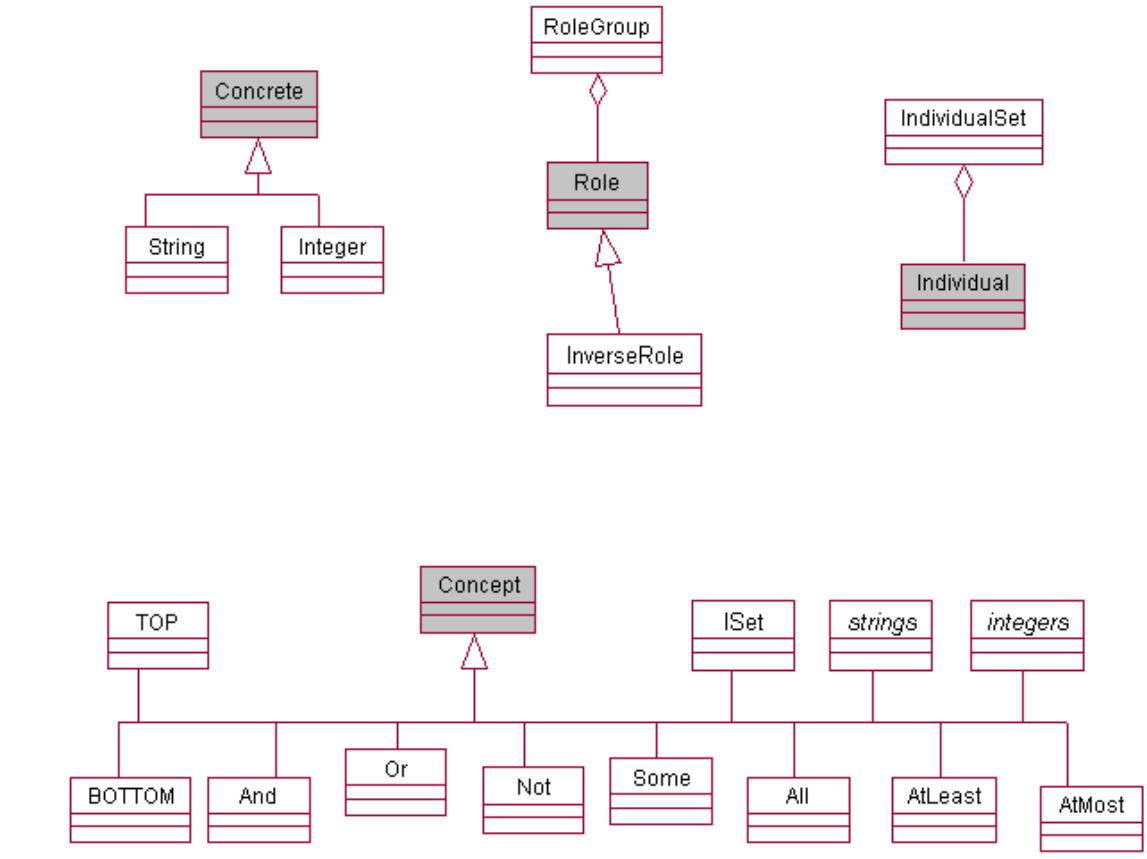


Рис. 5.5. Диаграмма классов для базовых понятий протокола DIG

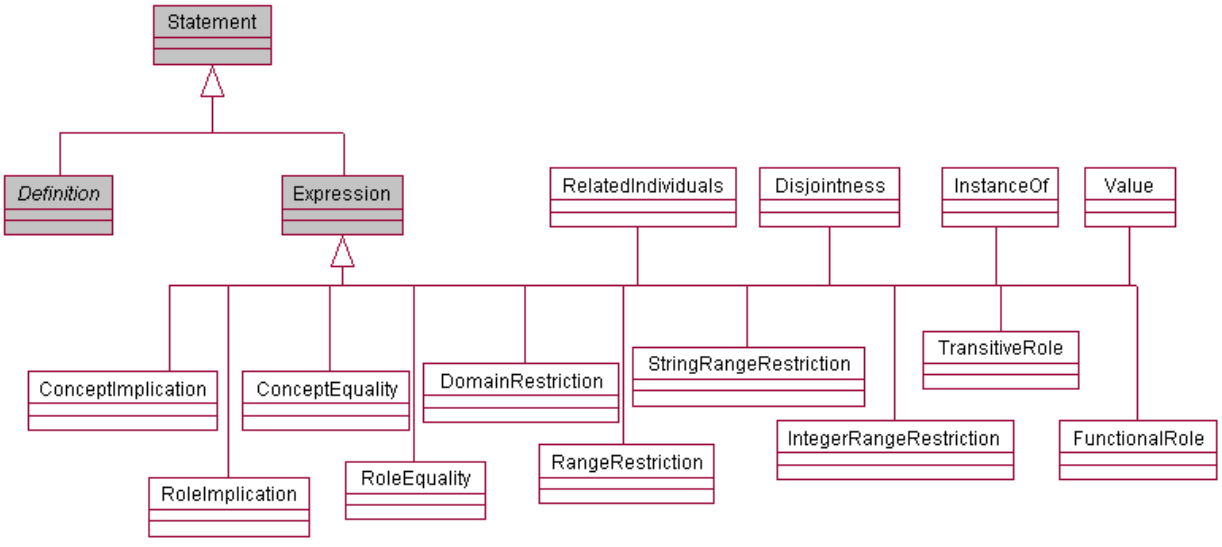


Рис. 5.6. Диаграмма классов для логических утверждений протокола DIG

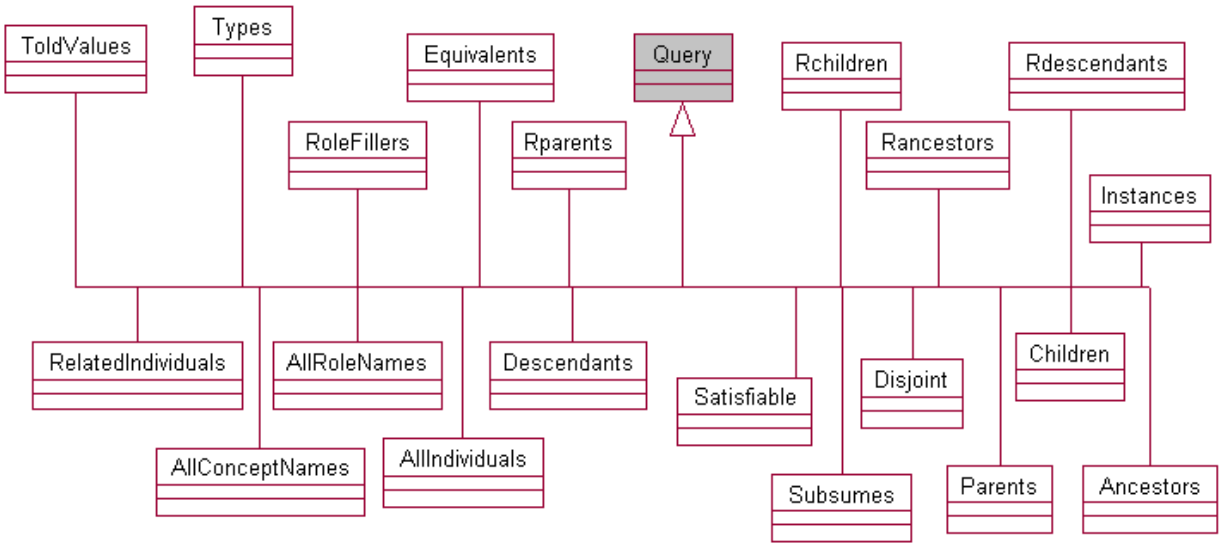


Рис. 5.7. Диаграмма классов для запросов по протоколу DIG

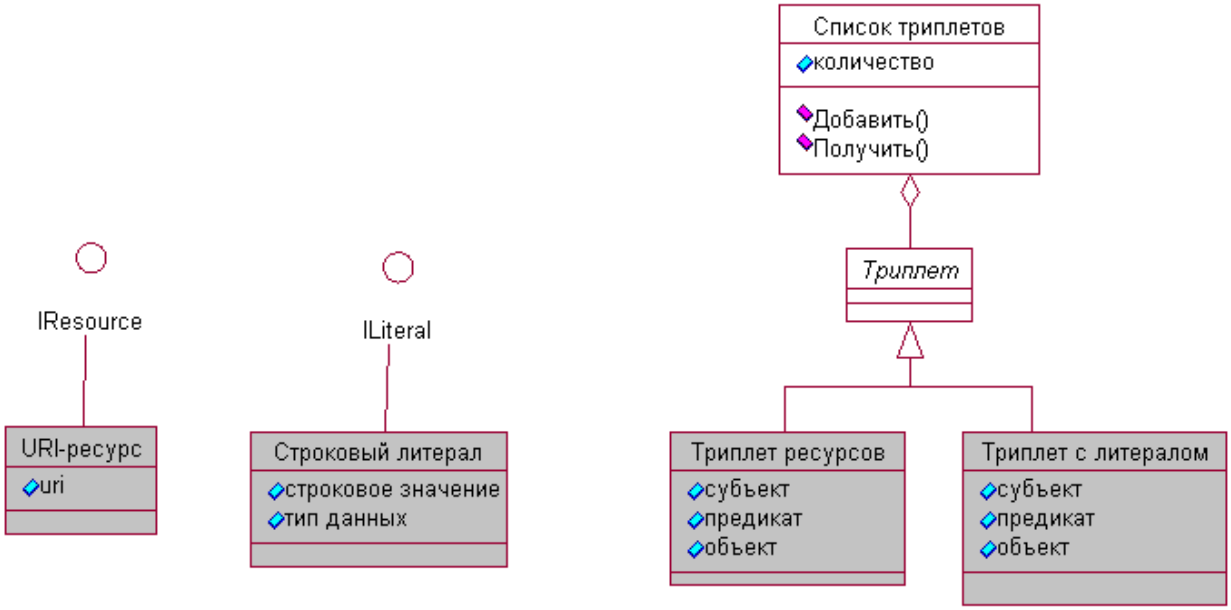


