

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ НАУКИ
ИНСТИТУТ СИСТЕМ ИНФОРМАТИКИ ИМ. А.П. ЕРШОВА
СИБИРСКОГО ОТДЕЛЕНИЯ РОССИЙСКОЙ АКАДЕМИИ НАУК

На правах рукописи



Загорулько Галина Борисовна

**МОДЕЛЬ, МЕТОДЫ И СРЕДСТВА
КОМПЛЕКСНОЙ ПОДДЕРЖКИ РАЗРАБОТКИ СППР
В СЛАБОФОРМАЛИЗОВАННЫХ ПРЕДМЕТНЫХ ОБЛАСТЯХ**

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

Массель Людмила Васильевна

Новосибирск - 2020

Оглавление

Введение	6
Глава 1. Анализ современного состояния области разработки интеллектуальных СППР в слабоформализованных предметных областях	15
1.1. Поддержка принятия решений. СППР в слабоформализованных предметных областях, определения, классификация, основные составляющие	15
1.1.1. Теория принятия решений. Этапы и альтернативы.....	15
1.1.2. Область знаний «Поддержка принятия решений»	16
1.1.3. Определения СППР.....	17
1.1.4. Классификация СППР	18
1.1.5. Специфика слабоформализованных предметных областей	19
1.1.6. Структура интеллектуальной СППР	20
1.2. Базы знаний интеллектуальных СППР, модели представления знаний	21
1.2.1. Классические модели представления знаний.....	21
1.2.2. Комбинированные модели представления знаний	24
1.3. Принципы и подходы к разработке интеллектуальных СППР	26
1.3.1. Принципы разработки СППР	27
1.3.2. Универсальные подходы к разработке интеллектуальных СППР	28
1.3.3. Онтологический подход	29
1.3.4. Фрактальный подход	31
1.3.5. Срезы знаний	32
1.3.6. Сервис-ориентированный подход.	32
1.4. Методы поддержки принятия решений и реализующие их средства	35
1.4.1. Методы и средства визуализации информации	35
1.4.2. Методы извлечения и приобретения знаний.....	37
1.4.3. Методы семантического моделирования.....	38
1.4.4. Методы аналитического моделирования в слабоформализованных предметных областях.....	40
1.4.5. Метод рассуждений на основе прецедентов	41
1.4.6. Метод рассуждений на основе экспертных правил.....	43
1.4.7. Методы экспертных оценок	45
1.5. Технологии разработки интеллектуальных СППР	47
1.5.1. Фреймворки с открытым исходным кодом	47
1.5.2. Технология Semantic Web	48

1.5.3. Технология разработки интеллектуальных научных интернет-ресурсов	49
1.6. Средства разработки web-сервисов	50
1.6.1. Архитектурный стиль REST	50
1.6.2. Удаленный вызов процедур (RPC).....	51
1.6.3. SOAP/WSDL/UDDL	51
1.6.4. Протоколы передачи данных	53
1.7. Выводы.....	55
Глава 2. Модель комплексной поддержки разработки интеллектуальных СППР	56
2.1. Формальная система	57
2.2. Процесс комплексной поддержки разработки интеллектуальных СППР	59
2.3. Подходы, принципы, технологии, предлагаемые для разработки интеллектуальных СППР	62
2.3.1. Принципы разработки интеллектуальных СППР	62
2.3.2. Подходы к разработке интеллектуальных СППР	64
2.3.3. Технологии для разработки ИСППР	67
2.4. Онтология поддержки принятия решений	70
2.5. Информационно-аналитический интернет-ресурс по поддержке принятия решений.....	71
2.6. Репозиторий методов поддержки принятия решений	74
2.6.1. Модель репозитория	74
2.6.2. Методика разработки репозитория	77
2.7. Архитектура и методика разработки интеллектуальных СППР.....	78
2.7.1. Архитектура интеллектуальных СППР	78
2.7.2. Методика разработки интеллектуальных СППР	79
2.8. Выводы.....	82
Глава 3. Реализация предлагаемых средств комплексной поддержки разработки интеллектуальных СППР	84
3.1. Разработка онтологии Поддержки Принятия Решений.....	84
3.1.1. Конкретизация понятий базовых онтологий.....	84
3.1.2. Фрактально-стратифицированная структура онтологии	88
3.1.3. Использование технологий Semantic Web при разработке онтологии	89
3.2. Реализация информационно-аналитического интернет-ресурса по поддержке принятия решений.....	91
3.2.1. Контент информационно-аналитического интернет-ресурса по поддержке принятия решений.....	92

3.2.2. Создание контента с помощью технологий разработки интеллектуальных научных интернет-ресурсов и Semantic Web	93
3.3. Реализация репозитория методов поддержки принятия решений	95
3.3.1. Использование паттернов онтологического проектирования для описания методов поддержки принятия решений	96
3.3.2. Рассуждения на основе прецедентов	97
3.3.3. Метод недоопределенных вычислений	98
3.3.4. Рассуждения на основе правил	101
3.3.5. Метод онтологического моделирования	102
3.3.6. Когнитивное моделирование	103
3.3.7. Сервис обработки массивов данных	105
3.3.8. Сбор и анализ мнений экспертов	105
3.3.9. Методы экспертного оценивания	106
3.4. Разработка сервисов, реализующих взаимодействие с внешними хранилищами данных	108
3.4.1. Архитектура системы доступа к внешним данным и схема её функционирования ..	108
3.4.2. Особенности реализации взаимодействия интеллектуальных научных интернет-ресурсов и системы доступа к внешним данным	110
3.5. Выводы	113
Глава 4. Использование предлагаемых средств комплексной поддержки разработки интеллектуальных СППР	114
4.1. Интеллектуальная СППР по угрозам энергетической безопасности	114
4.1.1. Этап Идентификации	114
4.1.2. Этап Концептуализации	116
4.1.3. Этап Формализации	118
4.1.4. Этап Реализации компонентов интеллектуальной СППР	121
4.1.5. Этап Реализации интеллектуальной СППР	122
4.2. Интеллектуальная СППР «Элементозы»	125
4.2.1. Этап Идентификации	125
4.2.2. Этап Концептуализации	125
4.2.3. Разработка онтологии ПО	126
4.2.4. Этап Формализации	127
4.2.5. Этап Реализации компонентов интеллектуальной СППР	131
4.2.6. Этап Реализации интеллектуальной СППР	131

4.3. Выводы.....	132
Заключение.....	134
Список сокращений и условных обозначений	137
Список литературы	139
Приложение А. Описание синтаксиса и семантики дескрипционной логики SOIN(D)	163
Приложение Б. Базовые онтологии. Паттерны онтологического проектирования ..	165
Приложение В. Онтология ППР. Визуальное представление методов поддержки принятия решений.....	168
Приложение Г. Программная реализация методов. Фрагменты программного кода	173
Приложение Д. Свидетельство о государственной регистрации программ для ЭВМ. Акты о внедрении результатов диссертационного исследования	188

Введение

Актуальность темы исследования. Системы поддержки принятия решений (СППР) используются во многих сферах человеческой деятельности. Их популярность неуклонно возрастает. Одной из причин этого явления является массовая компьютеризация таких сфер нашей жизни, как экономика, политика, здравоохранение, образование, юриспруденция, соцзащита, культура и прочее, т.е. тех областей, которые слабо формализуемы по своей сути [11,95,109,110]. Накопленные в этих областях знания и данные требуют специальных (интеллектуальных) средств для их представления, организации доступа к ним и использования. Владение этими средствами необходимо для принятия грамотных, взвешенных решений. Вместе с тем всё больше людей вовлекаются в процесс принятия важных решений. Одновременно растет ответственность за их последствия. Поэтому так важны системы, способные оказать помощь лицам, принимающим решения, в особенности, в слабоформализованных предметных областях (СФПО).

Разработка СППР – важная задача, решение которой осложняется отсутствием доступных инструментариев и систематизированной информации о методах и аспектах поддержки принятия решений. Таким образом, процесс разработки СППР также нуждается во всесторонней поддержке, а создание методов и средств такой поддержки является актуальной проблемой.

Степень разработанности проблемы. Теория принятия решений является хорошо исследованной научной дисциплиной. Вопросами развития этой теории и ее практических приложений занимаются многие ученые и научные коллективы. В работах Кулагина О.А., Орлова А.И., Петровского А.Б. дается описание этапов принятия решений, классификация задач принятия решений, описывается ряд

классических методов принятия решений [116,146,148]. Методы построения и анализа когнитивных карт и их применение для принятия решений рассматриваются в работах Axelrod R., Абрамовой Н.А., Кузнецова О.П., Кулинича А.А. [5,72,118,119]. Мягкими вычислениями и их использованием в гибридных интеллектуальных системах занимаются Тарасов В.Б., Ярушкина Н.Г. и др. [78,169]. Методы рассуждений на основе прецедентов развивают Варшавский П.Р., Еремеев А.П. и др. [82]. Использованием методов анализа данных и машинного обучения для принятия решений занимаются Мельников А.В., Юсупова Н.И. [68,96] и др. Вопросы создания систем поддержки принятия управленческих решений рассмотрены в работах Saha C., Zulkafli Z., Трахтенгерца Э.А. [56,71,162]. Разработкой интеллектуальных СППР занимаются Геловани В.А., Массель Л.В., Черняховская Л.Р. (СППР в нештатных ситуациях) [93,130,166], Грибова В.В., Кобринский Б.А., Шалфеева Е.А. (медицинские СППР) [95,109] и др. Методологии, используемые подходы и принципы разработки СППР обсуждаются в работах Garaibeh N.K., Power D. J., Turban E., Вагина В.Н., Судакова В. А, Халина В.Г. [28,50,66,81,160,164].

Предложено и описано большое количество методов поддержки принятия решений (ППР) [124,146,148,156]. В ряде работ исследуются вопросы создания инструментария для разработки СППР [28,49,157,160]. Однако до сих пор не создано исчерпывающего описания и классификации существующих методов принятия решений. В то же время для алгоритмов и методов, имеющих реализации, весьма проблематично найти готовые к исполнению модули или хотя бы их подробные спецификации.

Наиболее полно методы ППР описаны для хорошо формализованных областей. Для слабоформализованных областей таких методов разработано и специфицировано намного меньше.

В работах Casanovas P., Rospocher M., Saremi A., Авдеенко Т.В., Виттиха В.А., Массель Л.В., Смирнова С.В. обсуждается использование онтологий при построении СППР в различных областях, в том числе слабоформализованных [11,54,55,58,73,84,134,140]. Однако в этих и близких к ним работах не ставилась

задача построения онтологии, на которой могла бы базироваться комплексная (всесторонняя) поддержка разработки СППР.

Общепризнано, что современная СППР является одновременно и информационной системой, интегрирующей информационные ресурсы своей области знаний. Вопросы обеспечения эффективного содержательного доступа к информационным ресурсам научной, образовательной и инженерной тематики посредством разработки специализированных информационных систем освещены в работах Edwards M.P., Scott-Morton M.S., Zhang, S., Васенина В.А., Глобы Л.С, Загорулько Ю.А., Копайгородского А.Н., Мельникова А.В., Серебрякова В. А., Тузовского А.Ф. [18,30,70,83,94,103,113,137,158]. Что касается информационных ресурсов, обслуживающих разработчиков СППР, то их практически нет. Существующие системы в основном ориентированы на лиц, принимающих решения, или специалистов в конкретной предметной области. Такие системы, большей частью, представляют собой структурированные и аннотированные каталоги ссылок на Интернет-ресурсы, составные элементы которых практически не связаны между собой, что затрудняет поиск нужной информации и делает невозможной навигацию по ним.

Предлагаемые в диссертационном исследовании методы и средства предназначены для решения проблемы отсутствия доступных инструментариев построения интеллектуальных СППР с требуемой функциональностью и оказания помощи в разработке таких систем в СФПО.

Таким образом, важность СППР в СФПО, отсутствие универсальных доступных широкому кругу пользователей инструментариев для их построения определяют актуальность диссертационного исследования.

В то же время, в рамках теории принятия решений и искусственного интеллекта разработано теоретически и реализовано программно большое количество методов, которые могут быть применены для поддержки принятия решений. Описано множество подходов, принципов построения СППР. Организация доступа к этой информации и к программным реализациям методов

является актуальной задачей, решение которой создает предпосылки для успешного выполнения задач диссертационного исследования.

Цель и задачи исследования. Целью диссертационного исследования является создание модели, методов и средств комплексной поддержки разработки интеллектуальных СППР для слабоформализованных предметных областей.

Достижение этой цели предполагает решение следующих **задач**:

1. Анализ современного состояния в области поддержки разработки СППР, позволяющий выявить существующие принципы, подходы и средства разработки интеллектуальных СППР, методы принятия решений в слабоформализованных предметных областях.
2. Создание модели комплексной поддержки разработки интеллектуальных СППР в слабоформализованных областях.
3. Разработка онтологии области знаний «Поддержка принятия решений в слабоформализованных областях».
4. Создание информационно-аналитического ресурса по поддержке принятия решений в слабоформализованных областях на основе разработанной онтологии.
5. Разработка репозитория методов поддержки принятия решений в слабоформализованных областях, в том числе, программная реализация ряда методов.
6. Разработка и апробирование методики создания интеллектуальных СППР на основе разработанных методов и средств.

Объектом исследования в данной работе является процесс разработки СППР в слабоформализованных предметных областях.

Предмет исследования – современные методы и средства поддержки разработки СППР на всех этапах их жизненного цикла, методы принятия решений в слабоформализованных предметных областях.

Методы и средства исследования. При структуризации объекта и предмета исследования были использованы методы системного анализа, когнитивной психологии, искусственного интеллекта. При разработке модели

комплексной поддержки разработки СППР использовались методы инженерии знаний, онтологического моделирования, технологии Semantic Web. При создании методики разработки интеллектуальных СППР использовались методы объектно-ориентированного анализа, системного и прикладного программирования.

Научная новизна диссертационной работы:

1. Предложена оригинальная модель комплексной поддержки разработки интеллектуальных СППР для слабоформализованных предметных областей, отличающаяся интеграцией методов и средств, обеспечивающих концептуальную, информационную, компонентную и методическую поддержку разработчиков СППР, а также применением дескрипционной логики, используемой как для формального, так и содержательного описания интегрируемых в модели методов и средств.
2. Впервые выполнена систематизация области знаний «Поддержка принятия решений в слабоформализованных областях» и построена её онтология.
3. На основе онтологии поддержки принятия решений разработан информационно-аналитический интернет-ресурс, предоставляющий содержательный доступ к знаниям и данным этой предметной области. Отличительной особенностью данного ресурса является его интеграция с репозиторием методов поддержки принятия решений.
4. Разработан репозиторий методов поддержки принятия решений, который отличается от аналогичных библиотек и хранилищ методов, созданных для решения задач в других предметных областях, семантической систематизацией методов и организацией содержательного доступа как к их описаниям, так и к их реализациям в виде сервисов, за счет использования комбинации сервис-ориентированного и онтологического подходов.
5. Разработана новая методика создания ИСППР, предлагающая использовать в качестве каркаса создаваемой системы информационно-аналитический интернет-ресурс моделируемой предметной области и обеспечивать её функциональность за счет включения в ее состав сервисов из репозитория.

Теоретическая и практическая значимость диссертационной работы.

Теоретическая значимость диссертационной работы заключается в создании модели комплексной поддержки разработки интеллектуальных СППР. За счет использования дескрипционной логики для представления данной модели обеспечивается как формальное, так и содержательное описание методов и средств концептуальной, информационной, компонентной и методической поддержки, а также подходов и принципов разработки интеллектуальных СППР, хорошо зарекомендовавших себя на практике. Определение границ, систематизация и формализация области знаний «Поддержка принятия решений в СФПО» в виде онтологии, разработка на ее основе информационно-аналитического интернет-ресурса способствуют более быстрому освоению данной области и привлечению к ней новых исследователей, а значит, ее дальнейшему развитию.

Практическая значимость работы заключается в создании средств, предназначенных для облегчения и ускорения процесса разработки интеллектуальных СППР в СФПО. Кроме того, разработанные методы и средства могут быть использованы в процессе обучения студентов, специализирующихся в данной области знаний.

Результаты диссертационной работы вошли в наиболее значимый результат ИСИ СО РАН за 2019 год. Отдельные результаты были применены в базовом проекте ИСИ СО РАН «IV.39.1.4. Методы и технология создания и сопровождения интеллектуальных информационных систем и систем поддержки принятия решений», в проектах, поддержанных грантами РФФИ (проекты № 13-07-00422 и № 16-07-00569) и Президиума РАН (интеграционный проект СО РАН № 15/10 «Математические и методологические аспекты интеллектуальных информационных систем», проект П.2П/IV.39-1 «Информационные, управляющие и интеллектуальные технологии и системы»).

Основные положения работы, выносимые на защиту:

1. Модель комплексной поддержки процесса разработки интеллектуальных СППР.

2. Онтология области знаний «Поддержка принятия решений в слабоформализованных областях».
3. Информационно-аналитический интернет-ресурс по поддержке принятия решений в слабоформализованных областях.
4. Расширяемый репозиторий методов поддержки принятия решений. Ряд сервисов, реализующих интеллектуальные методы поддержки принятия решений в слабоформализованных областях.
5. Методика создания интеллектуальных СППР на основе разработанных методов и средств.

Достоверность результатов диссертационной работы подтверждается корректным применением методов и средств исследования, апробацией основных положений работы на международных и российских конференциях, актами о внедрении результатов, свидетельством о регистрации программного комплекса, включающего результаты диссертационного исследования.

Область исследования соответствует специальности 05.13.11 – «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей» (технические науки) по пунктам:

Направление исследования	Выносимое на защиту положение
1. Модели, методы и алгоритмы проектирования и анализа программ и программных систем.	1, 5
3. Модели, методы, алгоритмы, языки и программные инструменты для организации взаимодействия программ и программных систем.	2, 3, 4
4. Системы управления базами данных и знаний.	2, 3

Апробация работы. Результаты работы докладывались на XV, XVII, XVIII, XIX, XX, XXI, XXIII-й Байкальских Всероссийских конференциях «Информационные и математические технологии в науке и управлении», г. Иркутск (2010, 2012-2016, 2018 гг.); Всероссийской конференции с международным участием «Знания – Онтологии – Теории» (ЗОНТ–2013), Новосибирск, 2013 г.; III, VI и VIII-й Международных научно-технических

конференциях «Открытые семантические технологии проектирования интеллектуальных систем», Минск (2013, 2016, 2018 гг.); 13, 14, 16-й национальных конференциях по искусственному интеллекту с международным участием (КИИ-2012, КИИ-2014, КИИ-2018); IV Всероссийском симпозиуме «Инфраструктура научных информационных ресурсов и систем», Санкт-Петербург, 2014 г.; Международной конференции «Информационные технологии в науке, образовании и управлении», Гурзуф, 2015 г.; (9, 10, 14, 17-й международных конференциях «New Trends in Software Methodologies, Tools, and Techniques» (SoMet) (2010, 2011, 2015, 2018 гг.), 6 и 7-й всероссийской научной конференции «Информационные технологии и системы» (2017 и 2019 гг.).

Публикации. По теме диссертации опубликовано более 40 работ, в том числе, 7 – в изданиях, рекомендуемых ВАК РФ для опубликования результатов диссертационных работ на соискание ученой степени доктора и кандидата наук и 7 – в изданиях, индексируемых в Scopus и Web of Science.

Личный вклад. Положения 1, 2, 5, составляющие новизну и выносимые на защиту, получены автором лично. Положения 3, 4 получены совместно с сотрудниками лаборатории искусственного интеллекта ИСИ СО РАН и дипломниками НГУ под руководством автора. При этом вклад автора в получение этих результатов является определяющим. В совместных работах и публикациях автору принадлежат результаты, относящиеся к разработке онтологий, концепции и методике комплексной поддержки разработки интеллектуальных СППР, разработке и использованию методов ППР, методике построения интеллектуальных СППР, разработке конкретных СППР.

Структура и объем диссертационной работы. Диссертация состоит из введения, четырех глав, заключения, списка литературы, содержащего 196 наименований, и 5 приложений. Полный объем диссертации составляет 191 страницу, включая 56 рисунков и 2 таблицы.

Содержание диссертации

Во введении обоснована актуальность темы, определены цель и задачи, объект, предмет и методы исследования, сформулированы научная новизна,

выносимые на защиту положения, теоретическая и практическая значимость, представлены сведения об апробации результатов научной работы.

В **главе 1** выполняется анализ современного состояния в области разработки ИСППР в слабоформализованных предметных областях.

Глава 2 посвящена описанию предлагаемой модели комплексной поддержки разработки интеллектуальных СППР, формализованному средствами дескрипционной логики.

В **главе 3** описывается реализация предлагаемых методов и средств комплексной поддержки разработки интеллектуальных ИСППР.

В **главе 4** представлены примеры использования предлагаемых методики и средств при разработке двух СППР – в области энергетики и медицины.

В **заключении** сформулированы основные научные результаты диссертационной работы.

В **приложениях** представлены описание синтаксиса и семантики дескрипционной логики SOIN(D), базовые онтологии и ряд паттернов онтологического проектирования, визуальное представление фрагментов онтологии, описывающих методы поддержки принятия решений, фрагменты программного кода, реализующего ряд методов, свидетельство о государственной регистрации программного комплекса, включающего результаты диссертационного исследования, акты об их внедрении.

Глава 1. Анализ современного состояния области разработки интеллектуальных СППР в слабоформализованных предметных областях

1.1. Поддержка принятия решений. СППР в слабоформализованных предметных областях: определения, классификация, основные составляющие

1.1.1. Теория принятия решений. Этапы и альтернативы

Принятие решений есть особый вид человеческой деятельности, направленный на нахождение наилучших из возможных вариантов разрешения проблемной ситуации. Конечный результат решения проблемы определяется многими участниками, имеющими различные функции. Главное место принадлежит человеку или группе людей, которые фактически осуществляют выбор предпочтительного решения и которых называют лицом, принимающим решение (ЛПР). Обычно в роли ЛПР выступает руководитель или группа компетентных в своей области специалистов, обладающих соответствующими знаниями и опытом деятельности, наделенных необходимыми полномочиями для принятия решения и несущих ответственность за реализацию принятого решения [148].

Разработкой методов и средств принятия решений занимается комплексная научная дисциплина «Теория принятия решений» (ТПР). ТПР рассматривает широкий спектр вопросов, связанных с процессом принятия решений и решаемыми на каждом из его этапов задачами.

Г. Саймон выделял 3 этапа принятия решений: Поиск информации, Поиск и Формирование альтернатив, Выбор альтернативы [59]. В настоящее время принято рассматривать этот процесс с большей степенью детализации. На Рисунке 1.1 показаны основные этапы принятия решений, выделяемые многими исследователями [116,122,148].



Рисунок 1.1. Этапы принятия решений

В литературе, посвященной ТПР, содержатся разные формулировки её объекта и предмета исследования [168]. Однако исследователи сходятся во мнении, что основным назначением ТПР является поиск и обоснованный выбор наиболее предпочтительных для человека вариантов решения проблемы [75, 122, 148]. Таким образом, в фокус исследований ТПР попадают методы, связанные с формированием, оцениванием и выбором лучших или просто приемлемых вариантов разрешения проблемной ситуации, т.е. с поддержкой ЛПР на заключительных этапах принятия решений.

1.1.2. Область знаний «Поддержка принятия решений»

Рассматриваемая в данном диссертационном исследовании область знаний (ОЗ) «Поддержка принятия решений» (ППР) является расширением классической теории принятия решений. Данная ОЗ, помимо методов поиска и выбора альтернатив решения проблемы, рассматривает также и методы, применяемые на начальных этапах принятия решений. В работах О.И. Ларичева [122, 124] говорится, что в процессе принятия решений всегда переплетены наука и искусство, однако на начальных этапах научные методы не играют важной роли. Решаемые на этих этапах задачи не рассматриваются в рамках различных теорий

выбора, поскольку они в основных своих чертах не формализованы и решаются благодаря навыкам и умениям консультанта и ЛПР.

«Однако результаты, достигнутые в области искусственного интеллекта, когнитивной психологии и системного анализа в последние 10-15 лет, позволяют привести научную составляющую в ранее не формализуемые ОЗ.

В рамках смежных научных дисциплин были разработаны подходы и методы, позволяющие структурировать предметные области, диагностировать проблемные ситуации, анализировать объективные и субъективные факторы, находить варианты решения проблемы. При этом делается упор на тех методах, которые используются в слабоформализованных областях, для которых характерны такие факторы, как неопределённость и неполнота исходных данных, отсутствие аналитической модели решаемой задачи, качественный, декларативный характер используемой информации.

Наряду с понятиями, относящимися к процессу принятия решений и методами его поддержки, ОЗ «ППР» рассматривает и вопросы, связанные с программной реализацией методов, а также с осуществляемой в её рамках деятельностью» [174].

1.1.3. Определения СППР

История СППР, насчитывающая более 60 лет, описывается в работе [50]. Первые системы, управляющие принятием решений, появились ещё в середине прошлого века, однако термин СППР (Decision Support System, DSS) ввел Скотт-Мортон в 1971 г. [30]. Существует множество определений СППР, которые отражают точки зрения разных научных дисциплин и школ. Приведем два определения, наиболее точно описывающих класс СППР, исследованию и разработке которых посвящено данное диссертационное исследование:

«СППР – интерактивные компьютерные системы и подсистемы, направленные на помощь лицам, принимающим решения, путем предоставления доступа к коммуникационным технологиям, данным, документам, знаниям и/или моделям с целью определения, решения проблем и принятия решений» [50].

«Системы поддержки принятия решений являются человеко-машинными системами, которые позволяют лицам, принимающим решения, использовать данные, знания, объективные и субъективные модели для анализа и решения слабоструктурированных проблем» [123].

1.1.4. Классификация СППР

Существует множество классификаций СППР, предложенных разными авторами на основании разных характеристик. Рассмотрим некоторые из них:

- **По характеру взаимодействия с пользователем:** *пассивные* СППР – помогают в процессе принятия решений, но не могут предлагать варианты решения; *активные* СППР – непосредственно участвуют в процессе принятия решений и могут делать предположения, какое из решений следует выбрать; *кооперативные* СППР – предполагают взаимодействие системы с пользователем [35].
- **По цели использования и характеру используемых данных:** *оперативные* СППР – предназначены для немедленного реагирования на изменения текущей ситуации; *стратегические* СППР – ориентированы на анализ значительных объемов разнородной информации, собираемых из различных источников. *СППР уровня предприятия* – подключена к большим хранилищам информации, может одновременно работать с несколькими пользователями; *настольная* СППР – предназначена для работы на персональном компьютере пользователя с локальным хранилищем информации [47, 147].
- **По типовым операциям:** файло-ориентированные системы; системы анализа данных; информационно-аналитические системы; бухгалтерские и финансовые модели; имитационные модели; оптимизационные модели; логические модели [2].
- **По используемой концептуальной модели:** СППР, *управляемые сообщениями* (Communication-Driven DSS); СППР, *управляемые данными* (Data-Driven DSS); СППР, *управляемые документами* (Document-Driven DSS); СППР, *управляемые знаниями* (Knowledge-Driven DSS); СППР, *управляемые моделями* (Model-Driven DSS); *Web-ориентированные* СППР (Web-based DSS) [50].

Если в СППР используются методы искусственного интеллекта, то такие системы называются **интеллектуальными СППР (ИСППР)**.

Предлагаемые в диссертационной работе методы и средства ориентированы на разработку *кооперативных, стратегических, информационно-аналитических, Web-ориентированных интеллектуальных систем, управляемых знаниями*. Такие системы предназначены для использования в *слабоформализованных предметных областях*.

1.1.5. Специфика слабоформализованных предметных областей

Одним из наиболее важных свойств прикладной предметной области является степень структурированности используемых в ней знаний. В работе [86] *структурированность* определяется как «степень теоретического осмысления и выявления основных закономерностей и принципов, действующих в данной предметной области» и выделяются хорошо, средне и слабоструктурированные ПО. При этом *«средне структурированная* предметная область – область с определившейся терминологией, явными взаимосвязями между явлениями, но развивающейся теорией. *Слабо структурированная* предметная область – область с размытыми определениями, скрытыми взаимосвязями между явлениями, богатой эмпирикой, с большим количеством «белых пятен»».

Назовем *слабоформализованными* предметные области со средней и слабой структуризацией. Помимо указанных выше свойств таких областей для них также характерно отсутствие строгих математических моделей, аналитических зависимостей, преобладание декларативных описаний понятий, существующих между ними связей и процессов вывода новых знаний из имеющихся. В СФПО используется преимущественно качественная информация, обладающая свойствами с множеством НЕ-факторов [143]. К таким областям относятся энергетика, медицина, лингвистика, фольклор, юриспруденция, социология, политика и многие другие. Но и области, традиционно относящиеся к хорошо формализованным, как, например, экономика, сейсмология, материаловедение, могут иметь черты СФПО. В них зачастую возникают новые задачи и

необходимость работать с качественными, неполными и слабо структурированными данными [172,179].

1.1.6. Структура интеллектуальной СППР

В работе [123] отмечались направления развития СППР: объединение СППР с автоматизированными информационными системами, сближение СППР с экспертными системами и появление интеллектуальных СППР. Современная прикладная ИСППР, как правило, состоит из нескольких взаимосвязанных подсистем (Рисунок 1.2): информационной, моделирующей, аналитической, оценивающей и интеллектуальной (экспертной). Эти подсистемы решают разные задачи и используют для этого разные методы.

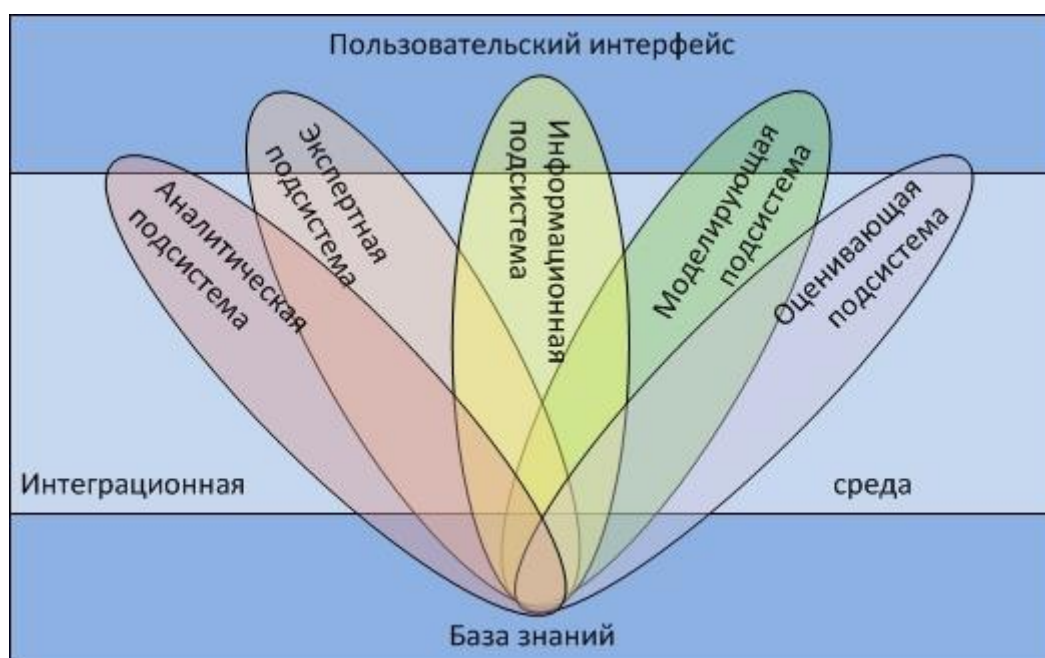


Рисунок 1.2. Структура интеллектуальной системы поддержки принятия решений

Таким образом, ИСППР можно представить в виде:

$IDSS = \langle GUI, KN [, IS, \{AS_i\}, \{ES_j\}, \{MS_k\}, \{EvS_l\}] \rangle$, где

GUI – пользовательский интерфейс;

KN – база знаний;

IS – информационная подсистема, обеспечивающая доступ к хранимой в СППР информации;

$\{AS_i\}$ – множество аналитических подсистем, предназначенных для выполнения расчетов, анализа содержащейся в СППР информации;

$\{ES_j\}$ – множество экспертных подсистем, осуществляющих вывод информации, не содержащейся в СППР в явном виде, на основе экспертных знаний;

$\{MS_k\}$ – множество моделирующих подсистем, позволяющих строить компьютерные модели для решаемых СППР задач и проводить на них эксперименты;

$\{EvS_l\}$ – множество оценивающих подсистем, предназначенных для сбора и анализа экспертных оценок вариантов разрешения проблемной ситуации.

Обязательными элементами ИСППР являются пользовательский интерфейс и база знаний.

1.2. Базы знаний интеллектуальных СППР, модели представления знаний

Разработка базы знаний является наиболее сложным этапом, «узким местом» создания любых интеллектуальных систем, в частности ИСППР [86].

Для представления знаний в системах искусственного интеллекта используются классические и комбинированные модели представления знаний (МПЗ).

1.2.1. Классические модели представления знаний

К классическим МПЗ относят сетевые, логические и продукционные модели.

Сетевая МПЗ. В основе сетевых моделей лежит понятие сети. Данный класс МПЗ объединяет семантические сети, функциональные сети и фреймы [151].

Семантическая сеть – это ориентированный граф, вершины которого представляют объекты (понятия, конкретные объекты, события, процессы, явления и т.п.), а дуги – отношения, существующие между объектами. [127].

Функциональная сеть представляет собой двудольный ориентированный граф, включающий вершины двух типов – объекты (переменные) и операторы (функции). Дуги отражают функциональные связи между операторами и объектами. Дуга, направленная от объекта к оператору, предписывает рассматривать этот объект как аргумент данного оператора, дуга обратной ориентации указывает на то, что объект выступает по отношению к оператору в качестве результата. Функциональные сети отражают некоторую декомпозицию определенной вычислительной процедуры, а дуги показывают функциональную связь, устанавливаемую между частями, возникшими в результате декомпозиции [141,163,190].

Фрейм-представления были предложены М. Минским в 1970-е годы. «Фрейм - это структура данных, представляющая стереотипную ситуацию» [138]. Фрейм представляет собой сеть узлов и отношений, которые организованы иерархически. Узлы сети – это понятия, определяемые набором атрибутов, называемых слотами, и значениями этих атрибутов. Каждый слот может быть связан с процедурами, выполняемыми при изменении информации в слотах. Группы родственных фреймов образуют систему фреймов. Поскольку различные фреймы одной и той же системы используют общее множество слотов, становится возможным управлять информацией, полученной из разных источников.

Логическая МПЗ. В основе логических моделей лежит понятие формальной системы. Формальная система представляет собой совокупность абстрактных объектов, в которой представлены правила оперирования множеством символов в чисто синтаксической трактовке без учета их семантики (или смыслового содержания). Формальная система (ФС) определена, если:

- 1) задан конечный алфавит (или конечное множество символов);
- 2) определена процедура построения правильных формул;
- 3) выделено некоторое множество формул, называемых аксиомами;
- 4) задано конечное множество правил вывода, которые позволяют получать из некоторого конечного множества формул другое множество формул.

Справедливость утверждения (или формулы) устанавливается или опровергается на основании аксиом и правил вывода этой формальной системы.

На практике чаще всего в качестве формальной системы используется *исчисление предикатов первого порядка* [149].

Обеспечивая высокий уровень формализации и механизм проверки выполнимости формул, исчисление предикатов имеет ограниченные выразительные возможности и не слишком удобно для описания предметных областей.

Дескрипционные логики (ДЛ) – это формализм, предназначенный для представления знаний о предметной области [6]. Дескрипционные логики сочетают в себе высокие выразительные возможности семантических сетей и фреймов и высокий уровень формализации описания знаний, характерный для исчисления предикатов.

Базовыми элементами ДЛ являются атомарные концепты и роли. Каждый концепт представляет некую сущность, класс описываемой предметной области. Роли служат для описания свойств концептов и отношений между ними. В ДЛ также входят конструкторы и синтаксические правила, с помощью которых из базовых элементов можно строить сложные концепты. В соответствии с набором конструкторов определяется класс конкретной ДЛ.

Базовой ДЛ, обладающей минимальным набором конструкторов и представляющей практический интерес, является логика ALC (Attributive Language with Complement). Она включает такие конструкторы, как объединение, пересечение, отрицание концептов, ограничения на значения концепта, задаваемые с помощью ролей, кванторов всеобщности и существования, а также численных ограничений мощности.

С помощью языка ДЛ можно задавать аксиомы общего вида о понятиях и их взаимосвязях (терминологические аксиомы) и частного вида – утверждения о конкретных объектах и их связях с другими объектами.

Семантика ДЛ задается путем определения функции интерпретации, которая каждому атомарному концепту сопоставляет множество конкретных

объектов, принадлежащих некому фиксированному множеству (домену), и каждой атомарной роли – множество пар объектов (или бинарное отношение на домене). Для определения семантики составных концептов вводится расширение функции интерпретации.

С помощью терминологических аксиом и утверждений о конкретных объектах задаются знания об описываемой предметной области.

Продукционная МПЗ предполагает такой способ организации вычислительного процесса, при котором программа преобразования некоторой информационной структуры S задается в виде системы правил вида: Условие $\rightarrow$ Действие, где Условие задает некоторые требования к текущему состоянию структуры S , а Действие содержит описание тех операций над S , которые надо выполнить, если S удовлетворяет этим требованиям [152,190].

Системы, использующие продукционную МПЗ делятся на *формальные* и *программные* системы. Наибольший практический интерес представляют программные системы продукций, т.е. системы, имеющие программную реализацию. Программная система продукций состоит из трех основных частей: базы данных, множества правил-продукций и интерпретатора [150]. База данных (БД) является рабочей памятью с разной структурной организацией, над которой работает множество правил. Правила могут иметь произвольную сложность, но их структура остается прежней: левая часть – условие применимости, а правая часть – действие, которое данное правило выполняет. Интерпретатор представляет собой поисковый процесс, который включает два обязательных действия – выбор продукции и ее применение.

1.2.2. Комбинированные модели представления знаний

Классические МПЗ широко используются в современных интеллектуальных системах. Однако выразительных и вычислительных возможностей каждой отдельно взятой модели часто оказывается недостаточно для решения прикладных задач. Поэтому на практике, как правило, используется сочетание нескольких таких моделей.

Онтологическая МПЗ. В основе данной модели лежит понятие *онтологии*, которое появилось в научном сообществе на рубеже столетий и к настоящему времени завоевало всеобщее признание. Существует множество определений онтологий.

Одним из самых известных является ставшее классическим определение Томаса Грубера [32]: «Онтология – это явная спецификация концептуализации». Под концептуализацией понимается упрощенное представление некоторого фрагмента реального мира.

Никола Гуарино [33] определяет онтологию как формальную теорию, ограничивающую возможные концептуализации мира.

Известные российские специалисты в области онтологического инжиниринга Гаврилова Т.А., Хорошевский В.Ф. дают неформальное определение: «Онтология – это БЗ специального типа, которые могут «читаться», пониматься и отчуждаться от разработчика и/или физически разделяться их пользователями». Более формально они определяют онтологию как $O = \langle C, R, F \rangle$, где C – конечное множество концептов (понятий) предметной области; R – конечное множество отношений между концептами; F – конечное множество функций интерпретации, заданных на концептах и/или отношениях [86].

В работе Доброва Б.В., Лукашевич Н.В. и их соавторов дается следующее определение: «онтология – это система, состоящая из набора понятий и набора утверждений об этих понятиях, на основе которых можно строить классы, объекты, отношения, функции и теории» [99].

Основная цель разработки онтологий заключается в обеспечении поддержки деятельности по накоплению, распределению (совместному использованию) и повторному использованию знаний [190].

Онтологии, предоставляя формализм для представления знаний, в свою очередь, опираются на формализмы семантических сетей и дескрипционных логик.

Интегрированная МПЗ, описанная в работе [103], представляет собой модель, объединяющую в рамках одного формализма различные

взаимодополняющие друг друга средства и методы представления и обработки знаний следующих моделей – онтологической, сетевой, продукционной, императивной и недоопределенной [69]. Недоопределенная МПЗ [141,142] включает аппарат недоопределенных типов данных и разработанный на его основе метод недоопределенных вычислительных моделей, который, является универсальным методом программирования в ограничениях. Комбинация методов и средств представления знаний, входящих в состав данной МПЗ, является достаточной для использования в современных интеллектуальных системах при решении практических задач.

Признаковые пространства. В ИСППР очень часто используются методы экспертного оценивания альтернатив разрешений проблемной ситуации (п.1.4.7.). В этих методах используются понятия критериев (признаков), по которым оценивают альтернативы. Каждая альтернатива задается вектором, состоящим из ее оценок по выбранным критериям, представляя собой точку в пространстве признаков [125]. Критериев может быть очень много, их выбор совсем неочевиден, и здесь большую роль играют знания и опыт эксперта. Таким образом, сложности построения признаковых пространств и использование при этом знаний экспертов позволяют говорить о «пространственно-признаковых» МПЗ.

1.3. Принципы и подходы к разработке интеллектуальных СППР

Разработка ИСППР в СФПО является сложной задачей. Для ее решения предложено много подходов, как в России, так и за рубежом. Современные ИСППР должны обладать достаточным набором функциональных возможностей, чтобы удовлетворять потребностям пользователей. Исследователи в области ИСППР формулируют принципы, подходы, методологии, которые должны закладываться при разработке этих систем.

1.3.1. Принципы разработки СППР

В работе [66] сформулированы принципы, которым должна удовлетворять «идеальная» СППР:

1. Оперирование слабоструктурированными решениями;
2. Возможность использования для ЛПР различного уровня;
3. Возможность группового или индивидуального использования;
4. Поддержка как взаимозависимых, так и последовательных решений;
5. Поддержка трех фаз процесса решения: интеллектуальной части, разработки альтернатив и самого выбора;
6. Поддержка различных методов и стилей решения, характерных для решения задач группой лиц;
7. Гибкость и адаптивность к изменениям как внутренних, так и внешних факторов;
8. Простота в эксплуатации и модернизации;
9. Способность повышать эффективность процесса принятия решения;
10. Возможность для ЛПР компьютерного управления процессом принятия решений;
11. Поддержка эволюционного использования и способность легко адаптироваться к изменяющимся требованиям;
12. Простота разработки СППР при формулировании логики ее конструкции;
13. Поддержка моделирования;
14. Использование знаний.

Эти принципы, сформулированные более 20 лет назад, являются актуальными и в настоящее время.

В работе [160] говорится об унификации программного обеспечения прикладных СППР, которая подразумевает использование каркасной архитектуры программного обеспечения и следующих принципов проектирования: принцип свободы от субъективизма разработчиков, позволяющий системе быть максимально гибкой и подстраиваться под нужды и потребности ЛПР; принцип инвариантности по отношению к предметной области; принцип множественности

методов, при котором СППР обеспечивает легкую и прозрачную интеграцию требуемых методов, советует пользователю, какой метод лучше применим и почему.

В настоящее время широко признанным является мнение, что современная ИСППР должна быть информационной системой с развитыми средствами анализа и обработки хранящихся в них данных. В статье [164] делается попытка философского осмысления назначения и функциональных возможностей СППР без привязки к конкретной предметной области или особенностям технической реализации. Авторы подчеркивают важность обеспечения ЛПР информацией об особенностях реализации процессов управления в предметной области, о порядке выработки и принятия решений, о методах и инструментальных средствах принятия решений, а также представления этой информации в виде, удобном для подготовки принятия решения.

1.3.2. Универсальные подходы к разработке интеллектуальных СППР

Ретроспективный обзор методологий и подходов к разработке СППР дан в работах [28], [50]. В работе [48] подчеркивается, что не существует наилучшего стандартного подхода или методологии для разработки СППР. Power выделяет три основных подхода, указывая их плюсы и минусы: 1) подход, обеспечивающий полный жизненный цикл системы (SDLC); 2) быстрое прототипирование и 3) подход конечного пользователя, при котором ЛПР может сам выбирать средства для решения своих задач. Второй и третий подходы позволяют быстро получить рабочую систему. Наиболее распространенным, универсальным и практически значимым автор считает подход быстрого прототипирования, который дает возможность быстро оценить возможности полученной системы и при необходимости провести её модификацию. Такой подход позволяет объединять усилия ЛПР и аналитика.

В последнее время много говорится о целесообразности использования подхода гибкой разработки программного обеспечения (Agile software development) для создания СППР [27]. Такие принципы этого подхода, как

использование итеративной разработки, динамическое формирование требований, тесное взаимодействие разработчиков различного профиля, перекликаются с подходом быстрого прототипирования.

В работах [90,160] рассматривается подход каркасной разработки СППР подразумевающий создание каркаса СППР, который обеспечивает базовую функциональность системы и поддерживает вариативность поведения пользователя через соответствующие точки расширения.

Перечисленные подходы являются универсальными для создания прикладных программных систем и с успехом используются на практике при разработке СППР.

О создании прикладных (медицинских) СППР говорится в [100]. Авторы считают перспективными как разработку специализированных систем, так и использование гибких, легких сервисов, позиционируемых как отдельные приложения и интегрируемых между собой.

Примером использования подхода конечного пользователя, в котором СППР строится для решения определенного класса задач, содержит множество методов и позволяет выбирать методы для решения конкретных задач, является описанная в работе [115] экспертная СППР (ЭСППР). ЭСППР содержит около 50 методов принятия решений, а также базу знаний, представленную в виде последовательности вопросов об элементах задачи принятия решения и их различных реализациях (ответах). ЭСППР является web-приложением, что делает ее доступной широкому кругу пользователей и позволяет привлекать экспертов для оценки альтернативных вариантов решения. Данная система ориентирована на решение экономических задач.

1.3.3. Онтологический подход

В современных СППР онтологии широко используются на всех стадиях их жизненного цикла. В работе [183] дается обзор экспертных систем и СППР, созданных на основе онтологий, и определяется роль онтологии. В этой работе говорится, что «при разработке СППР онтология может использоваться для:

- формирования и фиксации общего разделяемого всеми экспертами знания о предметной области (ПО);
- явной концептуализации ПО, позволяющей описывать семантику данных при разработке баз знаний;
- обеспечения возможности переиспользования знаний;
- описания функциональных возможностей СППР (типов решаемых задач).

В процессе функционирования СППР онтология может использоваться для обеспечения:

- совместного использования разнородных данных и знаний в рамках одной системы;
- процесса решения задач, на которые ориентирована система;
- лучшего понимания предметной области пользователями системы.»

В настоящее время онтологии де-факто являются основой баз знаний интеллектуальных систем. В работе [88] указываются преимущества использования онтологий при проектировании БЗ:

- Онтологии являются мощным инструментом познания. Представляя информацию в виде сети связанных отношениями понятий, онтологии позволяют сделать видимыми семантические сети памяти человека.
- Онтологии обеспечивают системный подход к изучению предметной области, что существенно облегчает её структурирование.

При разработке интеллектуальных систем онтологии являются платформо- и предметно-независимым средством организации взаимодействия между компьютерными системами, поскольку они представляют собой системы описания знаний в формате, совместимом со многими языками представления.

Разработка онтологий хорошо поддержана методически и инструментально. Существует целый ряд языков и программных систем для разработки онтологий [16,46,51,61,107,145], предложены методики их построения [88, 102], и оценивания [87, 167].

1.3.4. Фрактальный подход

Фрактальный подход к структурированию знаний был предложен и описан в работах [128,135]. Автором было сделано предположение, что знаниям присуще свойство фрактальности, характерной чертой которой является самоподобие. Предложенная концептуальная модель структурирования знаний, названная фрактальной стратифицированной моделью (ФС-модель), основана на представлении разных видов знаний как объектов расслоенного (стратифицированного) информационного пространства, представляющего собой известную нам часть знаний о мире. Объектом в этом мире является некоторая совокупность свойств, проявляющаяся в виде реального объекта или процесса. «Сущность объекта – это инвариант, совокупность существенных и неизменяющихся свойств объекта. Тогда информация – это совокупность описаний сущностей и отношений между ними». При построении ФС-модели вводится понятие информационных миров – подпространств, объединяющих однотипные информационные объекты. Тогда ФС-модель определяется как «совокупность непересекающихся слоев (информационных миров) и их отображений в информационном пространстве. Каждому уровню соответствует свой слой (страта), а последовательность отображений отражает процесс познания». «Графически ФС-модель представляется в виде вложенных сферических оболочек, информационные объекты – точками на той или иной сфере» в зависимости от степени детализации их рассмотрения (Рисунок 1.3). Каждый исследователь практически работает со своим «фракталом» знаний – «вырезкой» из информационного пространства, которую можно представить в виде конуса или пирамиды.

Данный подход прошел проверку временем, он позволяет обосновать введение мета-мета-...уровней при проектировании баз знаний [112,129].

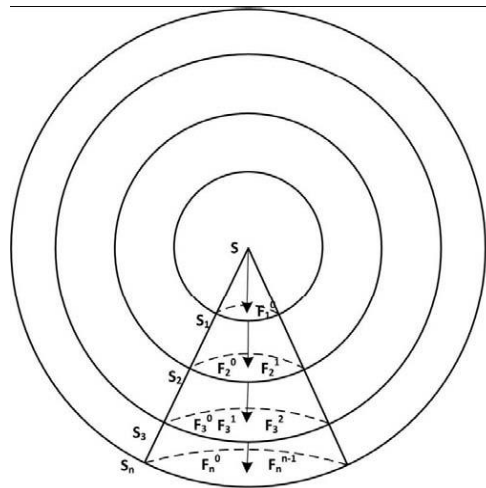


Рисунок 1.3 Графическое представление ФС-модели

1.3.5. Срезы знаний

Концепция «срезов знаний» предложена в работе [80]. «Срезы представляют собой множества связей онтологии и концептов, относящихся к этим связям. Деление онтологической схемы на такие множества предоставляет пользователю возможность рассматривать тот или иной вопрос с разных точек зрения, например, с точки зрения истории, практического применения, научного обоснования». «К историческому срезу отнесены такие свойства, как «объяснил», «провел», «открыл», «доказал», «ввел понятие». Этот срез позволяет взглянуть на некоторое явление в его историческом контексте – выяснить какие ученые его изучали, какие опыты они проводили. К практическому срезу отнесены такие свойства, как «основано на», «использует». Этот срез позволяет оценить практическое применение понятий». «К теоретическому срезу отнесены такие свойства, как «есть условие», «является параметром», «является частью». Этот срез позволяет увидеть концепты, с помощью которых рассматриваемое понятие может быть объяснено». «Одно свойство может входить в несколько срезов, что обеспечивает связь между ними».

1.3.6. Сервис-ориентированный подход

По определению организации OASIS (Organization for the Advancement of Structured Information Standards), сервис-ориентированный подход – это «парадигма организации и использования распределенных возможностей,

которые могут принадлежать различным владельцам» [44]. Данный подход основан на использовании распределённых, слабо связанных заменяемых компонентов, называемых *сервисами*, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам. Интерфейсы компонентов инкапсулируют детали их реализации. Это делает сервисы автономными, независимыми от используемых платформ и инструментов разработки. Системы, построенные с использованием таких компонентов, можно легко масштабировать и модифицировать. Использование стандартизированных интерфейсов обеспечивает интеграцию компонентов для построения сложных программных систем и решения нетривиальных задач. Сетевые протоколы, которые могут использоваться для взаимодействия между компонентами, позволяют получить доступ к информации и функциональным компонентам, размещенным на удаленных серверах.

Сервис-ориентированная архитектура [19,20] может быть реализована с использованием разных технологий – REST [23], CORBA [12], RPC [155], web-сервисы и другие. Наиболее популярными средствами разработки web-сервисов являются тройка стандартов SOAP/WSDL/UDDI [13,34] и архитектурный стиль REST. Применение этих технологий требует от программиста знания специализированных языков и протоколов (п. 1.6). Более просты в использовании подходы, основанные на WFMS (work flow management system) – системах управления потоками работ [14,64,111,161]. Такие системы позволяют создавать сервисы из готовых программных модулей разных типов, предоставляя для этого простые в использовании редакторы и графические языки. В настоящее время активно разрабатываются подходы к созданию WFMS. Существует ряд платформ, поддерживающих их разработку [108]. Они решают как технологические, так и организационные проблемы, связанные с вовлечением исследователей в процесс разработки проблемно-ориентированных сервисов.

Совокупность основных функциональных возможностей и свойств указанных выше платформ представлена на Рисунке 1.4.



Рис. 1.4. Функциональные возможности и свойства платформ для построения распределенных систем

«Системы, разрабатываемые с помощью подобных платформ, ориентированы на прикладного пользователя и предоставляют графический desktop- или web- интерфейс. Такие инструментарины поддерживают широкий спектр ресурсов. Они могут быть развернуты на локальных или удаленных серверах, использовать грид- или облачную инфраструктуру, предоставлять разнообразное программное обеспечение (сервисы) – web-сервисы, локальные или удаленные вызовы модулей из командной строки, библиотеки Java, пакетные или API приложения. Платформы ориентированы на поддержку моделей обслуживания SaaS (Software as a Service) или AaaS (Application as a Service)». Они предоставляют «возможность публикации сервисов и их повторного использования для разработки композитных приложений на основе концепции workflow (WF) – потоков работ или сценариев. Для управления WF (разработки, редактирования, исполнения, преобразования в новый сервис) инструментальные системы предоставляют редакторы с графическим интерфейсом или специализированным языком. Для передачи данных между модулями WF могут быть задействованы разные механизмы: через машину пользователя или распределенную файловую систему, либо путем использования сервисов-посредников или встроенных механизмов на стороне сервера» [186].

1.4. Методы поддержки принятия решений и реализующие их средства

К настоящему времени разработано и описано большое количество методов поддержки принятия решений. Для многих из них есть программные реализации и/или подробные методики использования.

Рассмотрим методы, используемые для поддержки принятия решений в СФПО, и реализующие их программные средства.

1.4.1. Методы и средства визуализации информации

Методы визуализации информации играют важную роль при разработке и использовании баз знаний. Визуализация является мощным инструментом познания. Визуальные модели (графы, диаграммы, карты и другие) обладают особенной когнитивной силой при структурировании информации [86]. Одним из наиболее эффективных визуальных представлений являются графы. Считается, что они являются инструментом, позволяющим сделать видимыми понятийные (семантические) сети памяти человека, поскольку память наиболее адекватно представляется именно сетевой структурой.

В зависимости от целей и типа информации ее визуализация выполняется путем построения различных карт.

Интеллект-карты (mind maps) – один из самых доступных методов отображения понятийных структур. Автором этого метода считается Тони Бьюзен [10,79]. Идея интеллект-карт заключается в использовании и совмещении функции левого и правого полушарий для достижения целостного и наглядного представления идеи. Фактически это означает переход от последовательного (текстового) изложения к сетевому (образному). Интеллект-карты имеют 4 отличительные черты:

- объект внимания/изучения кристаллизован в центральном образе;
- основные темы, связанные с объектом изучения, расходятся от центрального образа в виде ветвей, которые поясняются ключевыми словами или образами;

- вторичные идеи также ветвятся;
- ветви формируют связанную узловую структуру.

Таким образом, интеллект-карты показывают связи и древовидную структуру произвольных фрагментов знаний.

Для построения интеллект-карт можно использовать любой программный графический пакет. Однако сейчас все большую популярность приобретают специализированные методы и инструменты. На рынке программного обеспечения имеется большое число средств для быстрого и красивого формирования интеллект-карт [25,40,67].

Построение **концептуальных карт** (concept maps) – метод, позволяющий глубоко рассмотреть предметную область, который был предложен Новаком в начале 70-х гг. [43]. Концептуальная карта – это граф, в узлах которого представлены понятия (концепты) ПО, а направленные поименованные дуги представляют отношения, или связи между узлами-понятиями. Разработка такого графа подразумевает анализ структурных взаимодействий между отдельными понятиями предметной области. В процессе создания концептуальной карты эксперт и аналитик анализируют структуру отношений предметной области, что помогает им глубже понимать ее природу. Зачастую приходится генерировать новые, ранее невербализованные связи. Таким образом, концептуальная карта – не только цель, но и средство. В процессе построения, то есть при взаимодействии семантических связей памяти человека с визуальной информацией, связи перестраиваются, порождая, в свою, очередь новые знания [139].

Для построения концептуальных карт чаще всего используется специализированный графический редактор Сmap [37].

Построение **когнитивных карт** – метод, предназначенный для отображения причинно-следственных связей между факторами, влияющими на развитие некоторой ситуации реального мира. Когнитивная карта отображает упрощенную субъективную модель функциональной организации некоторой наблюдаемой системы. Она представляет собой ориентированный граф, в узлах

которого представлены факторы, влияющие на развитие ситуации, а ребра отражают причинно-следственные связи между факторами. На ребра навешиваются весовые коэффициенты, отражающие степень влияния факторов. Знак коэффициента означает, что увеличение фактора-посылки влечет за собой увеличение фактора-следствия при положительном коэффициенте, и уменьшение – при отрицательном.

Создать когнитивную карту можно в любом графическом редакторе. Однако она является вспомогательным средством для дальнейших исследований и преобразований – проведения когнитивного моделирования, требующего привлечения специальных методов и инструментов (п. 1.4.3).

1.4.2. Методы извлечения и приобретения знаний

При построении баз знаний интеллектуальных систем, и в частности, ИСППР, важную роль играют методы извлечения знаний.

В работе [85] предложена классификация методов извлечения знаний на основании их источника. Эти методы делятся на две основные группы – коммуникативные и текстологические.

Коммуникативные методы охватывают все виды контактов с живым источником знаний – экспертом. Текстологические методы предполагают извлечение знаний из документов (методик, пособий, руководств) и специальной литературы (статьи, монографии, учебники).

Коммуникативные методы делятся на активные и пассивные. В активных методах ведущая роль, инициатива принадлежит инженеру знаний, который активно контактирует с экспертом различными способами – в играх, диалогах и т.п. В пассивных методах ведущую роль играет эксперт, а инженер знаний только протоколирует рассуждения эксперта во время его реальной работы по принятию решений или записывает то, что эксперт считает нужным сказать во время беседы или лекции.

В работе [150] дается определение процесса приобретения знаний интеллектуальной системой как получения (извлечения) знаний и передачи их

системе. Автор описывает фазы этого процесса (предварительная, начальная, накопительная) и выделяет исторически сложившиеся его модели, которые различаются разной степенью вовлеченности экспертов в процесс приобретения знаний, а иногда и в сам процесс разработки интеллектуальной системы.

1.4.3. Методы семантического моделирования

Методы семантического моделирования активно развиваются многими исследователями и научными коллективами. К этой группе относятся методы онтологического, когнитивного, событийного, вероятностного моделирования [133], в основе которых лежит сетевая МПЗ (п. 1.2.1.). Наиболее известными и востребованными являются методы онтологического и когнитивного моделирования.

Онтологическое моделирование – это процесс построения, сопровождения и использования онтологий. В работе [92] описывается последовательность разработки онтологической модели.

На начальных этапах построения онтологий широко используется ряд методов и средств, облегчающих и ускоряющих этот процесс.

Методы визуализации информации (п. 1.4.1) позволяют легче и качественнее выполнить концептуализацию моделируемой ПО.

Предметные онтологии разного уровня детализации, имеющиеся в свободном доступе, а также методы их *доработки, слияния и выравнивания* [120], позволяют переиспользовать ранее созданные фрагменты знаний.

Метод разработки онтологий на основе *паттернов онтологического проектирования* (Ontology Design Patterns) позволяет упорядочить и облегчить процесс построения онтологий и помочь разработчикам избежать некоторых часто повторяющихся ошибок онтологического моделирования [9,22,180,191].

Основными средствами «спецификации концептуализации» являются *языки описания онтологий и редакторы построения онтологий*.

Одним из самых распространенных языков онтологического моделирования является язык OWL (Web Ontology Language) [3], обеспечивающий большую

выразительность и более богатую семантику, чем другие языки описания онтологий.

Язык OWL имеет три диалекта: OWL Full, OWL DL и OWL Lite, которые отличаются разным соотношением выразительных возможностей и разрешимости. Наиболее оптимальным соотношением обладает диалект OWL DL, в основе которого лежит дескрипционная логика SHOIN(D) [6], являющаяся расширением логики ALC (п. 1.2.1).

Язык OWL предназначен, в первую очередь для машинной обработки онтологий и достаточно сложен для восприятия человеком. Для разработки и модификации онтологий (в том числе, с последующим их сохранением на языке OWL) существует большое количество редакторов. В работе [88] приводится обзор существующих редакторов онтологий. Наибольшую популярность завоевал редактор Protégé [51].

Когнитивное моделирование – процесс генерации и проверки гипотез о функциональной структуре наблюдаемой ситуации до получения функциональной структуры, способной объяснить поведение наблюдаемой ситуации. Данный метод был предложен Р. Аксельродом [5]. Когнитивное моделирование позволяет решать такие задачи, как структурирование и анализ проблемной ситуации, прогнозирование развития проблемной ситуации, разработка альтернатив разрешения проблемной ситуации.

Когнитивное моделирование включает два основных этапа – построение модели, роль которой играет когнитивная карта (п. 1.4.1.), и взаимодействие с ней.

В зависимости от типа когнитивной карты методы работы с ней можно подразделить на две основные группы [91] – статические и динамические.

Статические методы анализа когнитивной карты позволяют проследить и изучить причинно-следственные связи, влияние одного фактора на другой, исследовать структурную устойчивость системы, рассмотреть достижимость целевых факторов. К данной группе относятся методы, основанные на теории графов: анализ путей и циклов; анализ связности.

Динамические методы основываются на импульсном моделировании и сценарном анализе. Импульсное моделирование – моделирование распространения возмущения на когнитивной карте. Возмущение достигается внесением импульса или управляющего воздействия в вершину (множество вершин) графа. Сценарный анализ позволяет по результатам импульсного моделирования рассуждать о поведении системы, о её возможном развитии.

Анализ устойчивости представляет собой исследование устойчивости по значению и устойчивости по возмущению модели по мере её эволюции.

Для построения и анализа когнитивных карт был создан ряд программных систем: КОМПАС [117], Канва [118], grMCogMap [131] и др.

1.4.4. Методы аналитического моделирования в слабоформализованных предметных областях

В контексте поддержки принятия решений в СФПО методы аналитического моделирования используются для решения задач в условиях НЕ-факторов (неполноты, неточности, неопределенности используемых знаний и данных) [143]. Одним из таких методов является *метод недоопределенных вычислений* (МНВ) Нариньяни [141].

МНВ относится к методам программирования в ограничениях [65,142, решаящих задачу, которую упрощенно можно сформулировать следующим образом: найти значения переменных, удовлетворяющие определенным ограничениям. Концепция программирования в ограничениях очень привлекательна для конечного пользователя, поскольку оно является наиболее декларативным, т.е. основано на описании модели задачи, а не алгоритма ее решения. Модель специфицируется как совокупность параметров задачи и связывающих их отношений. Отношения задают ограничения на значения параметров, которые могут иметь вид уравнений, неравенств, логических выражений и т.п. Они записываются в привычной математической нотации и не предполагают разделения связываемых ими объектов на входные и выходные. Это позволяет решать на одной и той же модели как прямые, так и обратные

задачи. Используя модель задачи и исходную информацию о значениях ее параметров, методы программирования в ограничениях обеспечивают автоматическое нахождение решения.

МНВ реализован в ряде прикладных систем, ориентированных на определенные классы задач и обладающих специализированными пользовательскими интерфейсами: системы UniCalc [114] и NeMo+[182] являются наиболее универсальными, предназначенными для решения широкого спектра прикладных задач; Time-Ex используется при решении задач календарного планирования [144]; технологический комплекс Интегра [188] и входящая в ее состав система Финплан [170] предназначены для поддержки принятий решений в таких областях, где исходную информацию удобно представлять в виде таблиц.

Подробнее остановимся на технологическом комплексе Интегра. Модель решаемой средствами Интегры задачи, ее параметры и связывающие их ограничения, хранятся в ячейках таблицы. Однако, в отличие от известных электронных таблиц (например, Excel [41]), Интегра позволяет задавать интервальные значения параметров и приписывать ячейкам таблицы одновременно со значениями совокупность формул, по которым можно вычислить эти значения. Возможность пошагового вычисления с сохранением промежуточных результатов, средства графического представления, позволяют позиционировать Интегру как оболочку для разработки систем поддержки принятия решений, дающих возможность ЛПР проанализировать все возможные альтернативы развития ситуации и определить, какие действия нужно предпринять сейчас, чтобы ситуация развивалась по наиболее благоприятному сценарию.

1.4.5. Метод рассуждений на основе прецедентов

В основе данного метода лежит понятие Прецедента (от латинского praecedens – предшествующий) – имевшего ранее место случая, на примере которого можно делать выводы в последующих случаях. Вывод на основе прецедентов (CBR – Case-Based Reasoning) является подходом, позволяющим

решить новую, неизвестную задачу, используя или адаптируя решение уже известной задачи, т.е. используя уже накопленный опыт решения подобных задач [57,82].

Решение задач методом CBR выполняется на основе так называемого CBR-цикла, основными этапами которого являются:

1. извлечение наиболее похожего на сложившуюся ситуацию прецедента из библиотеки прецедентов;
2. попытка использования извлеченного прецедента для решения текущей проблемы;
3. при необходимости пересмотр и адаптация полученного решения в соответствии с текущей ситуацией;
4. сохранение вновь принятого решения как части нового прецедента.

Метод рассуждений на основе прецедентов имеет как достоинства, так и ограничения в его применимости: К основным достоинствам CBR можно отнести 1) возможность использовать накопленный системой опыт без привлечения экспертов; 2) ускорение поиска решения за счет использования уже имеющегося решения подобной задачи и отсутствия необходимости глубокого изучения предметной области.

Недостатком этого метода является то, что его невозможно применить в случаях, для которых нет прецедентов или степень их близости меньше заданного порогового значения. Кроме того большой объем базы прецедентов может привести к снижению производительности системы. Также имеются проблемы при определении критериев для индексации и сравнения прецедентов.

Одной из наиболее популярных систем, реализующих метод CBR, является **Colibri** – фреймворк с открытым исходным кодом для разработки программного обеспечения на основе прецедентов, разработанный группой IAI Мадридского университета Комплутенсе [38,53]. Его основное назначение заключается в обеспечении инфраструктуры, необходимой для разработки новых систем CBR и связанных с ними программных компонентов. Colibri предлагает среду, где пользователи могут поделиться своими достижениями в реализации приложений

CBR. Это – открытая платформа, где пользователи могут внести свой вклад различными конструкциями или компонентами, которые будут повторно использоваться другими пользователями. Фреймворк Colibri предлагает широкий выбор стратегий поиска и выбора прецедента, а также метрик сходства.

1.4.6. Метод рассуждений на основе экспертных правил

Метод рассуждений на основе экспертных правил предназначен для моделирования действий эксперта при решении его профессиональных задач. Он основан на использовании двух МПЗ: сетевой и продукционной (п.1.2.1.). Предметная область и проблемная ситуация, как правило, задаются в виде семантической сети, над которой работает система продукционных правил, в которых «аккумулируется» опыт эксперта.

Программные продукционные системы используются для решения ряда слабоструктурированных проблем: диагностики, прогнозирования, классификации и составляют ядро экспертной подсистемы в составе ИСППР (п. 1.1.6.).

Для разработки экспертных подсистем существует ряд программных инструментов.

G2 – платформа компании Gensym для разработки и внедрения экспертных приложений [29]. Данная платформа предоставляет богатый спектр возможностей и с успехом используется для разработки экспертных систем и СППР. Однако доступность данного продукта существенно ограничена из-за его высокой стоимости.

CLIPS (C Language Integrated Production System) — программная среда для разработки экспертных систем [98]. В настоящее время CLIPS является одной из наиболее широко используемых инструментальных сред для разработки экспертных систем благодаря своей скорости, эффективности и бесплатности [165]. Являясь open source разработкой, она до сих пор обновляется и поддерживается своим автором, Гэри Райли (Gary Riley).

CLIPS поддерживает продукционную МПЗ, а для реализации вывода использует эффективный алгоритм Rete [24]. Для разработки экспертных систем CLIPS предоставляет полноценный объектно-ориентированный язык COOL. Расширения можно создавать на языке Си, кроме того, можно интегрировать CLIPS в программы на языке Си.

Существует ряд разработок на основе базового CLIPS, в которых поддерживается обратный вывод и нечеткая логика [105].

В России использование CLIPS несколько ограничено отсутствием полноценной поддержки кириллицы.

Semp-TAO – технологический комплекс для разработки систем, основанных на знаниях, разработанный в лаборатории искусственного интеллекта Института систем информатики им. А.П. Ершова [69,101]. С его помощью эффективно решаются задачи, которые требуют сочетания логического вывода и вычислений над неточно или частично заданными значениями [171,177].

В основе комплекса SemP-TAO лежит интегрированная МПЗ (п. 1.2.2.), которая объединяет такие классические средства и методы представления и обработки знаний, как фреймы, семантические сети, продукционные системы, а также недоопределенные модели и распространение ограничений на основе объектно-ориентированного подхода.

Основным средством представления декларативных знаний служит семантическая сеть, состоящая из объектов, которые связаны бинарными отношениями. Процессы вывода и обработки информации задаются главным образом в виде системы продукций, работающей над такой семантической сетью.

Для спецификации приложений в различных предметных областях SemP-TAO предлагает язык представления и обработки знаний, разработанный на основе интегрированной МПЗ. Этот язык обладает богатым набором типов данных, позволяет представлять понятия предметной области в виде классов объектов и отношений и специфицировать требуемые в приложениях процессы вывода и обработки информации. Он также включает средства, необходимые для разработки пользовательских интерфейсов и работы с текстовыми файлами.

Программа на языке системы SemP-ТАО состоит из двух частей: объектно-ориентированного описания предметной области и описания логики разрабатываемого приложения в виде продукционных правил и операторов управления их активацией.

1.4.7. Методы экспертных оценок

Методы экспертных оценок (МЭО) – это большой класс методов организации работы со специалистами-экспертами и обработки их мнений. Экспертные исследования проводят с целью подготовки информации для принятия решений ЛПР [146].

МЭО предполагают проведение процедуры, состоящей из нескольких шагов:

1. Принятие ЛПР решения о необходимости проведения экспертного опроса и формулировка его цели.
2. Подбор и назначение основного состава Рабочей группы (РГ).
3. Разработка Рабочей группой технического задания на проведение экспертного опроса.
4. Разработка Рабочей группой подробного сценария проведения сбора и анализа экспертных мнений (оценок). На этой стадии фиксируются конкретный вид информации, которая будет получена от экспертов, и конкретные методы анализа собранной информации.
5. Подбор экспертов в соответствии с их компетентностью.
6. Формирование экспертной комиссии (ЭК).
7. Проведение сбора экспертной информации.
8. Оцифровка экспертной информации и проведение её компьютерного анализа с помощью включенных в сценарий методов.
9. При применении экспертной процедуры из нескольких туров – повторение двух предыдущих этапов.

10. Итоговый анализ экспертных мнений, интерпретация полученных результатов аналитической группой РГ и подготовка заключительного документа ЭК для ЛПР.

11. Официальное окончание деятельности РГ.

МЭО делятся на две основные группы: индивидуальные методы, в которых решение принимается одним экспертом, и коллективные, применяемые в сложных случаях, требующих учета мнений нескольких специалистов.

К индивидуальным методам, используемым в СФПО для решения слабоструктурированных задач, относятся методы рационального выбора [124,148,156]. Они предназначены для решения следующей задачи: имеется несколько альтернатив $A_1, \dots, A_n$ разрешения проблемной ситуации. Каждая альтернатива оценивается множеством критериев $K_1, \dots, K_n$ ($n \geq 1$), которые могут принимать как количественные, так и качественные значения, так, что каждая альтернатива A_i представляется точкой $(x_{i1}, x_{i2}, \dots, x_{in})$ в признаковом пространстве (п. 1.2.2.). Основываясь на предпочтениях ЛПР, необходимо решить одну из задач: 1) выделить лучший вариант; 2) упорядочить варианты от лучшего к худшему; 3) отнести каждый вариант к одному из заранее определенных классов.

Для решения задачи рационального выбора используются разные модели формализации предпочтений ЛПР. Модель может иметь формализованное представление и состоять из определенных требований и правил описания альтернатив, процедур их сравнения и выделения лучших, может задаваться функциональными зависимостями, либо быть неформализованной.

В работе [148] в зависимости от характера информации, используемой для оценки критериев, выделено четыре группы методов рационального выбора, характеризующихся:

- количественными показателями и сравнением вариантов по числовому значению их ценности;

- качественными показателями, которые переводятся в числовые оценки альтернатив и числовые ценности альтернатив;
- количественными показателями и сравнением альтернатив без вычисления их числовой ценности.
- качественными показателями и сравнением альтернатив без вычисления их числовой ценности.

Для классификации коллективных методов выделяются дополнительные основания:

- Число туров. Экспертизы могут включать один тур, некоторое фиксированное число туров или неопределенное число туров.
- Организация общения экспертов. Возможны следующие способы общения: отсутствие общения, заочное анонимное общение, заочное общение без анонимности, очное общение с ограничениями, очное общение без ограничений.

В реальных экспертизах, как правило, используются комбинации методов разных типов.

1.5. Технологии разработки интеллектуальных СППР

1.5.1. Фреймворки с открытым исходным кодом

Известный исследователь в области разработки СППР D. J. Power считает, что не существует наилучшего стандартного подхода или методологии для разработки этих систем [48]. Это тем более относится и к современным ИСППР, имеющим в своем составе несколько подсистем, для разработки которых должны применяться разные средства и технологии. Довольно распространенным является подход, когда такие системы разрабатываются на основе технологии, предлагающей средства для реализации основной функциональности ИСППР и позволяющей встраивать и/или интегрировать средства, обеспечивающие

остальные возможности разрабатываемой системы. Такой подход реализован в рассмотренных выше программных системах – фреймворке Colibri (п.1.4.5) и программной среде CLIPS (п.1.4.6).

1.5.2. Технология Semantic Web

В настоящее время одной из наиболее востребованных технологий для разработки интеллектуальных систем, и в частности ИСППР, является технология Semantic Web [36]. Эта технология предлагает методы и средства для представления знаний и данных в пригодном для машинной обработки виде. Инструменты Semantic Web предназначены, в первую очередь, как это следует из названия, для отображения семантики объектов, размещенных в сети Интернет. Однако удобство использования, теоретическая, программная и методическая поддержка, а также усилия по продвижению способствовали росту популярности этой технологии среди разработчиков широкого класса интеллектуальных приложений.

В статье [104] дается обзор современных средств формализации семантики областей знаний и манипулирования знаниями, предоставляемых технологией Semantic Web.

Наиболее известным средством описания онтологий является язык OWL (п. 1.4.3). Этот язык разработан на основе других популярных языков: RDF (Resource Description Framework), позволяющего описывать ресурсы утверждениями вида «субъект-предикат-объект», и RDFS (RDF Schema), в котором имеются средства для определения словаря терминов и отношений между терминами, а также их семантики [121].

Благодаря тому, что язык OWL основан на RDF, OWL-онтологии можно хранить в RDF-хранилищах, в которые, как правило, встраиваются машины вывода (Pellet, FaCT++, HermiT и др.).

Для поиска информации в онтологии используются языки DL Query и SPARQL [60]. Язык DL Query основан на Манчестерском синтаксисе, понятном

пользователям, хорошо владеющим логикой. SQL-подобный язык SPARQL поддерживает не только выполнение запросов, но и редактирование данных.

Логический вывод в OWL-онтологиях осуществляется машинами вывода на основе представленных в онтологиях аксиом, а также правил, записанных на языке SWRL (Semantic Web Rule Language) [62].

Для работы с OWL-онтологиями в программных приложениях используются различные программные библиотеки. Наиболее популярными из них являются OWL API [46] и Jena [4].

Для разработки онтологий создано множество редакторов. Наиболее популярным из них является Protégé [51]. Онтологии, построенные в Protégé, могут быть экспортированы в различные форматы, включая RDF, RDFS, OWL, Turtle, Clips и др.

Одной из проблем, ограничивающих использование средств технологии Semantic Web, является их высокий «порог входа» для пользователей, не знакомых с дескрипционными логиками.

1.5.3. Технология разработки интеллектуальных научных интернет-ресурсов

Вопросы важности информационной поддержки ЛПР для принятия ответственных решений обсуждаются в публикациях [126,164]. Разработке информационных систем на основе онтологий посвящены работы [84,113]. В работе [173] представлена одна из самых развитых технологий для создания систем такого класса – технология разработки интеллектуальных научных интернет-ресурсов (ИНИР).

ИНИР представляет собой доступную через Интернет информационную систему. Эта система предназначена для систематизации и интеграции научных знаний и информационных ресурсов определенной области знаний, а также для обеспечения содержательного эффективного доступа к ним, к методам обработки информации, используемым в данной области знаний, и к методам решения типовых для данной области задач.

ИНИР базируется на формализмах онтологий и семантических сетей. Онтология при этом составляет ядро системы знаний ИНИР, а в семантической сети, структура которой определяется онтологией, накапливается информация об основных сущностях моделируемой области знаний.

Технология разработки ИНИР предоставляет оболочку типового ресурса со встроенными в нее средствами работы с онтологией и контентом (редакторы, сервисы поиска, навигации, обработки данных), подсистему сбора информации из сети Интернет, а также методологию построения онтологий, включающую набор базовых онтологий (приложение Б, Рисунок Б.1). Таким образом, разработка тематического ИНИР в определенной ПО, фактически сводится к разработке онтологии этой области.

Предоставляемые данной технологией средства реализованы с использованием технологий Semantic Web.

1.6. Средства разработки web-сервисов

Программные системы, разработанные на основе сервис-ориентированного подхода, обычно реализуются как набор веб-сервисов (п. 1.3.6). Наиболее популярными средствами их разработки являются архитектурный стиль REST, технологии RPC и тройка стандартов SOAP/WSDL/UDDI.

1.6.1. Архитектурный стиль REST

REST (сокращение от англ. Representational State Transfer — «передача состояния представления») — архитектурный стиль взаимодействия компонентов распределённого приложения в сети, впервые описанный в работе [21]. Приложения, спроектированные в соответствии с этим стилем, должны удовлетворять следующим условиям-ограничениям: 1) модель клиент-сервер; 2) отсутствие состояния; 3) кэширование ответов сервера; 4) единообразие интерфейса, подразумевающее идентификацию ресурсов (например, URI), манипуляцию ресурсами через представление, «самоописываемые» сообщения,

изменения состояния приложения посредством гипермедиа; 5) иерархическая структура вычислительной сети; 6) возможность загрузки кода с сервера в виде апплетов или сценариев.

REST-приложения обладают следующими преимуществами [19,21]: надёжность, производительность, масштабируемость, прозрачность системы взаимодействия, простота интерфейсов, портативность компонентов, лёгкость внесения изменений, способность эволюционировать.

1.6.2. Удаленный вызов процедур (RPC)

Удалённый вызов процедур (Remote Procedure Call, RPC) – подход, позволяющий компьютерным программам вызывать функции (процедуры), расположенные на удаленных компьютерах. Реализации данного подхода включают спецификацию сетевого протокола для обмена в режиме клиент-сервер и язык сериализации объектов / структур. Различные реализации RPC имеют очень отличающуюся друг от друга архитектуру и разнятся в своих возможностях: одни реализуют архитектуру SOA, другие — CORBA или DCOM. На транспортном уровне RPC используют в основном протоколы TCP и UDP, однако, некоторые построены на основе HTTP [8].

RPC используется совместно с разными протоколами передачи данных, которые позволяют их сериализовать при передаче и десериализовать при получении. Под сериализацией понимается перевод состояния объекта в последовательность битов (чаще всего, бинарный или XML-файл), после чего его копия может быть передана в другой процесс. Обратный процесс – десериализация – это восстановление состояния объекта из принятой последовательности битов.

1.6.3. SOAP/WSDL/UDDL

К основным стандартам разработки и функционирования веб-сервисов можно отнести:

- SOAP – протокол взаимодействия веб-сервисов, основанный на XML;

- WSDL (Web Services Description Language – язык описания веб-сервисов) – методология описания ресурсов, предоставляемых веб-сервисом;
- UDDI (Universal Description Discovery and Integration – универсальный метод поиска и интеграции) – метод описания, поиска, взаимодействия и использования веб-сервисов.

Протокол SOAP предназначен для организации взаимодействия удаленных систем при помощи асинхронного обмена XML-отформатированными документами, состоящими из трех частей: конверта (обертки), заголовка и тела. SOAP формирует базовый слой стека протоколов веб-сервисов, обеспечивая инфраструктуру обмена сообщениями между ними [34].

Язык WSDL описывает сервисы в виде неких абстрактных ресурсов, способных принимать на вход документы определенных типов и инициировать отправку документов других типов. WSDL используется для описания веб-сервисов и для определения их расположения [13].

WSDL определяет сервис с двух точек зрения: абстрактной и конкретной. На абстрактном уровне сервис задается в терминах посылаемых и принимаемых им сообщений, которые описываются средствами XML Schema в виде, независимом от конкретного транспортного протокола. На конкретном уровне определяются привязки к транспортным форматам и точкам физического размещения.

Протокол UDDI представляет собой стандарт на внутреннее устройство и внешние интерфейсы базы данных, хранящей описания сервисов. Все описания в БД хранятся в виде XML-записей. UDDI обеспечивает репликацию баз данных со сложными моделями их подчиненности друг другу, построение репозитория из нескольких БД (и репликацию данных между ними), глобальную уникальность записей и ключей, API публикации описаний и подписки на изменения, средства обеспечения целостности данных, интернационализации записей, шифрования содержимого.

1.6.4. Протоколы передачи данных

XML (eXtensible Markup Language) — расширяемый язык разметки, рекомендован консорциумом W3C. [42]. Этот язык ориентирован на использование в Интернете и имеет простой формальный синтаксис, удобный как для создания и обработки документов программами, так и чтения и создания документов человеком. XML не фиксирует используемую в документах разметку, и пользователи могут в рамках синтаксических правил языка создавать собственную. Расширение XML — это конкретная грамматика, которая создается на базе XML и представляется словарём тегов, атрибутов тегов, и набором правил, определяющих структуру элементов языка. Сочетание простого формального синтаксиса, удобства для человека, расширяемости, а также базирование на кодировках Юникод для представления содержания документов привело к широкому использованию как собственно XML, так и множества производных специализированных языков на базе XML в самых разнообразных программных средствах.

JSON (JavaScript Object Notation) — текстовый формат обмена данными, основанный на JavaScript. Как и многие другие текстовые форматы, JSON легко читается людьми. Формат JSON был разработан Дугласом Крокфордом [15].

Несмотря на происхождение от JavaScript, формат считается независимым от языка и может использоваться практически с любым языком программирования. Для многих языков существует готовый код для создания и обработки данных в формате JSON.

За счёт своей лаконичности по сравнению с XML, формат JSON может быть более подходящим для сериализации сложных структур. Применяется в web-приложениях как для обмена данными между браузером и сервером (AJAX), так и между серверами (программные HTTP-сопряжения).

Поскольку формат JSON является подмножеством синтаксиса языка JavaScript, то он может быть быстро десериализован встроенной функцией `eval()`.

JSON-текст представляет собой (в закодированном виде) одну из двух структур: 1) Набор пар ключ: значение. Ключом может быть только строка,

значением — любая форма. 2) Упорядоченный набор значений, в качестве которых могут быть использованы записи, одномерные массивы, числа, литералы true, false и null, строки.

Структуры данных, используемые JSON, поддерживаются любым современным языком программирования, что и позволяет применять JSON для обмена данными между различными языками программирования и программными системами.

Protocol Buffers — протокол сериализации и передачи структурированных данных, предложенный Google [52]. Данный протокол, в отличие от текстового формата XML является бинарным. Он оптимизирован под минимальный размер сообщения, что делает его проще, компактнее и быстрее XML.

По заявлению компании Google, Protocol Buffers по сравнению с XML: проще; от 3 до 10 раз компактнее; от 20 до 100 раз быстрее; более однозначный; позволяет создавать классы, которые в дальнейшем легче использовать программно.

Для сериализации данных в формат Protocol Buffers необходимо, прежде всего, определить их структуру. Для этого необходимо создать proto-файл с исходным кодом этой структуры в объектно-ориентированном стиле, который легко создавать и читать человеком. После того как файл с нужной структурой данных создан, нужно скомпилировать его компилятором для того языка программирования, на котором ведется прикладная разработка, в результате чего будет сгенерирован класс доступа к этим данным. Класс будет содержать простейшие методы доступа ко всем полям типа get/set, а также методы для сериализации и десериализации структуры данных в бинарный формат и обратно.

На данный момент компанией Google, а также сторонними разработчиками созданы компиляторы практически для всех более или менее используемых языков программирования.

1.7. Выводы

Проведенный обзор литературы позволяет сделать следующие выводы:

- Основная часть областей практической деятельности современного человека является слабоформализованной. Разработка ИСППР для СФПО является актуальной и сложной задачей.
- Нет готовых доступных универсальных инструментов разработки ИСППР. Существующие инструментарины либо недоступны из-за высокой стоимости, либо не предназначены для СФПО, либо решают ограниченный класс задач. Следовательно, необходимо разработать методы и средства, которые бы оказывали всестороннюю поддержку разработки ИСППР от проектирования до непосредственной реализации.
- И в российских, и в зарубежных публикациях сформулированы принципы «идеальных» СППР, описаны подходы к их разработке, технологии, которые могут использоваться для создания отдельных подсистем СППР.
- Разработано много методов поддержки принятия решений. Для многих из них есть программные реализации и/или подробные методики использования.
- Несмотря на наличие разработанных методов принятия решений до сих пор не построена достаточно полная их классификация, нет формализованного описания этих методов, не обеспечен доступ к готовым реализациям и к инструкциям по их применению и встраиванию.

Таким образом, анализ, проведенный в первой главе, подтверждает актуальность диссертационной работы, обосновывает её цели и задачи, представленные во введении.

Глава 2. Модель комплексной поддержки разработки интеллектуальных СППР

Проведенный аналитический обзор показал, что в свободном доступе универсальных инструментов для разработки ИСППР нет, а существующие решения, использующие разные принципы и подходы, обладают своими достоинствами и недостатками. Данная глава посвящена описанию и формализации предлагаемой модели комплексной поддержки разработки (КПР) ИСППР [175]. Основные элементы модели КПР представлены на Рисунке 2.1.

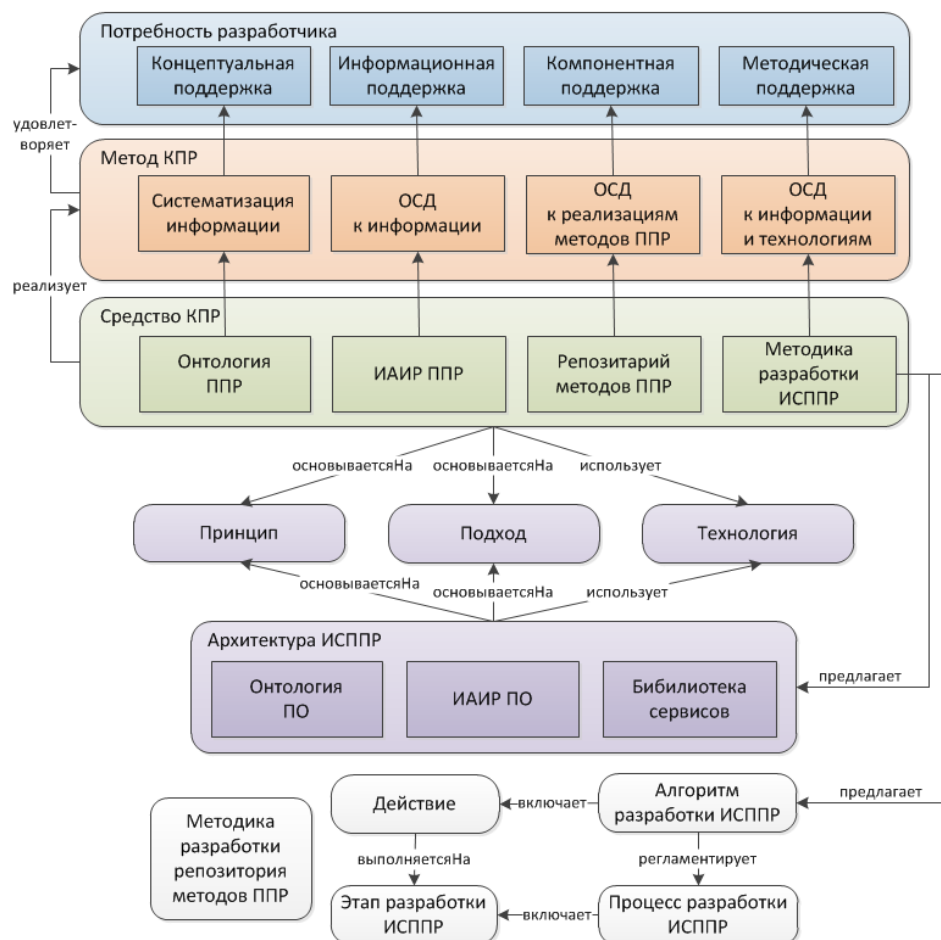


Рисунок 2.1. Основные элементы модели КПР.

(здесь: ОСД – обеспечение содержательного доступа; ИАИР – информационно-аналитический интернет-ресурс)

2.1. Формальная система

Для описания модели КПР рассмотрим формальную систему $FM = \langle C, R, A \rangle$, где C – множество атомарных концептов КПР, R – множество атомарных ролей, описывающих свойства концептов и отношения между ними, A – множество аксиом, устанавливающих связь между концептами и ролями. Аксиомы могут содержать описания концептов, составленные из атомарных концептов и ограничений ролей с использованием конструкторов определенной логики. Рассматриваемая система FM задается в рамках дескрипционной логики $SOIN(D)$ [6]. Синтаксис и семантика конструкторов этой логики представлены в Таблице 1 приложения А. Для описания элементов FM были введены дополнительные символы $(,), =, \{, \}, [,]$, семантика которых, при необходимости, поясняется по мере введения использующих их выражений. Использование дополнительных символов не выходит за рамки логики $SOIN(D)$, а просто позволяет описывать элементы модели КПР более компактно.

Множества C и R включают следующие основные концепты и роли (Рисунок 2.1):

$C = \{ \text{Разработчик_ИСППР}, \text{Потребность_разработчика_ИСППР}, \text{Метод_КПР},$
 $\text{Средство_КПР}, \text{Методика_разработки_ИСППР}, \text{Принцип_разработки_ИСППР},$
 $\text{Подход_к_разработке_ИСППР}, \text{Технология_для_разработки_ИСППР},$
 $\text{Методика_разработки_репозитория_методов_ППР},$
 $\text{Архитектура_ИСППР}, \text{Алгоритм_разработки_ИСППР}, \text{Действие},$
 $\text{Процесс_разработки_ИСППР}, \text{Этап_разработки_ИСППР}$
 $\}$

$R = \{ \text{удовлетворяет}, \text{реализует}, \text{заимствует}, \text{использует}, \text{основываетсяНа},$
 $\text{предлагает}, \text{включает} \}$

Множество A включает аксиомы, задающие отношение «общее-частное» между концептами, определяющие области определения левых и правых аргументов бинарных отношений, соответствующих ролям, описывающие свойства некоторых концептов и ролей. Аксиомы содержат составные концепты и

ограничения ролей, которые определяются в соответствии с конструкторами логики SOIN(D). Типы аксиом дескрипционных логик представлены в Таблице 2 приложения А.

Рассмотрим в качестве примера аксиому, которая описывает свойства концепта «Средство_КПР»:

Средство_КПР $\sqsubseteq \exists$ основываетсяНа . [Принцип_разработки_ИСППР $\sqcap$
 Подход_к_разработке_ИСППР] $\sqcap$
 $\exists$ использует . Технология_для_разработки_ИСППР

Интерпретация I формальной системы, задающая ее семантику, представляется парой $I = \langle \Delta, f \rangle$, где Δ – домен, множество всех экземпляров (индивидов) модели КПР, f – функция интерпретации
 $f: C \cup R \rightarrow (2^\Delta) \cup (2^\Delta \times 2^\Delta)$.

$f(C_i) \subseteq \Delta$, $C_i \in C$ – каждому концепту сопоставляется множество его экземпляров (индивидов).

$f(R_i) \subseteq \Delta \times \Delta$, $R_i \in R$ – каждой роли сопоставляется бинарное отношение, т.е. множество пар связываемых им индивидов.

Функция f расширяется для интерпретации составных концептов в соответствии с индуктивными правилами логики SOIN(D).