Рис. 5.8. Диаграмма классов для элементов языка RDF

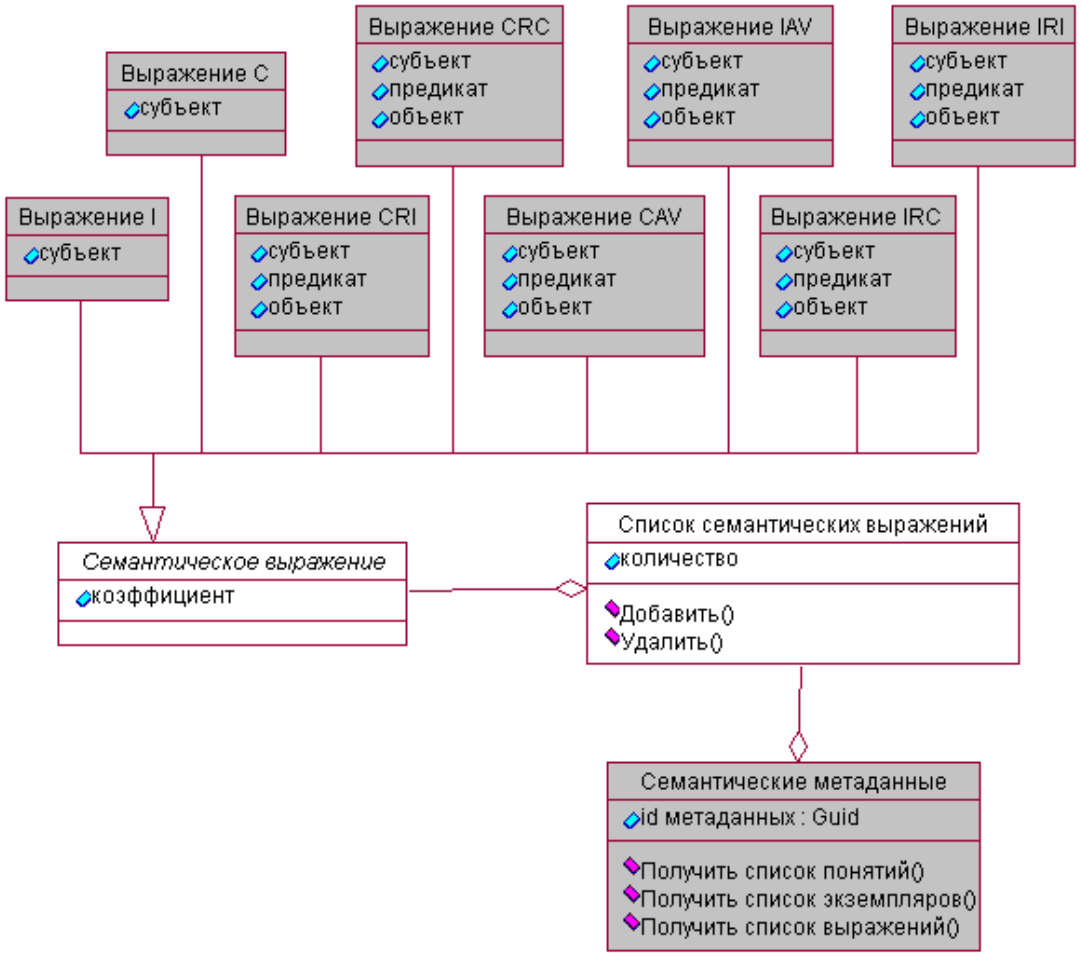


Рис. 5.9. Диаграмма классов для семантических метаданных

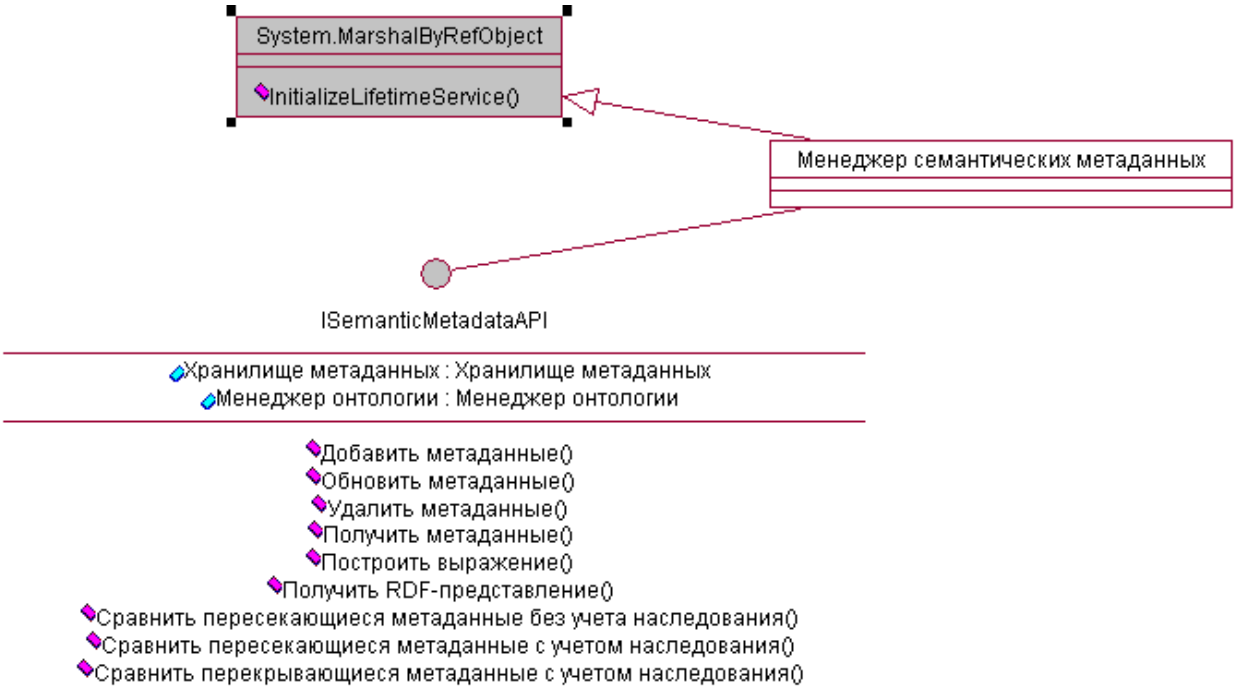


Рис. 5.10. Диаграмма наследования для класса «Менеджер семантических метаданных»

Приложение 5. Состав и структура тестового рубрикатора документов

1. **Приборы и средства автоматизации**
   - 1.1. Измерительные преобразователи
   - 1.2. Модули ввода/вывода
   - 1.3. Коммуникационное оборудование
   - 1.4. Программируемые логические контроллеры
   - 1.5. Управляемый электропривод
   - 1.6. Преобразователи энергии и системы электропитания
   - 1.7. Промышленные компьютеры и серверы
   - 1.8. Конструктивы для средств автоматизации

2. **Системы и комплексы АСУ ТП**
   - 2.1. Системы автоматического регулирования
   - 2.2. Автоматизация добычи нефти
   - 2.3. Автоматизация резервуарных парков
   - 2.4. Автоматика нефтеперекачивающих станций
   - 2.5. Комплексы телемеханики магистральных трубопроводов
   - 2.6. Автоматизация узлов учета нефти и газа
   - 2.7. Автоматизация переработки нефти
   - 2.8. Системы автоматического пожаротушения