Тогда интерпретация формальной системы FM будет задавать **модель КПР** как множество конкретных объектов, принадлежащих определенным классам, связанных определенными отношениями и удовлетворяющих аксиомам А.

Отметим, что термин «модель» можно рассматривать не только в логическом или алгебраическом смысле, но и как упрощенное информационное представление совокупности объектов и процессов реального мира [Горшков, 2016], описывающих процесс КПР.

При описании концептов, ролей, аксиом и их интерпретаций используются следующие соглашения по именованию [6]:

1) Имена концептов пишутся с прописной буквы, за которой следуют строчные. Если название состоит из нескольких слов, они соединяются символом подчеркика (Концепция_КПР, Потребность_разработчика_ИСППР).

2) Названия ролей пишутся строчными буквами. Если название состоит из нескольких слов, они «приклеиваются» друг к другу, и каждое слово, начиная со второго, пишется с прописной буквы (основываетсяНа, выполняетсяНаЭтапе).

3) Имена объектов полностью состоят из прописных букв. Если имя состоит из нескольких слов, они соединяются символом подчеркика (ИНЖЕНЕР_ЗНАНИЙ, КОНЦЕПТУАЛЬНАЯ_ПОДДЕРЖКА).

Рассмотрим более подробно основные понятия, роли и аксиомы модели КПР и их интерпретации.

2.2. Процесс комплексной поддержки разработки интеллектуальных СППР

Модель КПР рассматривает комплексную поддержку как процесс удовлетворения потребностей разработчиков ИСППР, протекающий параллельно с процессом разработки ИСППР. Она предлагает методы удовлетворения потребностей и средства реализации этих методов.

В разработке ИСППР участвуют специалисты разного профиля – эксперты той предметной области (ПО), для которой создается ИСППР, инженеры знаний, являющиеся специалистами в области представления знаний и поддержки принятия решений, и программисты.

$f(\text{Разработчик_ИСППР}) = \{ \text{ИНЖЕНЕР_ЗНАНИЙ, ЭКСПЕРТ, ПРОГРАММИСТ} \}$

Все они говорят на своем профессиональном языке, используют свою терминологию. Для их успешной совместной работы, в особенности на начальных этапах разработки, необходима единая система понятий, *общий концептуальный базис*.

Помимо общих представлений об области знаний разработчикам необходима *информация о конкретных методах* поддержки принятия решений (ППР), о классах задач, решаемых этими методами, о возможностях и ограничениях каждого из них. Кроме этого, разработчикам должна быть доступна информация об основных этапах принятия решений и используемых на каждом из них методах, а также об инструментальных средствах, реализующих эти методы.

На этапе реализации ИСППР важную роль играет *компонентная поддержка* разработчиков. Возможность выбрать и опробовать готовые программные компоненты, реализующие необходимые методы поддержки принятия решений или организующие интерфейс с пользователями, может существенно облегчить и ускорить процесс создания ИСППР.

Еще одной важной потребностью разработчиков на этапе реализации является *методическая поддержка*, включающая описание подходов и принципов разработки ИСППР, средства разработки и методику их применения.

$f(\text{Потребность_разработчика}) = \{$
 КОНЦЕПТУАЛЬНАЯ_ПОДДЕРЖКА, ИНФОРМАЦИОННАЯ_ПОДДЕРЖКА,
 КОМПОНЕНТНАЯ_ПОДДЕРЖКА, МЕТОДИЧЕСКАЯ_ПОДДЕРЖКА
 $\}$

Согласно предлагаемой концепции концептуальный базис КПР обеспечивается путем *систематизации информации об области знаний* «Поддержка принятия решений», результатом которой является *онтология* этой области. Информационная поддержка осуществляется *путем обеспечения содержательного доступа (ОСД) к структурированным на основе онтологии знаниям, информационным ресурсам и методам*, относящимся к области знаний «Поддержка принятия решений». Средством такой поддержки является специализированный *информационно-аналитический интернет-ресурс (ИАИР)* по ППР. Компонентная поддержка процесса разработки ИСППР осуществляется путем *обеспечения непосредственного содержательного доступа к реализациям методов* ППР и обеспечивается *репозиторием* методов ППР. Методическая

поддержка обеспечивается *предоставлением информации о подходах, принципах, технологиях, алгоритме разработки ИСППР*, а также об их архитектуре.

Таким образом, основными методами КПР являются способы работы с информацией о методах и аспектах ППР – их систематизация и обеспечение содержательного доступа к ним, а также ОСД к реализациям методов ППР и методике разработки ИСППР.

$f(\text{Метод_КПР}) = \{$
 СИСТЕМАТИЗАЦИЯ_ИНФОРМАЦИИ,
 ОСД_К_ИНФОРМАЦИИ_О_МЕТОДАХ_ППР, ОСД_К_РЕАЛИЗАЦИЯМ_МЕТОДОВ_ППР,
 ОСД_К_ИНФОРМАЦИИ_О_ПОДХОДАХ_ПРИНЦИПАХ_ТЕХНОЛОГИЯХ
 $\}$

Средствами КПР являются онтология области знаний (ОЗ) ППР, информационно-аналитический интернет-ресурс (ИАИР) по ППР, репозиторий методов ППР и методика разработки ИСППР.

$f(\text{Средство_КПР}) = \{$
 ОНТОЛОГИЯ_ОЗ_ППР, ИАИР_ППР, РЕПОЗИТОРИЙ_МЕТОДОВ,
 МЕТОДИКА_РАЗРАБОТКИ_ИСППР
 $\}$

Методы КПР удовлетворяют потребности разработчиков ИСППР.

$f(\text{удовлетворяет}) =$
 $\{ (\text{СИСТЕМАТИЗАЦИЯ_ИНФОРМАЦИИ, КОНЦЕПТУАЛЬНАЯ_ПОДДЕРЖКА}),$
 $(\text{ОСД_К_ИНФОРМАЦИИ_О_МЕТОДАХ_ППР,}$
 $\text{ИНФОРМАЦИОННАЯ_ПОДДЕРЖКА}),$
 $(\text{ОСД_К_РЕАЛИЗАЦИЯМ_МЕТОДОВ_ППР, КОМПОНЕНТНАЯ_ПОДДЕРЖКА}),$
 $(\text{ОСД_К_ИНФОРМАЦИИ_О_ПОДХОДАХ_ПРИНЦИПАХ_ТЕХНОЛОГИЯХ,}$
 $\text{МЕТОДИЧЕСКАЯ_ПОДДЕРЖКА})$
 $\}$

Средства КПР реализуют методы КПР.

$f(\text{реализует}) =$
 $\{ (\text{ОНТОЛОГИЯ_ОЗ_ППР, СИСТЕМАТИЗАЦИЯ_ИНФОРМАЦИИ}),$

```
( ИАИР_ППР, ОСД_К_ИНФОРМАЦИИ_О_МЕТОДАХ ),
( РЕПОЗИТОРИЙ_МЕТОДОВ, ОСД_К_РЕАЛИЗАЦИЯМ_МЕТОДОВ ),
( МЕТОДИКА_РАЗРАБОТКИ_ИСППР,
  ОСД_К_ИНФОРМАЦИИ_О_ПОДХОДАХ_ПРИНЦИПАХ_ТЕХНОЛОГИЯХ )
}
```

Средства комплексной поддержки (Рисунок 2.1), являющиеся индивидами в модели КПР, – это сложные объекты. Для них построены собственные подмодели, которые будут рассмотрены в п. 2.4-2.6.

2.3. Подходы, принципы, технологии, предлагаемые для разработки интеллектуальных СППР

В Главе 1 были рассмотрены принципы, на которых строятся современные ИСППР, подходы и технологии, используемые при их разработке. Рассматриваемая модель КПР включает ряд наиболее важных, хорошо зарекомендовавших себя на практике принципов, а также подходы и технологии, которые обеспечивают соблюдение этих принципов, упрощают и ускоряют разработку ИСППР, определяют свойства и функциональные возможности тех ИСППР, которые будут разрабатываться в рамках данной модели (Рисунки 2.2, 2.4).

Заметим, что предлагаемые средства КПР можно рассматривать как ИСППР по разработке ИСППР. Поэтому они разрабатываются с применением тех же принципов, подходов и технологий, которые будут использоваться и для конкретных ИСППР.

2.3.1. Принципы разработки интеллектуальных СППР

В настоящее время в научных и практических целях разработано большое количество методов принятия решений и реализующих их программных продуктов – библиотек, пакетов, приложений. И сейчас во многих областях знаний складывается такая ситуация, когда трудно придумать что-то новое, а то,

что кажется таковым, либо хорошо забытое или не известное старое, либо комбинация или вариация уже известных разработок. Поэтому при обеспечении комплексной поддержки очень важно предоставить доступ к имеющимся реализациям и при разработке ИСППР обеспечить *максимальное использование готовых решений*. Вместе с тем, жизнь не стоит на месте. Появляются новые методы, модифицируются, улучшаются старые. Еще одной важной задачей является *обеспечение масштабируемости и актуальности* предлагаемых средств и конкретных ИСППР. Соблюдение этих принципов позволит легко подключать новые методы ППР и интерфейсные компоненты, модифицировать имеющиеся, решая проблему устаревания разрабатываемых систем.

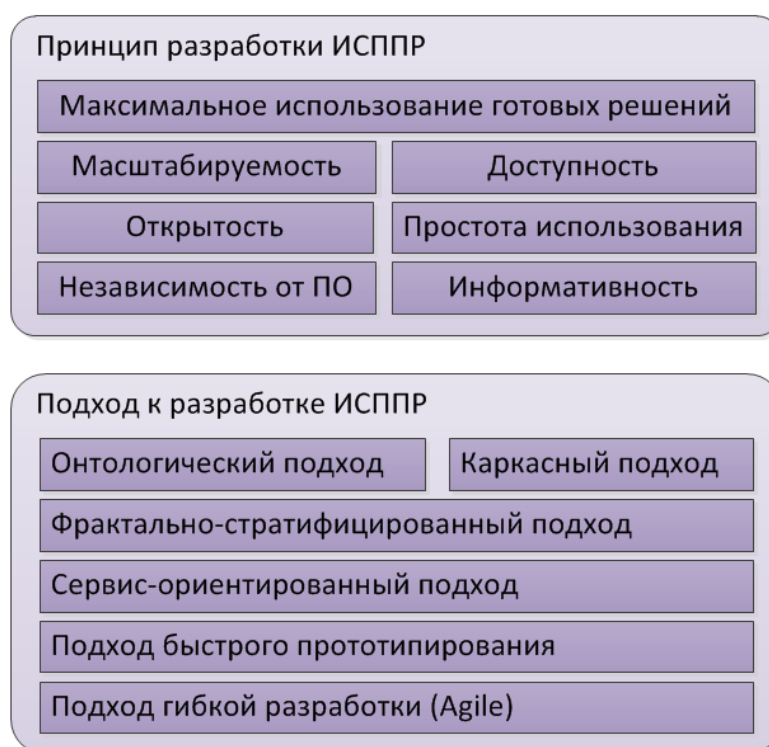


Рисунок 2.2. Принципы, подходы к разработке ИСППР

Используемые средства и разрабатываемые системы должны быть *открыты и доступны* для применения всем желающим. Только в таком случае будет обеспечено их широкое использование. Несмотря на возможность выбора предлагаемых средств разработки ИСППР и решения конкретных задач, невозможно полностью исключить необходимость их модификации. Принцип открытости позволит сделать ее безболезненно.

Для снижения квалификационных требований к разработчикам и пользователям ИСППР предлагаемые средства должны быть *просты в использовании* и, по возможности, *независимы от предметной области*.

Несмотря на то, что многие продукты находятся в свободном доступе, разработчики ИСППР вместо того, чтобы воспользоваться ими целиком или хотя бы частично, зачастую «изобретают велосипед». Основными причинами такого положения являются отсутствие общего систематизированного описания программных продуктов в данной области, неполнота авторских спецификаций, устаревание платформ, на которых реализовывались эти продукты. Поэтому важным принципом и обеспечения КПР, и создания ИСППР, является принцип *информативности*, т.е. обеспечение содержательного доступа к информации обо всех аспектах разработки и функционирования ИСППР.

$$f(\text{Принцип_разработки_ИСППР}) = \{$$

МАКСИМАЛЬНОЕ_ИСПОЛЬЗОВАНИЕ_ГОТОВЫХ_РЕШЕНИЙ,
 МАСШТАБИРУЕМОСТЬ, ДОСТУПНОСТЬ, ОТКРЫТОСТЬ,
 ПРОСТОТА_ИСПОЛЬЗОВАНИЯ, НЕЗАВИСИМОСТЬ_ОТ_ПО,
 ИНФОРМАТИВНОСТЬ

$$\}$$

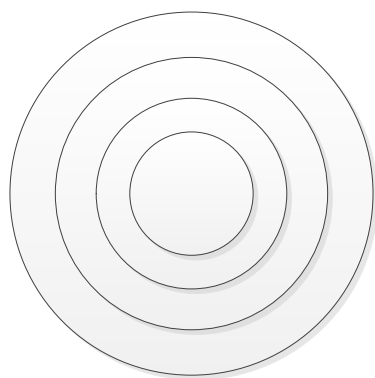
2.3.2. Подходы к разработке интеллектуальных СППР

Одним из основных элементов интеллектуальных систем являются базы знаний (БЗ), роль которых в настоящее время играют онтологии. При разработке методов и средств КПР *онтологический подход* также был использован для разработки баз знаний (БЗ). Онтология ППР является концептуальной основой КПР. Согласно предлагаемой модели, ИСППР также должны разрабатываться на основе онтологии.

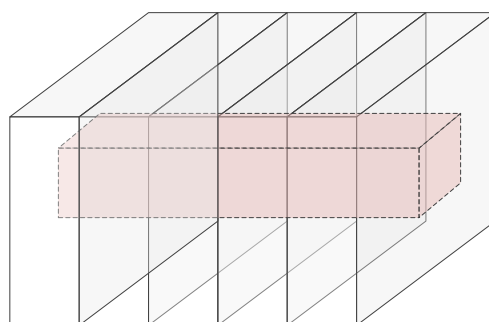
При разработке БЗ, помимо онтологического, предлагается использовать и *фрактально-стратифицированный подход* (ФС-подход), который базируется на фрактальном подходе, предложенном Л.В. Массель [135] (п. 1.3.4). Данный подход позволяет учесть подобие системы знаний в целом и ее фрагментов.

ФС-подход был дополнен автором возможностью использования в ФС-моделях разных видов расслоения (стратификации). На Рисунке 2.3 показаны виды возможной стратификации пространства знаний.

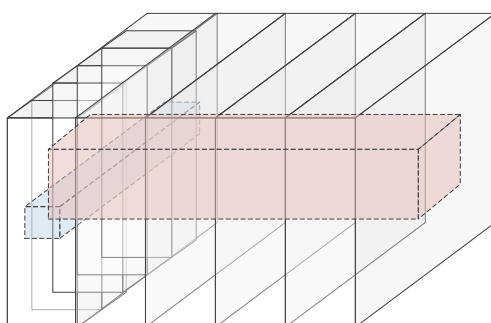
Сферическое расслоение (а), предложенное автором подхода, удобно использовать для представления разных уровней детализации знаний; линейное расслоение (брусочное, параллелепипедное) (б) представляет невложенные фрагменты знаний. Брус, «пронизывающий» расслоенный параллелепипед знаний, это инвариант, т.е. фрагмент знаний, присутствующий во всех стратах. Страты в таком расслоении могут представлять, например, разные аспекты понятий, используемых разными специалистами.



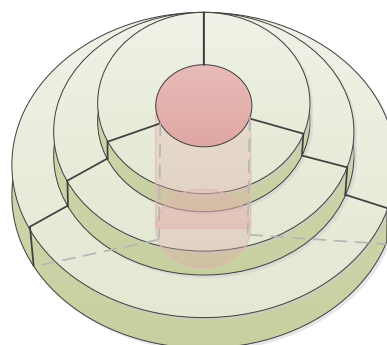
(а) Сферическое расслоение



(б) Линейное расслоение



(в) Линейно – иерархическое расслоение



(г) Пирамидальное расслоение

Рисунок 2.3. Виды стратификации пространства знаний.

Комбинированное расслоение – линейно-иерархическое (в), пирамидальное (г), удобно использовать для представления невложенных фрагментов знаний, имеющих собственную иерархическую структуру. В комбинированных

стратифицированных моделях инвариант – это метаонтология, содержащая наиболее общие понятия пространства знаний, которые конкретизируются как на разных уровнях детализации, так и в разных стратах.

Использование ФС-подхода регулирует и упрощает систематизацию знаний, позволяет многократно использовать ранее разработанные фрагменты знаний.

При разработке прикладных программных систем хорошо зарекомендовал себя *сервис-ориентированный подход* [39]. Все функциональные возможности ИСППР, методы, которые будут использоваться для решения поставленных задач, а также пользовательский интерфейс предлагается реализовывать в виде сервисов. Использование данного подхода создает предпосылки для выполнимости основных принципов КПР, упомянутых выше.

Подходы *быстрого прототипирования* [48], *гибкой разработки* [27] и *каркасный* подход [160] являются универсальными при разработке широкого класса программных систем и хорошо согласуются с принципами и другими подходами, включенными в модель КПР. Их достоинства, о которых говорилось в Главе 1, позволяют с успехом применять их и для разработки ИСППР.

Таким образом, для разработки ИСППР предлагается использовать следующие подходы:

$$f(\text{Подход_к_разработке_ИСППР}) = \{$$

ОНТОЛОГИЧЕСКИЙ_ПОДХОД,
 ФРАКТАЛЬНО_СТРАТИФИЦИРОВАННЫЙ_ПОДХОД,
 СЕРВИС-ОРИЕНТИРОВАННЫЙ_ПОДХОД,
 ПОДХОД_БЫСТРОГО_ПРОТОТИПИРОВАНИЯ,
 ПОДХОД_ ГИБКОЙ_РАЗРАБОТКИ, КАРКАСНЫЙ_ПОДХОД

$$\}$$

Используемые подходы обеспечивают соблюдение упомянутых выше принципов.

Подход_к_разработке_ИСППР $\sqsubseteq$

$\exists$ обеспечивает . Принцип_разработки_ИСППР

2.3.3. Технологии для разработки ИСППР

Модель КПР включает две технологии, используемые для разработки ИСППР (Рисунок 2.4):

$$f(\text{Технология_для_разработки_ИСППР}) = \{ \\ \text{ТЕХНОЛОГИЯ_SEMANTIC_WEB, ТЕХНОЛОГИЯ_РАЗРАБОТКИ_ИАИР} \}$$

Эти технологии содержат набор средств, достаточный для того, чтобы процесс разработки был простым, понятным, удовлетворяющим рассмотренным выше принципам [36,173]. На Рисунке 2.4 представлены их основные компоненты.

Технология_для_разработки_ИСППР $\sqsubseteq$

Э включает . [Средство_представления_знаний $\sqcup$ Средство_разработки_БЗ $\sqcup$ Хранилище_информации $\sqcup$ Сервис $\sqcup$ Машина_вывода $\sqcup$ Язык_запросов $\sqcup$ Оболочка_ИСППР]

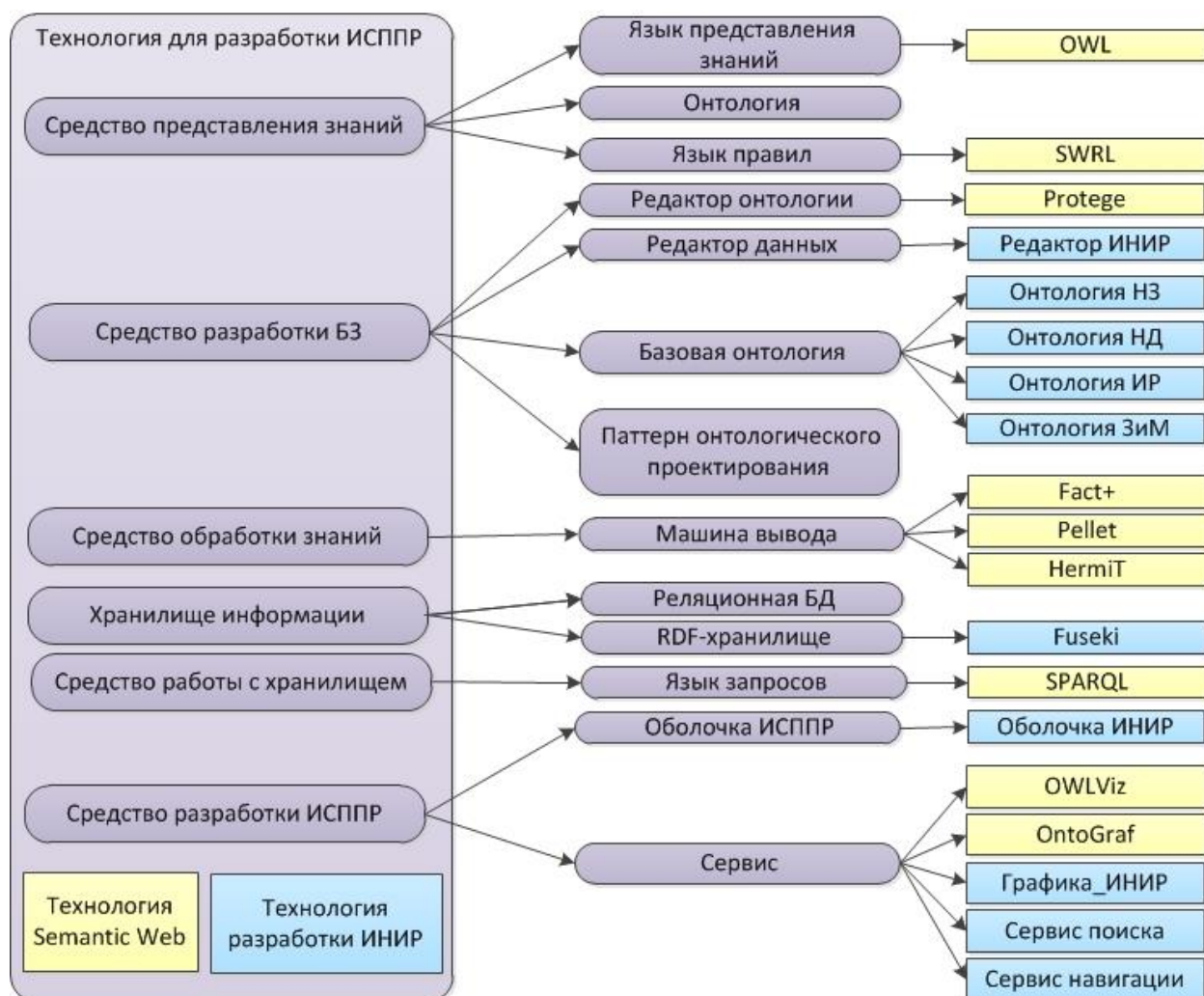


Рисунок 2.4. Технологии, используемые для разработки ИСППР

В современных интеллектуальных системах знания, как правило, представлены *онтологией* и *правилами вывода*, позволяющими получать новые знания, не содержащиеся в явном виде в БЗ, и описываются соответствующими *языками*.

Для разработки базы знаний удобно использовать *редакторы онтологий* и *редакторы данных*. Такие редакторы позволяют снизить квалификационные требования к разработчикам и привлечь к созданию БЗ экспертов предметных областей. Упростить разработку онтологий также может использование *базовых онтологий* и *паттернов онтологического проектирования*.

Для хранения информации могут использоваться различные типы *хранилищ*.

Функциональные возможности той или иной технологии определяются набором предоставляемых ею *сервисов*. Это могут быть сервисы *пользовательского интерфейса*, предназначенные для получения данных от пользователя и вывода результатов работы ИСППР; *аналитические сервисы*, позволяющие осуществлять поиск необходимой информации и навигацию по информационному наполнению ИСППР; *графические сервисы*, предназначенные для представления информации пользователю в графическом виде; *сервисы, реализующие* те или иные *методы ППР* и позволяющие решать поставленные перед ИСППР задачи.

Для организации логического вывода в онтологии на основе аксиом и правил, а также проверки онтологии на непротиворечивость, используются *машины вывода*.

Для поиска информации в хранилище используется *язык запросов*.

Технология разработки ИСППР предоставляет *оболочку* будущей ИСППР.

Следующие аксиомы задают иерархию понятий модели КПР, описывающих используемые технологии:

Средство_представления_знаний $\supseteq$ Язык_представления_знаний $\sqsubset$

Онтология $\sqsubset$ Язык_правил

Средство_разработки_БЗ $\supseteq$ Редактор_онтологии $\sqcup$ Редактор_данных $\sqcup$
 Базовая_онтология $\sqcup$ Паттерн_онтологического_проектирования

Хранилище_информации $\supseteq$ RDF_хранилище $\sqcup$ Реляционная_БД

Сервис $\supseteq$ Сервис_пользовательского_интерфейса $\sqcup$ Аналитический_сервис $\sqcup$
 Графический_сервис $\sqcup$ Сервис_реализующий_метод_ППР

Аналитический_сервис $\supseteq$ Сервис_поиска $\sqcup$ Сервис_навигации

Конкретные элементы технологии Semantic Web выделены на Рисунке 2.4 желтым цветом, элементы технологии разработки ИАИР – голубым.

$f(\text{включает}) = \{$
 (ТЕХНОЛОГИЯ_SEMANTIC_WEB,
 (OWL, PROTÉGÉ, ONTOGRAF, OWLVIZ, FACT+, PELLET, HERMIT, SPARQL)) }

Такая сокращенная запись означает множество всех пар отношения «включает», первым элементом которых является технология Semantic Web, а вторым – одна из компонент этой технологии, перечисленных в круглых скобках. Аналогично представлены экземпляры данного отношения для технологии разработки ИНИР:

$f(\text{включает}) = \{$
 (ТЕХНОЛОГИЯ_РАЗРАБОТКИ_ИНИР,
 (РЕДАКТОР_ИНИР, ОНТОЛОГИЯ_НАУЧНОГО_ЗНАНИЯ,
 ОНТОЛОГИЯ_НАУЧНОЙ_ДЕЯТЕЛЬНОСТИ, ОНТОЛОГИЯ_ЗАДАЧ_МЕТОДОВ,
 ОНТОЛОГИЯ_ИНФОРМАЦИОННЫХ_РЕСУРСОВ, FUSEKI, ГРАФИКА_ИАИР,
 ОБОЛОЧКА_ИНИР)) }

При этом технология разработки ИНИР использует компоненты и средства Semantic Web.

$f(\text{использует}) = \{$
 (ТЕХНОЛОГИЯ_РАЗРАБОТКИ_ИНИР, ТЕХНОЛОГИЯ_SEMANTIC_WEB) }

Содержательно конкретные компоненты данных технологий были рассмотрены в 1 главе и далее будут упомянуты при описании тех средств КПР, для разработки которых они использовались.

2.4. Онтология поддержки принятия решений

Средством концептуальной поддержки разработки ИСППР является онтология ППР [174]. На Рисунке 2.5. показаны основные компоненты онтологии, а также подходы, технологии и конкретные компоненты технологий, использовавшиеся для ее разработки.

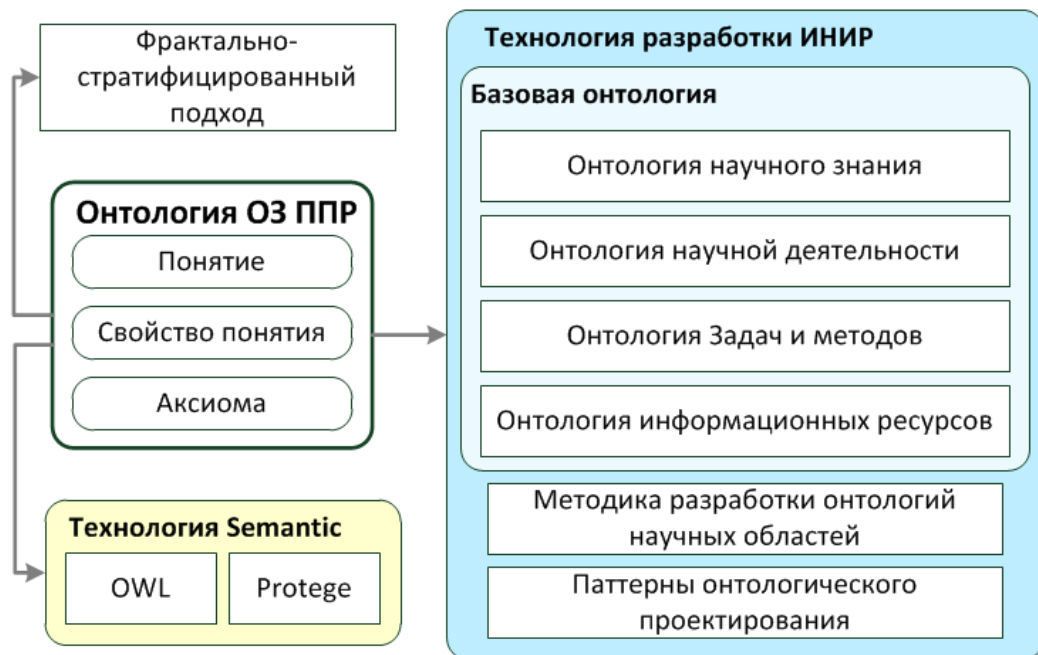


Рисунок 2.5. Модель онтологии ППР

Онтология_ППР $\sqsubseteq$ $\exists$ включает. Элемент_онтологии $\sqcap$

$\exists$ строитсяНаОснове. Базовая_онтология

Элемент_онтологии $\sqsupseteq$ Понятие $\sqcup$ Свойство_понятия $\sqcup$ Аксиома

$f(\text{используетсяДляРазработки}) =$

{ ((ФРАКТАЛЬНО_СТРАТИФИЦИРОВАННЫЙ_ПОДХОД,
МЕТОДИКА_РАЗРАБОТКИ_ОНТОЛОГИЙ_НАУЧНЫХ_ОБЛАСТЕЙ,
ОНТОЛОГИЯ_НАУЧНОГО_ЗНАНИЯ,

ОНТОЛОГИЯ_НАУЧНОЙ_ДЕЯТЕЛЬНОСТИ, ОНТОЛОГИЯ_ЗАДАЧ_МЕТОДОВ,
 ОНТОЛОГИЯ_ИНФОРМАЦИОННЫХ_РЕСУРСОВ),
 ОНТОЛОГИЯ_ОЗ_ППР)
 }

Такая сокращенная запись означает множество всех пар отношения «используетсяДляРазработки», первым элементом которых является подход, или компонент технологии, который был использован при разработке онтологии, а вторым – онтология ОЗ ППР.

$f(\text{Базовая_онтология}) = \{$
 ОНТОЛОГИЯ_НАУЧНОГО_ЗНАНИЯ, ОНТОЛОГИЯ_НАУЧНОЙ_ДЕЯТЕЛЬНОСТИ,
 ОНТОЛОГИЯ_ЗАДАЧ_И_МЕТОДОВ,
 ОНТОЛОГИЯ_ИНФОРМАЦИОННЫХ_РЕСУРСОВ }
 $\}$

Понятия и отношения базовых онтологий, предоставляемых технологией разработки ИНИР, проиллюстрированы в приложении Б, Рисунок Б.1. Конкретизация этих понятий для ОЗ ППР представлена в 3 главе.

2.5. Информационно-аналитический интернет-ресурс по поддержке принятия решений

Средством информационной поддержки разработки ИСППР является *информационно-аналитический интернет-ресурс по ППР* [184,192]. Информационная поддержка заключается в предоставлении пользователям содержательного доступа к онтологии ППР и к систематизированной в соответствии с онтологией информации о рассматриваемой ПО. Помимо оказания информационной поддержки ИАИР ППР предназначен также для обеспечения доступа к методам обработки информации и решения задач.

На Рисунке 2.6 представлена модель ИАИР. Ресурс строится на основе оболочки ИНИР, предоставляемой технологией разработки ИНИР.

$f(\text{строитсяНаОснове}) = \{ (\text{ИАИР_ППР}, \text{ОБОЛОЧКА_ИНИР}) \}$

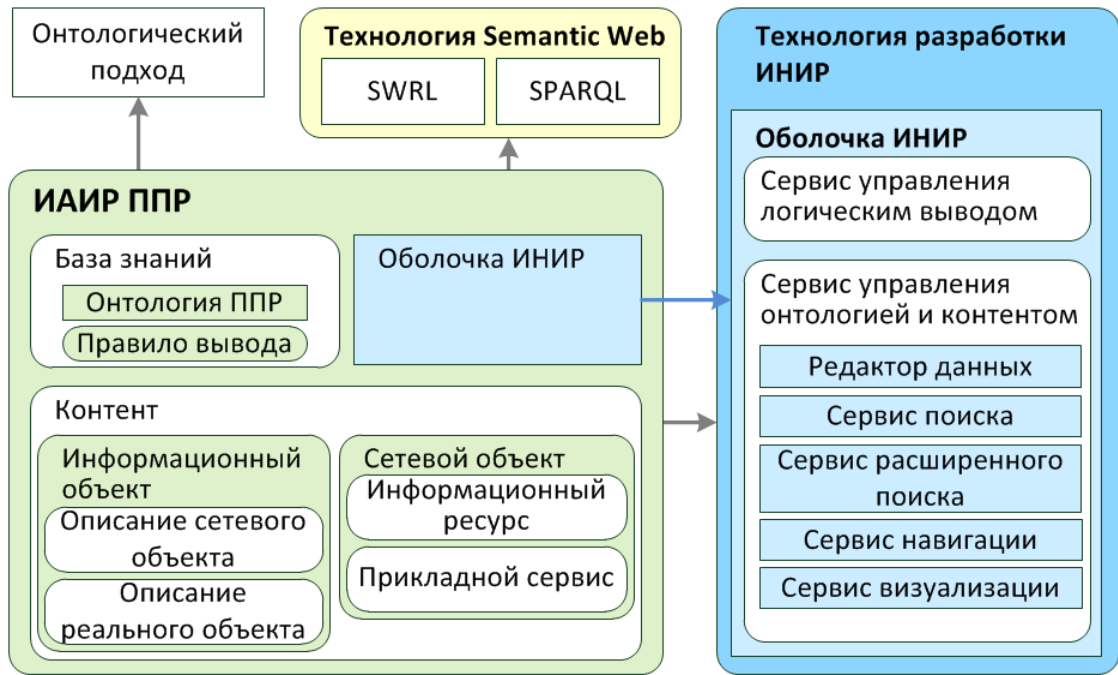


Рисунок 2.6. Модель ИАИР

Компоненты ИАИР делятся на три основных типа:

$\{ \text{ИАИР_ППР} \} \sqsubseteq \exists$ включает . Компонент_ИАИР

Компонент_ИАИР $\sqsupseteq$ База_знаний $\sqcup$ Контент $\sqcup$ Системный_сервис

База_знаний $\sqsubseteq \exists$ содержит . [Онтология $\sqcup$ Правило_вывода]

Информационное наполнение ресурса, его контент, образуют объекты двух видов – информационные и сетевые.

Контент $\sqsubseteq \exists$ содержит . [Информационный_объект $\sqcup$ Сетевой_объект]

Информационные объекты – это конкретные экземпляры понятий онтологии. Они представляют систематизированную в соответствии с онтологией информацию о рассматриваемой ПО. Информационные объекты могут описывать как реальные объекты (персона, город, организация, книга, метод исследования), так и объекты, «живущие» в сети – сетевые объекты (информационный ресурс, веб-сервис).

Информационный_объект $\sqsupseteq$ Описание_реального_объекта $\sqcup$

Описание_сетевого_объекта

Сетевые объекты могут быть как интегрированы в ИАИР, т.е. содержаться в его контенте, так и располагаться в другом месте (подобно тому, как реальные объекты, описываемые в ИАИР, не располагаются в его контенте). Каждый интегрированный в ресурс сетевой объект обязательно должен описываться соответствующим информационным объектом.

Сетевой_объект $\supseteq$ Информационный_ресурс $\sqcup$ Прикладной_сервис

Сетевой_объект $\sqsubseteq$ $\exists$ описывается . Описание_сетевого_объекта

Под информационными ресурсами понимаются произвольные ресурсы, доступные через компьютерные средства связи. Это могут быть базы данных, текстовые или мультимедиа документы, сайты и т.д. Все информационные объекты ИАИР могут быть связаны с информационными ресурсами разных типов.

Информационный_ресурс $\sqsubseteq$ $\exists$ имеетТип . Тип_информационного_ресурса

$f(\text{Тип_информационного_ресурса}) = \{$
 БАЗА_ДАННЫХ, ДОКУМЕНТ, КОЛЛЕКЦИЯ, САЙТ $\}$

Прикладные сервисы предназначены для решения задач ПО. Наличие таких сервисов в контенте является особенностью ИАИР ППР. Пользователь не просто получает информацию об интересующем его методе или ссылке на реализацию метода. Он может тут же, на ресурсе, посмотреть примеры использования метода, запустить его, проанализировать его работу с разными входными данными. ИАИР ППР предоставляет также доступ к сервисам, имеющим программные интерфейсы. Семантические описания этих сервисов существенно упрощают их использование в прикладных ИСППР.

Системные сервисы определяют базовую функциональность ИАИР ППР, характерную для подобных ресурсов.

Системный_сервис $\supseteq$ Сервис_управления_логическим_выводом $\sqcup$

Сервис_управления_онтологией_и_контентом

Сервис управления логическим выводом позволяет получать информацию, не представленную в явном виде в контенте ИАИР, за счет интерпретации аксиом и правил вывода, содержащихся в БЗ. Сервисы управления онтологией и контентом осуществляют проверку онтологии и контента на совместность, позволяют вводить и редактировать информационные и сетевые объекты, предоставляют возможность поиска, навигации по онтологии и контенту ИАИР. Сервисы визуализации позволяют отображать данные в табличном [172] или графическом виде [180]. Возможность отображения табличных зависимостей очень важна для научных областей, в которых значительная часть информации представляется именно в таком виде. Сервисы визуализации, отображая информацию в виде графиков и диаграмм, способствуют повышению ее наглядности и позволяют провести более качественный ее анализ.

$$f(\text{Сервис_управления_онтологией_и_контентом}) = \{ \text{РЕДАКТОР_ДАННЫХ}, \\ \text{СЕРВИС_ПОИСКА}, \text{СЕРВИС_РАСШИРЕННОГО_ПОИСКА}, \text{СЕРВИС_НАВИГАЦИИ}, \\ \text{СЕРВИС_ВИЗУАЛИЗАЦИИ} \}$$

Для разработки ИАИР ППР были использованы следующие подходы и технологии:

$$f(\text{используетсяДляРазработки}) = \\ \{ ((\text{ОНТОЛОГИЧЕСКИЙ_ПОДХОД}, \text{ТЕХНОЛОГИЯ_SEMANTIC_WEB}, \text{SWRL}, \\ \text{SPARQL}, \text{ТЕХНОЛОГИЯ_РАЗРАБОТКИ_ИНИР}, \text{ОБОЛОЧКА_ИНИР}), \\ \text{ИАИР_ППР}) \}$$

2.6. Репозиторий методов поддержки принятия решений

2.6.1. Модель репозитория

Средством компонентной поддержки разработчиков является *репозиторий методов ППР* [186]. Модель репозитория представлена на Рисунке 2.7.

Репозиторий включает три основных типа компонентов:

Репозиторий $\sqsubseteq \exists$ включает . Компонент_репозитория

Компонент_репозитория $\sqsupseteq$ Метод_ППР $\sqcup$ Информационный_объект $\sqcup$
Программная_система

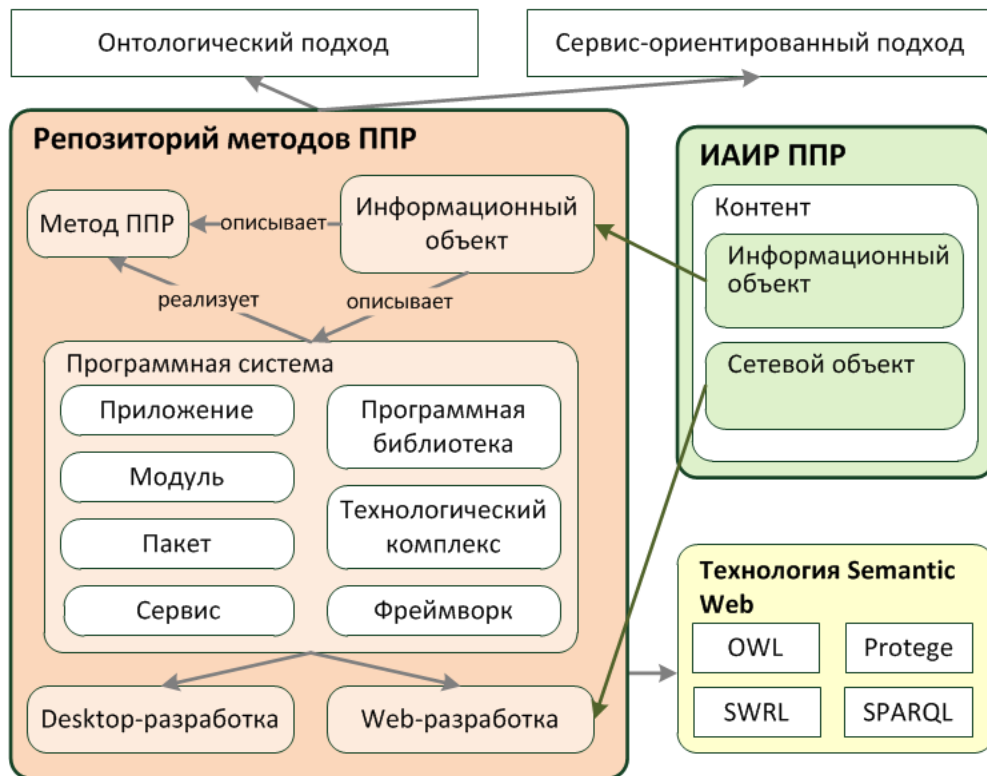


Рисунок 2.7. Модель репозитория методов ППР

Следующие аксиомы описывают подклассы и роли компонентов репозитория, а также их связи с контентом ИАИР:

Программная_система $\sqsupseteq$ Сетевой_объект $\sqcup$ Несетевой_объект

Программная_система $\sqsubseteq \exists$ имеетТип . Тип_программной_системы $\sqcap$
 $\exists$ реализует . Метод_ППР

Информационный_объект $\sqsubseteq \exists$ описывает . [Метод_ППР $\sqcup$
Программная_система]

Контент_ИАИР $\sqsubseteq \exists$ содержит . [Информационный_объект $\sqcup$
Сетевой_объект]

$f(\text{Тип_программной_системы}) = \{ \text{ДЕСКТОП_ПРИЛОЖЕНИЕ, WEB_ПРИЛОЖЕНИЕ,}$
 $\text{МОДУЛЬ, ПАКЕТ, ПРОГРАММНАЯ_БИБЛИОТЕКА, СЕРВИС,}$
 $\text{ТЕХНОЛОГИЧЕСКИЙ_КОМПЛЕКС, ФРЕЙМВОРК}$
 $\}$

В данный момент репозиторий включает представительный набор методов и реализующих их программных систем [185,188,189], который при необходимости может быть дополнен:

$$f(\text{Метод_ППР}) = \{ \text{АНКЕТИРОВАНИЕ, ВЕРОЯТНОСТНОЕ_МОДЕЛИРОВАНИЕ,} \\ \text{КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ,} \\ \text{МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ,} \\ \text{ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ,} \\ \text{РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ,} \\ \text{РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ,} \\ \text{СОБЫТИЙНОЕ_МОДЕЛИРОВАНИЕ, ЭКСПЕРТНОЕ_ОЦЕНИВАНИЕ} \\ \}$$

$$f(\text{Программная_система}) = \{ \text{СМАР_TOOLS, COLIBRI, GOOGLE_FORMS,} \\ \text{SEMP_ТАО, NETICA, НЕМО+, ИНТЕГРА, PROTEGE, CLIPS, КО_МОД,} \\ \text{GRM_ONTO_МАР, GRM_COG_МАР, GRM_EVENT_МАР, МЭО, SEMP_НЕМО,} \\ \text{UNICALC, СЕРВИС_ОБРАБОТКИ_МАССИВОВ_ДАННЫХ,} \\ \text{СЕРВИС_ВЗАИМОДЕЙСТВИЯ_С_ВНЕШНИМИ_ДАННЫМИ} \\ \}$$

$$f(\text{реализует}) = \{ \\ ((\text{СМАР_TOOLS, PROTEGE, GRM_ONTO_МАР}), \\ \text{ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ}), \\ (\text{COLIBRI, РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ}), \\ ((\text{SEMP_ТАО, CLIPS}), \text{РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ}), \\ (\text{NETICA, ВЕРОЯТНОСТНОЕ_МОДЕЛИРОВАНИЕ}), \\ ((\text{GRM_COG_МАР, КО_МОД}), \text{КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ}), \\ (\text{GRM_EVENT_МАР, СОБЫТИЙНОЕ_МОДЕЛИРОВАНИЕ}), \\ ((\text{НЕМО+, ИНТЕГРА, SEMP_НЕМО, UNICALC}), \\ \text{МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ}), \\ (\text{СИСТЕМА_ЭКСПЕРТНЫЕ_ОЦЕНКИ, ЭКСПЕРТНОЕ_ОЦЕНИВАНИЕ}), \\ (\text{GOOGLE_FORMS, АНКЕТИРОВАНИЕ}) \\ \}$$

Для разработки репозитория были использованы следующие подходы и технологии:

$$f(\text{используетсяДляРазработки}) = \{ ((\text{ОНТОЛОГИЧЕСКИЙ_ПОДХОД}, \text{СЕРВИС_ОРИЕНТИРОВАННЫЙ_ПОДХОД}, \text{ТЕХНОЛОГИЯ_SEMANTIC_WEB}), \text{РЕПОЗИТОРИЙ}) \}$$

Доступ к репозиторию, к описанию методов и реализующим их программным системам, предоставляется через ИАИР ППР:

$$f(\text{обеспечиваетДоступ}) = \{ (\text{ИАИР_ППР}, \text{РЕПОЗИТОРИЙ}) \}$$

2.6.2. Методика разработки репозитория

Разработка репозитория состоит в разработке новых методов ППР, реализующих их сервисов/программных систем и интеграции этих методов и систем, а также находящихся в свободном доступе реализованных методов с ИАИР ППР [187]. Интеграция каждого реализованного метода, в свою очередь, состоит в создании в контенте ИАИР ППР информационного объекта, описывающего реализующую его программную разработку, ее тип, связи с другими объектами, с инструкциями по ее применению, формат используемых данных, IP-адрес, по которому можно запустить сервис или скачать программную систему.

$$\begin{aligned} \{ \text{МЕТОДИКА_РАЗРАБОТКИ_РЕПОЗИТОРИЯ_МЕТОДОВ_ППР} \} &\sqsubseteq \\ &\exists \text{предлагает} . \{ \text{АЛГОРИТМ_РАЗРАБОТКИ_РЕПОЗИТОРИЯ} \} \sqcap \\ &\exists \text{основываетсяНа} . [\text{Принцип_разработки_ИСППР} \sqcup \\ &\quad \{ \text{СЕРВИС-ОРИЕНТИРОВАННЫЙ_ПОДХОД} \}] \sqcap \\ &\exists \text{использует} . \text{Технология_реализации_сервиса} \\ \{ \text{АЛГОРИТМ_РАЗРАБОТКИ_РЕПОЗИТОРИЯ} \} &\sqsubseteq \\ &\exists \text{включаетДействие} . \text{Действие} \\ \text{Действие} &\sqsubseteq \text{Разработка_метода_ППР} \sqcup \text{Разработка_сервиса} \sqcup \\ &\text{Интеграция_метода_с_ИАИР_ППР} \end{aligned}$$

2.7. Архитектура и методика разработки интеллектуальных СППР

2.7.1. Архитектура интеллектуальных СППР

Согласно предлагаемой модели основным компонентом разрабатываемой ИСППР, её каркасом, на который в виде сервисов будут «нанизаны» интерфейсные и функциональные компоненты, является тематический ИАИР выбранной ПО. Он разрабатывается по той же технологии, что и ИАИР ППР, с использованием предлагаемой ею оболочки ИНИР. База знаний строится на основе онтологии ПО. Сервисы, включаемые в архитектуру ИСППР, заимствуются из репозитория методов ППР, а описывающие их информационные объекты – из онтологии ППР и контента ИАИР ППР.

На Рисунке 2.8 показаны архитектурные компоненты типовой ИСППР (выделены жирным контуром) и схема их взаимодействия.

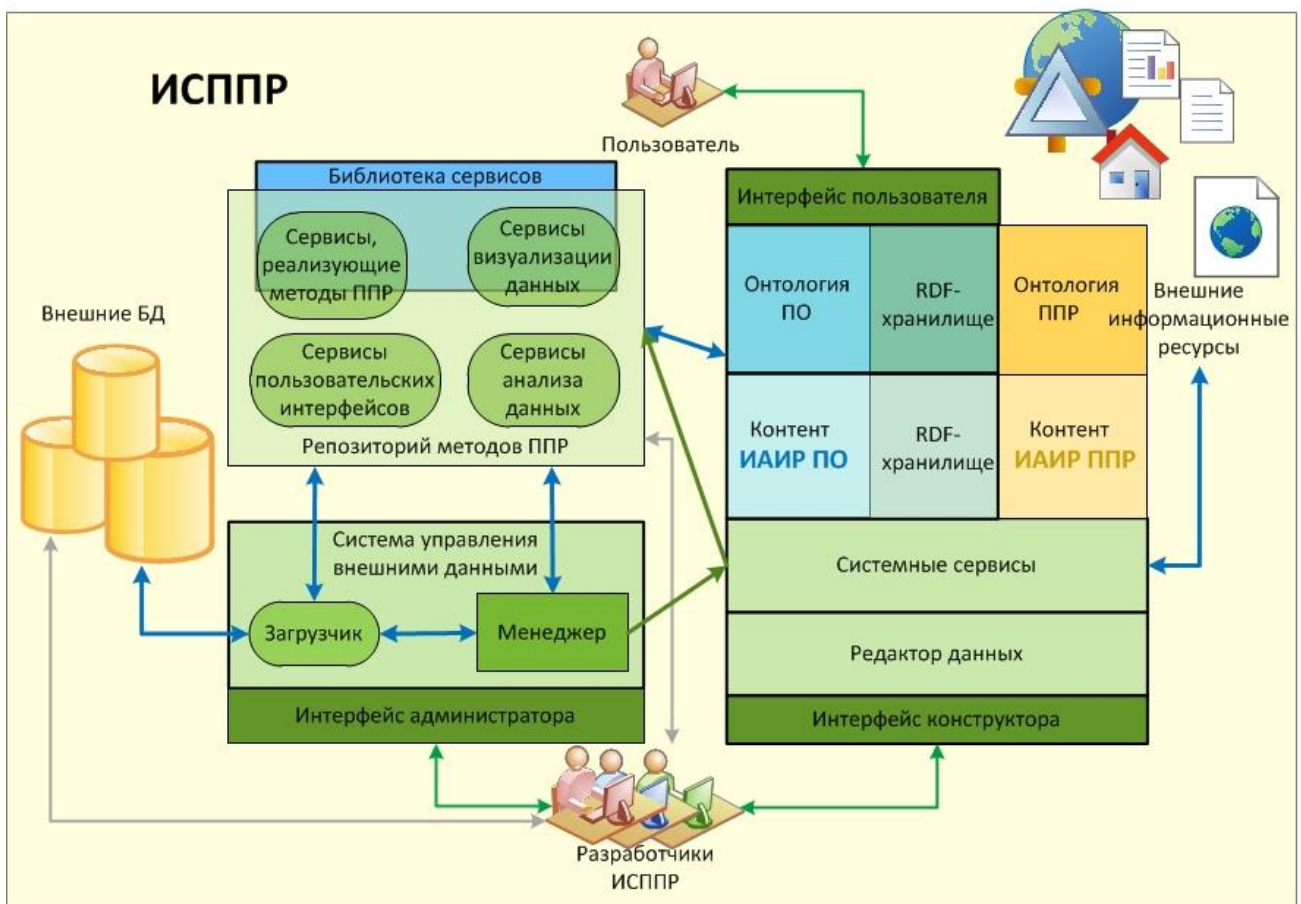


Рисунок 2.8. Архитектура типовой ИСППР

$$f(\text{включает}) = \{ \\
\quad (\text{ИСППР}, (\text{ОНТОЛОГИЯ_ПО}, \text{ИАИР_ПО}, \text{РЕПОЗИТОРИЙ_МЕТОДОВ_ППР})), \\
\quad (\text{ИАИР_ПО}, (\text{ОНТОЛОГИЯ_ПО}, \text{КОНТЕНТ_ИАИР_ПО})) \\
\}$$

$$f(\text{заимствуетИз}) = \{ \\
\quad (\text{ОНТОЛОГИЯ_ПО}, \text{ОНТОЛОГИЯ_ППР}), \\
\quad (\text{КОНТЕНТ_ИАИР_ПО}, \text{КОНТЕНТ_ИАИР_ППР}), \\
\quad (\text{БИБЛИОТЕКА_СЕРВИСОВ}, \text{РЕПОЗИТОРИЙ}) \}$$

2.7.2. Методика разработки интеллектуальных СППР

Методика разработки ИСППР (Рисунок 2.1) предлагает архитектуру (Рисунок 2.8) и алгоритм разработки ИСППР (Рисунок 2.9), основывается на рассмотренных выше принципах и подходах и использует указанные выше технологии [194,195]. Обеспечивая содержательный доступ к информации о подходах и принципах разработки ИСППР, а также к используемым технологиям, данная методика является средством методической поддержки.

$$\{ \text{МЕТОДИКА_РАЗРАБОТКИ_ИСППР} \} \sqsubseteq \\
\quad \exists \text{предлагает} . [\{ \text{АРХИТЕКТУРА_ИСППР} \} \sqcup \\
\quad \quad \{ \text{АЛГОРИТМ_РАЗРАБОТКИ_ИСППР} \}] \sqcap \\
\quad \exists \text{основываетсяНа} . [\text{Принцип_разработки_ИСППР} \sqcup \\
\quad \quad \text{Подход_к_разработке_ИСППР}] \sqcap \\
\quad \exists \text{использует} . \text{Технология_для_разработки_ИСППР} \sqcap \\
\quad \exists \text{реализует} . \text{Метод_КПР}$$

Рассматриваемая методика предлагает алгоритм, регламентирующий весь процесс разработки ИСППР и включающий последовательность действий:

$$\{ \text{АЛГОРИТМ_РАЗРАБОТКИ_ИСППР} \} \sqsubseteq \\
\quad \exists \text{регламентирует} . \text{Процесс_разработки_ИСППР} \sqcap \\
\quad \exists \text{включает} \text{Действие} . \text{Действие}$$

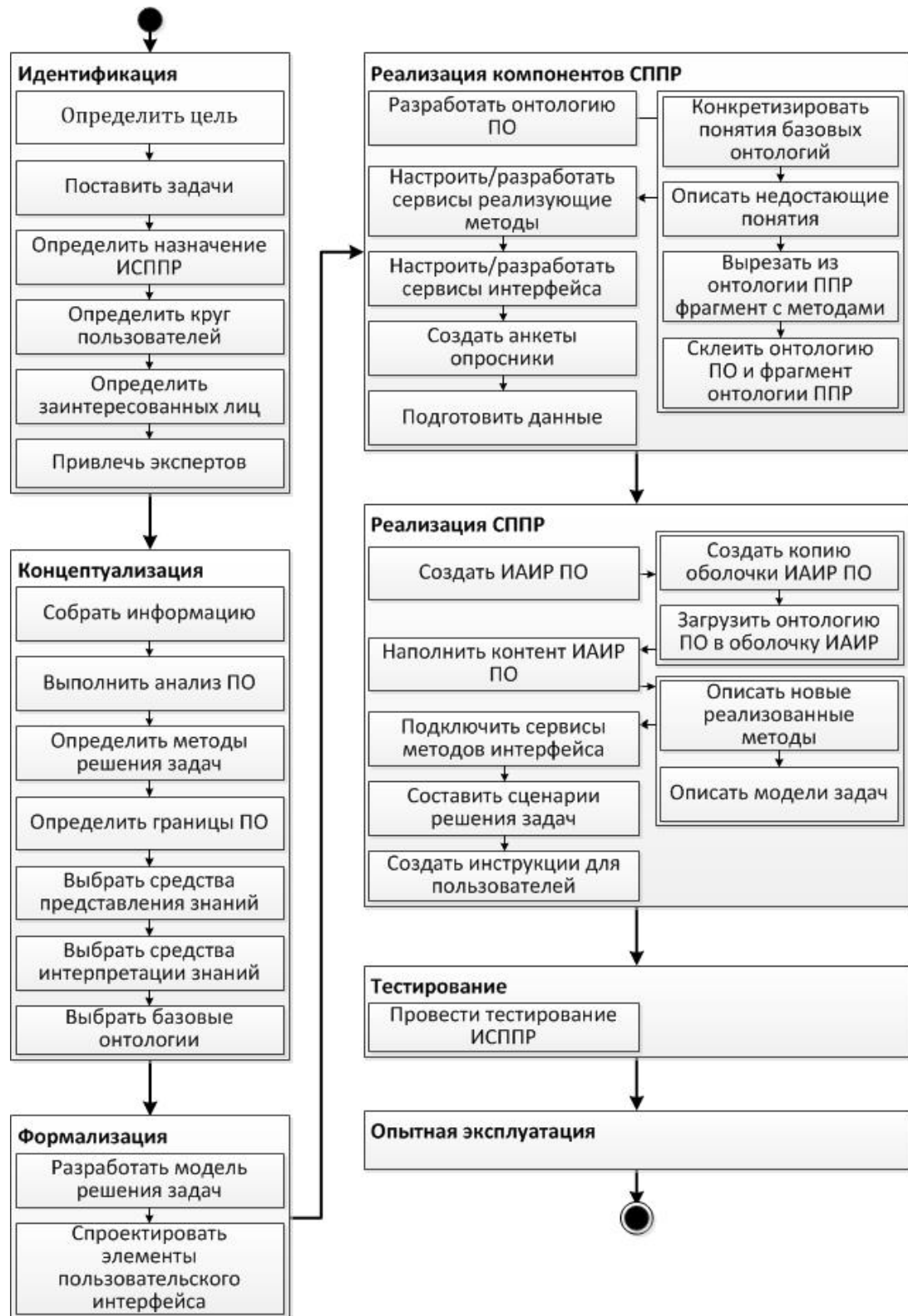


Рисунок 2.9. Этапы и алгоритм разработки ИСППР

$f(\text{Действие}) = \{$
 ОПРЕДЕЛИТЬ_ЦЕЛЬ, ПОСТАВИТЬ_ЗАДАЧИ,
 ОПРЕДЕЛИТЬ_НАЗНАЧЕНИЕ_ИСППР,
 ОПРЕДЕЛИТЬ_КРУГ_ПОЛЬЗОВАТЕЛЕЙ,

ОПРЕДЕЛИТЬ_ЗАИНТЕРЕСОВАННЫХ_ЛИЦ, ПРИВЛЕЧЬ_ЭКСПЕРТОВ,
 СОБРАТЬ_ИНФОРМАЦИЮ, ВЫПОЛНИТЬ_АНАЛИЗ_ПО,
 ОПРЕДЕЛИТЬ_МЕТОДЫ_РЕШЕНИЯ_ЗАДАЧ, ОПРЕДЕЛИТЬ_ГРАНИЦЫ_ПО,
 ВЫБРАТЬ_СРЕДСТВА_ПРЕДСТАВЛЕНИЯ_ЗНАНИЙ,
 ВЫБРАТЬ_СРЕДСТВА_ИНТЕРПРЕТАЦИИ_ЗНАНИЙ,
 ВЫБРАТЬ_БАЗОВЫЕ_ОНТОЛОГИИ,
 РАЗРАБОТАТЬ_МОДЕЛЬ_РЕШЕНИЯ_ЗАДАЧ,
 СПРОЕКТИРОВАТЬ_ЭЛЕМЕНТЫ_ПОЛЬЗОВАТЕЛЬСКОГО_ИНТЕРФЕЙСА,
 РАЗРАБОТАТЬ_ОНТОЛОГИЮ_ПО,
 КОНКРЕТИЗИРОВАТЬ_ПОНЯТИЯ_БАЗОВЫХ_ОНТОЛОГИЙ,
 ОПИСАТЬ_НЕДОСТАЮЩИЕ_ПОНЯТИЯ,
 ВЫРЕЗАТЬ_ИЗ_ОНТОЛОГИИ_ППР_ФРАГМЕНТ_С_МЕТОДАМИ,
 СКЛЕИТЬ_ОНТОЛОГИЮ_ПО_И_ФРАГМЕНТ_ОНТОЛОГИИ_ППР,
 НАСТРОИТЬ_РАЗРАБОТАТЬ_СЕРВИСЫ_РЕАЛИЗУЮЩИЕ_МЕТОДЫ,
 НАСТРОИТЬ_РАЗРАБОТАТЬ_СЕРВИСЫ_ИНТЕРФЕЙСА,
 СОЗДАТЬ_АНКЕТЫ_ОПРОСНИКИ, ПОДГОТОВИТЬ_ДАННЫЕ,
 СОЗДАТЬ_ИАИР_ПО, СОЗДАТЬ_КОПИЮ_ОБОЛОЧКИ_ИАИР_ПО,
 ЗАГРУЗИТЬ_ОНТОЛОГИЮ_ПО_В_ОБОЛОЧКУ_ИАИР,
 НАПОЛНИТЬ_КОНТЕНТ_ИАИР_ПО,
 ОПИСАТЬ_НОВЫЕ_РЕАЛИЗОВАННЫЕ_МЕТОДЫ,
 ОПИСАТЬ_МОДЕЛИ_ЗАДАЧ,
 ПОДКЛЮЧИТЬ_СЕРВИСЫ_МЕТОДОВ_ИНТЕРФЕЙСА,
 СОСТАВИТЬ_СЦЕНАРИИ_РЕШЕНИЯ_ЗАДАЧ,
 СОЗДАТЬ_ИНСТРУКЦИИ_ДЛЯ_ПОЛЬЗОВАТЕЛЕЙ,
 ПРОВЕСТИ_ТЕСТИРОВАНИЕ_ИСППР

}

Процесс разработки ИСППР включает те же этапы, что и разработка любых интеллектуальных систем [150].

{ ПРОЦЕСС_РАЗРАБОТКИ_ИСППР } $\equiv$

Э включает Этап . Этап_разработки_ИСППР

f(Этап_разработки_ИСППР) = {

ИДЕНТИФИКАЦИЯ, КОНЦЕПТУАЛИЗАЦИЯ, ФОРМАЛИЗАЦИЯ,
РЕАЛИЗАЦИЯ, ТЕСТИРОВАНИЕ, ОПЫТНАЯ_ЭКСПЛУАТАЦИЯ }

На множестве всех действий определено отношение « $\leq$ », задающее порядок следования действий алгоритма, а также отношение, сопоставляющее этапам разработки ИСППР набор действий, выполняющихся на этих этапах, и отношение, сопоставляющее действию тип разработчика, ответственного за выполнение этого действия:

Действие $\sqsubseteq$ $\exists \leq$. Действие

Действие $\sqsubseteq \exists$ выполняетсяНаЭтапе . Этап_разработки_ИСППР

Разработчик $\sqsubseteq \exists$ отвечаетЗа . Действие

2.8. Выводы

Модель комплексной поддержки разработки ИСППР – это совокупность методов, средств, принципов, подходов, технологий, алгоритмов, предназначенных для удовлетворения потребностей разработчиков, которые предлагаются им на всех этапах создания систем данного класса.

Основными методами КПР являются способы работы с информацией о методах и аспектах ППР – их систематизация и обеспечение содержательного доступа к ним, а также к реализациям методов ППР и методике разработки ИСППР. Средствами, реализующими методы концептуальной, информационной, компонентной и методической поддержки являются, соответственно, онтология ППР, ИАИР ППР, репозиторий методов ППР и методика разработки ИСППР.

Онтология систематизирует информацию об области знаний «Поддержка принятия решений» и обеспечивает разработчиков ИСППР единым концептуальным базисом.

Информационно-аналитический ресурс, базирующийся на онтологии ППР, является средством содержательного доступа как к информации о данной ОЗ, так и к репозиторию методов ППР.

Предложенная методика разработки репозитория основывается на сервис-ориентированном подходе и принципе максимального использования готовых решений. Для включения метода ППР в репозиторий необходимо найти готовую, либо разработать реализующую его программную систему, создать в контенте ИАИР ППР информационные объекты, описывающие метод и программную систему, а также все аспекты их взаимодействия и использования.

Методика разработки ИСППР предлагает архитектуру и алгоритм разработки ИСППР. Принципы и подходы, заложенные в основу методики, обуславливают следующие её основные особенности: нацеленность на слабоформализованные ПО; независимость от ПО; нацеленность на максимальное использование готовых разработок (как авторских, так и сторонних); использование технологий Semantic Web и сервис-ориентированного подхода, технологии разработки ИАИР; использование оболочки ИАИР в качестве каркаса будущей ИСППР; открытость и масштабируемость предлагаемых средств; удобство и низкий порог входа для использования предлагаемых средств.

Для формализации модели использован язык дескрипционной логики SOIN(D).

Глава 3. Реализация предлагаемых средств комплексной поддержки разработки интеллектуальных СППР

3.1. Разработка онтологии Поддержки Принятия Решений

3.1.1. Конкретизация понятий базовых онтологий

Онтология, являющаяся базисом концептуальной поддержки, модель которой была рассмотрена в главе 2, в свою очередь, представляет собой формальную систему, задающую концепты, роли и аксиомы ОЗ ППР.

На Рисунке 3.1 показана иерархия основных концептов (понятий) этой области.

Онтология ППР, согласно методике, строится на основе базовых онтологий научного знания, научной деятельности, задач и методов, научных информационных ресурсов путем дополнения и конкретизации содержащихся в них понятий, а также паттернов онтологического проектирования [191]. Базовые онтологии предоставляются технологией разработки ИНИР [173], описанной в п.1.5.3.

Рассмотрим описание онтологии ППР с использованием формализма дескрипционной логики SHOIN(D), который был использован и для описания модели КПР (глава 2).

Пусть $FO = \langle OC, OR, OA \rangle$ – формальная система, где OC – множество концептов ОЗ ППР, OR – множество атомарных ролей, описывающих свойства концептов и отношения между ними, OA – множество аксиом, устанавливающих связи между концептами и ролями.

Рассмотрим конкретизации понятий базовой онтологии и некоторые роли и аксиомы формальной системы FO .

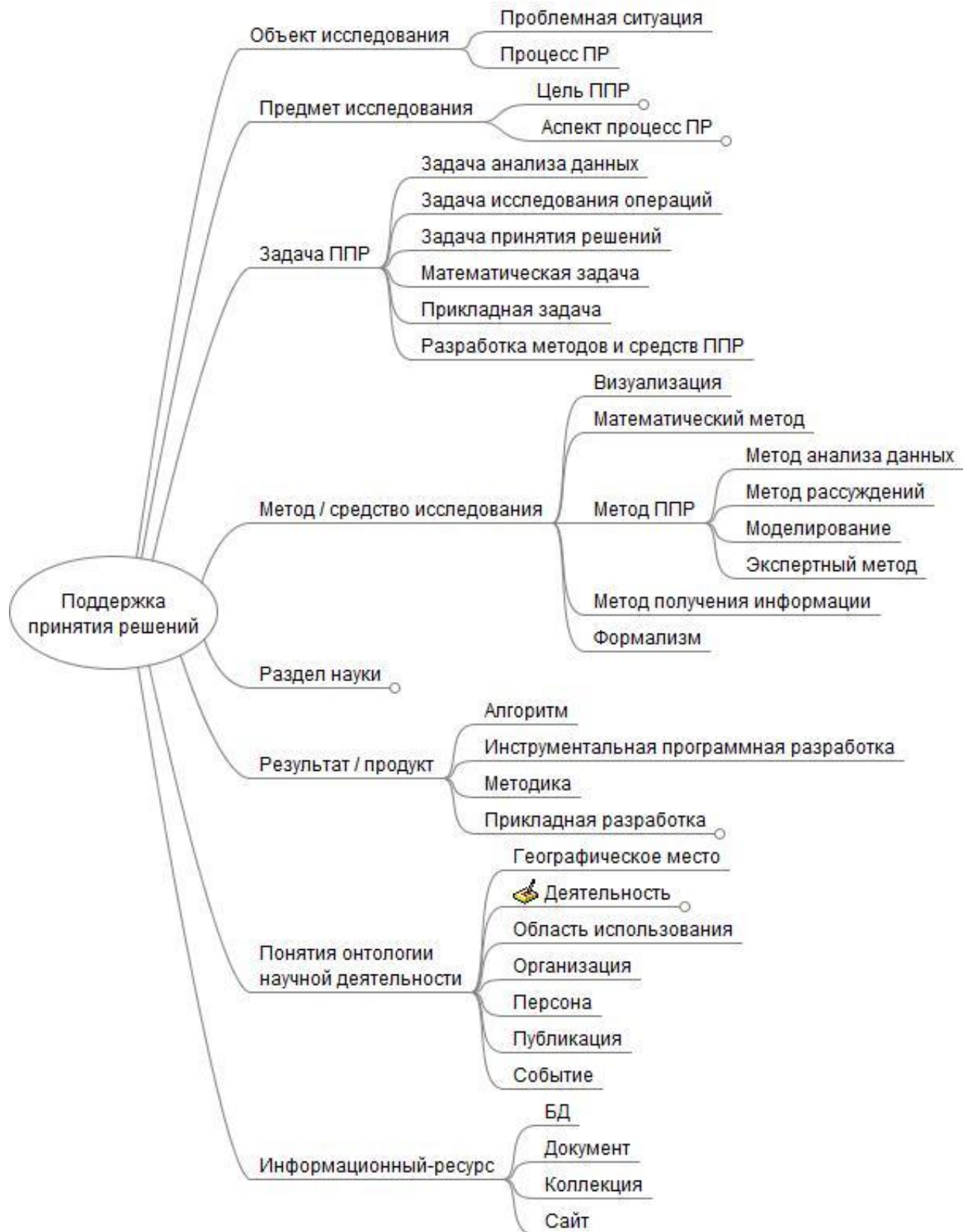


Рисунок 3.1 – Область знаний «Поддержка принятия решений»

ОС = { Географическое_место, Деятельность, Задача_ППР,
Информационный_ресурс, Метод/средство_исследования,
Область_использования, Объект_исследования,

Организация, Область_использования, Персона,
Предмет_исследования, Публикация, Раздел_науки,
Результат/продукт, Событие }

Эти понятия базовых онтологий конкретизируются для рассматриваемой ОЗ
ППР следующим образом:

Объект_исследования $\supseteq$ Проблемная_ситуация $\sqcup$ Процесс_ПР

Предмет_исследования $\supseteq$ Аспект_процесса_ПР $\sqcup$ Информация $\sqcup$
Представление_информации $\sqcup$ Цель_ППР

Аспект_процесса_ПР $\supseteq$ Альтернатива $\sqcup$ Критерий_оценки $\sqcup$

Метрика/мера_близости $\sqcup$ Предпочтение_ЛПР $\sqcup$ Процесс_работы_метода $\sqcup$
Шкала Этап_ПР $\sqcup$ Этап_работы_метода

Информация $\supseteq$ Данные $\sqcup$ Знания

Элемент_знания $\supseteq$ Закономерность $\sqcup$ Понятие $\sqcup$ Свойство

Представление информации $\supseteq$ Визуальное_представление $\sqcup$
Математическое_представление

Цель_ППР $\supseteq$ Разрешение проблемной ситуации $\sqcup$ Помощь ЛПР

Задача_ППР $\supseteq$ Задача_анализа_данных $\sqcup$ Задача_исследования_операций $\sqcup$
Задача_принятия_решений $\sqcup$ Математическая_задача $\sqcup$
Прикладная_задача $\sqcup$ Разработка_методов_средств_ППР

Метод/средство_исследования $\supseteq$ Визуализация $\sqcup$ Математический_метод $\sqcup$
Метод_ППР $\sqcup$ Метод_получения_информации $\sqcup$ Формализм

Метод_ППР $\supseteq$ Метод_анализа_данных $\sqcup$ Метод_рассуждений $\sqcup$
Метод_Моделирования $\sqcup$ Экспертный метод

Результат/продукт $\supseteq$ Алгоритм $\sqcup$ Программная_система $\sqcup$ Методика $\sqcup$
Прикладная_разработка

Прикладная_разработка $\supseteq$ База_данных $\sqcup$ Онтология $\sqcup$ СППР $\sqcup$
Экспертная_система

OR = { включает_Задача_Задача, включает_ПроцессРМ_ЭтапРМ,

включает_ЭтапРМ_ЭтапРМ, использует_Информация_Информация,
 использует_Информация_Представление,
 использует_Информация_Формализм, использует_Метод_Информация,
 использует_Метод_Метод, использует_Метод_МетрикаМера,
 использует_Представление_Представление,
 используется_Метод_Деятельность, используется_Метод_РазделНауки,
 поддерживает_РезультатПродукт_МетодСредство,
 предназначенаДляРазработки_ПрограммнаяСистема_Знания,
 предоставляетДоступ_РезультатПродукт_РезультатПродукт,
 решает_Метод_Задача, решаетсяВ_Задача_Раздел, решаетсяНа_Задача_ЭтапПР,
 решаетсяНа_Задача_ЭтапРМ, сводитсяК_Задача_Задача,
 содержит_Информация_Информация
 }

Множество АО, помимо рассмотренных выше аксиом, задающих иерархию классов ОЗ ППР, включает также аксиомы, определяющие левые и правые аргументы соответствующих ролям отношений, например:

Метод_ППР $\equiv \exists$ решает_Метод_Задача . Задача_ППР

Ограничения ролей могут быть и более сложными, как в следующей аксиоме, которая задает ограничение на роль «предоставляет_доступ_к_Продукту»:

РезультатПродукт $\sqcap \exists$ Продукт_Тип . {сервис, web-приложение} $\sqcap$

≤ 1 Продукт_Тип.Т $\equiv$

$\exists$ предоставляет_доступ_к_Продукту . РезультатПродукт

Т.е. левым аргументом отношения, соответствующего данной роли, может быть продукт, для которого задан один из типов «сервис» или «web-приложение».

Онтология ППР должна быть связана с понятиями той предметной области, для которой создается ИСППР. Для облегчения работы пользователей могут разрабатываться базовые предметные онтологии (онтология медицинской диагностики, онтология технической диагностики, онтология мониторинга технического объекта и т.д.), которые затем будут использоваться для построения онтологии конкретных ПО. Связь онтологии ППР и онтологии конкретной ПО осуществляется путём отождествления таких сущностей, как Объект

исследования и Предмет исследования, описанных в первой онтологии, с сущностями онтологии ПО.

3.1.2. Фрактально-стратифицированная структура онтологии

Для того чтобы онтология была практическим средством концептуальной поддержки разработчиков ИСППР (ЛПР, экспертов, инженеров знаний, программистов), она должна предоставлять каждому типу разработчиков интересующие именно их аспекты этих понятий. Особенно это касается информации, связанной с разработкой и применением методов ППР для решения прикладных задач. Учитывая особую важность этих понятий для рассматриваемой ОЗ, их описание выделено в отдельную, устроенную особым образом, онтологию задач и методов ППР. Для ее разработки был использован фрактально-стратифицированный подход [135] и пирамидальный способ расслоения (глава 2). При таком способе онтология имеет так называемую «вертикально-горизонтальную» структурную организацию, что позволяет представлять интересующие понятия в двух измерениях (Рисунок 3.2).



Рисунок 3.2. Вертикально-горизонтальная организация онтологии

Первое измерение («вертикальная структуризация») представляется в виде традиционной иерархической структуры, на каждом уровне которой понятия описываются с разной степенью детализации.

Второе измерение («горизонтальная структуризация») задаёт описание понятий с точки зрения разных типов специалистов, работающих над решением

общей задачи. В качестве общего ядра, на котором строится рассматриваемая онтология, выступает метаонтология задач и методов ППР (Рисунок 3.3).

3.1.3. Использование технологий Semantic Web при разработке онтологии

Онтология ППР разрабатывалась на языке OWL средствами редактора Protégé. Для представления вертикально-горизонтальной организации онтологии был разработан структурный паттерн, показанный на Рисунке 3.4., реализованный средствами языка OWL и редактора Protégé.

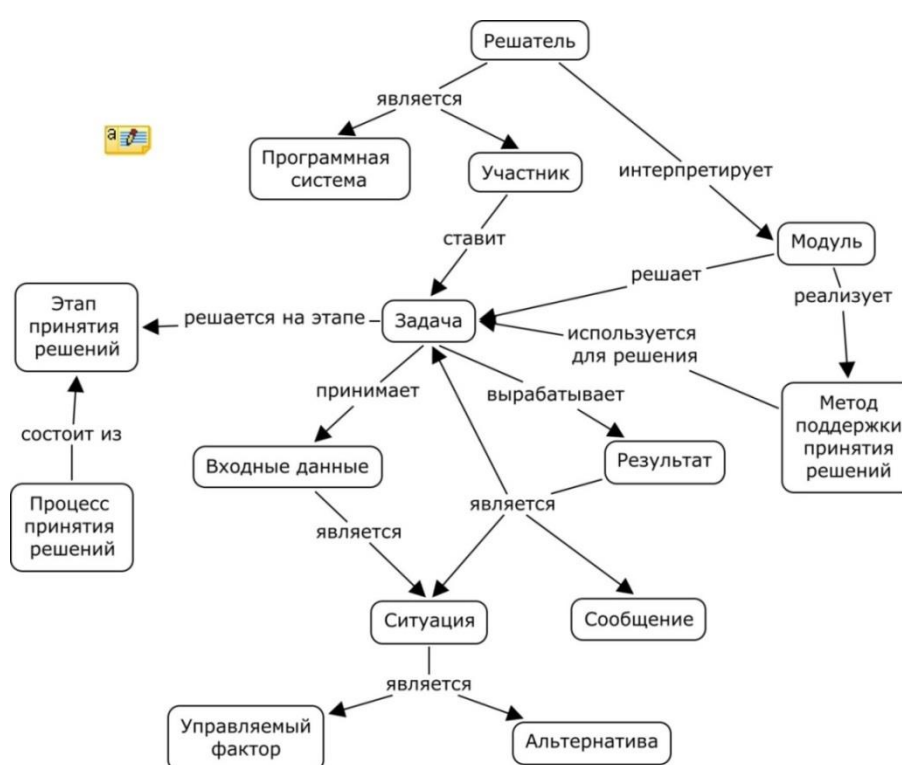


Рисунок 3.3. Метаонтология задач и методов ППР

Для реализации данного паттерна было использовано аннотационное свойство языка OWL Annotation Property, которое можно приписать любому элементу онтологии или объекту. Список встроенных аннотационных свойств был дополнен новым свойством typeDesigner со значениями {ЛПР, ИНЖЕНЕР ЗНАНИЙ, ПРОГРАММИСТ}. Это свойство, наряду со встроенным свойством label, приписывается всем элементам онтологии и объектам (Рисунок 3.5).

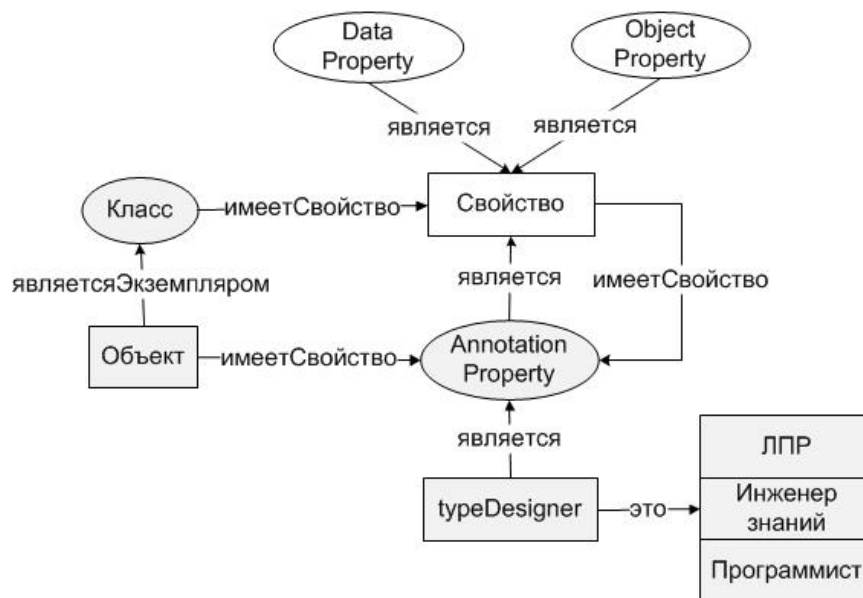


Рисунок 3.4. Паттерн для указания типа разработчика

В процессе работы ИАИР ППР пользователь может выбрать, к какому типу разработчика он относится, и в зависимости от этого ему будут показываться только классы, объекты и свойства этого типа.

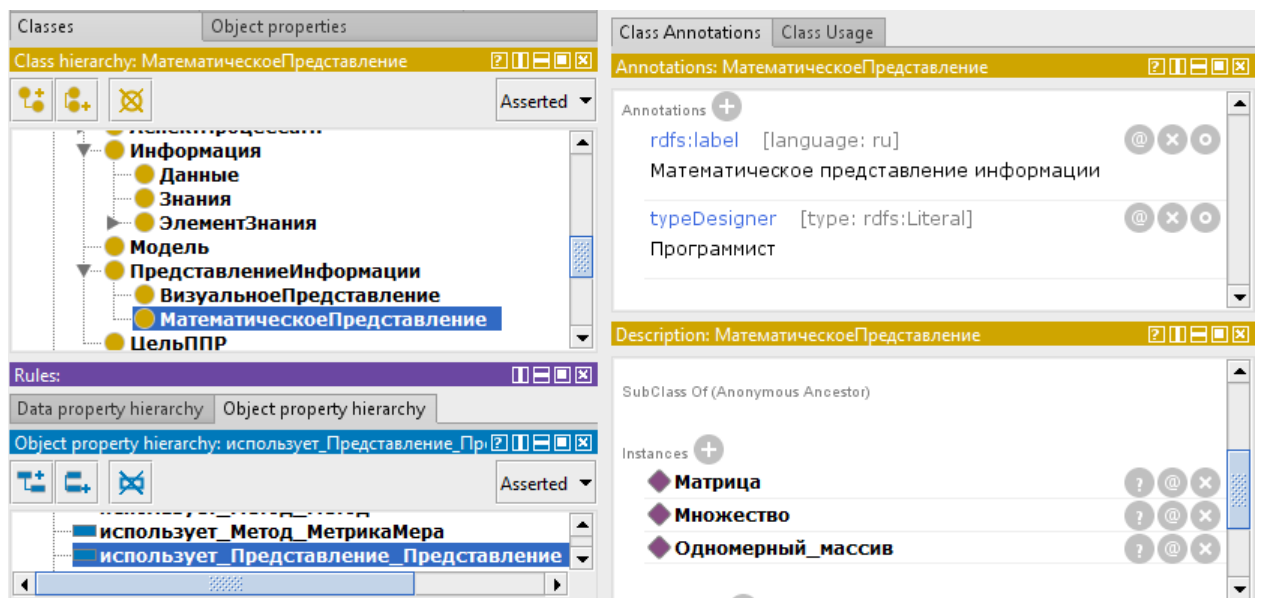


Рисунок 3.5. Реализация паттерна в редакторе Protégé

3.2. Реализация информационно-аналитического интернет-ресурса по поддержке принятия решений

Информационно-аналитический интернет-ресурс по поддержке принятия решений является средством информационной поддержки разработчиков ИСППР. Для его создания была использована технология разработки ИНИР [173]. Данная технология предоставляет оболочку ресурса, методику построения онтологии и контента ресурса, редактор данных, а также средства извлечения научной информации из сети Интернет [1,77].

Для реализации ИАИР была создана копия оболочки ИНИР и настроены элементы пользовательского интерфейса. В оболочку была загружена онтология ППР. На Рисунке 3.6 представлена страница ИАИР ППР, в левой части которой показаны понятия онтологии, организованные в иерархию «общее-частное».

The screenshot displays the IAIR PPR web interface. At the top, there is a navigation bar with links: Главная, **Онтология**, and О ресурсе. A search icon and 'Вход' are on the right. Below the navigation bar, the main heading reads 'ПОДДЕРЖКА ПРИНЯТИЯ РЕШЕНИЙ В СЛАБОФОРМАЛИЗОВАННЫХ ОБЛАСТЯХ'. To the right of the heading are three 3D figures with puzzle pieces. The main content area is divided into two columns. The left column contains a hierarchical ontology tree. The right column displays detailed information for a selected object, including its properties, relationships, and reverse relationships.

Онтология

- Географическое место
- + Деятельность
- + Интернет ресурс
- Метод / средство исследования
 - + Вспомогательный Метод
 - Метод поддержки принятия решений
 - + АнализДанных
 - + Метод рассуждений
 - Моделирование
 - Аналитическое моделирование
 - + Математическое моделирование
 - Программирование в ограничениях**
 - Имитационное моделирование
 - Семантическое моделирование
 - + Экспертный метод
- Объект исследования
- Организация
- Персона
- + Предмет исследования
- Публикация
- Раздел науки
- + Результат / продукт
- Событие

Свойства объекта

Название	Метод недоопределенных вычислений
Описание	Метод, предложенный А.С. Нариньяни. Является методом программирования в ограничениях в самой общей постановке
Дата возникновения	1980

Связи объекта

имеет Автор-Персону
Персона
Нариньяни А.С.

Обратные связи объекта

описывает Метод
Публикация
А.С. Нариньяни НЕ-факторы: неточность и недоопределенность – различие и взаимосвязь 2000
А.С. Нариньяни Недоопределенность в системах представления и обработки знаний 1986
В.А. Сидоров, Г.Б. Загоруйко NeMo+: Объектно-ориентированная система программирования в ограничениях на основе недоопределенных моделей 1998
В.А. Сидоров, Г.Б. Загоруйко Метод недоопределенных вычислений как средство поддержки принятия решений в слабоформализованных предметных областях 2016
реализует Программный продукт Метод
Результат / продукт
Решатель UniCalc
Система Интегра
реализует Сервис Метод
Проблемно-ориентированный сервис
WebUniCalc

Рисунок 3.6. ИАИР «Поддержка принятия решений»

3.2.1. Контент информационно-аналитического интернет-ресурса по поддержке принятия решений

Информационное наполнение ИАИР ППР, его контент, образуют объекты – конкретные экземпляры классов онтологии и конкретные факты отношений между ними. С формальной точки зрения, контент – это модель ОЗ ППР, интерпретация формальной системы FO (п. 3.1.1), удовлетворяющая ее аксиомам.

Рассмотрим в качестве примера часть модели, описывающую задачи ППР и некоторые связи объекта Структуризация_Предметной_Области:

$f(\text{Задача_ППР}) = \{$
 АНАЛИЗ_ТЕКУЩЕЙ_СИТУАЦИИ, АНАЛИЗ_ФАКТОРОВ,
 ВЫЯВЛЕНИЕ_ПРОБЛЕМНОЙ_СИТУАЦИИ, ВЫЯВЛЕНИЕ_ФАКТОРОВ,
 ДЕКОМПОЗИЦИЯ_ЦЕЛИ, ДИАГНОСТИКА_ПРОБЛЕМНОЙ_СИТУАЦИИ,
 ОБЕСПЕЧЕНИЕ_СИТУАЦИОННОЙ_ОСВЕДОМЛЕННОСТИ,
 ПОСТАНОВКА_ЦЕЛИ_ПРИНЯТИЯ_РЕШЕНИЙ, ПОСТРОЕНИЕ_ОНТОЛОГИИ_ПО,
 ПРОГНОЗИРОВАНИЕ, РАЗРАБОТКА_СТРАТЕГИЧЕСКОГО_ПЛАНА,
 РАЗРАБОТКА_ТЕКУЩЕГО_ПЛАНА, РАЗРАБОТКА_АЛЬТЕРНАТИВ,
 СТРУКТУРИЗАЦИЯ_ПРЕДМЕТНОЙ_ОБЛАСТИ,
 СТРУКТУРИЗАЦИЯ_ПРОБЛЕМНОЙ_СИТУАЦИИ }
 $\}$

$f(\text{Задача_анализа_данных}) = \{$
 ВЫБОР_ИНФОРМАТИВНЫХ_ПРИЗНАКОВ,
 ВЫЯВЛЕНИЕ_АССОЦИАЦИЙ, ВЫЯВЛЕНИЕ_ОТКЛОНЕНИЙ_ОТ_ТРЕНДА,
 ВЫЯВЛЕНИЕ_ПОСЛЕДОВАТЕЛЬНОСТЕЙ, ВЫЯВЛЕНИЕ_ТРЕНДОВ,
 КЛАССИФИКАЦИЯ, КЛАСТЕРИЗАЦИЯ }
 $\}$

$f(\text{Задача_исследования_операций}) = \{$
 ЗАДАЧА_О_НАЗНАЧЕНИЯХ,
 ЗАДАЧА_О_РАЗМЕЩЕНИИ_ОБЪЕКТОВ, ЗАДАЧА_О_РАНЦЕ,
 ЗАДАЧА_ОТБОРА_ПРОЕКТОВ, ЗАДАЧА_РАСПРЕДЕЛЕНИЯ_РЕСУРСОВ,
 ТРАНСПОРТНАЯ_ЗАДАЧА, ФОРМИРОВАНИЕ_ПОРТФЕЛЯ_ПРОЕКТОВ }
 $\}$

$f(\text{Задача_математическая}) = \{$
 ВЫЧИСЛЕНИЕ_РАССТОЯНИЯ,
 $\}$

ЗАДАНИЕ_ВЕКТОРНОГО_ПРОСТРАНСТВА, ЗАДАНИЕ_ФУНКЦИИ_ПОЛЕЗНОСТИ,
ЗАДАЧА_УДОВЛЕТВОРЕНИЯ_ОГРАНИЧЕНИЙ }

f(Задача_принятия_решений) = {
АНАЛИЗ_МНЕНИЙ_НЕСКОЛЬКИХ_ЭКСПЕРТОВ,
АНАЛИЗ_МНЕНИЯ_ОДНОГО_ЭКСПЕРТА, ВЫБОР_ЛУЧШЕЙ_АЛЬТЕРНАТИВЫ,
ИЗВЛЕЧЕНИЕ_ПРЕЦЕДЕНТА, КЛАСТЕРИЗАЦИЯ_АЛЬТЕРНАТИВ,
ОПРЕДЕЛЕНИЕ_КРИТЕРИЕВ, ОПРЕДЕЛЕНИЕ_ПРЕДПОЧТЕНИЙ_ЛПР,
ОЦЕНКА_АЛЬТЕРНАТИВ, РАЗРАБОТКА_АНКЕТ,
РАНЖИРОВАНИЕ_АЛЬТЕРНАТИВ }

f(Разработка_методов_средств_ППР) = {
РАЗРАБОТКА_МЕТОДОВ_ППР,
РАЗРАБОТКА_ОНТОЛОГИИ_ОЗ_ППР }

решаетсяЗадачаНаЭтапе (СТРУКТУРИЗАЦИЯ_ПРОБЛЕМНОЙ_СИТУАЦИИ,
ВЫЯВЛЕНИЕ_ПРОБЛЕМЫ)
включаетЗадачаЗадача (ПОСТРОЕНИЕ_ОНТОЛОГИИ_ПО,
СТРУКТУРИЗАЦИЯ_ПРОБЛЕМНОЙ_СИТУАЦИИ)
решаетМетодЗадача (МЕТОД_ОНТОЛОГИЧЕСКОГО_МОДЕЛИРОВАНИЯ,
СТРУКТУРИЗАЦИЯ_ПРОБЛЕМНОЙ_СИТУАЦИИ)

Для наполнения контента ИАИР ППР были использованы три способа: с помощью редактора Protégé, с помощью встроенного в оболочку редактора данных [76], с помощью системы извлечения научных данных [1], входящей в состав технологии разработки ИНИР.