3. **Теория, методы и программное обеспечение для создания средств и систем автоматизации**
   - 3.1. Теория, методы и ПО для проведения научных исследований по проблемам автоматизации
   - 3.2. Теория, методы и ПО для проектирования приборов и средств автоматизации
   - 3.3. Теория, методы и ПО для разработки приборов и средств автоматизации
   - 3.4. Теория, методы и ПО для конструирования приборов и средств автоматизации
   - 3.5. Теория, методы и ПО для производства приборов и средств автоматизации
   - 3.6. Теория, методы и ПО для тестирования приборов и средств автоматизации
   - 3.7. Теория, методы и ПО для моделирования и управления технологическими процессами
   - 3.8. Теория, методы и ПО для моделирования и управления производственными процессами

Приложение 6. Документы по апробации результатов диссертационного исследования

Директор Центра профессиональной переподготовки специалистов нефтегазового дела ТПУ

/Кошовкин И. Н./

АКТ ВНЕДРЕНИЯ
результатов диссертационной работы Васильева И. А.

В 2002 году в рамках хозяйственного договора, заключенного между Институтом «Кибернетический Центр» ТПУ (ИКЦ ТПУ) и Центром профессиональной переподготовки специалистов нефтегазового дела ТПУ (ЦППС НД ТПУ), сотрудниками группы КСУЗ ИКЦ ТПУ был разработан для ЦППС НД ТПУ и внедрен портал для работы с явными и неявными знаниями «Petroleum Engineers Virtual Network».

Васильев И. А. принимал непосредственное участие в разработке информационного, функционального и программного обеспечения портала. Портал представляет собой программную систему управления явными и неявными знаниями для коллектива специалистов в области разработки нефтяных месторождений и позволяет:

- работать с явными знаниями в виде библиотеки файлов и ссылок на источники информации
- работать с неявными знаниями в виде описаний профилей компетенции сотрудников ЦППС НД
- использовать русско-английский тезаурус для унифицированного описания явных и неявных знаний
- выполнять поиск явных и неявных знаний

Лично Васильевым И. А. разработана архитектура программной системы, разработана часть программного обеспечения для администрирования портала, а также разработано программное обеспечение по работе с явными знаниями – ведение дискуссий, управление рубрикатором файлов и описание файлов и внешних источников информации с помощью тезауруса.

В созданном портале были использованы методы и алгоритмы работы с семантическими моделями, разработанные в диссертационной работе Васильева И. А.

Внедрение указанной программной системы, в которую вошли результаты работы Васильева И. А., позволило мотивировать сотрудников ЦППС НД ТПУ к обмену знаниями за счет введения системы рейтингов, повысить эффективность сбора и распространения знаний, сократить временные затраты на поиск экспертов по интересующему вопросу, и в целом активизировать обмен знаниями как между сотрудниками центра, так и между сотрудниками и учащимися центра.

Ведущий специалист ЦППС НД ТПУ /Брусницын Д. Н./

11.10.05

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ГОСУДАРСТВЕННЫЙ КООРДИНАЦИОННЫЙ ЦЕНТР ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОТРАСЛЕВОЙ ФОНД АЛГОРИТМОВ И ПРОГРАММ

СВИДЕТЕЛЬСТВО ОБ ОТРАСЛЕВОЙ РЕГИСТРАЦИИ РАЗРАБОТКИ

№ 4608

Настоящее свидетельство выдано на разработку:

Web-портал для работы с явными и неявными знаниями организации

зарегистрированную в Отраслевом фонде алгоритмов и программ.

Дата регистрации: **08 апреля 2005 года**

Авторы: **Тузовский А.Ф., Васильев И.А., Козлов С.В., Панченко А.Г., Усов М.В.**

Организация-разработчик: **Институт «Кибернетический центр» Томского политехнического университета**

Директор _____ Е.Г. Калинкевич

Руководитель ОФАП _____ А.И. Галкина

Дата выдачи 29.04.2005

Диплом

КОНКУРСА

"СИБИРСКИЕ АФИНЫ"

9-й Всероссийской научно-производственной
инновационной выставки-ярмарки
«ИНТЕГРАЦИЯ-2004»

НАГРАЖДАЕТСЯ

в номинации «Новые научные разработки и технологии»

Институт
«Кибернетический центр» ТПУ
г.Томск

за создание корпоративного Web-портала для организации взаимодействия групп экспертов и создания рабочего пространства распределенных проектных групп

Заместитель Главы Администрации
(Губернатора) Томской области — В. Зинченко

7-й Международный Всесибирский инновационный форум
Томск-2004

ТЕХНОПАРК
ОАО ТОМСКИЙ МЕЖДУНАРОДНЫЙ ДЕЛОВОЙ ЦЕНТР