3.2.2. Создание контента с помощью технологий разработки интеллектуальных научных интернет-ресурсов и Semantic Web

Контент ИАИР может быть описан непосредственно в онтологии как ее A-Box – набор утверждений об индивидах [6]. Технология разработки ИНИР позволяет своим пользователям самим настроить некоторые элементы отображения контента, а именно – порядок следования атрибутов объектов и имя, под которым объект будет представлен на ресурсе. Для этого был разработан

структурный паттерн, показанный на Рисунке 3.7. Данный паттерн был реализован аналогично паттерну, позволяющему указать тип разработчика (п.3.1.3). Для настройки имени используется встроенное аннотационное свойство `rdfs:label`, которое можно приписать любому объекту онтологии – классу, свойству, объекту. Для настройки порядка следования атрибутов вводится новое аннотационное свойство `order`.

Для загрузки в оболочку ИАИР онтология ППР должна быть сохранена на языке OWL в синтаксисе turtle.

При создании новых информационных объектов был также использован встроенный в оболочку редактор данных (Рисунок 3.8).

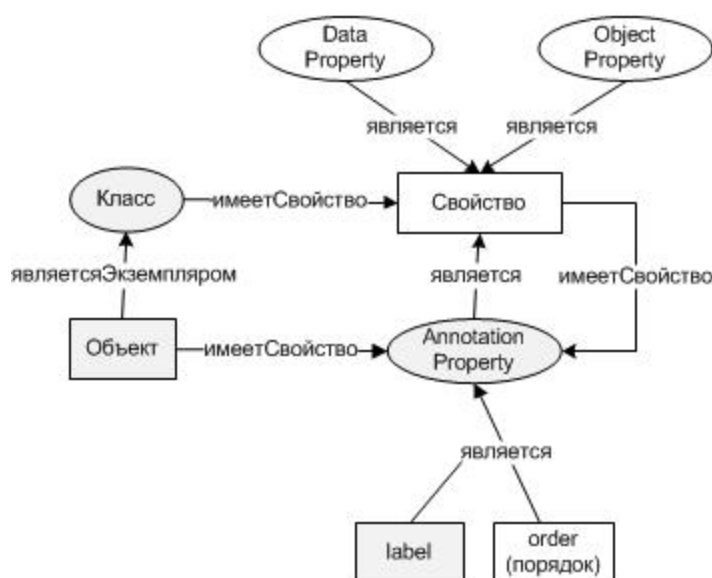


Рисунок 3.7. Паттерн настройки отображения объектов

Данный редактор является альтернативой редактору Protégé. Помимо ввода информационных объектов он позволяет добавлять в контент ИАИР такие сетевые объекты, как полнотекстовые и мультимедиа документы. После ввода и редактирования объекты сохраняются в ttl-файле, содержащем онтологию. Этот файл может в дальнейшем быть открыт в редакторе Protégé.

Для наполнения контента ИАИР, в частности для поиска и создания информационных объектов, описывающих публикации, также была использована система извлечения научных данных [1].

Метка

Похолодание

Генерировать автоматически

Свойства

Название	+	ru Похолодание	×
Описание	+		
Длительность реализации	+	тактическая	×
Количество решений	+	несколько	×
Известность вариантов решения	+	варианты известны	×
Описание	+ -	Задача "Угроза похолодания" Для исследования выбрана природная угроза "Б" в виде аномального проявления нормальных природных процессов: ситуации с похолоданием. В случае похолодания и/или снижения температуры наружного воздуха относительно среднесуточных в определенные месяцы отопительного периода повышается потребление тепло- и электроэнергии. Подробное описание задачи дано в публикациях, указанных на странице данной задачи. Рассматриваемая задача может быть решена разными способами: методом Байеса, методом неопределенных вычислений (МНВ), методом когнитивного моделирования и т.д. /threats/uploads/59/a6/348819942c12b3dd724eee4adb2d.pdf	

Связи

включает Задачу

сводится к Задаче

связана с Задачей

Задача Энергетической безопасности

Похолодание

Задача Энергетической безопасности

Разработка и реализация превентивных мероприятий

Рисунок 3.8 Встроенный в оболочку ИНИР редактор данных

3.3. Реализация репозитория методов поддержки принятия решений

В главе 2 была описана модель и методика построения репозитория методов ППР. С точки зрения реализации репозиторий представляет собой совокупность информационных объектов ИАИР ППР, которые описывают методы, решаемые ими задачи и реализующие их программные системы. Если программная система имеет реализацию в виде сервиса или находится в свободном доступе и доступна для скачивания, то она тоже может стать частью репозитория. Доступ к репозиторию осуществляется средствами ИАИР ППР. Выбрав на ресурсе

соответствующий метод, пользователь получает ссылку, по которой он может либо запустить сервис, либо скачать соответствующую программную систему.

3.3.1. Использование паттернов онтологического проектирования для описания методов поддержки принятия решений

Для описания методов был создан соответствующий паттерн содержания. Его использование облегчает этот процесс и позволяет сократить время и количество возможных ошибок. Паттерн содержания представляет собой фрагмент онтологии, описывающий интересующие понятия и их взаимосвязи. На Рисунке 3.9 показано визуальное представление паттерна содержания для описания методов и задач.

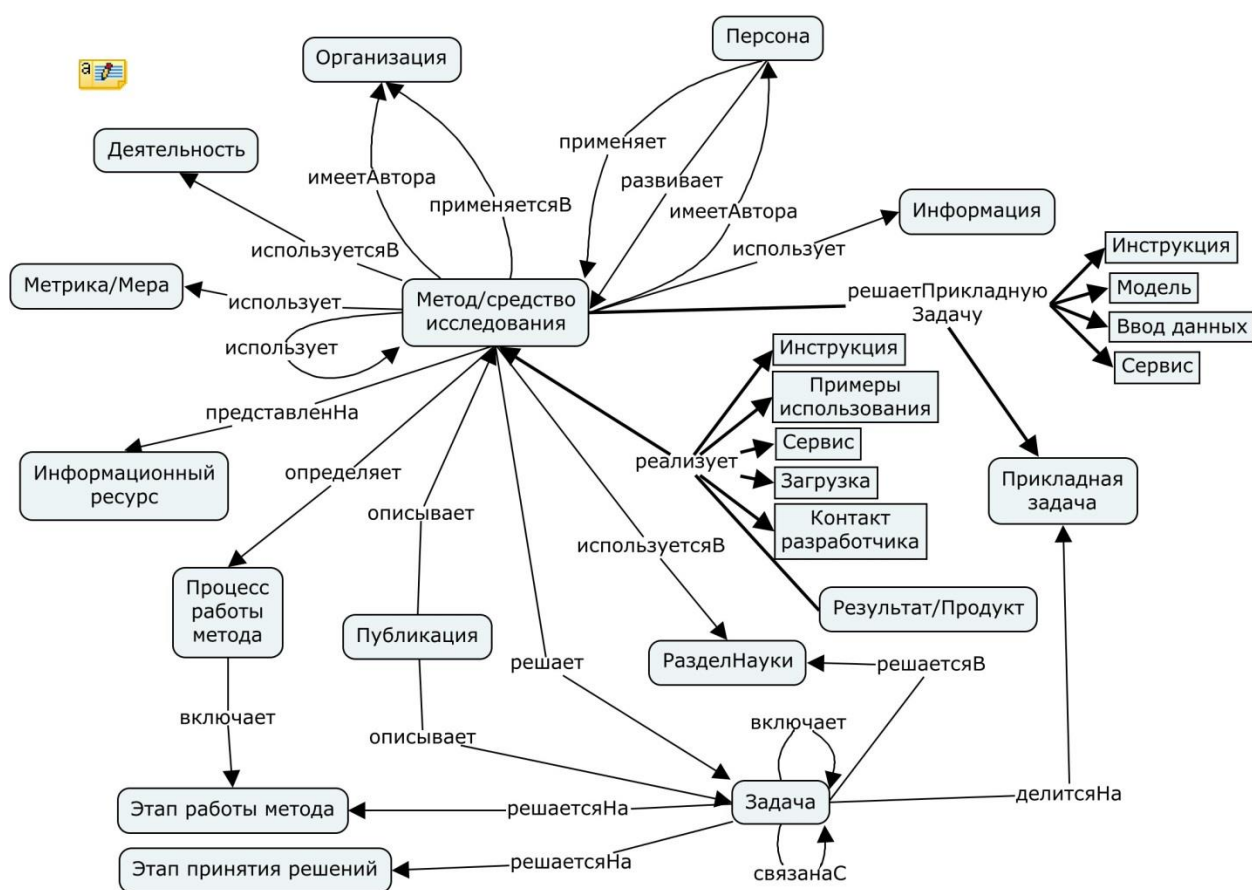


Рисунок 3.9. Паттерн содержания для описания методов и задач

Два отношения, включенные в этот паттерн – реализует и решаетПрикладнуюЗадачу – являются отношениями, которым приписаны атрибуты. Технология разработки ИНИР предлагает для реализации таких отношений специальные паттерны и методику их реализации с использованием

технологий Semantic Web [191]. В приложении Б приведены визуальные представления паттернов для задания **доменов** и **атрибутированных отношений** (Рисунки Б.2 и Б.3).

В п. 1.4 был описан ряд методов и реализующих их программных систем, которые используются при принятии решений. В главе 2 (п. 2.6.1) был приведен перечень методов ППР и систем, включенных в репозиторий. Рассмотрим более подробно те методы ППР, реализации которых были выполнены либо автором диссертационного исследования лично, либо при участии автора, либо являются сторонними разработками, но использовались для реализации авторских систем. При разработке методов использовался рассмотренный паттерн содержания. Для каждого метода с помощью графического редактора SMapTols [37] построены концептуальные карты, в которых конкретизируются понятия и отношения паттерна (Приложение В, Рисунки В.1-В.5).

Для включения этих методов в репозиторий были разработаны соответствующие информационные объекты. Опишем их в соответствии с паттерном содержания в виде совокупности утверждений (аксиом) о связях данных объектов с объектами других классов, формализованных средствами языка дескрипционной логики SOIN(D).

3.3.2. Рассуждения на основе прецедентов

имеетАвтораПерсону (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ, { SCHANK_R, ABELSON_R, KOLODNER_J })

применяетсяВ (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ, { ИСИ_СО_РАН, УГАТУ, ИСЭМ_СО_РАН, COMPLUTENCE_UNIVERSITY })

развивает (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ, { ВАРШАВСКИЙ_ПР, ЕРЕМЕЕВ_АП, RECIO-GARCÍA_JA, }),

использует (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ, { ВАРШАВСКИЙ_ПР, ЕРЕМЕЕВ_АП, ЮСУПОВА_НИ, ГЕЛОВАНИ_ВА, ЧЕРНЯХОВСКАЯ_ЛР, ЗАГОРУЛЬКО_ГБ, МАССЕЛЬ_АГ })

реализует (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ, COLIBRI,

[DOWNLOAD: <https://sourceforge.net/projects/jcolibri-cbr/>,

КОНТАКТ_РАЗРАБОТЧИКА: <http://gaia.fdi.ucm.es/research/colibri/#about>])

решает_Метод_Задача (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ,

{ ДИАГНОСТИКА_ПРОБЛЕМНОЙ_СИТУАЦИИ, ПРОГНОЗИРОВАНИЕ })

используетИнформацию (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРЕЦЕДЕНТОВ,

БАЗА_ПРЕЦЕДЕНТОВ)

3.3.3. Метод недоопределенных вычислений

имеетАвтораПерсону (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ,

НАРИНЬЯНИ_АС)

имеетАвтораОрганизацию (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ,

ИСИ_СО_РАН)

применяетсяВ (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, ИСИ_СО_РАН)

развивает (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, { НАРИНЬЯНИ_АС,

УШАКОВ_ДМ, СИДОРОВ_ВА, ТЕЛЕРМАН_ВВ, КАШЕВАРОВА_ТП, СЕМЕНОВ_АЛ,

ЗАГОРУЛЬКО_ГБ }),

использует (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, { ЗАГОРУЛЬКО_ГБ,

НАПРЕЕНКО_ВВ, ЛИПАТОВ_А })

реализует (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, UNICALC,

КОНТАКТ_РАЗРАБОТЧИКА: zagor@iis.nsk.su])

реализует (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, WEBUNICALC,

[СЕРВИС: <http://uniserv.iis.nsk.su/unicalc>

ПРИМЕРЫ_ИСПОЛЬЗОВАНИЯ:

/ rdms /uploads/bf/c3/9a51dc369a7b90cf2b0cfb835575.uni

КОНТАКТ_РАЗРАБОТЧИКА: zagor@iis.nsk.su])

реализует (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, SEMP_НЕМО ,

[ИНСТРУКЦИЯ: /rdms /uploads/bf/c3/20047fb20adf221095b7eea2c309.pdf

ПРИМЕРЫ_ИСПОЛЬЗОВАНИЯ:

/rdms /uploads/bf/c3/9a51dc369a7b90cf2b0cfb835575.uni

КОНТАКТ_РАЗРАБОТЧИКА: zagor@iis.nsk.su])

решает_Метод_Задача (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ,

ЗАДАЧА_УДОВЛЕТВОРЕНИЯ_ОГРАНИЧЕНИЙ)

использует Информацию (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ,

{ МНОЖЕСТВО_ОБЪЕКТОВ, МНОЖЕСТВО_ЭКЗЕМПЛЯРОВ_ОГРАНИЧЕНИЙ })

Как видно из формализованного описания, метод недоопределенных вычислений реализуется несколькими программными системами. Для предоставления удаленного доступа к этому методы были разработаны два сервиса – SempNemo-сервис и Приложение WebUnicalc.

SempNemo-сервис разработан с использованием протоколов SOAP/WSDL. Была использована платформа .Net, язык C#. Сервис имеет клиент-серверную архитектуру.

Его серверная часть позволяет настроить соединение с сервером (Рисунок 3.10).

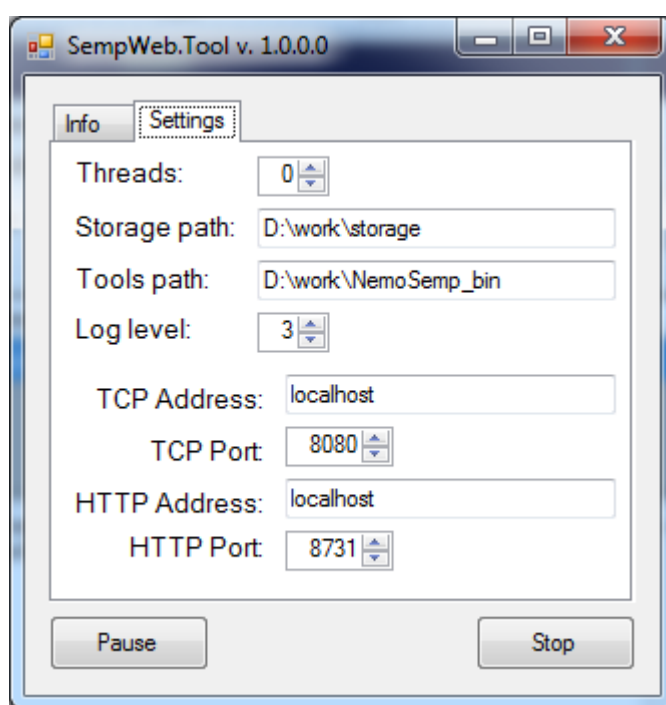


Рисунок 3.10. Интерфейс для установления соединения с сервером

Для этого необходимо выбрать метод (помимо метода недоопределенных вычислений данный сервис предоставляет доступ и к методу рассуждений на основе экспертных правил), указать папку на сервере, в которой будет размещена модель для решения задачи, и папку, в которой находится программная система, реализующая выбранный метод.

Клиентская часть сервера позволяет пользователю задать модель (или загрузить готовую) и выполнить вычисления.

На Рисунке 3.11 показан пользовательский интерфейс сервиса. В левой части окна представлен список файлов, в которых описывается модель решения задачи на языке NeMo+ [182], в центральной – та часть модели, которая содержится в выбранном файле, в правой – результаты вычислений.

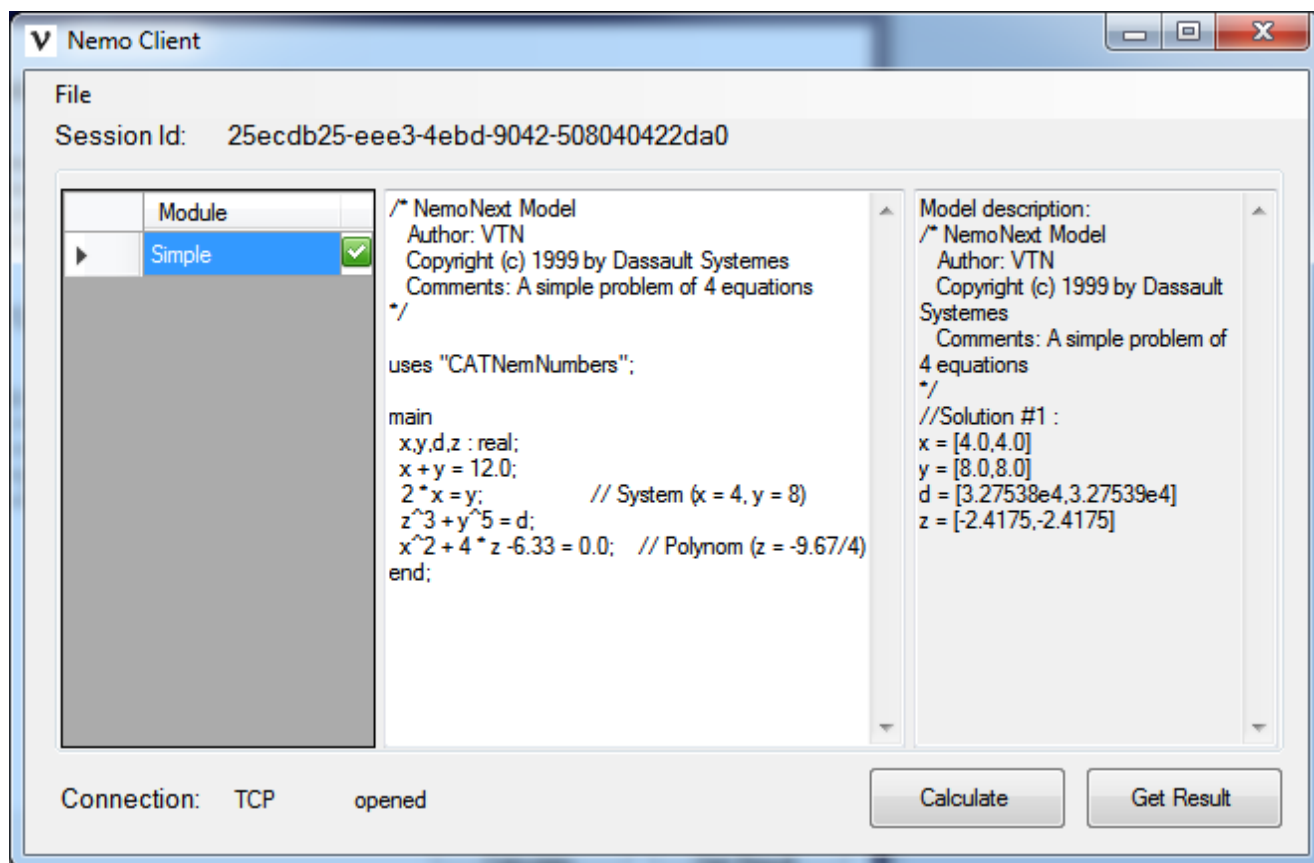


Рисунок 3.11. Клиентская часть сервиса

Приложение WebUnicalc (Рисунок 3.12) – еще один сервис, который позволяет описывать и запускать на счет недоопределенные модели. В отличие от SempNemo-сервиса данный инструмент не позволяет вводить пользовательские типы данных и описывать сложные недоопределенные модели. Однако, он более прост в использовании. Модель задачи, решаемой с помощью WebUnicalc, описывается в привычной математической нотации в виде набора переменных с интервальными значениями и связывающих их уравнений и неравенств. Реализация данного приложения выполнена на языке PHP.

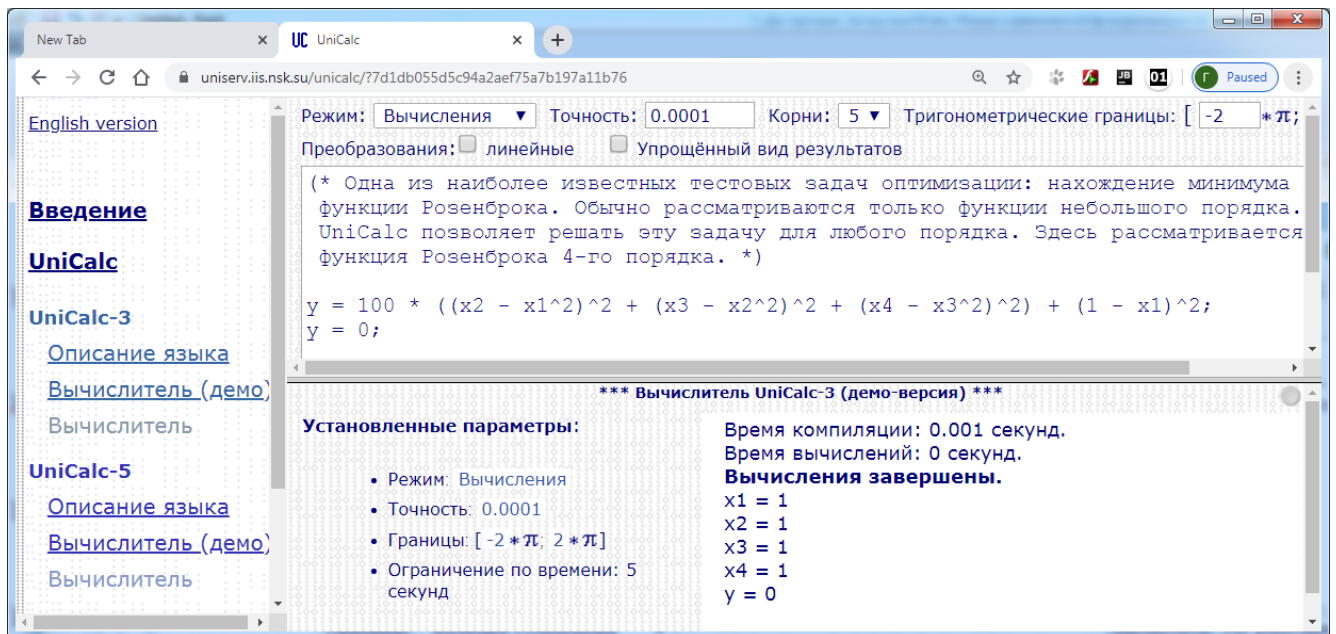


Рисунок 3.12. Приложение WebUnicalc

3.3.4. Рассуждения на основе правил

применяетсяВ (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ, ИСИ_CO_PAH)

реализует(РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ, SEMP_NEMO,

[ИНСТРУКЦИЯ: /rdms /uploads/bf/c3/20047fb20adf221095b7eea2c309.pdf

ПРИМЕРЫ_ИСПОЛЬЗОВАНИЯ:

/rdms /uploads/f1/af/386a9f2400167dc81077abb1265a.zip

КОНТАКТ_РАЗРАБОТЧИКА: zagor@iis.nsk.su])

реализует (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ, SEMP_TAO,

КОНТАКТ_РАЗРАБОТЧИКА: zagor@iis.nsk.su)

реализует (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ, CLIPS,

[DOWNLOAD: <https://sourceforge.net/projects/clipsrules/files/CLIPS/6.30/>

ПРИМЕРЫ_ИСПОЛЬЗОВАНИЯ: <https://sourceforge.net/projects/clipsrules/>])

решает_Метод_Задача (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ,

{ ДИАГНОСТИКА_ПРОБЛЕМНОЙ_СИТУАЦИИ,

ВЫЯВЛЕНИЕ_ПРОБЛЕМНОЙ_СИТУАЦИИ, РАЗРАБОТКА_АЛЬТЕРНАТИВ,

АНАЛИЗ_ТЕКУЩЕЙ_СИТУАЦИИ })

используетМетод (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ,

{ ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ })

используетИнформацию (РАССУЖДЕНИЯ_НА_ОСНОВЕ_ПРАВИЛ,

{ СЕМАНТИЧЕСКАЯ_СЕТЬ, СИСТЕМА_ПРОДУКЦИОННЫХ_ПРАВИЛ })

Метод рассуждений на основе правил реализован в технологическом комплексе Semp-ТАО [101]. Этот комплекс распространяется по ограниченной лицензии после связи с разработчиками. Для доступа к некоторым его возможностям можно использовать сервис SempNemo (п. 3.3.3). В серверной части, помимо настройки соединения с сервером, на котором развернут комплекс Semp-ТАО, можно задать метод, к которому будет предоставлен доступ. На Рисунке 3.13 показан пользовательский интерфейс сервиса. В левой части окна представлен список модулей, в которых описывается модель решения задачи на языке Semp-ТАО, в центральной – та часть модели, которая содержится в выбранном модуле, в правой – результаты вычислений.

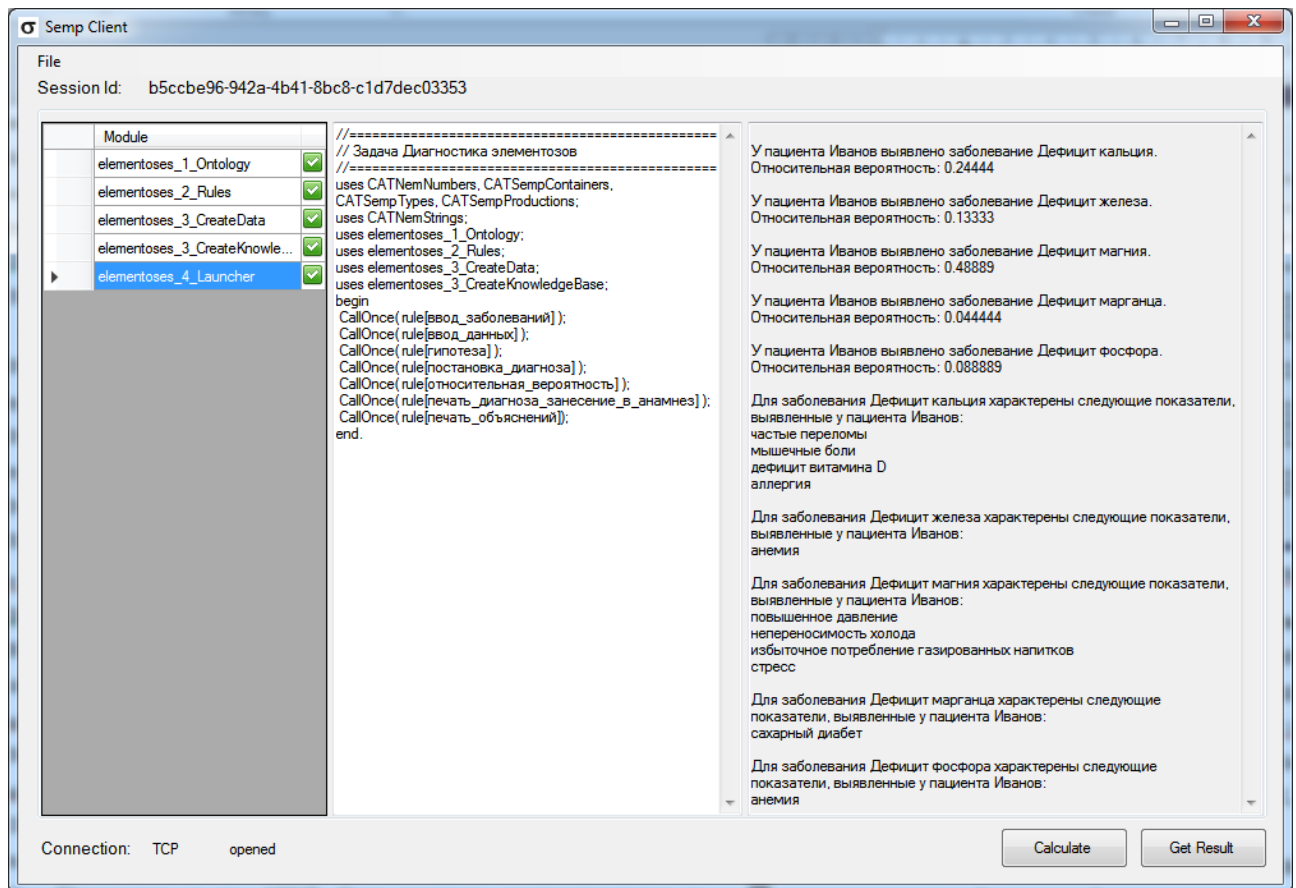


Рисунок 3.13. SempNemo – сервис

3.3.5. Метод онтологического моделирования

применяетсяВ (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ, ИСИ_СО_РАН, ИСЭМ_СО_РАН, УГАТУ, ИАПУ_ДВО_РАН, ШВМ, ИМ_СО_РАН, НГТУ, ИВМиМГ_СО_РАН, ИВТ_СО_РАН, ИФФиП_СО_РАН)

развивает (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ, { GRUBER_N, GUARINO_N, FENSEL_D, ГАВРИЛОВА_ТА, КЛЕЩЕВ_АС, АРТЕМЬЕВА_ИЛ, ШАЛФЕЕВА_ЕФ, ЛУКАШЕВИЧ_НВ, КУДРЯВЦЕВ_Д })

использует (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ, { МАССЕЛЬ_ЛВ, МАССЕЛЬ_АГ, КОПАЙГОРОДСКИЙ_АН, ВОРОЖЦОВА_ТН, ЗАГОРУЛЬКО_ГБ, ЗАГОРУЛЬКО_ЮА, АРТЕМЬЕВА_ИЛ, ШАЛФЕЕВА_ЕФ, ЛУКАШЕВИЧ_НВ, ГРИБОВА_ВВ, ЮСУПОВА_НИ, ЧЕРНЯХОВСКАЯ_ЛР, СМИРНОВ_СВ, БОРГЕСТ_НМ, МУРОМЦЕВ_ДИ, НЕВЗОРОВА_ОА })

реализует (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ, PROTEGE,
[СЕРВИС: <https://webprotege.stanford.edu/>,
DOWNLOAD: <https://protege.stanford.edu/download/protege/5.2/>])

реализует (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ, CMAP_TOOLS,
DOWNLOAD: <https://cmap.ihmc.us/cmaptools/cmaptools-download/>])

реализует (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ, GRM_ONTO_MAP,
[СЕРВИС: <http://repository.sei.irk.ru/rfbr160700569/app/grMontoMap.jnlp>,
КОНТАКТ_РАЗРАБОТЧИКА: lvmassel@gmail.com])

решает_Метод_Задача (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ,
{ ОБЕСПЕЧЕНИЕ_СИТУАЦИОННОЙ_ОСВЕДОМЛЕННОСТИ,
СТРУКТУРИЗАЦИЯ_ПРЕДМЕТНОЙ_ОБЛАСТИ,
СТРУКТУРИЗАЦИЯ_ПРОБЛЕМНОЙ_СИТУАЦИИ,
ПОСТРОЕНИЕ_ОНТОЛОГИИ_ПРЕДМЕТНОЙ_ОБЛАСТИ,
ВЫЯВЛЕНИЕ_ПРОБЛЕМНОЙ_СИТУАЦИИ, РАЗРАБОТКА_АЛЬТЕРНАТИВ,
АНАЛИЗ_ТЕКУЩЕЙ_СИТУАЦИИ })

используетИнформацию (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ
ОНТОЛОГИЯ_ПРЕДМЕТНОЙ_ОБЛАСТИ)

используетМетод (ОНТОЛОГИЧЕСКОЕ_МОДЕЛИРОВАНИЕ
ПОСТРОЕНИЕ_ИНТЕЛЛЕКТ_КАРТ)

3.3.6. Когнитивное моделирование

имеетАвтораПерсону (МЕТОД_НЕДООПРЕДЕЛЕННЫХ_ВЫЧИСЛЕНИЙ, AXELROD_R)

применяетсяВ (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, ИСИ_СО_РАН, ИСЭМ_СО_РАН, УГАТУ, ИПУ)

развивает (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, { МАССЕЛЬ_АГ, ЗАГОРУЛЬКО_ГБ, КУЛИНИЧ_АА }),

использует (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, { МАССЕЛЬ_ЛВ, МАССЕЛЬ_АГ, ЗАГОРУЛЬКО_ГБ, ЧЕРНЯХОВСКАЯ_ЛР })

реализует (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, КАНВА,

[КОНТАКТ_РАЗРАБОТЧИКА: kulinich@ipu.rssi.ru

DOWNLOAD: <http://www.raai.org/razrabotki/rgupiim/imlist.html>])

реализует (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, GR_COGNITIVE_MAP,

[СЕРВИС: <http://repository.sei.irk.ru/rfbr160700569/app/grMCogMap.jnlp>,

КОНТАКТ_РАЗРАБОТЧИКА: lvmassel@gmail.com])

реализует (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, КО_МОД,

[СЕРВИС: [http:// uniserv.iis.nsk.su/komod](http://uniserv.iis.nsk.su/komod)

КОНТАКТ_РАЗРАБОТЧИКА: gal@iis.nsk.su])

использует Представление Информации (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, ГРАФ)

использует Метод (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, { АНАЛИЗ_ПУТЕЙ,

АНАЛИЗ_ЦИКЛОВ, АНАЛИЗ_СВЯЗНОСТИ, ИМПУЛЬСНОЕ_МОДЕЛИРОВАНИЕ,

СЦЕНАРНЫЙ_АНАЛИЗ, АНАЛИЗ_УСТОЙЧИВОСТИ })

решает_Метод_Задача (КОГНИТИВНОЕ_МОДЕЛИРОВАНИЕ, {

СТРУКТУРИРОВАНИЕ_ПРЕДМЕТНОЙ_ОБЛАСТИ,

СТРУКТУРИРОВАНИЕ_ПРОБЛЕМНОЙ_СИТУАЦИИ,

АНАЛИЗ_ПРОБЛЕМНОЙ_СИТУАЦИИ, ПРОГНОЗИРОВАНИЕ,

РАЗРАБОТКА_АЛЬТЕРНАТИВ, РАЗРАБОТКА_ТЕКУЩЕГО_ПЛАНА })

Для реализации метода когнитивного моделирования было разработано Web-приложение «Когнитивное моделирование» (Ко_Мод) (Рисунок 3.14).

Для построения когнитивных карт был использован сервис grMCogMap [131] (продукт ИСЭМ СО РАН), позволяющий сохранять карту в XML-формате. Был написан парсер, преобразующий XML-представление в матрицу. Для анализа когнитивных карт был использован сервис, реализующий методы обработки с матриц (п.3.3.7). Разработка приложения выполнялась на языке PHP.

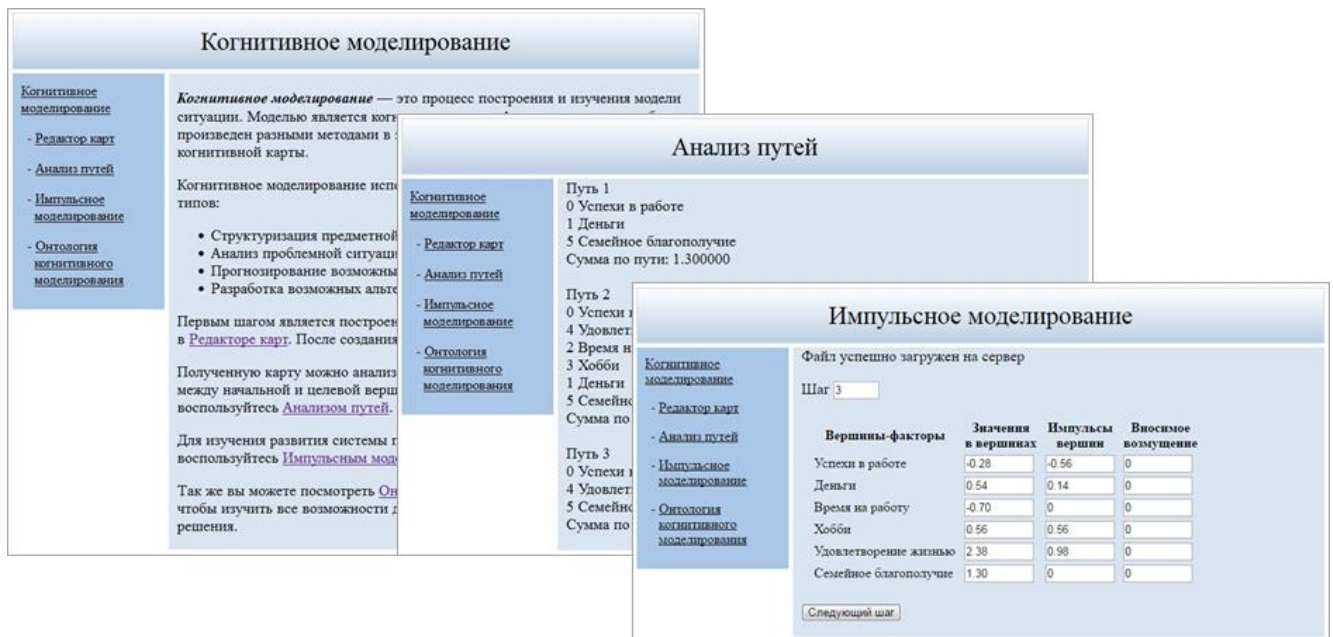


Рисунок 3.14. Web-приложение Ко_Мод

3.3.7. Сервис обработки массивов данных

Сервис обработки массивов данных предназначен для реализации методов, использующихся как вспомогательные для тех методов принятия решений, в которых данные представляются в матричном или векторном виде.

имеетАвтораПерсону (СЕРВИС_ОБРАБОТКИ_МАССИВОВ_ДАННЫХ, ЗАГОРУЛЬКО_ГБ)

применяетсяВ (СЕРВИС_ОБРАБОТКИ_МАССИВОВ_ДАННЫХ, ИСИ_СО_РАН, ИСЭМ_СО_РАН)

реализуетМетод(СЕРВИС_ОБРАБОТКИ_МАССИВОВ_ДАННЫХ,

{ НОРМИРОВАНИЕ, АДДИТИВНАЯ_СВЕРТКА,
МУЛЬТИПЛИКАТИВНАЯ_СВЕРТКА,
ВЫЧИСЛЕНИЕ_ДЕКАРТОВА_РАССТОЯНИЯ, АНАЛИЗ_ПУТЕЙ,
АНАЛИЗ_ЦИКЛОВ, АНАЛИЗ_СВЯЗНОСТИ})

Реализация сервиса выполнена на языке C++ с использованием протоколов SOAP/WSDL.

3.3.8. Сбор и анализ мнений экспертов

Любой метод экспертного оценивания предполагает организацию сбора данных от эксперта. Для реализации этой процедуры в репозиторий был включен

сервис google-forms [31]. На Рисунке 3.15 показан пример его использования для составления анкеты. С помощью данного сервиса рабочая группа может создать анкету, которую в дальнейшем эксперты будут заполнять количественными и качественными данными. Сервис google-forms предоставляет возможность рассылки анкет экспертам и получения от них заполненных анкет. Для обработки результатов, которые google-forms представляет в Excel-таблице, создан специальный сервис, позволяющий считывать данные в матрицы и применять к ним методы, рассмотренные в п. 3.3.7.

Рисунок 3.15. Сервис Google Forms

3.3.9. Методы экспертного оценивания

Методы экспертного оценивания – группа методов организации работы со специалистами-экспертами для решения основных задач принятия решений – формирования, оценки, выбора, ранжирования альтернатив решения проблемной ситуации. Одним из наиболее используемых МЭО является метод анализа иерархий Саати [156].

имеетАвтораПерсону ((МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, СААТИ_ТЛ)
развивает (МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, {СААТИ_ТЛ, СЕРЕДЕНКО_НН,
КРАВЧЕНКО_ТК, ФУРЦЕВ_ДГ }),

использует (МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, { ЗАГОРУЛЬКО_ГБ, СЕРЕДЕНКО_НН, КРАВЧЕНКО_ТК, РАЕВСКАЯ_ЕА, ЛОМОВ_ЭО })

реализует(МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, МЭО,

[СЕРВИС: <http://uniserv.iis.nsk.su/meo>

КОНТАКТ_РАЗРАБОТЧИКА: lvmassel@gmail.com])

решает (МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, ОЦЕНКА_АЛЬТЕРНАТИВ)

используетИнформацию (МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, ПРИЗНАКОВОЕ_ПРОСТРАНСТВО)

используетПредставлениеИнформации (МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, МАТРИЦА)

используетМетод (МЕТОД_АНАЛИЗА_ИЕРАРХИЙ, { АНКЕТИРОВАНИЕ, ПОПАРНОЕ_СРАВНЕНИЕ, НОРМИРОВАНИЕ_МАТРИЦЫ, АДДИТИВНАЯ_СВЕРТКА, МУЛЬТИПРИКАТИВНАЯ_СВЕРТКА })

Для доступа к описаниям МЭО было разработано Web-приложение «Методы экспертного оценивания» (Рисунок 3.16).

Реализация метода Саати:
Для реализации данного метода лицу принимающему решение(ЛПР) нужно создать анкету с оцениванием характеристик. Для этого предлагается воспользоваться сервисом гугл google-forms. Плюсом данного сервиса является то, что анкетирование можно производить на расстоянии, отправив анкету по почте, а ответы сразу же появятся перед ЛПР-отправителем.

Для создания google-form нужно:

1. Зайти на Google-disk (нужно быть зарегистрированным на почте google). Прейти на сайт drive.google.com
2. Нажать кнопку "Мой диск" и выбрать "google-формы"
3. Создать анкету для опроса, в которой:
 - 3.1. Сначала идут разделы с точными(количественными) характеристиками
 - 3.2. Затем идут разделы с относительными(качественными - субъективное мнение ЛПР или эксперта заполняющего анкету) характеристиками
 - 3.3. Затем данные о важности одной характеристики относительно другой.
4. Ответить на анкету или отправить ее Эксперту для заполнения
5. Получить ответы зайдя во вкладку "Ответы"
6. Сохранить таблицу с ответами, для этого нажать "Просмотр ответов в таблицах", сохранить таблицу.

Для обработки полученных данных нужно:

1. Загрузить полученную таблицу excel в поле указанное ниже.
2. Указать количество альтернатив(вариантов элементов выбора)
3. Указать количество характеристик по которым сравниваются альтернативы

Введите таблицу: Файл не выбран

Введите количество альтернатив:

Введите количество характеристик: - точных

- относительных

Результат: 3-я альтернатива - 0.2545340939210943

Рисунок 3.16. Web-приложение МЭО

Помимо к информации о МЭО данное Web-приложение предоставляет доступ к реализации метода анализа иерархий.

Разработка приложения выполнена на языке PHP. При программной реализации МЭО используется сервис GoogleForms.

3.4. Разработка сервисов, реализующих взаимодействие с внешними хранилищами данных

3.4.1. Архитектура системы доступа к внешним данным и схема её функционирования

Для доступа к внешним данным была разработана система [181,193], предоставляющая пользователям ИНИР следующие функциональные возможности:

1. Организация взаимодействия с внешними источниками данных. Такими источниками могут быть сторонние БД, либо БД, созданные разработчиками конкретных ИНИР.
2. Описание информационных объектов ИНИР, включающих значения атрибутов из внешних БД.
3. Импорт значений свойств указанных объектов из внешних источников. Визуализация значений свойств объектов в виде таблиц или графиков.
4. Запуск сервисов анализа импортированных данных.
5. Использование импортированных данных для решения задач.

На Рисунке 3.17 представлены архитектурные компоненты системы доступа к данным и схема их взаимодействия. Основным компонентом данной системы является сервис загрузки данных – **Загрузчик**, осуществляющий непосредственное взаимодействие с внешними хранилищами. Для подключения к системе конкретных БД служит **Панель администратора**, имеющая пользовательский Web-интерфейс, который позволяет зарегистрировать новые источники данных и сформировать шаблоны запросов для доступа к ним. Загрузчик имеет собственную БД, которая содержит адреса зарегистрированных в нем БД и информацию, требующуюся для построения запроса к конкретному ресурсу. Здесь могут использоваться как шаблоны SQL запросов к реляционным БД, так и иные форматы запросов (REST API, SOAP, SPARQL и другие) к

внешним ресурсам. Для построения конкретных запросов Загрузчику передаются необходимые параметры, которые извлекаются из онтологии ИНИР.

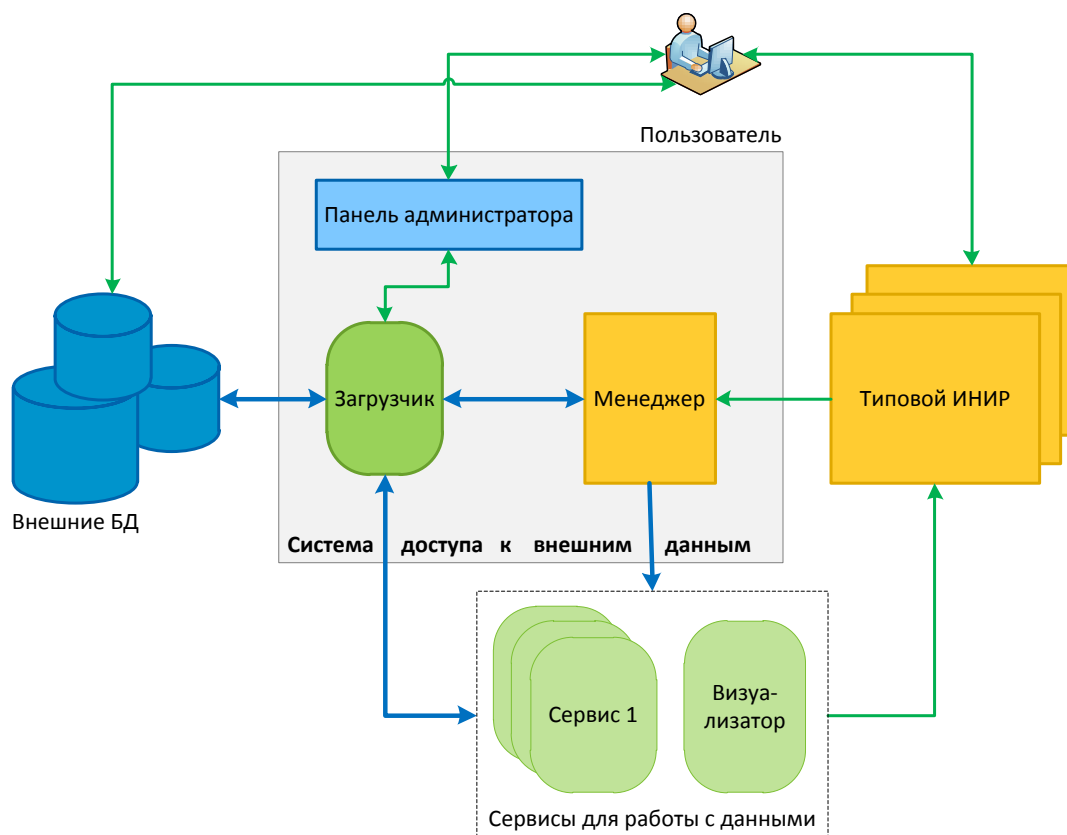


Рисунок. 3.17. Архитектура системы доступа к данным из внешних источников

Сервисы для работы с данными позволяют показывать их пользователю, выполнять их анализ или использовать для решения задач ОЗ ресурса.

Для организации взаимодействия ИНИР с Загрузчиком и Сервисами для работы с данными был разработан специальный плагин – **Менеджер**. Данный плагин предназначен для извлечения из онтологии ИНИР параметров, необходимых Загрузчику для построения запроса к внешней БД. Менеджер передает Загрузчику параметры, получает от него идентификатор запроса, который затем передает необходимому Сервису.

Рассмотрим **схему функционирования системы доступа к внешним данным**. Для того чтобы иметь возможность использовать в ИНИР внешние данные, необходимо предварительно через Панель администратора зарегистрировать в Загрузчике БД и шаблон запроса, делающего выборку

необходимых данных. При этом каждой тройке (шаблон, строка подключения, тип БД) присваивается уникальный идентификатор, который сообщается инженеру знаний. Инженер знаний должен определить в онтологии ОЗ класс объектов, свойства которых будут принимать значения из внешней БД, и связать с этим классом полученный идентификатор. Кроме того, он должен позаботиться о том, чтобы связать свойства таких объектов с параметрами шаблона запроса к внешней БД (их именами и порядком следования в шаблоне). Способ и средства осуществления этих предварительных действий описаны ниже.

При отображении в ИНИР свойств объектов, принадлежащих таким классам, появляется атрибут «Значение» с гиперссылкой «>>>». Переход по этой ссылке инициирует работу Менеджера, которому передается имя отображаемого объекта. По имени объекта Менеджер делает запрос к онтологии ИНИР, результатом которого будут связанный с классом данного объекта идентификатор и набор параметров шаблона запроса с их именами, значениями и порядком вхождения в запрос. Идентификатор и набор параметров передаются Менеджером Загрузчику. Загрузчик записывает эту информацию во временную кэш-память и также присваивает каждому такому новому запросу уникальный идентификатор. В зависимости от того, в какой момент работы ИНИР и для каких целей вызывался Менеджер, он передает идентификатор запроса определенному Сервису, способному обратиться к Загрузчику, и, предъявив идентификатор запроса, получить и обработать данные, полученные от Загрузчика.

3.4.2. Особенности реализации взаимодействия интеллектуальных научных интернет-ресурсов и системы доступа к внешним данным

Рассмотрим методику обеспечения связи классов объектов и их свойств с параметрами запроса к внешним БД на примере онтологии теплофизических свойств химических веществ, на основе которой был построен одноименный информационно-аналитический интернет-ресурс [172]. Был разработан паттерн онтологического проектирования (Рисунок 3.18), который относится к группе паттернов визуализации [191].

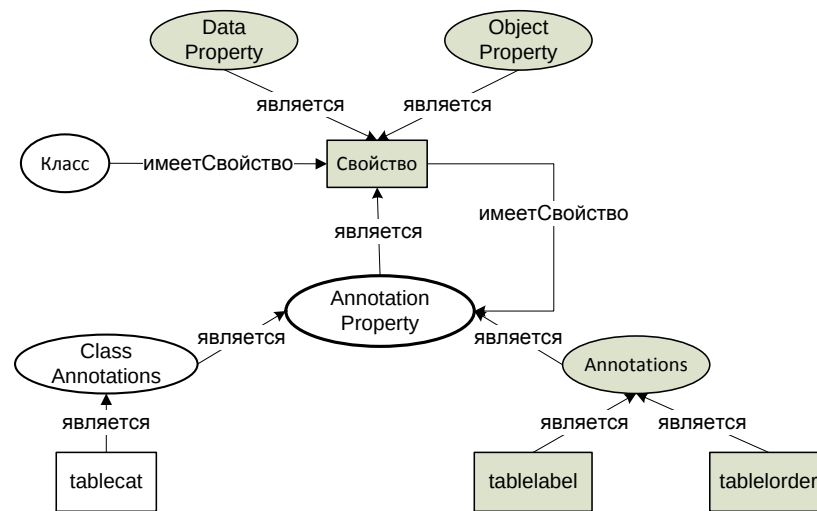


Рисунок. 3.18. Паттерн для установления связи класса и свойств объектов с параметрами запроса

Для задания связи объектов со значениями из внешней БД классу, к которому относятся эти объекты, приписывается аннотационное свойство `tablecat`, значением которого является уникальный идентификатор, сообщаемый инженеру знаний после регистрации в Загрузчике базы данных и шаблона запроса к ней. На Рисунке 3.19 показано, как в редакторе Protégé задается значение этого свойства для класса «Зависимость», который входит в состав онтологии теплофизических свойств химических веществ.

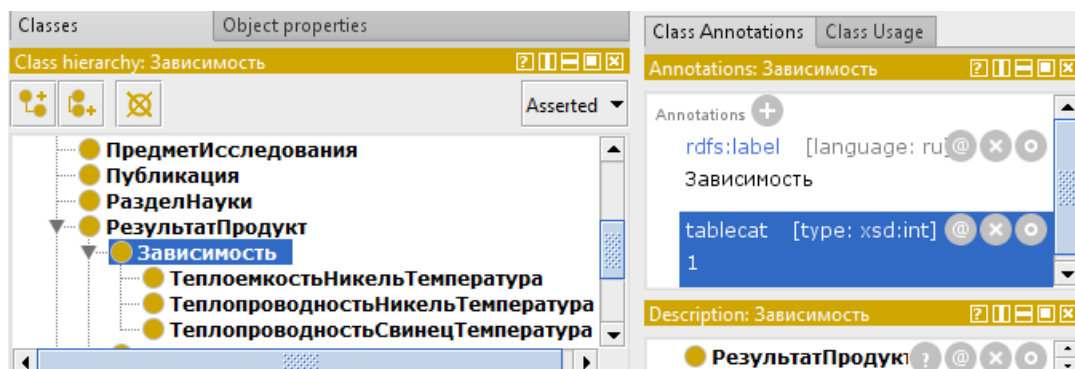


Рисунок. 3.19. Задание уникального идентификатора для класса объектов с данными из внешних источников

Тем свойствам объектов данного класса, которые входят в шаблон запроса, приписываются аннотационные свойства `tablelabel` и `tableorder`, задающие, соответственно, имя, под которым свойство объекта входит в шаблон запроса, и

порядок вхождения (Рисунок 3.20). Значение этого свойства извлекается из онтологии для каждого конкретного объекта.

Для класса «Зависимость» такими свойствами будут материал, для которого получена зависимость, фамилия получившего её автора, год получения и название публикации, в которой она была впервые описана. Так, год получения какой-либо зависимости теплофизических параметров будет входить в запрос под именем «year» и под номером 3 (Рисунок 3.20).

Использование сервис-ориентированного подхода, а также наличие Менеджера, управляющего взаимодействием сервисов, позволяют легко масштабировать систему доступа к внешним данным, наращивать её функциональные возможности, не производя изменений в её коде.

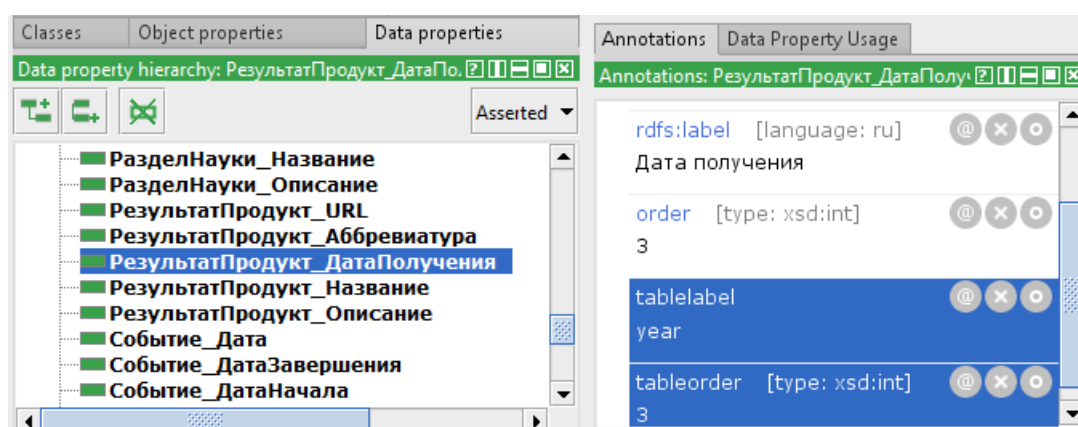


Рисунок. 3.20. Задание имен и порядка вхождения параметров шаблона запроса

В качестве протокола общения между компонентами системы была использована библиотека protobuf [52] компании Google (п. 1.6.4). Это позволило централизованно управлять протоколом общения компонентов, написанных на разных языках, а также оптимизировать трафик за счёт сжатия передаваемых данных.

Менеджер реализован на языке PHP с использованием пакета EasyRDF [17] для формирования запросов к онтологии.

3.5. Выводы

- Выполнена реализация средств КПР. Разработаны онтология ППР, ИАИР ППР, репозиторий методов ППР.
- Важным этапом разработки репозитория методов ППР является создание информационных объектов, описывающих методы и реализующие их программные разработки.
- Разработаны сервисы, реализующие ряд интеллектуальных методов ППР. Для разработки сервисов было использовано несколько подходов: разработка с использованием протоколов SOAP/WSDL; разработка с использованием технологии Java Web Start (JavaWS), основанной на протоколе Java Network Launching Protocol (JNLP); разработка сервиса как web-приложения, предоставляющего доступ к описанию метода и к его реализации.
- Разработан Менеджер, осуществляющий взаимодействие системы доступа к внешним данным и интеллектуальных научных интернет-ресурсов.
- Использование предлагаемых подходов и технологий при реализации средств КПР обеспечивает соблюдение принципов, которые должны, согласно модели КПР, закладываться в основу разрабатываемых этими средствами ИСППР.

Глава 4. Использование предлагаемых средств комплексной поддержки разработки интеллектуальных СППР

Предлагаемые средства КПР ИСППР [196] использовались при разработке ИСППР в технической и медицинской областях. В частности, с помощью этих средств были созданы две ИСППР: «Угрозы энергетической безопасности» [176] и «Элементозы» [194,195]. Разработка этих ИСППР выполнялась в соответствии с алгоритмом, описанным в п.2.7.2. Рассмотрим назначение и функциональные возможности этих систем и действия, которые выполнялись на каждом этапе их разработки.

4.1. Интеллектуальная СППР по угрозам энергетической безопасности

4.1.1. Этап Идентификации

Целью разработки ИСППР по угрозам энергетической безопасности (ЭнБ) является обеспечение достижения цели самой ЭнБ – «Обеспечение обоснованных потребностей потребителей в энергии экономически доступными для них энергетическими ресурсами приемлемого качества» [97].

Для достижения этой цели ИСППР должна решать следующие **задачи**:

- 1) Решение задач энергетической безопасности:
 - а) Обеспечение в полном объеме потребностей в энергии в нормальных условиях.
 - б) Обеспечение гарантированного минимального объема потребностей в энергии в экстремальных условиях.

- с) Разработка и реализация мероприятий, повышающих защищенность потребителей энергии от возникновения вероятных угроз нарушения бесперебойного энергоснабжения.
 - д) Разработка и реализация мероприятий, снижающих влияние угроз нарушения бесперебойного энергоснабжения в случае их наступления.
- 2) Решение конкретных задач по предотвращению угроз ЭНБ и ликвидации их последствий.

В настоящий момент рассматриваемая ИСППР решает две конкретные задачи: «Угроза похолодания» и «Авария у производителя энергоресурса». Подробно задачи и их аналитические модели описаны в публикациях [106,154] соответственно.

В первой задаче моделируется ситуация, когда из-за снижения температуры наружного воздуха относительно среднесезонной в определенные месяцы отопительного периода повышается потребление тепло- и электроэнергии. Модель задачи отражает аналитические зависимости между этими параметрами и позволяет оценить, насколько нужно увеличить выработку указанных видов энергии, чтобы не допустить ее дефицита.

Во второй задаче моделируется реакция сети энергоснабжения на угрозы технологического типа. В случае аварийной ситуации у какого-либо производителя подключаются пункты резервного хранения энергоресурсов. Суммарное содержание ресурса в таких пунктах должно компенсировать его сокращение у производителей в результате аварии. Любой потребитель может получать ресурс из любого пункта резервного хранения, т.е. между ними потенциально можно установить связь. В модели задачи задаются веса таких связей. Цель решаемой задачи – оптимизировать потоки ресурсов из резервных хранилищ потребителям с учетом весов их связей.

Назначением ИСППР по угрозам ЭНБ является (1) предоставление информации об угрозах ЭНБ и о деятельности по ее обеспечению и (2) решение конкретных практических задач для определения набора действий, необходимых для предотвращения угроз или минимизации ущерба в случае их реализации.

Основными **пользователями** данной ИСППР могут быть лица, принимающие решения в области энергетики на всех звеньях управленческой цепи. Кроме того, она может быть интересна обучающимся по различным направлениям энергетики и ЭнБ.

В качестве **экспертов** по ЭнБ были привлечены сотрудники лаборатории информационных систем в энергетике №34 ИСЭМ СО РАН.

4.1.2. Этап Концептуализации

Для **сбора информации** об угрозах ЭнБ и о решаемых в этой области задачах были использованы следующие публикации: [97,106,130,132,136,153,154]. На основе этих публикаций, а также непосредственного общения с экспертами и предоставленных ими концептуальных карт (Рисунки 4.1, 4.2) был выполнен анализ предметной области и определены ее границы. При этом были введены такие базовые концепты, как цели, задачи ЭнБ, объект, предмет, субъект ЭнБ (Рисунок 4.3).

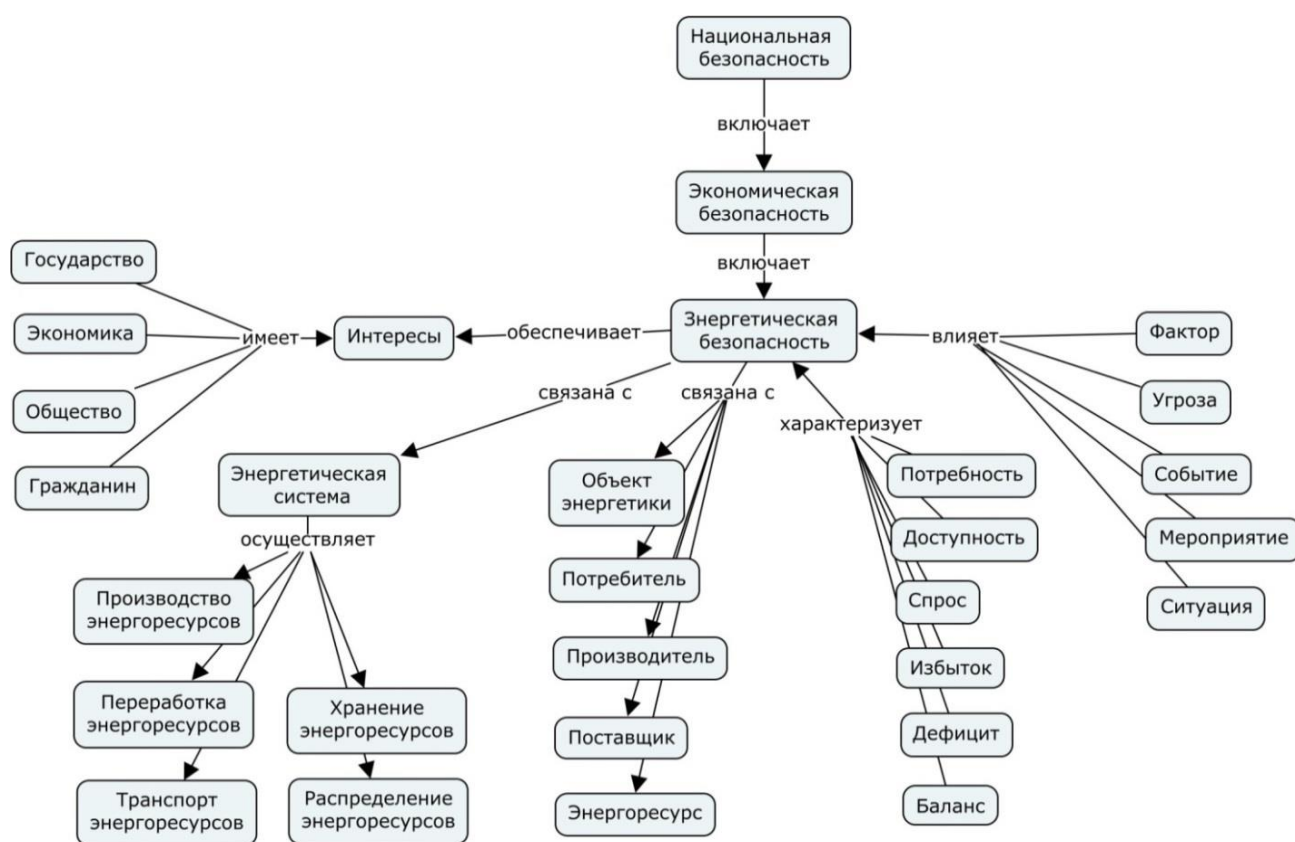


Рисунок 4.1. Основные понятия энергетической безопасности

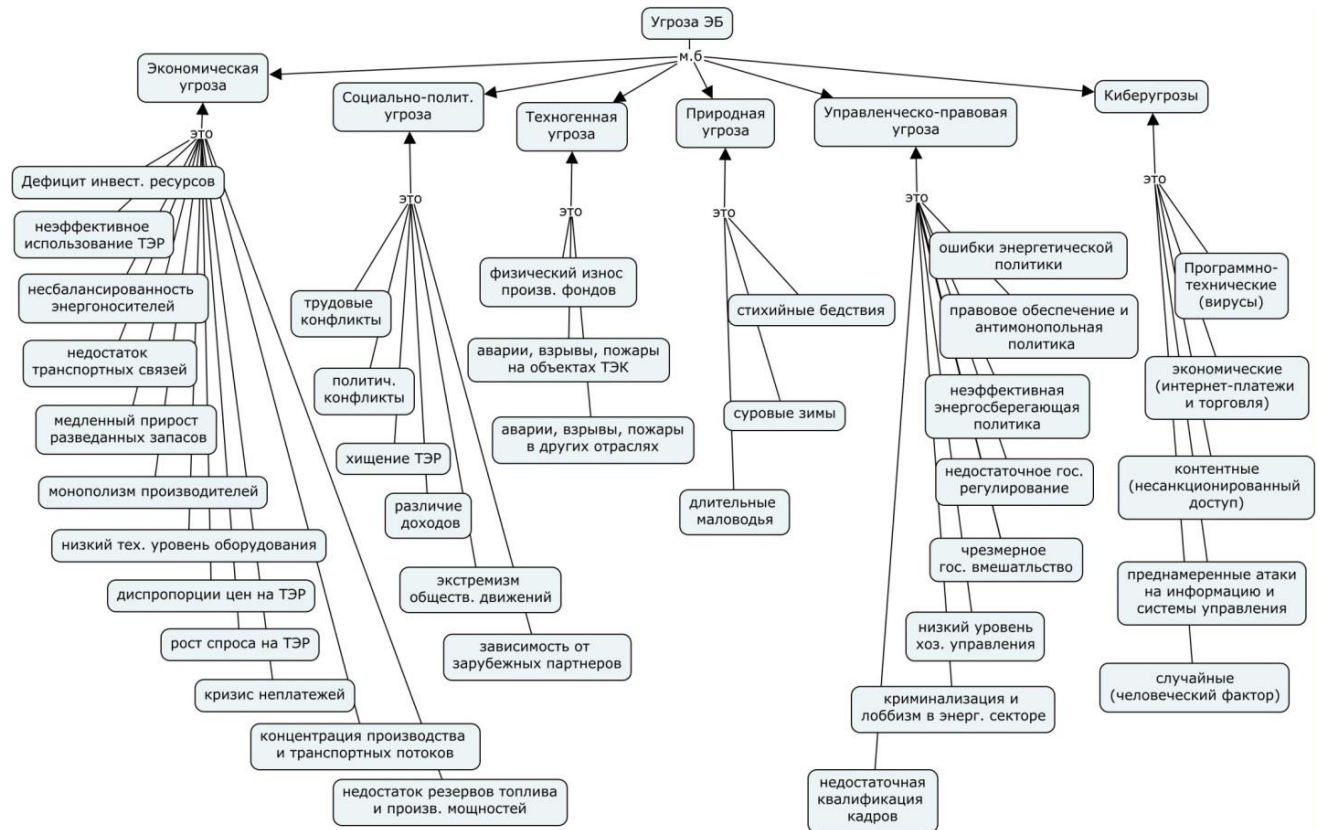


Рисунок 4.2. Иерархия угроз энергетической безопасности

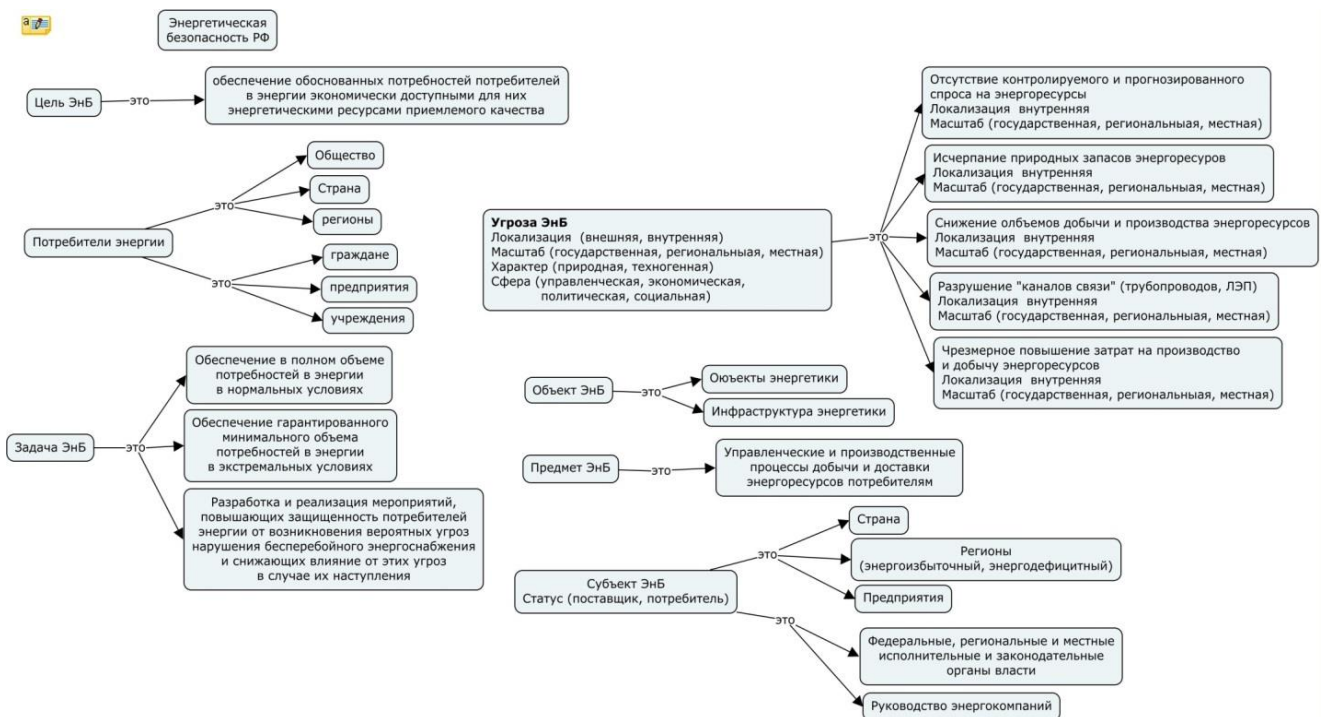


Рисунок 4.3. Базовые концепты энергетической безопасности

Поскольку конкретные задачи ИСППР имели аналитические модели, для их решения был **выбран метод** недоопределенных вычислений (МНВ).

Достоинством этого метода является его способность работать с не полностью определенными числовыми данными, значения которых могут представляться в модели интервалами. Еще одним существенным преимуществом является отсутствие деления параметров задачи на входные и выходные. Это позволяет на одной и той же модели решать как прямые, так и обратные задачи.

ИСППР по угрозам ЭНБ работает с разнородной информацией и использует разные **средства представления знаний**. Знания об угрозах ЭНБ представляются в виде онтологии. МНВ использует внутреннее представление знаний в виде функциональной сети. Пользователь представляет задачу в привычной математической нотации с интервальными значениями объектов модели.

Для **интерпретации знаний и решения задач** были применены средства технологии Semantic Web [36] и решатель UniCalc [114].

При разработке онтологии угроз ЭНБ в качестве **базовой онтологии** использовалась онтология научного знания [102], а также фрагменты онтологии ППР.

4.1.3. Этап Формализации

На данном этапе разрабатывались модели решения конкретных задач. Ниже представлены фрагменты этих моделей на языке, интерпретируемом решателем UniCalc:

(* Модель задачи «Угроза похолодания» *)

```
//среднемесячная наружная температура в Новосибирске по месяцам, гр. С
tsr[1] = -15.500; tsr[2] = -15.5; tsr[3] = -8; tsr[4] = 2; tsr[5] = 11; tsr[6] = 16.5;
tsr[7] = 18.5; tsr[8] = 16; tsr[9] = 10; tsr[10] = 2; tsr[11] = -8.5; tsr[12] = -15;
tvn = 18; // расчетная Т воздуха отапливаемых помещений, гр. С
i = 1; // рассматриваемый месяц
dt[i] = 10; //отклонение среднемесячной наружной Т в i-ом месяце, гр. С
bnt = 22000; //нормальное потребление т/э, тыс. ГКал
dbt = [-20, 20]; //изменения потребления т/э %
dtk = 1920; //увеличение выработки тепла на котельных, тыс. ГКал
dtt = [0,5000]; //увеличение выработки тепла на ТЭЦ, тыс. ГКал
```

```

bdeft = 0; //дефицит т/э, тыс. ГКал
bne =19000; //нормальное потребление э/э, млн кВт*ч
dbe= [-10, 10]; //изменения потребления э/э %
det = [0,2500]; //увеличение выработки э/э на ТЭС, млн кВт*ч
bdefe = 0; //дефицит э/э, млн кВт*ч
dbt= dt[i]/( tvn- tsr[i]); //Изменения в отпускаемой т/э в случае похолодания
dbe=0.3*dbt; //изменение потребления э/э
bdeft= bnt* dbt - dtk - dtt; // Дефицит т/э
bdefe= bne* dbe - det; //Дефицит э/э

```

(* Модель задачи расчета в энергосистеме потоков энергоресурсов из резервных хранилищ после аварии у некоторых производителей *)

```

M=4; //Количество производителей
a[1]=10000; a[2]=8500; a[3]=6800; a[4]=9700; //Объемы производства э/р
sumA=sum( i = 1, 1, M; a[i] );
N=10; //Количество потребителей
//Объемы потребления э/р, тыс кВт*ч
b[1]=1300; b[2]=800; b[3]=7300; b[4]=6000; b[5]=4800; b[6]=5400; b[7]=2200;
b[8]=2800; b[9]=3900; b[10]=[0,10000];
sumB=sum( i = 1, 1, N; b[i] );
sumA=sumB;
//Объемы э/р, поставляемых потребителям. Незаданные значения равны 0
ix[1,12]=[0,a[2]]; ix[2,11]=[0,a[1]]; ix[3,4]=[0,10000]; ix[3,10]=2000; ix[3,14]=1000;
ix[4,11]=[0,a[1]]; ix[4,12]=25; ix[4,13]=[0,a[3]]; ix[5,12]=15; ix[5,13]=3300;
ix[6,8]=1200; ix[6,13]=1000; ix[6,14]=3200; ix[7,14]=2200; ix[8,13]=700;
ix[8,14]=[0,a[4]]; ix[9,12]=3900, ix[10,11]=[0,a[1]];
//Вероятности аварии у производителей ресурсов
ip[1] = 1; ip[2] = 0; ip[3] = 0; ip[4] = 0;
T = 3; //Количество резервных хранилищ
//Объемы энергоресурсов в резервных хранилищах, тыс кВт*ч
iu[1]=[0, 10000]; iu[2]=[0, 10000]; iu[3]=[0, 10000];
//Балансы с учетом вероятности аварии.

```

```

for ( j = 1, 1, M; sum( i= 1, 1, N; ix[i,j+N]) = ( 1-ip[j] ) *a[j] );
//Веса связей между потребителями, поставщиками и п. резервного хранения
if[1,1]=1; if[1,2]=4; if[1,3]=4; if[2,1]=2; if[2,2]=5; if[2,3]=3; if[3,1]=2; if[3,2]=3;
if[3,3]=1; if[4,1]=1; if[4,2]=2; if[4,3]=3; if[5,1]=3; if[5,2]=1; if[5,3]=4; if[6,1]=4;
if[6,2]=1; if[6,3]=3; if[7,1]=4; if[7,2]=2; if[7,3]=1; if[8,1]=4; if[8,2]=2; if[8,3]=3;
if[9,1]=1; if[9,2]=4; if[9,3]=5; if[10,1]=3; if[10,2]=5; if[10,3]=2;
// iy[i] - поток ресурса по резервной дуге, тыс кВт*ч *)
for ( i = 1, 1, N;
    for( j = 1, 1, T; iy[i,j] = [0,10000] ) );
sum( i= 1, 1, T; iu[i] ) = sum( j = 1, 1, M; ip[j] *a[j] );
//Суммарный поток энергоресурса, исходящий из резервного хранилища
for ( i = 1, 1, T; sum( j = 1, 1, N; iy[j,i] ) = iu[i] );
//Флажок. 1-есть поток ресурса по дуге, 0-нет потока по дуге
for ( i = 1, 1, N;
    for( j = 1, 1, T; iz[i,j] = [0,1] ) );
for ( i = 1, 1, N;
    for( j = 1, 1, T; iy[i,j]=0 -> iz[i,j] = 0 ) );
for ( i = 1, 1, N;
    for( j = 1, 1, T; iy[i,j]>0 -> iz[i,j] = 1 ) );
//Основной баланс. Суммарный объем потребляемых ресурсов и
//перепоставленных другим потребителям равен суммарному объему //ресурсов,
полученных от производителей и из пунктов резервного хранения
for ( i = 1, 1, N;
    b[i] + sum( j = 1, 1, N; ix[j,i] ) =
    sum( j = 1, 1, N+M; ix[i,j] ) + sum( j = 1, 1, T; iy[i,j] ) );
//Целевая функция
igoal= sum( i= 1, 1, N; sum( j= 1, 1, T; if[i,j]*iz[i,j] ) );
igoal>=0;

```

Для решения данных задач был **спроектирован сервис пользовательского интерфейса**, который позволяет пользователю выбирать задачу, вводить данные, запускать модель задачи на решателе UniCalc и получать результаты вычислений.

4.1.4. Этап Реализации компонентов интеллектуальной СППР

На основе проведенного анализа ПО «Угрозы Энергетической безопасности» была построена ее онтология. Были конкретизированы такие понятия базовой онтологии, как Цель, Задача, Объект, Предмет ЭНБ, введены описания конкретных решаемых задач, введены понятия, специфические для рассматриваемой ПО.

Из онтологии ППР был выделен фрагмент, описывающий МНВ, и склеен с онтологией угроз ЭНБ. На Рисунке 4.4 представлены окно редактора Protégé и фрагмент разработанной в нем онтологии угроз ЭНБ.

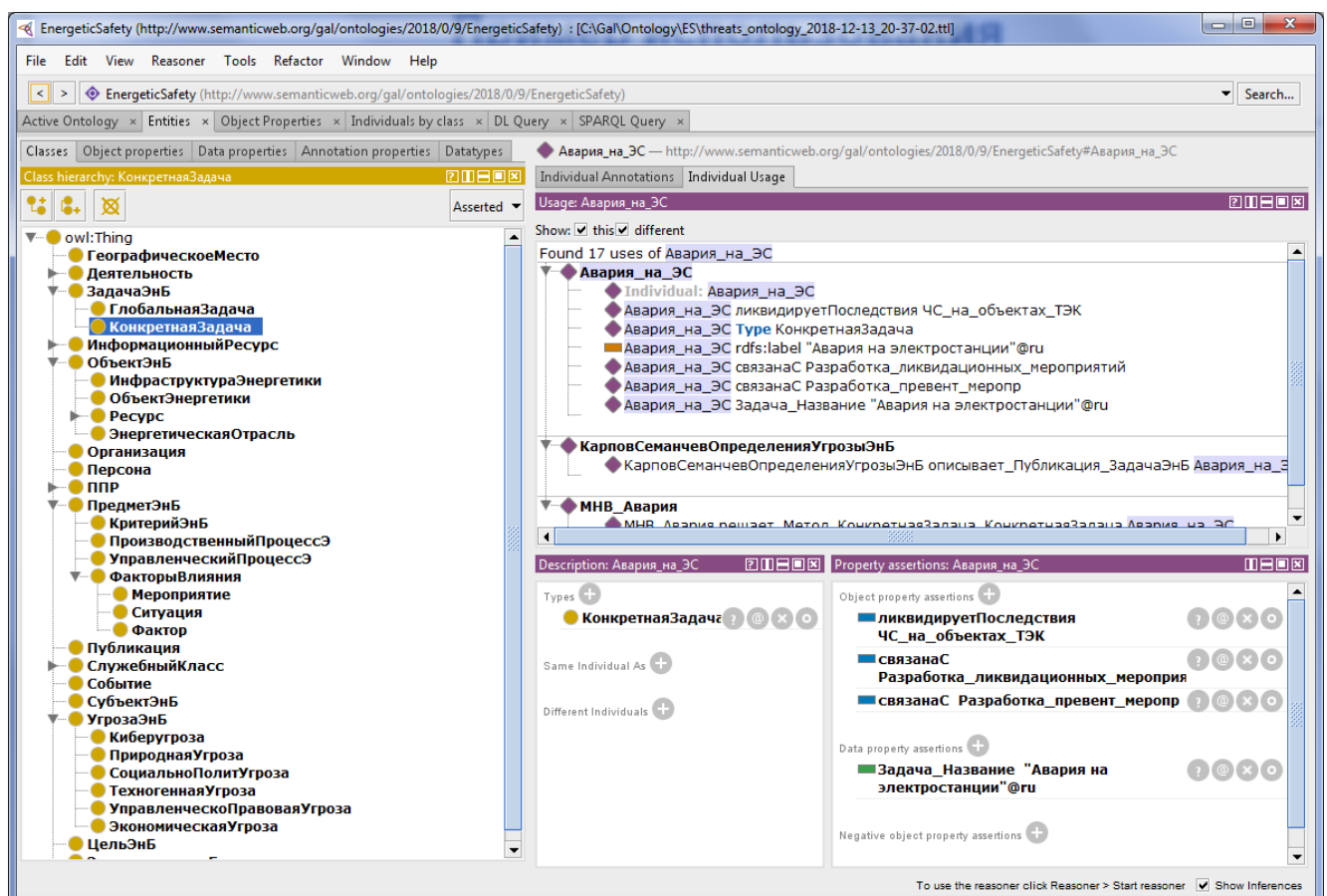


Рисунок 4.4. Разработка онтологии энергетической безопасности средствами редактора Protégé

Сервис пользовательского интерфейса был реализован с использованием фреймворка для разработки графических приложений Qt [7]. Запуск решателя UniCalc выполняется посредством его программного интерфейса.

4.1.5. Этап Реализации интеллектуальной СППР

Разработанная онтология угроз ЭНБ была загружена в копию оболочки ИНИР. На Рисунке 4.5 показана страница получившегося в результате этого действия ИАИР. Этот ресурс является каркасом разрабатываемой ИСППР.

В контент ресурса были добавлены описания конкретных объектов – экземпляров введенных в онтологию понятий. Для описания задач добавлены ссылки на их полнотекстовые описания (Рисунок 4.6), инструкции по их решению и использованию, тексты моделей (Рисунок 4.7), интерфейсы для запуска вычислений (Рисунки 4.8, 4.9).

Главная **Онтология** О СППР 🔍 Вход

УГРОЗЫ ЭНЕРГЕТИЧЕСКОЙ БЕЗОПАСНОСТИ

СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

Географическое место
+ Деятельность
- Задача Энергетической безопасности
 Глобальная задача
 Конкретная задача
+ Информационный ресурс
- Объект Энергетической безопасности
 Инфраструктура энергетики
 Объект энергетики
- Ресурс
 + Топливо-энергетический ресурс
 Энергетический ресурс
 Энергетическая отрасль
Организация
Персона
+ Поддержка принятия решений
- Предмет Энергетической безопасности
 Критерий Энергетической безопасности
 Производственный процесс энергетики
 Управленческий процесс энергетики
+ Факторы влияния на Энергетическую безопасность
 Публикация
 Событие
 Субъект Энергетической безопасности
+ Угроза Энергетической безопасности
 Цель Энергетической безопасности
 Энергетическая безопасность

Табличное представление Графовое представление

Свойства объекта

Название	Авария у производителя энергоресурса
Описание	/threats/uploads/59/fe/b4bee478833840f37443534842ae.pdf

Связи объекта

ликвидирует последствия

Угроза Энергетической безопасности

[ЧС на объектах ТЭК](#)

связана с Задачей

Задача Энергетической безопасности

[Разработка и реализация превентивных мероприятий](#)

[Разработка ликвидационных мероприятий](#)

Обратные связи объекта

описывает Задачу

Публикация

[Определение и угрозы энергетической безопасности, 2016, статья](#)

решает Конкретную Задачу

Метод / средство

[Метод недоопределенных вычислений](#)

Инструкция	/threats/uploads/75/83/1554f7f6be9cb89f724302899608.pdf
Ввод данных	http://uniserv.iis.nsk.su/threats-files/web/accident/Client.html
Модель Задачи	/threats/uploads/0a/1f/8913665556823c11ecfb7110e923.uni
Решатель	http://uniserv.iis.nsk.su/unicalc

© ИСИ, ИСЭМ 2018–2019 Ресурс разработан при финансовой поддержке РФФИ (проект №16-07-00569)

Рисунок 4.5. ИСППР по угрозам энергетической безопасности



Рисунок 4.6. Неформальное описание задачи

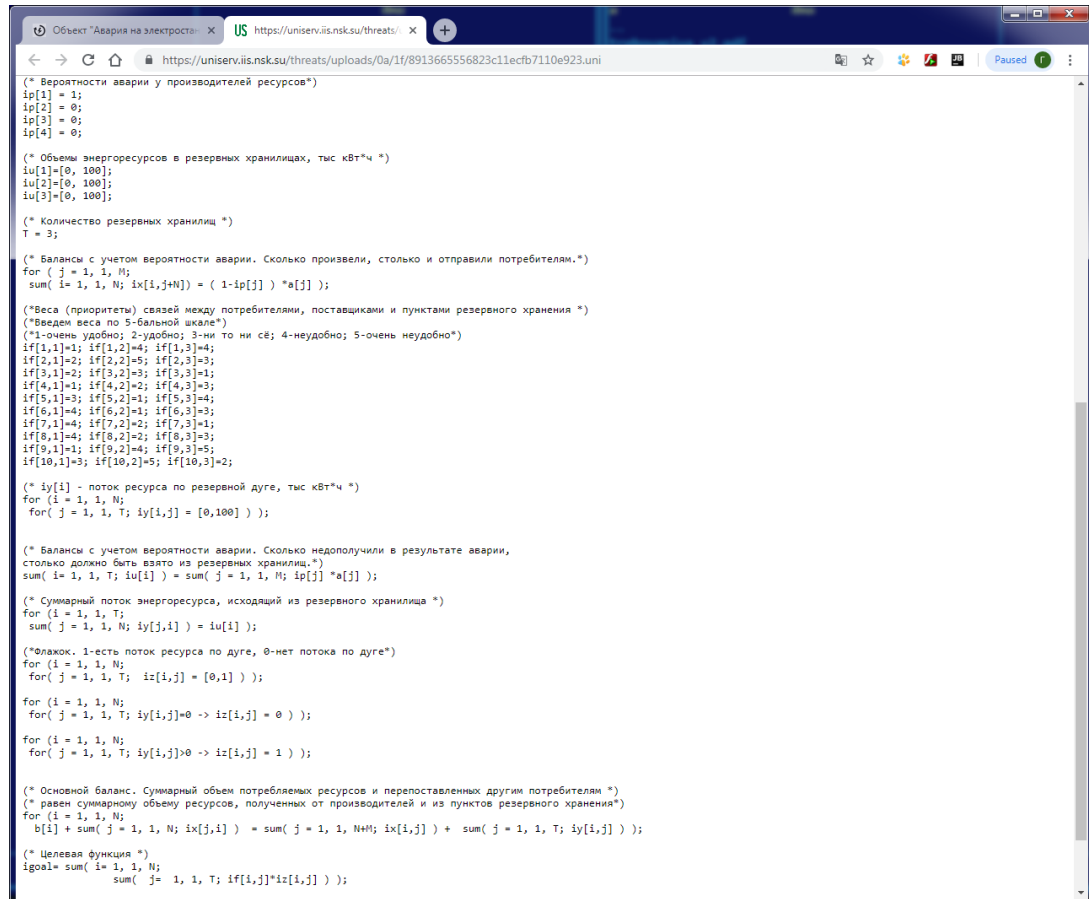


Рисунок 4.7. Фрагмент модели задачи

Результаты вычислений

Количество производителей 4

Количество потребителей 10

Количество резервных хранилищ 3

Объемы производства энергоресурсов ...

Вероятности аварии у производителей ресурсов ...

Объемы потребления энергоресурсов ...

Объемы энергоресурсов, поставляемых потребителям ...

Объемы энергоресурсов в резервных хранилищах ...

Приоритеты связей между потребителями и пунктами резервного хранения ...

☐ заменить текущую модель

Заккрыть

Рисунок 4.8. Фрагмент пользовательского интерфейса. Форма для ввода данных

Объемы энергоресурсов, поставляемых потребителям(ix[i,j]), тыс кВт*ч

	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	6	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	43	0	0	0	0	0	0	20	0	0	0	10
4	0	0	0	0	0	0	0	0	0	25	18	0
5	0	0	0	0	0	0	0	0	0	15	33	0
6	0	0	0	0	12	0	0	0	0	0	10	32
7	0	0	0	0	0	0	0	0	0	0	0	22
8	0	0	0	0	0	0	0	0	0	0	7	33
9	0	0	0	0	0	0	0	0	0	39	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0

Заккрыть

Рисунок 4.9. Фрагмент пользовательского интерфейса. Форма для вывода результатов

После проведенного **тестирования** ИСППР «Угрозы энергетической безопасности» доступна для использования по адресу: <https://uniserv.iis.nsk.su/threats/>

4.2. Интеллектуальная СППР «Элементозы»

4.2.1. Этап Идентификации

Целью разработки ИСППР «Элементозы» является оказание помощи при выявлении у человека элементозов – заболеваний, связанных с недостатком, избытком или дисбалансом в организме человека макро- и микроэлементов, а также при лечении элементозов, прежде всего, за счет коррекции питания [195].

Для достижения этой цели ИСППР должна решать следующие **задачи**:

- 1) Предоставление информации о рассматриваемой группе заболеваний.
- 2) Решение конкретных практических задач:
 - а) Диагностика элементозов путем проведения опроса пациента и предоставления системе данных объективных обследований (анализов, измерений).
 - б) Назначение пациенту препаратов (БАДов) и продуктов, предназначенных для коррекции содержания проблемных минералов.

Назначением ИСППР «Элементозы» является решение вышеупомянутых задач.

Пользователями данной ИСППР могут быть нутриологи, лица, профессионально занимающиеся проблемами питания и элементозов, а также люди, заботящиеся о своем здоровье, желающие узнать об элементозах или о возможных причинах испытываемых ими проблем со здоровьем.

В качестве **экспертов** по элементозам были привлечены сотрудники института микробиологии СО РАН.

4.2.2. Этап Концептуализации

Для **сбора информации** об элементозах были использованы публикации [89,159]. На основании этих публикаций, а также непосредственного общения с экспертами, был выполнен **анализ предметной области, определены ее границы**, построена базовая онтология медицинской диагностики (Рисунок 4.10).

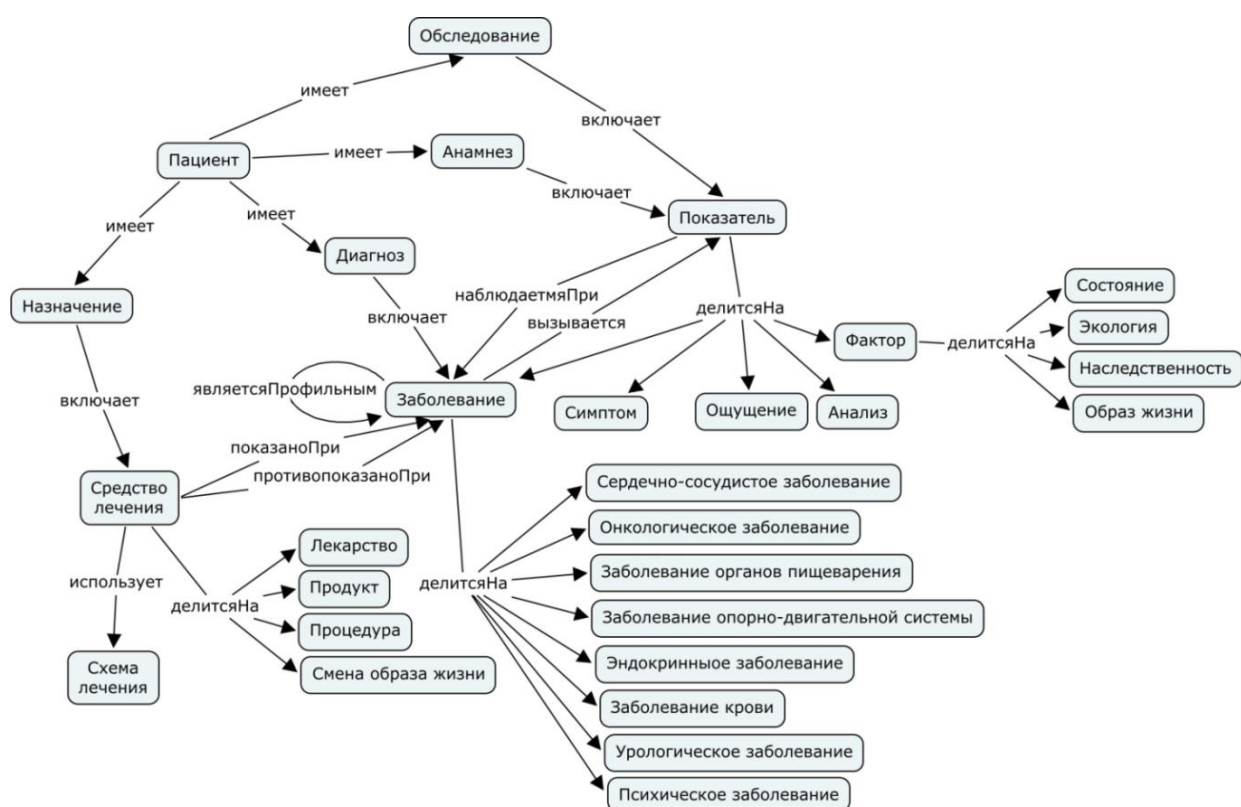


Рисунок 4.10. Базовая онтология ОЗ «Диагностика и лечение заболеваний»

Решаемые ИСППР практические задачи не имеют аналитической модели, а используемые ею информация и данные носят в основном качественный характер. Поэтому для решения таких задач был выбран **метод рассуждений на основе экспертных правил**.

Для структуризации информации о рассматриваемой ПО была использована **онтологическая модель представления знаний**, а для описания задач и механизмов их решения – **сетевая и продукционная модели**.

Для **интерпретации знаний и решения задач** были применены средства технологии Semantic Web [36] и программный комплекс Semp-TAO [101].

4.2.3. Разработка онтологии ПО

В данной задаче этапу формализации предшествовала **разработка онтологии ПО**. Понятия базовой онтологии медицинской диагностики были **конкретизированы** для рассматриваемой группы заболеваний и введены **недостающие понятия** (Элемент, Продукт питания). На Рисунке 4.11 представлены окно редактора Protégé и фрагмент разработанной в нем онтологии элементарных элементов.

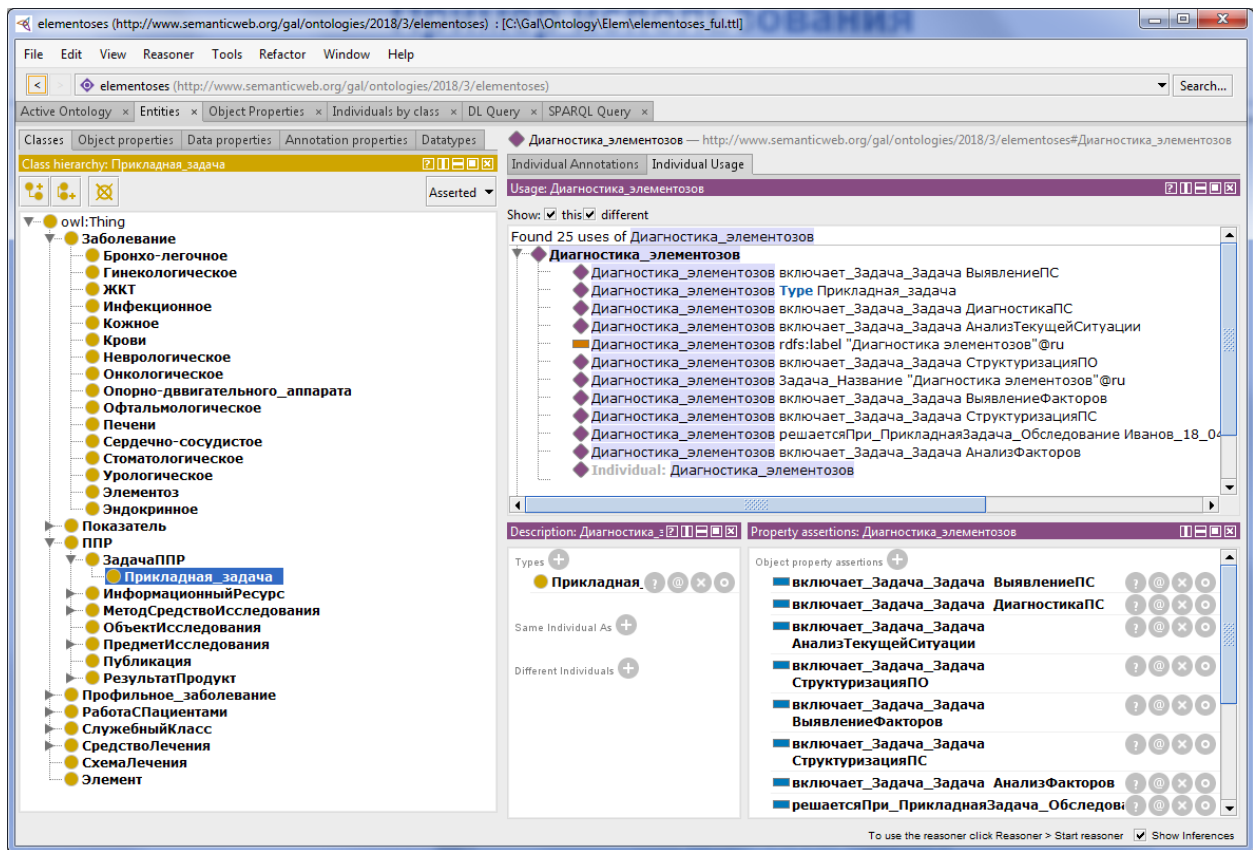


Рисунок 4.11. Онтология элементозов

4.2.4. Этап Формализации

Для решения задачи диагностики онтология элементозов и описания конкретных элементозов были сконvertированы в представления на языке Semp-ТАО. Ниже приведен фрагмент, содержащий описания основных классов:

class Заболевание

Название: string;

Симптом: set of ПСимптом;

Ощущение: set of ПОщущение;

Фактор: set of ПФактор;

Анализ: set of ПАнализ;

Измерение : set of ПокАнИз;

СопутЗабол: set of ПСопутЗабол;

end;

class Пациент

Фамилия: string;

class Обследование

Пациент: Пациент;

Показатель: set of Показатель;

ДатаОбследования: string;

end;

class Диагноз

Пациент: Пациент;

Заболевание: Заболевание;

Вес: integer:=0;

ДатаПостановкиДиагноза: string;

//список показателей для

```

Имя: string;                                     //подсистемы объяснений
Пол: ТПол;                                       ПокОб: tuple of string := string[];
Дата_рождения: string;                         end;
Болезни: set of Заболевание;
end;

```

Знания об элементах хранятся в виде объектов семантической сети.
Рассмотрим в качестве примера описание элемента «Дефицит меди»:

@Cup : Элемент

```

( Название: "Дефицит меди", Элемент: Ca, Статус: дефицит,
  Симпт: PSимптом {
    PSимптом ( Название: "частые переломы", Вес:2),
    PSимптом ( Название: "снижение иммунитета", Вес:1),
    PSимптом ( Название: "плохая свертываемость крови", Вес:1),
    PSимптом ( Название: "повышенное давление", Вес:1),
    PSимптом ( Название: "нарушение сердечного ритма", Вес:1) },

```

```

  Ощущ: ПОщущение{
    ПОщущение(Название: "судороги в мышцах", Вес:1),
    ПОщущение(Название: "мышечные боли", Вес:1),
    ПОщущение(Название: "онемение покаявание в конечностях", Вес:1),
    ПОщущение(Название: "повышенная возбудимость", Вес:1) },

```

```

  Факт: ПФактор {
    ПФактор (Название: "черепно-мозговые травмы", Вес:1),
    ПФактор (Название: "избыток фосфора", Вес:1),
    ПФактор (Название: "дефицит витамина D", Вес:1),
    ПФактор (Название: "беременность", Вес:1),
    ПФактор (Название: "лактация", Вес:1),
    ПФактор (Название: "применение мочегонных средств", Вес:1),
    ПФактор (Название: "применение слабительных средств", Вес:1),
    ПФактор (Название: "возраст - больше 40 лет", Вес:1) },

```

```

  Ан: ПАнализ {
    ПАнализ (Название: "микроэлементный анализ волос",

```



```

ПА: ПокАнИз { ПокАнИз (Название:"Са", Знач:real(0..220) ),
              ПокАнИз (Название:"Р", Знач:(150..32000) ) } },
СопутЗ: ПСопутЗабол { ПСопутЗабол (Название: "аллергия", Вес:1) } );

```

Рассуждения экспертов при постановке диагноза моделируются с помощью продукционных правил вида:

rule постановка_диагноза

```

forall $x: Диагноз( Заболевание:$з, ПокОб:$по ),
              $y: Обследование( Пациент: $пац, Показатель: $пок,
                              ДатаОбследования: $до )

when # $пок <> 0
=>
var $в : integer :=0;
var $setpok: set of Показатель; $setpok := показатели_Заболевания($з);
edit $x: Диагноз( Пациент:$пац, ДатаПостановкиДиагноза: $до);
for $i in $пок loop
    for $j in $setpok loop
        if $i.Название=$j.Название
        then
            $по := $по + string[$i.Название];
            $в:=$в+$j.Вес*$i.Вес;
edit $x: Диагноз( Вес:$в, ПокОб:$по );
        end; end; end; end;

```

Для ввода информации об элементах и данных о пациенте был **спроектирован пользовательский интерфейс**. На Рисунке 4.12 показаны графические формы для редактирования базы знаний. Форма а) позволяет пользователю-эксперту ввести показатели элементов (симптомы, ощущения, факторы, анализы). Форма б) служит для задания связи заболевания и его симптомов. Так, например, для дефицита магния характерны такие симптомы, как нарушение сердечного ритма, повышенное давление. Бегунок справа от названия симптома позволяет задать степень выраженности этого симптома для

выбранного элементоза. Крайнее левое положение бегунка говорит о том, что данный симптом нехарактерен для данного элементоза.

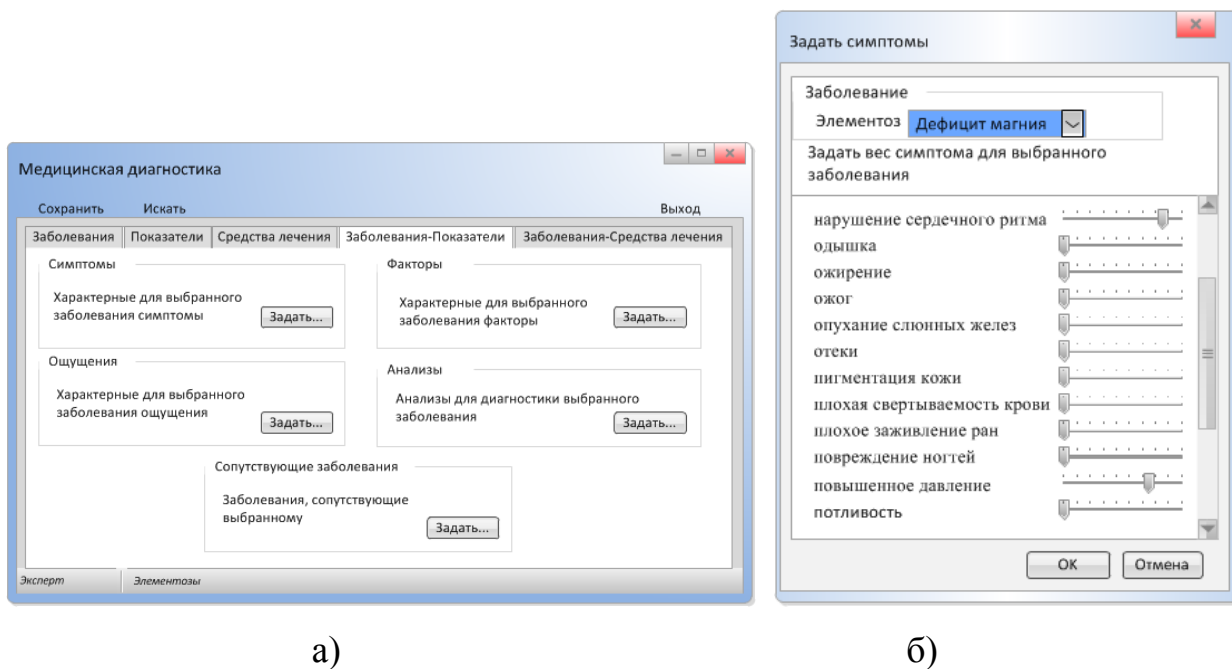


Рисунок 4.12. Формы для редактирования БЗ.

На Рисунках 4.13, 4.14 представлены формы для ввода некоторых данных о конкретном пациенте. Совокупность этих форм представляет собой электронную медицинскую карту пациента. Вкладка «Общие сведения» позволяет зарегистрировать пациента и ввести идентификационные данные о нем.

Медицинская диагностика	
Сохранить Искать Выход	
Пациенты Общие сведения Показатели Диагнозы	
Регистрационный номер	257
Фамилия	Иванова
Имя	Ирина
Отчество	Васильевна
Дата рождения	21.03.1954
Пол	женский
Рост	167
Вес	63.4
Адрес	Ул. Рубиновая 3-15
Телефон	+79134500304
Дата посещения	12.07.2018

Врач Элементозы 257

Рисунок 4.13. Форма для ввода общих сведений о пациенте

Во вкладке «Показатели» пользователь может выбрать тип показателей, которые он хочет задать для конкретного пациента. Это могут быть, например, «Симптомы». При выборе этого типа показателей откроется форма, аналогичная представленной на Рисунке 4.12 б), в которой можно будет выбрать симптомы, присутствующие у обследуемого пациента и задать степень их выраженности. При выборе показателей «Анализы» откроется форма, показанная на Рисунке 4.14 б), в которой можно выбрать тип анализа и ввести данные пациента.

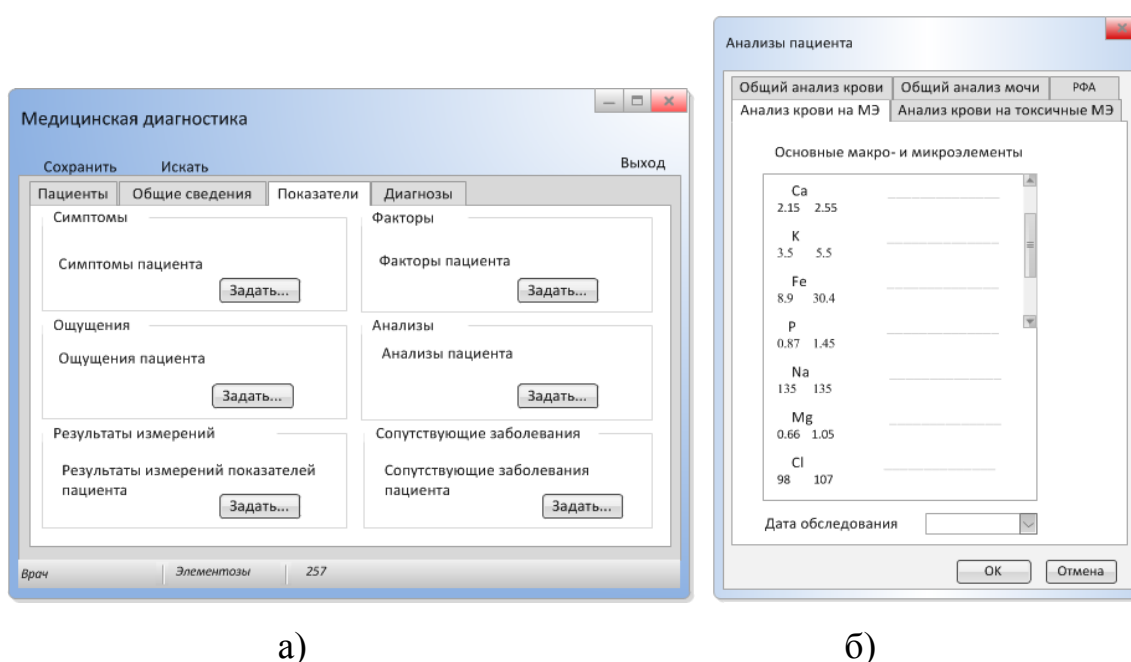


Рисунок 4.14. Формы для выбора типа показателей и ввода результатов анализа пациента

4.2.5. Этап Реализации компонентов интеллектуальной СППР

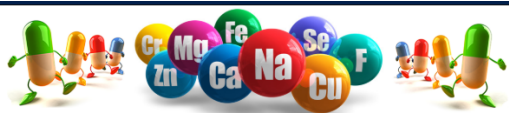
Онтология элементозов (п. 4.2.3) была **дополнена фрагментом онтологии ППР**, описывающим метод рассуждений на основе экспертных правил. Вырезка фрагмента и его склейка с онтологией элементозов выполнялась средствами редактора Protégé (Рисунок 4.11). Для доступа к системе Semp-TAO был использован сервис SempNemo (п.3.3.3).

4.2.6. Этап Реализации интеллектуальной СППР

Разработанная онтология элементозов была загружена в **копию оболочки ИНИР**. На Рисунке 4.15 показана страница получившегося в результате этого ИАИР.

ДИАГНОСТИКА, ПРОФИЛАКТИКА ЛЕЧЕНИЕ ЭЛЕМЕНТОЗОВ

СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ



- Заболeвание
 - Бронхо-легочное заболeвание
 - Гинекологическое заболeвание
 - Дерматологическое заболeвание
 - Заболeвание ЖКТ
 - Заболeвание крови
 - Заболeвание опорно-двигательного аппарата
 - Заболeвание печени
 - Инфекционное заболeвание
 - Неврологическое заболeвание
 - Онкологическое заболeвание
 - Офтальмологическое заболeвание
 - Сердечно-сосудистое заболeвание
 - Стоматологическое заболeвание
 - Урологическое заболeвание
 - Элементоз
 - Эндокринное заболeвание
- Поддержка принятия решений
 - Задача ППР
 - Прикладная задача**
 - Информационный ресурс
 - Метод / средство исследования
 - Объект исследования
 - Предмет исследования
 - Публикация
 - Результат / продукт
 - Показатель
 - Профильное заболeвание
 - Работа с Пациентами
 - Средство лечения
 - Схема лечения
 - Химический элемент

Табличное представление
Графовое представление

Свойства объекта

Название	Диагностика элементозов
Описание	/elem/uploads/36/5b/b7d1506002ba6e9e6d6670b0140e.pdf

Связи объекта

включает Задачу
Задача ППР

Выявление проблемной ситуации

Анализ факторов

Анализ текущей ситуации

Структуризация предметной области

Выявление факторов

Структуризация проблемной ситуации

Диагностика проблемной ситуации

решается при Обследовании
Обследование

Иванов 18.04.2018

Обратные связи объекта

решает Прикладную задачу
Метод / средство исследования

Рассуждения на основе продукционных правил

Инструкция	/elem/uploads/ba/cc/1fd955109c9ae75d874300659af2.pdf
Ввод данных	http://uniserv.iis.nsk.su/threats-files/web/cooling/Client.html
Доступ к сервису	https://uniserv.iis.nsk.su/sempr-files/SempWeb_Client.zip
Модель	https://uniserv.iis.nsk.su/sempr-files/elementoses1.zip

Онтологическое моделирование

Доступ к сервису	https://webprotege.stanford.edu/
------------------	----------------------------------

Рисунок 4.15. ИСППР «Элементозы»

Этот ресурс является каркасом разрабатываемой ИСППР «Элементозы». В контент ресурса добавлены описания конкретных объектов – заболеваний, химических элементов, конкретных пациентов. Для описания решаемых ИСППР задач добавлены ссылки на их полнотекстовые описания, инструкции по их решению и использованию, тексты моделей, сервис для запуска вычислений.

После проведенного **тестирования** ИСППР «Элементозы» доступна для использования по адресу: <https://uniserv.iis.nsk.su/elem/>.

4.3. Выводы

Предложенные средства комплексной поддержки разработки ИСППР были применены для разработки двух систем:

1. «Угрозы энергетической безопасности»
2. «Элементозы»

Разработка этих систем показала удобство и простоту использования предлагаемых средств. Основная сложность при разработке данных ИСППР заключалась в разработке онтологий соответствующих предметных областей и моделей задач.

Использование оболочки ИНИР позволило быстро получить действующие прототипы указанных ИСППР, которые строились как информационно-аналитические интернет-ресурсы с возможностью наращивания функциональных возможностей за счет подключения сервисов, реализующих методы решения практических задач, либо предоставляющих услуги пользовательского интерфейса для облегчения работы пользователей.

Используемые при разработке ИСППР средства позволили создать системы, отвечающие принципам, которые были сформулированы в п. 2.3.

Заключение

Разработка интеллектуальных СППР в слабоформализованных областях продолжает оставаться актуальной проблемой. В диссертационном исследовании была поставлена цель создания модели, методов и средств комплексной поддержки разработки систем такого класса. Для ее достижения были сформулированы задачи, в процессе решения которых получены следующие результаты:

1. Создана модель комплексной поддержки разработки интеллектуальных СППР в слабоформализованных областях. Эта модель интегрирует методы и средства, обеспечивающие концептуальную, информационную, компонентную и методическую поддержку разработчиков СППР. При создании модели использовался язык дескрипционной логики, позволяющий получить как формальное, так и содержательное её описание.
2. Определены границы предметной области «Поддержка принятия решений в слабоформализованных областях» и построена ее онтология.
3. Разработан информационно-аналитический интернет-ресурс по поддержке принятия решений в слабоформализованных областях, предоставляющий содержательный доступ к знаниям и данным этой предметной области, а также к методам поддержки принятия решений и реализующим их программным системам.
4. Создан репозиторий методов поддержки принятия решений в слабоформализованных областях. Разработаны и включены в репозиторий сервисы, реализующие ряд методов поддержки принятия решений в слабоформализованных областях, сервис для организации доступа к данным из внешних источников. Методы и программные системы репозитория

имеют семантические описания и доступны через информационно-аналитический интернет-ресурс.

5. Разработана методика создания интеллектуальных СППР. Данная методика предлагает использовать в качестве каркаса создаваемой системы информационно-аналитический интернет-ресурс моделируемой предметной области и обеспечивать её функциональность за счет подключения сервисов из репозитория. Выполнена апробация методики на примере разработки ИСППР из технической и медицинской областей, которая показала удобство и простоту использования предлагаемых средств.

Разработанные в диссертационном исследовании средства обеспечивают разные типы разработчиков ИСППР единой системой понятий за счет использования онтологии поддержки принятия решений. Созданный на основе онтологии информационно-аналитический интернет-ресурс предоставляет им информацию о конкретных методах поддержки принятия решений, о классах задач, решаемых этими методами, о реализующих их программных системах. Репозиторий предоставляет разработчикам возможность выбрать и опробовать готовые программные компоненты, реализующие необходимые методы поддержки принятия решений или организующие интерфейс с пользователями. Методика предлагает каркас разрабатываемой ИСППР и алгоритм её построения. Все эти средства упростят и ускорят процесс создания конкретных прикладных ИСППР.

Помимо поддержки разработки прикладных ИСППР полученные в диссертационном исследовании результаты могут быть использованы в процессе обучения студентов, специализирующихся в данной области знаний.

Дальнейшее развитие результатов диссертационной работы будет вестись в нескольких направлениях:

1. Развитие онтологии поддержки принятия решений.
2. Пополнение контента ИАИР ППР описаниями новых методов (методы машинного обучения, агрегационные методы многокритериального оценивания,

методы получения информации и др.), программных разработок, связанной с ними литературы.

3. Развитие репозитория, реализация новых сервисов, в частности, сервисов интерфейса с конечным пользователем.

4. Разработка конкретных ИСППР с применением разработанных модели и средств КПР.

Список сокращений и условных обозначений

БД	база данных
БЗ	база знаний
ИАИР	информационно-аналитический интернет-ресурс
ИНИР	интеллектуальный научный интернет-ресурс
ИСППР	интеллектуальная система поддержки принятия решений
КПР	комплексная поддержка разработки ИСППР
ЛПР	лицо, принимающее решение
МПЗ	модель представления знаний
МЭО	метод экспертного оценивания
ОЗ	область знаний
ОСД	обеспечение содержательного доступа
ПО	предметная область
ППР	поддержка принятия решений
СППР	система поддержки принятия решений
ПС	проблемная ситуация
РГ	рабочая группа
СФПО	слабоформализованная предметная область
ТПР	теория принятия решений
ФС-модель	фрактальная стратифицированная модель
ЭС	экспертная система
ALC	Attributive Language with Complement
CORBA	Common Object Request Broker Architecture
CBR	Case-Based Reasoning
CLIPS	C Language Integrated Production System
DL	Description Logics

JSON	JavaScript Object Notation
OASIS	Organization for the Advancement of Structured Information Standards
OWL	Web Ontology Language
REST	Representational State Transfer
RDF	Resource Description Framework
RDFS	RDF Schema
SPARQL	Simple Protocol and RDF Query Language
SWRL	Semantic Web Rule Language
W3C	World Wide Web Consortium
XML	eXtensible Markup Language
RPC	Remote Procedure Call
SOAP	Simple Object Access Protocol
UDDI	Universal Description Discovery and Integration
URI	Uniform Resource Identifier
WF	Work Flow
WFMS	Work Flow Management System
WSDL	Web Services Description Language

Список литературы

1. Akhmadeeva I. R., Zagorulko Y. A., Mouromtsev D. I. Ontology-Based Information Extraction for Populating the Intelligent Scientific Internet Resources // Communications in Computer and Information Science. – Springer International Publishing, 2016. – Vol. 649. – P. 119–128.
2. Alter S.L. Decision Support Systems: Current Practice and Continuing Challenge. Reading, MA: Addison-Wesley, 1980.
3. Antoniou G., Harmelen F. Web Ontology Language: OWL // Handbook on Ontologies. Berlin: Springer Verlag, 2004. pp. 67–92.
4. Apache Jena. A free and open source Java framework for building Semantic Web and Linked Data applications. URL: <https://jena.apache.org/index.html> (дата обращения 20.12.2019)
5. Axelrod R. The Structure of Decision: Cognitive Maps of Political Elites. Princeton University Press, 1976.
6. Baader F. The Description Logic Handbook: Theory, Implementation, Applications / F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider. Cambridge, 2003. 574 P.
7. Bacon Jono. Reviews: Qt. A multi-platform graphical toolkit. 2004. URL: <http://preserve.mactech.com/articles/mactech/Vol.20/20.03/QtReview/index.html> (дата обращения 20.11.2019)
8. Belshe M., Peon R., Thomson M. Hypertext Transfer Protocol version 2. 2015. URL: <https://tools.ietf.org/html/draft-ietf-httpbis-http2-17> (дата обращения 20.11.2019)

9. Blomqvist E., Hammar K., Presutti V. Engineering Ontologies with Patterns: The eXtreme Design Methodology, In *Ontology Engineering with Ontology Design Patterns*. Studies on the Semantic Web, Eds., Hitzler, P., and A. Gangemi, K. Janowicz, A. Krisnadhi, V. Presutti, IOS Press, pp: 23-50, 2016.
10. Buzan T. *The Mind Map Handbook: The Ultimate Thinking Tool*. - HarperCollins Publishers, 2005, -464с.
11. Casanovas P., Casellas N., Vallbe J.-J. An Ontology-Based Decision Support System for Judges. In *Proceeding of the 2009 conference on Law, Ontologies and the Semantic Web: Channelling the Legal Information Flood*. Amsterdam: IOS Press, 2009. – P. 165–175
12. CORBA. Официальный сайт. URL: <https://www.corba.org/index.html> (дата обращения: 20.11.2019)
13. Chinnici R., Moreau J.-J., Ryman A., Weerawarana S. *Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language*. URL: <http://www.w3.org/TR/wsdl20/> (дата обращения: 20.11.2019)
14. CLAVIRE – облачные решения. <http://clavire.com/>
15. Crockford D. *The application/json Media Type for JavaScript Object Notation (JSON)* — Internet Engineering Task Force, 2006. — 10 p.
16. De Nicola A., Missikoff M., Navigli R.A. Proposal for a Unified Process for Ontology Building: UPON. In: *Database and Expert Systems Applications*, Eds., Andersen, K.V., J. Debenham, R. Wagner DEXA 2005 // *Lecture Notes in Computer Science*, 2005. v. 3588. pp. 655–664.
17. EasyRDF. A PHP library designed to make it easy to consume and produce RDF. URL: <http://www.easyrdf.org/docs> (дата обращения: 15.10.2018)
18. Edwards M.P., Lippeveld T., Khatri U., Kunaka D., Mwebaze M., Tohouri R. *Building a Web-Based Decision Support System // MEASURE Evaluation*, 2018.

- URL: <https://www.measureevaluation.org/resources/publications/fs-18-288/> (дата обращения: 23.10.2019)
19. Erl T. Service-Oriented Architecture: Concepts, Technology, and Design. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2005. 792 p
 20. Erl Thomas, Carlyle Benjamin, Pautasso Cesare, Balasubramanian Raj. 5.1 // SOA with REST. — Prentice Hall, 2013.
 21. Fielding Roy. Dissertation: Architectural Styles and the Design of Network-based Software Architectures. CHAPTER 5: Representational State Transfer (REST). www.ics.uci.edu. (дата обращения: 2.12.2019).
 22. Gangemi A., Presutti V. Ontology Design Patterns // Handbook on Ontologies. Berlin: Springer, 2009. pp. 221–243.
 23. Fielding R. Architectural Styles and the Design of Network-based Software Architectures. URL: <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm> (дата обращения: 20.11.2019)
 24. Forgy Ch. Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem // Artificial Intelligence, 19, pp. 17—37, 1982.
 25. FreeMind – free mind mapping software – URL: http://freemind.sourceforge.net/wiki/index.php/Main_Page (дата обращения: 22.11.2019).
 26. Fuseki: serving RDF data over HTTP. – URL: http://jena.apache.org/documentation/serving_data/ (дата обращения: 03.08.2019)
 27. Garaibeh N.K. DSS Development and Agile Methods: Towards a new Framework for Software Development Methodology // International Journal of Machine Learning and Computing, Vol. 2, No. 4, August 2012 URL: https://www.researchgate.net/publication/267625331_Survey_of_DSS_Development_Methodologies (дата обращения: 14.01.2019).

28. Garaibeh N. K., Al-Raddadi A. Survey of DSS Development Methodologies // Encyclopedia of Business Analytics and Optimization. Publisher: IGI GlobalEditor: John Wang, 2014.
29. Gensym G2. The World's Leading Software Platform for Real-Time Expert System Applications. URL: <http://www.gensym.com/wp-content/uploads/Gensym-l-G2.pdf> (дата обращения: 20.02.2019).
30. Goiry A, Scott-Morton M.S. A framework for information FFsystems. Sloan ManagementReview. 1971, 13 (1): 56-79.
31. Google forms. URL: <https://www.google.com/forms/about/> (дата обращения: 20.02.2019).
32. Gruber T. R., Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal Human-Computer Studies, 43(5-6):907-928, 1995.
33. Guarino N. Formal Ontology in Information Systems//Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998. - Amsterdam: IOS Press, 1998. -P. 3-15
34. Gudgin M., Hadley M., Mendelsohn N., Moreau J.-J., Nielsen H.F., Karmarkar A., Lafon Y. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition). URL: <http://www.w3.org/TR/soap12-part1/>(дата обращения: 20.11.2019)
35. Haettenschwiler P. Neues anwenderfreundliches Konzept der Entscheidungsunterstützung. Gutes Entscheiden in Wirtschaft, Politik und Gesellschaft. Zurich: Hochschulverlag AG, 1999. — S. 189—208.
36. Hitzler P., Krötzsch V., Rudolph S. Foundations of Semantic Web Technologies. Chapman & Hall/CRC, 2009. 455 p.
37. IHMC CmapTools – свободно распространяемый инструментарий для построения концептуальных карт знаний – URL: <http://cmap.ihmc.us/> (дата обращения: 22.11.2019)

38. jColibri: CBR Framework jColibri 2.1 Tutorial. URL:
<http://gaia.fdi.ucm.es/research/colibri/> (дата обращения: 18.11.2019)
39. C. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, R. Metz. Reference Model for Service Oriented Architecture 1.0. Committee Specification 1, 2 August 2006. URL: <https://www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf> (дата обращения: 20.11.2019)
40. Map-it is a tool for participatory and conversation. URL: <http://map-it.be> (дата обращения: 22.11.2019)
41. Microsoft Excel. Официальный сайт. URL: <https://products.office.com/ru-ru/excel> (дата обращения: 20.11.2019).
42. Murata M., Kohn D., Lilley C. Internet Drafts: XML Media Types. IETF (24 сентября 2009). (дата обращения 20.11. 2019).
43. Novak Joseph D., Cañas Alberto J. The Theory Underlying Concept Maps and How To Construct and Use Them. Technical Report IHMC CmapTools 2006-01 Rev 2008-01. Institute for Human and Machine Cognition. URL:
<http://cmap.ihmc.us/docs/theory-of-concept-maps.php> (дата обращения: 19.06.2019).
44. OASIS. Reference Model for Service Oriented Architecture V 1.0. Committee Specification 1, 2 August 2006
URL: <https://www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf>
45. Портал Ассоциации ODPa (Association for Ontology Design & Patterns). URL: <http://ontologydesignpatterns.org> (дата обращения: 16.06.2017).
46. OWL API is a Java API and reference implementation for creating, manipulating and serialising OWL Ontologies. URL: <http://ontologydesignpatterns.org> (дата обращения: 06.11.2019).
47. Power D. J. What is a DSS? DSStar, vol. 1, no. 3, October 21, 1997. URL:
<http://dssresources.com/papers/whatisadss> (дата обращения: 20.10.2019).

48. Power D J., Decision Support Systems Hyperbook. Cedar Falls, IA:DSSResources.COM, HTML version, 2000. URL: <http://dssresources.com/subscriber/password/dssbook> (дата обращения: 20.10.2019).
49. Power D. J. Decision support systems: Concepts and resources for managers. NY: Greenwood Publishing Group, 2002.
50. Power D.J. A Brief History of Decision Support Systems. DSSResources.COM, version 4.0, March 10, 2007, URL: <http://DSSResources.COM/history/dsshhistory.html> (дата обращения: 20.10.2019).
51. Protégé. A free, open-source ontology editor and framework for building intelligent systems. URL: <http://protege.stanford.edu/>(дата обращения: 20.11.2019).
52. Protocol Buffers. Developer Guide — // Google Developers, April 2, 2012. URL: <https://developers.google.com/protocol-buffers/docs/overview> (дата обращения: 26.10.2019)
53. Recio-García J.A., Díaz-Agudo B., González-Calero P.A. (2014) The COLIBRI Platform: Tools, Features and Working Examples. In: Montani S., Jain L. (eds) Successful Case-based Reasoning Applications-2. Studies in Computational Intelligence, vol 494. Springer, Berlin, Heidelberg. URL: <https://books.google.ru/books?id=k1q5BQAAQBAJ&pg=PA55&lpg=PA55&dq=The+COLIBRI+Platform:+Tools,+Features+and+Working+Examples&source=bl&ots=pu38vpD8Nj&sig=ACfU3U1PDUbHKLqDm0HD7bSrdUwYTWluHg&hl=ru&sa=X&ved=2ahUKEwj8lOW71sngAhWR6KYKHRZDDFsQ6AEwA3oEACQAQ#v=onepage&q=The%20COLIBRI%20Platform%3A%20Tools%2C%20Features%20and%20Working%20Examples&f=false> (дата обращения: 20.10.2019).
54. Rospocher M, Serafini L. 2012. An Ontological Framework for Decision Support. In: Proceedings of the 2nd Joint International Semantic Technology Conference.

55. Rospocher M. 2014. An ontology for personalized environmental decision support. In: Formal Ontology in Information Systems - Proceedings of the Eighth International Conference, FOIS2014, September, 22-25, 2014, Rio de Janeiro, IOS Press, volume 267, 2014.
56. Saha C., Aqlan F., Lam S. S., Boldrin W. A decision support system for real-time order management in a heterogeneous production environment. *Expert Systems with Applications*, 60:16–26, 2016
57. Schank R.C., Abelson R.P. *Scripts, Plans, Goals and Understanding*. Erlbaum, Hillsdale, New Jersey, US, 1977, - 248 p
58. Saremi A., Esmaeili M., Habibi J., Ghaffari A. 2008. O2dss: A framework for ontologybased decision support systems in pervasive computing environment. In: *Proceedings of the 2008 Second Asia International Conference on Modeling & Simulation (AMS)*
59. Simon H.A. *The New Science of Management Decision* / H.A. Simon – N. Y.: Harper and Row Publishers, 1960.
60. SPARQL Query Language for RDF. W3C Recommendation 15 January 2008. URL: <http://www.w3.org/TR/rdf-sparql-query/> (дата обращения: 20.10.2019).
61. Sure Y., Staab S., Studer R. *Ontology Engineering Methodology // Handbook on Ontologies*, Eds., Staab S., Studer R. Berlin: Springer Verlag. 2009. pp. 135-152.
62. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission 21 May 2004. URL: <http://www.w3.org/Submission/SWRL/> (дата обращения: 20.10.2019).
63. Tagliaferri Lisa. *An Introduction to JSON*, 2016 URL: <https://www.digitalocean.com/community/tutorials/an-introduction-to-json> (дата обращения 27.09.2019).
64. Taverna Workflow Management System. URL: <http://www.taverna.org.uk/> (дата обращения: 20.10.2019).

65. Telerman V.V., Ushakov D.M. Subdefinite Models as a Variety of Constraint Programming // Proc. of the Intern. Conf. Tools of Artificial Intelligence (ICTAI'96), Toulouse, 1996.
66. Turban E. Decision Support and Expert Systems: Management Support Systems. 4th Edition, Macmillan Publishing Company, New York (1995)
67. Visual Mind – Mind Technologies. Support human performance. URL: www.visual-mind.com (дата обращения: 20.10.2019).
68. Vokhmintcev A., Melnikov A., Timchenko M., Kozko A., Makovetskii A., Kober A. Development of methods for selecting features using deep learning techniques based on autoencoders // Proceedings of SPIE - The International Society for Optical Engineering 41. Сер. "Applications of Digital Image Processing XLI" 2018. С. 1075227
69. Zagorulko Yu.A., Popov I.G., 1998. Knowledge representation language based on the integration of production rules, frames and a subdefinite model. Joint Bulletin of the Novosibirsk Computing Center and Institute of Informatics Systems. Series: Computer Science, 8: 81–100
70. Zhang S., Goddard S. A software architecture and framework for Web-based distributed decision systems. Decision Support Systems 43, 1133—1150.
71. Zulkafli Z., Perez K., Vitolo C., Buytaert W., Karpouzoglou T., Dewulf A., Bievre B. D., Clark J., Hannah D.M., Shaheed S. User-driven design of decision support systems for polycentric environmental resources management // Environmental Modelling & Software. 2017. Vol. 88. P. 58–73.
72. Абрамова Н.А., Порцев Р.Ю. Аналитические методы для анализа слабоструктурированных ситуаций на основе функциональных когнитивных карт. В сборнике: Информационные и математические технологии в науке и управлении Ответственный редактор Л.В. Массель. 2015. С. 62-70.
73. Авдеенко Т.В., Макарова Е.С. Система поддержки принятия решений в it-подразделениях на основе интеграции прецедентного подхода и онтологии.

- Вестник Астраханского государственного технического университета. 2017. № 3. С. 85.
74. Артемьева И.Л. Метод построения многоуровневых онтологий сложно структурированных предметных областей // Всероссийская конф. с межд. Участием «Знания-Онтологии-Теории». Новосибирск, 14-16 сентября 2007. Новосибирск: Институт математики. –Т.1. С.132-137.
75. Афоничкин А. И. Управленческие решения в экономических системах / А. И. Афоничкин, Д.Г. Михаленко // Учебник для вузов. — СПб.: Питер, 2009. — 480 с.
76. Ахмадеева И.Р., Серый А.С., Шестаков В.К. Некоторые особенности реализации платформы для построения информационно-аналитических интернет-ресурсов Информационные и математические технологии в науке и управлении. 2017. № 3 (7). С. 168-175.
77. Ахмадеева И.Р. Использование онтологии для автоматизации поиска научной информации в сети интернет Информационные и математические технологии в науке и управлении. 2018. № 4 (12). С. 42-49.
78. Батыршин И.З., Недосекин А.О., Стецко А.А., Тарасов В.Б., Язенин А.В., Ярушкина Н.Г. Нечеткие гибридные системы. Теория и практика Москва, 2007.
79. Бьюзен Т. Научите себя думать! 3-е издание. – Минск: «Попурри», 2008. – 192 с.
80. Баландин Е.А., Катков Ю.В., Муромцев Д.И., Починок И.Н. Опыт использования онтологий верхнего уровня при проектировании базы знаний музея оптических технологий // Материалы Всероссийской конференции «Знания-Онтологии-решения» (ЗОНТ-09).
81. Вагин В.Н., Еремеев А.П. Некоторые базовые принципы построения интеллектуальных систем поддержки принятия решений реального времени // Известия РАН. Теория и системы управления. 2001. №6.

82. Варшавский П.Р., Еремеев А.П. Моделирование рассуждений на основе прецедентов в интеллектуальных системах поддержки принятия решений// Искусственный интеллект и принятие решений, 2009, №2.
83. Васенин В.А., Голомазов Д.Д., Ганкин Г.М. Архитектура, методы и средства базовой составляющей системы управления научной информацией «Истина-Наука МГУ». // Программная инженерия, №9, 2014, с.3–12.
84. Виттих В.А., Ситников П.В., Смирнов С.В. Онтологический подход к построению информационно-логических моделей в процессах управления социальными системами // Вестник компьютерных и информационных технологий. - 2009. - №5. - С. 45-53.
85. Гаврилова Т.А., Червинская К.Р. Извлечение и структурирование знаний для экспертных систем. Москва, «Радио и связь», 1992.
86. Гаврилова Т. А., Хорошевский В. Ф. Базы знаний интеллектуальных систем. СПб.: Питер, 2001. 384 с.
87. Гаврилова Т.А., Горовой В.А., Болотникова Е.С. Оценка когнитивной эргономичности онтологии на основе анализа графа // Журнал РАН "Искусственный интеллект и принятие решений", 2009. № 03. – С. 33-41.
88. Гаврилова Т. А., Муромцев Д. И. Интеллектуальные технологии в менеджменте: инструменты и системы: Учеб. пособие. 2-е изд. / Высшая школа менеджмента СПбГУ : СПб : Изд-во «Высшая школа менеджмента»; Издат. дом С.-Петербур. гос. ун-та, 2008. -488 с.
89. Горбачев А.Л., Луговая Е.А. Введение в биоэлементологию. Учебное пособие. Магадан: СВГУ, 2018. 115 с.
90. Горбунов-Посадов М.М. Расширяемые программы. М.: Полиптих, 1999.
91. Горелова Г. В., Захарова Е. Н., Гинис Л. А. Когнитивный анализ и моделирование устойчивого развития социально-экономических систем // Ростов-на-Дону: Изд-во Рост. ун-та, 2005.

92. Горшков С. Введение в онтологическое моделирование // ООО «ТриниДата», 2016. URL: <http://trinidata.ru> (дата обращения: 20.10.2019).
93. Геловани В.А., Башлыков А.А., Бритков В.Б., Вязилов Е.Д. Интеллектуальные системы поддержки принятия решений в нештатных ситуациях. М.: Эдиториал УРСС, 2001. - 304 с.
94. Глоба Л.С., Новогрудская Р.Л. Модели и методы интеграции информационных и вычислительных ресурсов // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2012): материалы II Междунар. научн.-техн. конф. (Минск, 16-18 февраля 2012 г.) / редкол. : В. В. Голенков (отв. ред.) . – Минск : БГУИР, 2012. –С. 447-452.
95. Грибова В.В., Петряева М.В., Окунь Д.Б., Шалфеева Е.А. Онтология медицинской диагностики для интеллектуальных систем поддержки принятия решений// Онтология проектирования. 2018. Т. 8. № 1 (27). С. 58-73.
96. Гузаиров М.Б., Юсупова Н.И., Сметанина О.Н., Галеева Н.И. Инструментарий нейронных сетей при поддержке принятия решений по управлению образовательным маршрутом //Нейрокомпьютеры: разработка, применение. 2013. № 3. С. 021-025.
97. Доктрина энергетической безопасности России <https://minenergo.gov.ru/node/14766> (дата обращения: 16.07.2019).
98. Джарратано Джозеф, Райли Гари. «Экспертные системы: принципы разработки и программирование» = Expert Systems: Principles and Programming. — М.: «Вильямс», 2007. — 1152 с. — ISBN 978-5-8459-1156-8.
99. Добров Б. В., Иванов В. В., Лукашевич Н. В., Соловьев В. Д. Онтологии и тезаурусы: модели, инструменты, приложения. –М.: БИНОМ, 2009. 173 с.
100. Ефименко И.В., Хорошевский В.Ф. Интеллектуальные системы поддержки принятия решений в медицине: ретроспективный обзор состояния

- исследований и разработок и перспективы // Открытые семантические технологии проектирования интеллектуальных систем. 2017. № 7. С. 251-260.
101. Загорулько Ю.А., Попов И.Г.. Представление знаний в интегрированной технологической среде Semp-ТАО. // Проблемы представления и обработки не полностью определенных знаний. –Москва-Новосибирск, 1996. –С.59–74.
 102. Загорулько Ю.А., Боровикова О.И. Технология построения онтологий для порталов научных знаний // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2007. Т. 5. № 2. С. 42-52.
 103. Загорулько Ю.А. Семантические модели и технологии разработки информационных и интеллектуальных систем, ориентированные на экспертов //В сборнике: Информационные и математические технологии в науке и управлении Труды XIX Байкальской Всероссийской конференции. 2014. С. 131-138.
 104. Загорулько Ю.А. Современные средства формализации семантики областей знаний на основе онтологий // Информационные и математические технологии в науке и управлении. 2018. № 3 (11). С. 27-36.
 105. Заде Л. Понятие лингвистической переменной и его использование в принятии приближенных решений. М., Мир, 1976.
 106. Карпов В.В., Симанчев Р.Ю. Определение и угрозы энергетической безопасности. – Вестник Омского университета. Серия «Экономика». -2016. - №4. - С. 30-38.
 107. Клещев А.С., Артемьева И.Л. Необогатенные системы логических соотношений // Ч.1. Научно-техническая информация, серия 2. – 2000. – №7.– С. 18-28.
 108. Князьков К.В. Технология разработки композитных приложений с использованием предметно-ориентированных программных модулей. Автореферат. СПб, 2012. – 19 с.

109. Кобринский Б.А. Системы поддержки принятия решений в здравоохранении и обучении // Врач и информационные технологии. 2010. № 2. С. 39-45.
110. Кобринский Б.А. Особенности медицинских интеллектуальных систем // Информационно-измерительные и управляющие системы. 2013. Т. 11. № 5. С. 58-64.
111. Ковальчук С.В., Маслов В.Г. Интеллектуальная поддержка процесса конструирования композитных приложений в распределенных проблемно-ориентированных средах. // Известия вузов. Приборостроение. – 2011. – №10 (54). – С. 29-35.
112. Копайгородский А.Н., Массель Л.В. Фрактальный подход к проектированию архитектуры информационных систем // Вестник Иркутского государственного технического университета. 2010. № 6 (46). С. 8-12.
113. Копайгородский А.Н. Применение онтологий в семантических информационных системах // Онтология проектирования. 2014. № 4 (14). С. 78-89
114. Костов Ю.В., Липовой Д.А., Мамонтов П.Г., Петров Е.С. Новый UNICALC: версия 5 — возможности и перспективы // Труды 9-й национальной конференции по искусственному интеллекту — КИИ-2004. Тверь, 2004. -С. 915-922.
115. Кравченко Т.К. Развитие экспертной системы поддержки принятия решений // Искусственный интеллект и принятие решений. 2013. № 4. С. 72-80.
116. Кулагин О.А. Принятие решений в организациях. – СПб, Издательский дом «Сентябрь», 2001
117. Кулинич А.А., Максимов В.И. Система концептуального моделирования социально-политических ситуаций ПК «КОМПАС». Сборник докладов: Современные технологии управления. Москва. ИПУ. 21-22 мая 1998 г.

118. Кулинич, А. А. Компьютерные системы анализа ситуаций и поддержки принятия решений на основе когнитивных карт: подходы и методы // Проблемы управления. 2011. N 4. С. 3145
119. Кузнецов О.П. Интеллектуализация поддержки управляющих решений и создание интеллектуальных систем. Проблемы управления, 2009, № 3.1, 64–72
120. Кудрявцев Д.В. Практические методы отображения и объединения онтологий / Д.В. Кудрявцев // Труды 11-й Национальной конференции по искусственному интеллекту с международным участием (КИИ-08), семинар «Знания и Онтологии *ELSEWHERE*», 29 сент.-3 окт. 2008, г. Дубна, Россия. – М.: URSS, 2008. – Т. 3. – С.164-173.
121. Лапшин В. А. Онтологии в компьютерных системах. М.: Научный мир, 2010. 224 с.
122. Ларичев О. И. Наука и искусство принятия решений / О. И. Ларичев – М.: Наука, 1979. – 200 с.
123. Ларичев О.И., Петровский А.Б. Системы поддержки принятия решений. Современное состояние и перспективы развития // Итоги науки. Техническая кибернетика, том 21, Москва, ВИНТИ, 1987, с. 131-164.
124. Ларичев О. И. Теория и методы принятия решений, а также хроника событий в Волшебных странах / О. И. Ларичев // Учебник. - М.: Логос, 2000. - 296 с.
125. Ларичев О.И. Вербальный анализ решений. Ин-т системного анализа РАН. - М.: Наука, 2006. - 181 с.
126. Лежнина М.В. Информационная поддержка управленческих решений // Актуальные проблемы экономики современной России. Изд. Марийский гос. ун-т. Йошкар-Ола Т. 2 №2. 2015, с. 57-62
127. Лозовский В.С. Семантические сети//Представление знаний в человеко-машинных и робототехнических системах. -М.: ВИНТИ, 1984. -Т.А. -С. 84-121.

128. Массель Л.В. Фрактальная модель структурирования знаний // Сб. науч. трудов Национальной конференции с международным участием «Искусственный интеллект-94». Т.1 – Рыбинск, 1994. – С.46-49.
129. Массель Л.В., Васильев И.В. Разработка информационной системы для социологических исследований с использованием стратифицированной фрактальной модели // Вестник Иркутского государственного технического университета. 2004. № 2 (18). С. 98-104.
130. Массель Л.В. Применение онтологического, когнитивного и событийного моделирования для анализа чрезвычайных ситуаций в энергетике // Проблемы безопасности и чрезвычайных ситуаций. -2010, №2, -С. 34-43.
131. Массель А.Г., Копайгородский А.Н., Паскал К.К. Применение системы когнитивного моделирования в исследованиях проблем энергетической безопасности. Вестник Иркутского государственного технического университета. 2012. № 10 (69). С. 29-33.
132. Массель Л.В., Пяткова Е.В. Применение байесовских сетей доверия для интеллектуальной поддержки исследований проблем энергетической безопасности. – Вестник ИрГТУ.-№2.- 2012.- С. 8-13.
133. Массель Л.В., Массель А.Г. Семантические технологии на основе интеграции онтологического, когнитивного и событийного моделирования // Открытые семантические технологии проектирования интеллектуальных систем. 2013. № 3. С. 247-250.
134. Массель, Л.В. Применение онтологий в исследованиях и поддержке принятия решений в энергетике / Л.В. Массель, Т.Н. Ворожцова, А.Н. Копайгородский, Н.Н. Макагонова, С.К. Скрипкин // Информационные и математические технологии в науке и управлении // Всероссийская конференция «Знания-Онтологии-Теория (ЗОНТ-13)»: труды. Новосибирск. ИМ СО РАН. Т. 2. С. 29-38.

135. Массель Л.В. Фрактальный подход к структурированию знаний и примеры его применения // Онтология проектирования. 2016. Т. 6. № 2 (20). С. 149-161.
136. Массель Л.В., Ворожцова Т.Н., Пяткова Н.И. Онтологический инжиниринг для поддержки принятия стратегических решений в энергетике // Онтология проектирования. 2017. Т. 7. № 1 (23). С. 66-76.
137. Мельников А.В., Кузенбаев Б.А., Кузенбаева А.А. Онтологическая модель автоматизированной информационной системы вуза В сборнике: Информационные технологии и системы. Труды Четвертой Международной научной конференции. Ответственные редакторы: Ю.С. Попков, А.В. Мельников. 2015. С. 177-178.
138. Минский М. Структура для представления знания//Психология машинного зрения. -М.: Мир. -1978. -С. 249-338
139. Муромцев Д.И. Концептуальное моделирование знаний в системе Стар Tools - Санкт-Петербург: , 2009. - 83 с.
140. Муртазина М.Ш., Авдеенко Т.В. Онтологический подход к поддержке процесса инженерии требований в scrum. В сборнике: Информационные технологии и нанотехнологии. Сборник трудов ИТНТ-2018. Самарский национальный исследовательский университет имени академика С.П. Королева. 2018. С. 2610-2620.
141. Нариньяни, А.С. Недоопределённость в системах представления и обработки знаний / А.С. Нариньяни // Изв. АН СССР. Техн. кибернетика. – 1986. – №5. – С. 3-28.
142. Нариньяни А.С., Телерман В.В., Ушаков Д.М., Швецов И.Е. Программирование в ограничениях и недоопределённые модели//Информационные технологии, 1998. -№7. -С. 13-22

143. Нариньяни А.С. НЕ-факторы: неточность и недоопределенность различие и взаимосвязь// Известия РАН. Теория и системы управления. 2000. №5. - С.44-56.
144. Нариньяни А.С., Гофман И.Д., Инишев Д.А., Банасюкевич Д.В. Развитие интеллектуальной технологии недоопределенного планирования и управления проектами Time-EX// Труды 7-й национальной конференции по искусственному интеллекту КИИ-2000, 23 - 26 октября 2000 г., Переславль-Залесский, С.617-624.
145. Овдей О.М., Проскудина Г.Ю. Обзор инструментов инженерии онтологий. Труды 6-ой Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» - RCDL2004, Пушкино, Россия, 2004.
146. Орлов А.И. Экспертные оценки. Учебное пособие. - Москва, 2002.
147. Петровский А.Б., Стернин М.Ю., Моргоев В.К. Системы поддержки принятия решений. Препринт. М. ВНИИСИ, 1987, 42 с
148. Петровский А.Б.. Теория принятия решений. Москва, Издательский центр «Академия», 2009.
149. Плесневич Г.С. Логические модели//Искусственный интеллект. В 3-х кн. Кн.2. Модели и методы: Справочник/Под. Ред. Д.А. Поспелова. -М.: Радио и связь, 1990. -С. 14-28.
150. Попов Э.В. Экспертные системы: Решение неформализованных задач в диалоге с ЭВМ. –М.: Наука. 1987. – 288 с.
151. Поспелов Д.А. Представление знаний //Искусственный интеллект. В 3-х кн. Кн.2. Модели и методы: Справочник/Под. Ред. Д.А. Поспелова. -М.: Радио и связь, 1990. -С. 7-13

152. Поспелов Д.А. Продукционные модели //Искусственный интеллект. В 3-х кн. Кн.2. Модели и методы: Справочник/Под. Ред. Д.А. Поспелова. -М.: Радио и связь, 1990. -С. 49-56
153. Пяткова Е. В. Методика моделирования угроз энергетической безопасности с помощью байесовских сетей доверия // Современные технологии. Системный анализ. Моделирование. 2013. №3 (39) с. 133-139.
154. Энергетическая безопасность России: проблемы и пути решения / Н.И. Пяткова [и др.]; отв. ред. Воропай Н.И.; Рос. акад. наук, Сиб. отд-ние, Ин-т систем энергетики им. Л.А. Мелентьева. – Новосибирск, Изд-во СО РАН, 2011. – 198 с.
155. Радченко Г.И. Распределенные вычислительные системы: учебное пособие / Г.И. Радченко. – Челябинск: Фотохудожник, 2012. – 184 с.
156. Саати Т. Л. Принятие решений. Метод анализа иерархий. – М.: Радио и связь, 1993.
157. Семенова И.И. К вопросу разработки инструментальных средств для типового проектирования систем поддержки принятия решений // Современные наукоемкие технологии. 2010. № 4. С. 94-95.
158. Серебряков В. А. Работы Вычислительного центра РАН в области распределенных информационных систем // Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии. 2014. Т. 12, вып. 3. С. 100–123.
159. Скальный А. В. Микроэлементозы человека : (диагностика и лечение): практ. рук. для врачей и студентов мед. вузов / А.В. Скальный. - Москва, 1999. - 96 с.
160. Судаков В. А., Осипов В. П. Унификация разработки программного обеспечения прикладных СППР // Материалы XII Всероссийское совещание по проблемам управления. ВСПУ-2014 г. Москва, с. 8855-8863.

161. Сухорослов О.В. Архитектура и реализация сервис-ориентированной научной среды MathCloud/ XIII Российская конференция с участием иностранных ученых "Распределенные информационные и вычислительные ресурсы" (DICR'2010): материалы конф. – Электрон. дан. – Новосибирск: ИВТ СО РАН, 2010. – 1 электрон. опт. диск (CD-ROM).
162. Трахтенгерц Э.А, Степин Ю. П., Андреев А. Ф. Компьютерные методы поддержки принятия управленческих решений в нефтегазовой промышленности. – М.: СИНТЕГ, 2005, 592 стр.
163. Тыгу Э.Х. Концептуальное программирование. М.: Наука, 1984. 256 с.
164. Халин В.Г., Чернова Г.В., Юрков А.В. Методологические аспекты создания и функционирования систем поддержки принятия решений // Экономический анализ: теория и практика. 2015. № 7 (406). С. 20-33.
165. Частиков А. П., Гаврилова Т. А., Белов Д. Л. «Разработка экспертных систем. Среда CLIPS.». — СПб.: «БХВ-Петербург», 2003. — 608 с. — ISBN 5-94157-248-4.
166. Черняховская Л.Р., Никулина Н.О., Федорова Н.И., Малахова А.И. Разработка системы диагностических знаний с использованием интеллектуального анализа данных // В сборнике: Проблемы управления и моделирования в сложных системах. Труды XIX Международной конференции. Под редакцией Е.А. Федосова, Н.А. Кузнецова, В.А. Виттиха. 2017. С. 519-525.
167. Шалфеева Е.А. Классификация свойств онтологий. Свойства онтологий и их классификации // Научно-техническая информация, серия 2, 2005, №11. – С. 9-16
168. Яресь О. Б., Паньшин И. В. Методы принятия управленческих решений : учеб. пособие / Владим. гос. ун-т имени Александра Григорьевича и Николая Григорьевича Столетовых. – Владимир : Изд-во Владим. гос. ун-та имени

Александра Григорьевича и Николая Григорьевича Столетовых, 2011. – 66 с.
ISBN 978-5-9984-0174-9

169. Ярушкина Н.Г. Основы теории нечетких и гибридных систем. – Финансы и статистика, 2004.-320 с.

Основные публикации автора по теме диссертационной работы

Публикации в изданиях, рекомендованных ВАК

170. Загорулько Г.Б. Система ФинПлан: интеллектуальная технология решения финансовых и экономических задач // Приложение к журналу «Информационные технологии» № 6/2011. –С. 22-24.
171. Загорулько Ю.А., Загорулько Г.Б. Онтологический подход к разработке системы поддержки принятия решений на нефтегазодобывающем предприятии // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2012. Том.10, выпуск 1. –С. 121-128
172. Загорулько Г.Б., Молородов Ю.И., Федотов А.М. Систематизация знаний по теплофизическим свойствам веществ.//Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2014 - Том 12, Выпуск № 3. -С.48-56.
173. Загорулько Ю.А., Загорулько Г.Б., Боровикова О.И. Технология создания тематических интеллектуальных научных интернет-ресурсов, базирующаяся на онтологии // Программная инженерия. 2016. № 2. С. 51–60.
174. Загорулько Г.Б. Разработка онтологии для интернет-ресурса поддержки принятия решений в слабоформализованных областях // Онтология проектирования. – 2016. – Т. 6, №4(22). - С. 485-500.
175. Загорулько Г.Б. Модель комплексной поддержки разработки интеллектуальных СППР // Онтология проектирования. – 2019. – Т. 9, №4(34). - С. 462-479.

176. Загорулько Г.Б., Массель Л.В. Разработка интеллектуальной СППР по предотвращению угроз энергетической безопасности //Вестник СибГУТИ. 2019. №3. С.70-79.

Публикации в изданиях, индексируемых в SCOPUS и Web of Science

177. Zagorulko Yury, Zagorulko Galina. Ontology-Based Approach to Development of the Decision Support System for Oil-and-Gas Production Enterprise // New Trends in Software Methodologies, Tools and Techniques. Proceedings of the 9th SoMeT_10. Hamido Fujita (Eds.) –IOS Press, -Amsterdam,. –2010. –P.457-466.
178. Zagorulko Yury, Zagorulko Galina. Architecture of Extensible Tools for Development of Intelligent Decision Support Systems // New Trends in Software Methodologies, Tools and Techniques. Proceedings of the 10th SoMeT_11. Hamido Fujita (Eds.) –IOS Press, -Amsterdam. –2011. –P.457-466.
179. Braginskaya Ludmila; Kovalevsky Valery; Grigoryuk Andrey; Zagorulko Galina. Ontological approach to information support of investigations in active seismology // Proceedings of the 2nd Russian-Pacific Conference on Computer Technology and Applications, Vladivostok, Russky Island, Russia, 25-29 September, 2017 – P. 27–29. ISBN: 978-1-5386-1206-4
180. Zagorulko Yu., Borovikova O., Zagorulko G. Pattern-Based Methodology for Building the Ontologies of Scientific Subject Domains. In: New Trends in Intelligent Software Methodologies, Tools and Techniques. Proceedings of the 17th International Conference SoMeT_18. H. Fujita and E. Herrera-Viedma (Eds.). Series: Frontiers in Artificial Intelligence and Applications, vol. 303. Amsterdam: IOS Press, 2018. P. 529–542. DOI 10.3233/978-1-61499-900-3-529. (Scopus)
181. Molorodov Yu.I., Zagorulko G.B., Vishnev K.E. Tools for integrating intelligent scientific internet resources with distributed data sources // Eurasian Journal of Mathematical and Computer Applications. - 2018. - Vol.6. - Iss. 3. - P.45-52. - ISSN 2306-6172. - EISSN 2308-9822.

Доклады на конференциях и публикации в отечественных журналах

182. Загорулько Г. Б., Сидоров В. А., Телерман В. В., Ушаков Д. М. NeMo+: Объектно-ориентированная система программирования в ограничениях на основе недоопределенных моделей // Труды 6-й национальной конференции по искусственному интеллекту — КИИ'98. Пущино, 1998. -С.524-530.
183. Загорулько Ю.А., Загорулько Г.Б. Использование онтологий в экспертных системах и системах поддержки принятия решений // Труды Второго симпозиума «Онтологическое моделирование» (Казань, октябрь 2010 г.) – М.: ИПИ РАН, 2011. –С. 321-351.
184. Загорулько Г.Б. Обеспечение информационной поддержки разработчиков СППР // Информационные и математические технологии в науке и управлении / Труды XVIII Байкальской Всероссийской конференции «Информационные и математические технологии в науке и управлении». Часть III. – Иркутск: ИСЭМ СО РАН, 2013. –С. 137–142.
185. Загорулько Г.Б., Шмаков Е.С. Онтологический Подход к Разработке Интеллектуальных СППР на Основе Прецедентов // Материалы Всероссийской конференции с международным участием «Знания – Онтологии – Теории» (ЗОНТ–2013), 8-10 октября 2013 г., Новосибирск. – Новосибирск: Институт математики им. С.Л. Соболева СО РАН, 2013. –Т1. – С. 157-164.
186. Загорулько Г.Б., Григорьев С.Ю. Подход к разработке репозитария методов поддержки принятия решений // Тр. XIX Байкальской Всероссийской конференции "Информационные и математические технологии в науке и управлении". – Иркутск: Институт систем энергетики им. Л.А. Мелентьева СО РАН. –2014. –Т.2. –С.149-156.
187. Загорулько Г.Б. Сервис-ориентированный подход к разработке интеллектуальных научных интернет-ресурсов // Информационные и математические технологии в науке и управлении / Труды XX Байкальской

- Всероссийской конференции «Информационные и математические технологии в науке и управлении». Часть III. – Иркутск: ИСЭМ СО РАН, 2015. –С. 97-104.
188. Загорулько Г.Б., Сидоров В.А. Метод недоопределенных вычислений как средство поддержки принятия решений в слабоформализованных предметных областях // Информационные и математические технологии в науке и управлении. Иркутск: ИСЭМ СО РАН. 2016. № 4-1. – С. 27-36
 189. Загорулько Г.Б. Разработка методов экспертного оценивания для интеллектуального научного интернет-ресурса по поддержке принятия решений // 26-я Международная Крымская конференция «СВЧ-техника и телекоммуникационные технологии» (КрыМиКо'2016). Севастополь, 4–10 сентября 2016 г. : материалы конф. в 13 т. – Москва ; Минск ; Севастополь, 2016. – Т. 4. –С. 852-858.
 190. Загорулько Ю.А., Загорулько Г.Б. Инженерия знаний : учеб. пособие. / Ю. А. Загорулько, Г. Б. Загорулько; Новосиб. гос. ун-т. – Новосибирск : РИЦ НГУ, 2016. – 93 с.
 191. Загорулько Ю.А., Боровикова О.И., Загорулько Г.Б. Применение паттернов онтологического проектирования при разработке онтологий научных предметных областей // Аналитика и управление данными в областях с интенсивным использованием данных: сборник научных трудов XIX Международной конференции DAMDID/RCDL'2017 (10-13 октября 2017 года, Москва, МГУ, Россия) , 2017. С. 332– 340.
 192. Загорулько Ю.А., Загорулько Г.Б., Шестаков В.К. Подход к разработке информационно-аналитического интернет-ресурса по поддержке принятия решений // Информационные технологии и системы [Электронный ресурс]: тр. Шестой Междунар. науч. конф., Банное, Россия, 1-5 марта 2017 г. (ИТиС - 2017) : науч. электрон. изд. (1 файл 28,0 Мб) / отв. ред. Ю. С. Попков, А. В.

Мельников. – Челябинск : Изд-во Челяб. гос. ун-та, 2017. – С. 113–116.
(РИНЦ)

193. Загорулько Г.Б., Молородов Ю.И., Вишнев К.Е. Разработка системы доступа к данным из внешних источников для интеллектуальных научных интернет-ресурсов// Информационные и математические технологии в науке и управлении. —2018.— № 3 (7). — С. 144–153.
194. Загорулько Г.Б. Методические аспекты разработки интеллектуальных СППР в слабоформализованных предметных областях на основе информационно-аналитических ресурсов // Шестнадцатая Национальная конференция по искусственному интеллекту с международным участием КИИ-2018 (24–27 сентября 2018 г., г. Москва, Россия). Труды конференции. В 2-х томах. М.: РКП, 2018. Т. 1, С. 36–43.
195. Загорулько Г.Б. Методология разработки интеллектуальных СППР и ее применение для задач медицинской диагностики. // Информационные технологии и системы. Труды Седьмой Всероссийской научной конференции с международным участием Ханты-Мансийск, Россия, 12–16 марта 2019 года. –с.211-217

Свидетельство о государственной регистрации программы для ЭВМ

196. Свидетельство о государственной регистрации программы для ЭВМ №2019619698. Программный комплекс для разработки интеллектуальных систем поддержки принятия решений в слабоформализованных предметных областях СПОРА+ / Загорулько Ю.А., Загорулько Г.Б., Боровикова О.И., Шестаков В.К., Серый А.С. – №2019618680; заявл. 16.07.2019; зарег. в реестре программ для ЭВМ 23.07.2019 г.

Приложение А. Описание синтаксиса и семантики дескрипционной логики SOIN(D)

Таблица 1. Конструкторы и правила интерпретации (семантика) составных концептов и ролей

Название	Синтаксис	Семантика	Описание
Top	$\top$	Δ	Универсальный класс (сущность) – множество всех индивидов
Bottom	$\perp$	$\emptyset$	Универсальный класс – пустое множество
Complement	$\neg C$	$\Delta \setminus f(C)$	Дополнение концепта
Intersection	$C \sqcap D$	$f(C) \cap f(D)$	Пересечение концептов
Union	$C \sqcup D$	$f(C) \cup f(D)$	Объединение концептов
Value restriction	$\forall R.C$	$\{ a \in \Delta \mid \forall b (a, b) \in f(R) \rightarrow b \in f(C) \}$	Ограничение значений концепта
Limited Existential quantification	$\exists R.C$	$\{ a \in \Delta \mid \exists b (a, b) \in f(R) \wedge b \in f(C) \}$	Ограничение значений концепта
Number restrictions	$\geq n R$	$\{ a \in \Delta \mid \{ b \in \Delta \mid (a, b) \in f(R) \} \geq n \}$	Ограничение значений концепта
	$\leq n R$	$\{ a \in \Delta \mid \{ b \in \Delta \mid (a, b) \in f(R) \} \leq n \}$	
	$= n R$	$\{ a \in \Delta \mid \{ b \in \Delta \mid (a, b) \in f(R) \} = n \}$	
Complement	$\neg R$	$\Delta \times \Delta \setminus f(R)$	Дополнение роли
Inverse	R^{-}	$\{ (b, a) \in \Delta \times \Delta \mid (a, b) \in f(R) \}$	Обратная роль
Role restriction	$R _C$	$f(R) \cap (\Delta \times f(C))$	Ограничение роли
Nominal	I	$f(I) \subseteq \Delta \wedge f(I) = 1$	Индивиды

Таблица 2. Терминологические аксиомы и аксиомы-утверждения

Название	Синтаксис	Семантика	Описание
Concept inclusion	$C \sqsubseteq D$	$C^I \subseteq D^I$	
Role inclusion	$R \sqsubseteq S$	$R^I \subseteq S^I$	Универсальный класс – пустое множество
Concept equality	$C \equiv D$	$C^I = D^I$	Дополнение концепта
Role equality	$R \equiv S$	$R^I = S^I$	Пересечение концептов
Concept assertion	$C(a)$	$a^I \in C^I$	Объединение концептов
Role assertion	$R(a, b)$	$(a^I, b^I) \in R^I$	Ограничение значений концепта

Приложение Б. Базовые онтологии. Паттерны онтологического проектирования

Технология разработки интеллектуальных научных интернет-ресурсов включает технологию разработки онтологий, на основе которых строятся эти ресурсы. Согласно технологии эти онтологии, в свою очередь, строятся на основе взаимосвязанных базовых онтологий научного знания, научной деятельности, задач и методов и онтологии Информационных ресурсов.

Онтология научного знания содержит такие важные понятия для любой научной области знаний понятия, как Объект исследования, Предмет исследования, Метод исследования, Научный результат.

Онтология научной деятельности является онтологией верхнего уровня и включает базовые понятия, относящиеся к организации научно-исследовательской деятельности, такие как Персона, Организация, Событие, Деятельность, Публикация, используемые для описания участников научной деятельности, мероприятий, научных программ и проектов, различного типа публикаций.

Онтология Задач и методов включает понятия, описывающие решаемые в рамках рассматриваемой предметной области задачи – Задача, Метод, Результат, Входные данные, Сервис.

Во всех научных областях большую роль играет понятие Интернет-ресурса. Интернет-ресурсы могут быть связаны со всеми понятиями онтологии ОЗ. Их описание строится на основе базовой онтологии научных информационных ресурсов, включающих класс Информационный ресурс в качестве основного класса. Набор атрибутов и связей этого класса основан на стандарте Dublin core.

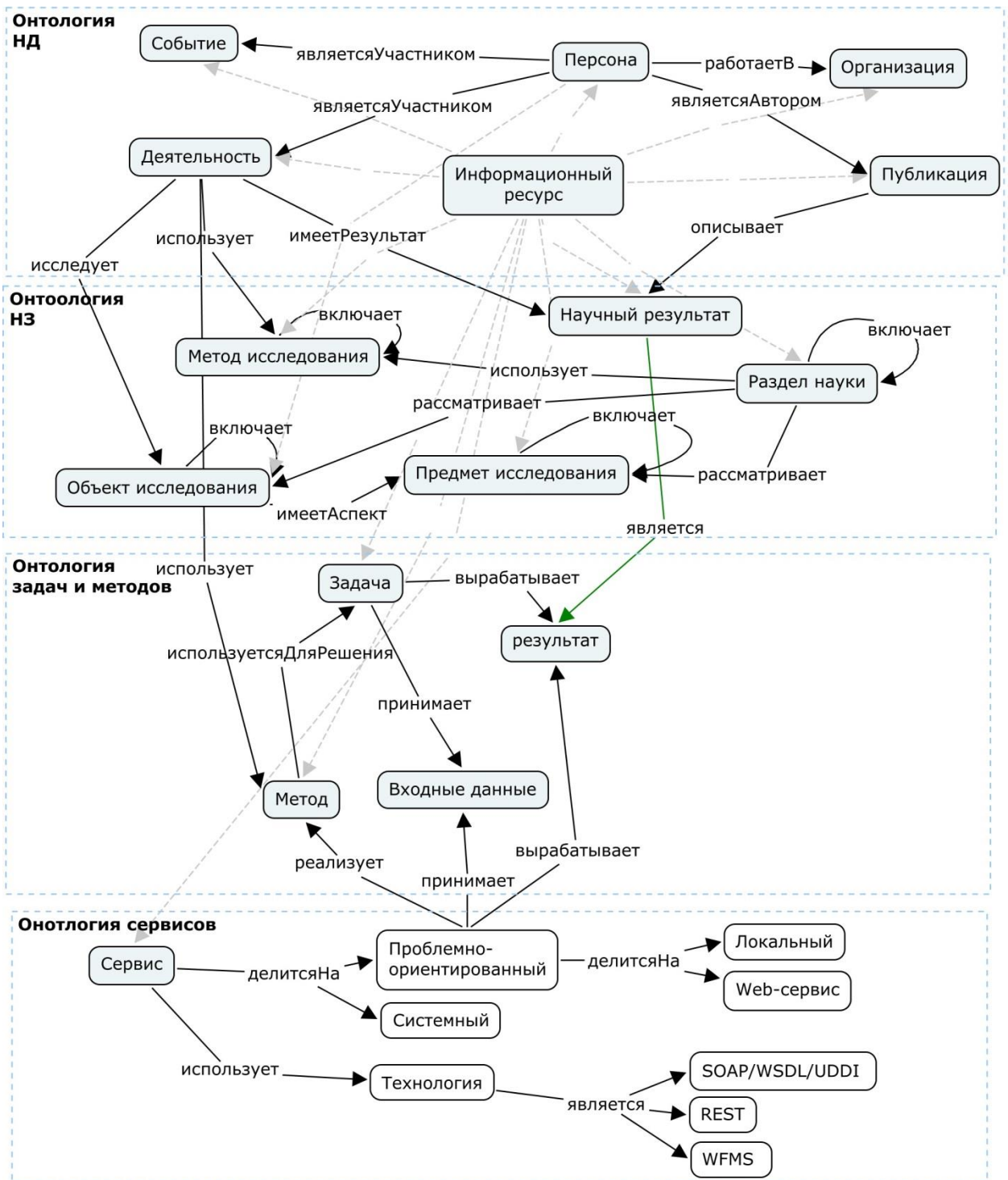


Рисунок Б.1. Базовые онтологии

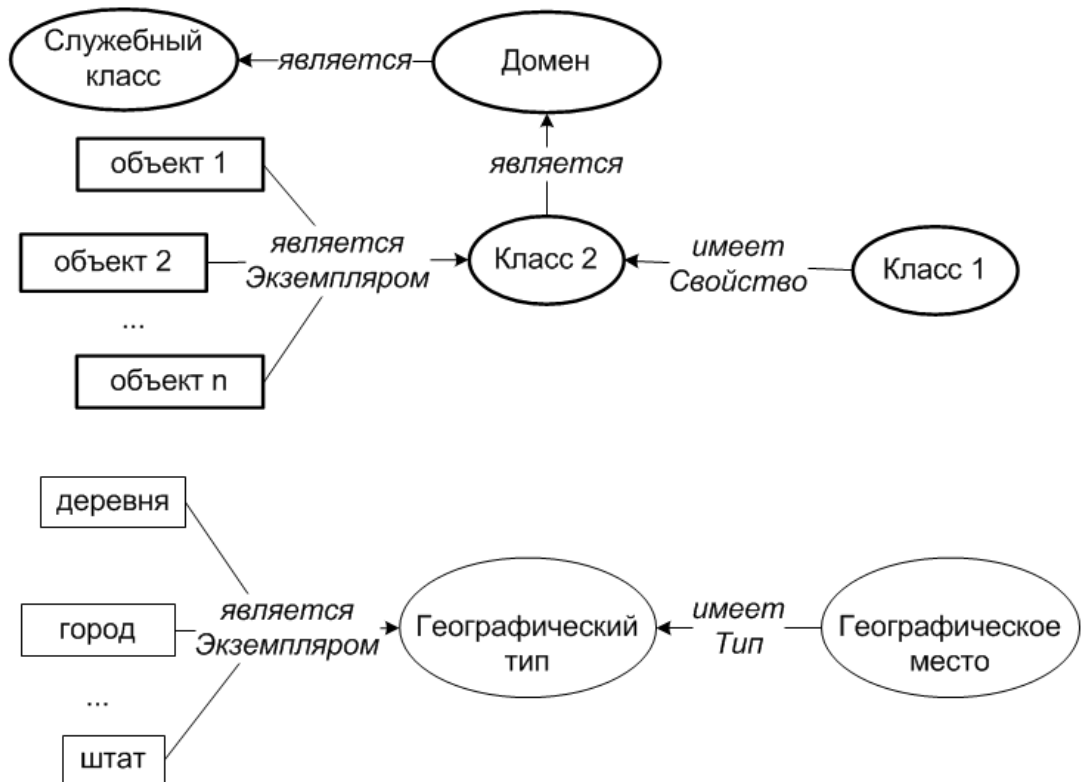


Рисунок Б.2. Паттерн представления доменов

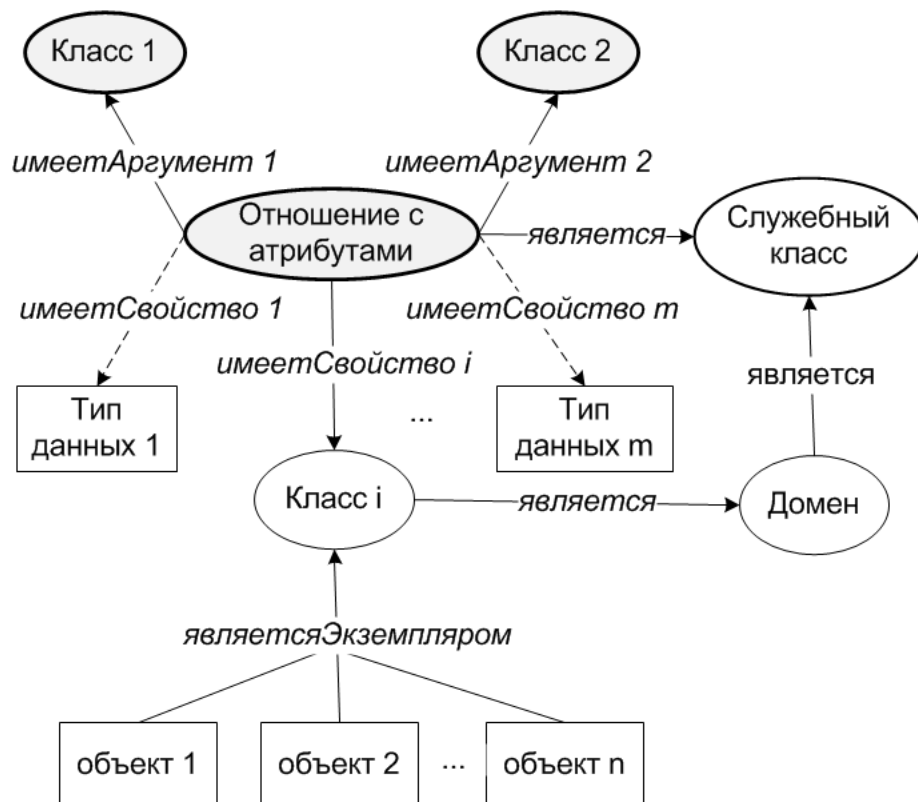


Рисунок Б.3. Паттерн представления отношения с атрибутами

Приложение В. Онтология ППР. Визуальное представление методов поддержки принятия решений

В данном приложении представлены фрагменты онтологии методов поддержки принятия решений, описывающие отдельные методы. Визуальное представление фрагментов выполнено средствами графического редактора СМар [37].

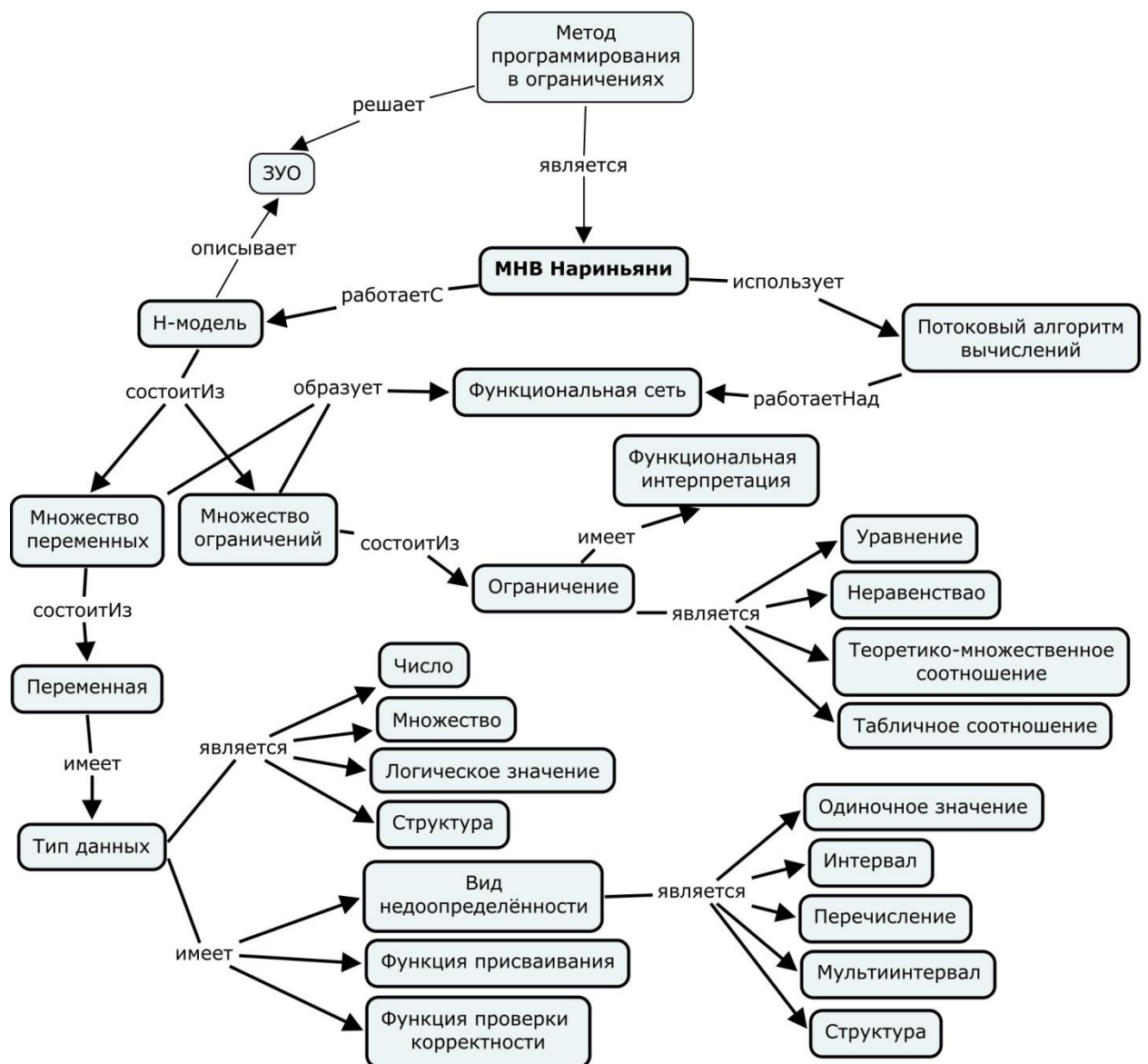


Рисунок В.1. Метод неопределенных вычислений

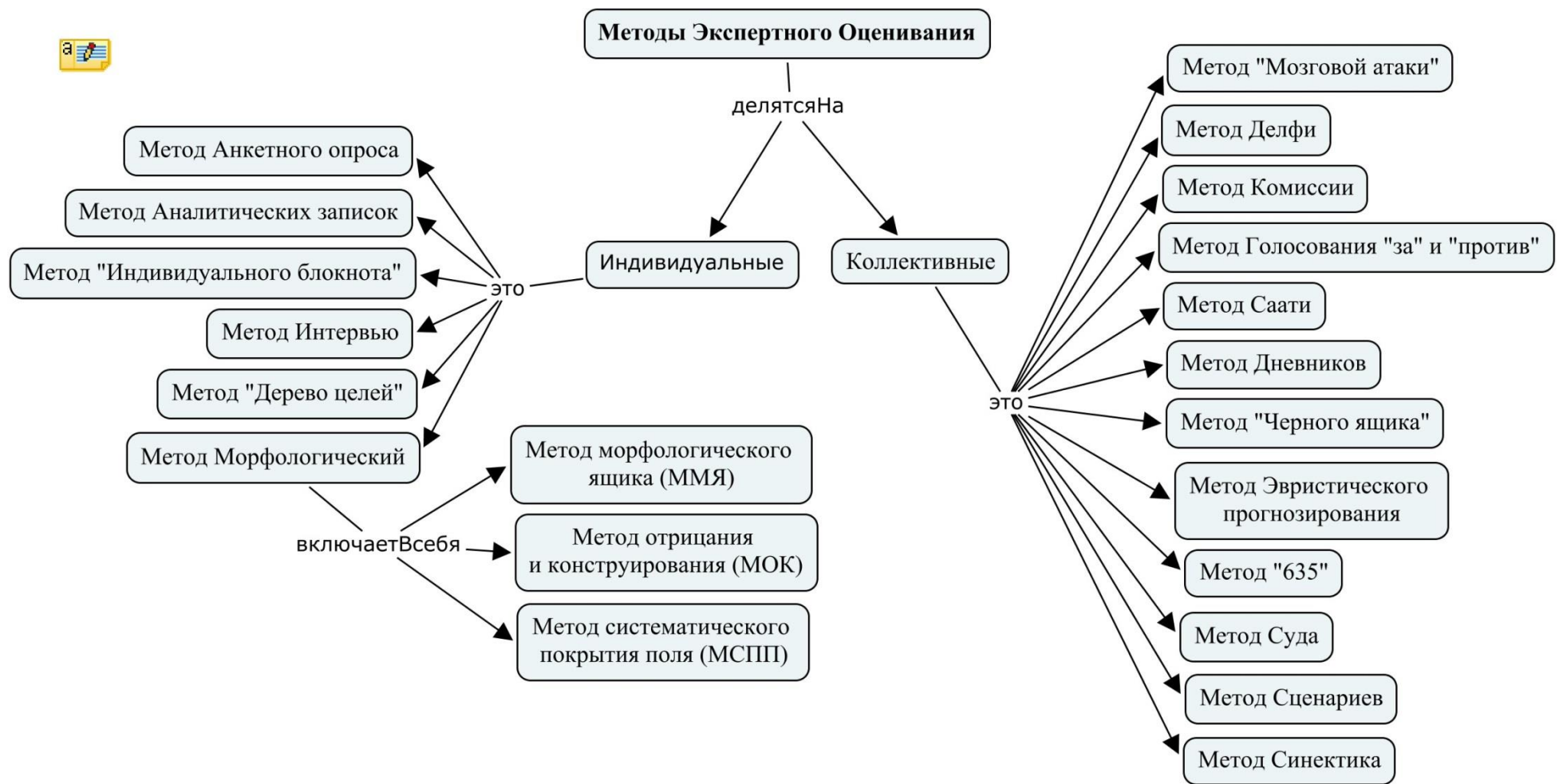


Рисунок В.2. Иерархия методов экспертного оценивания

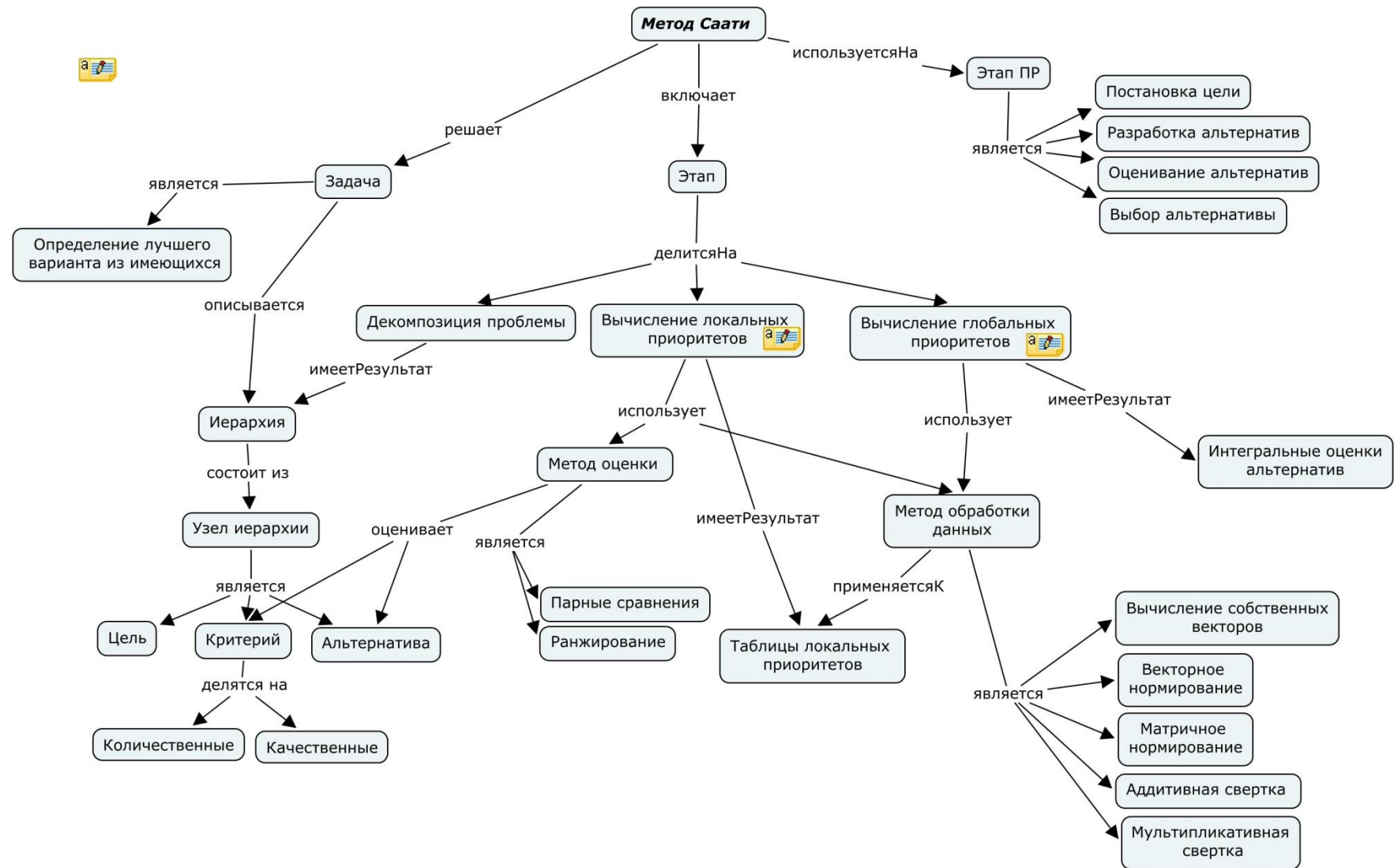


Рисунок В.3. Метод Саати

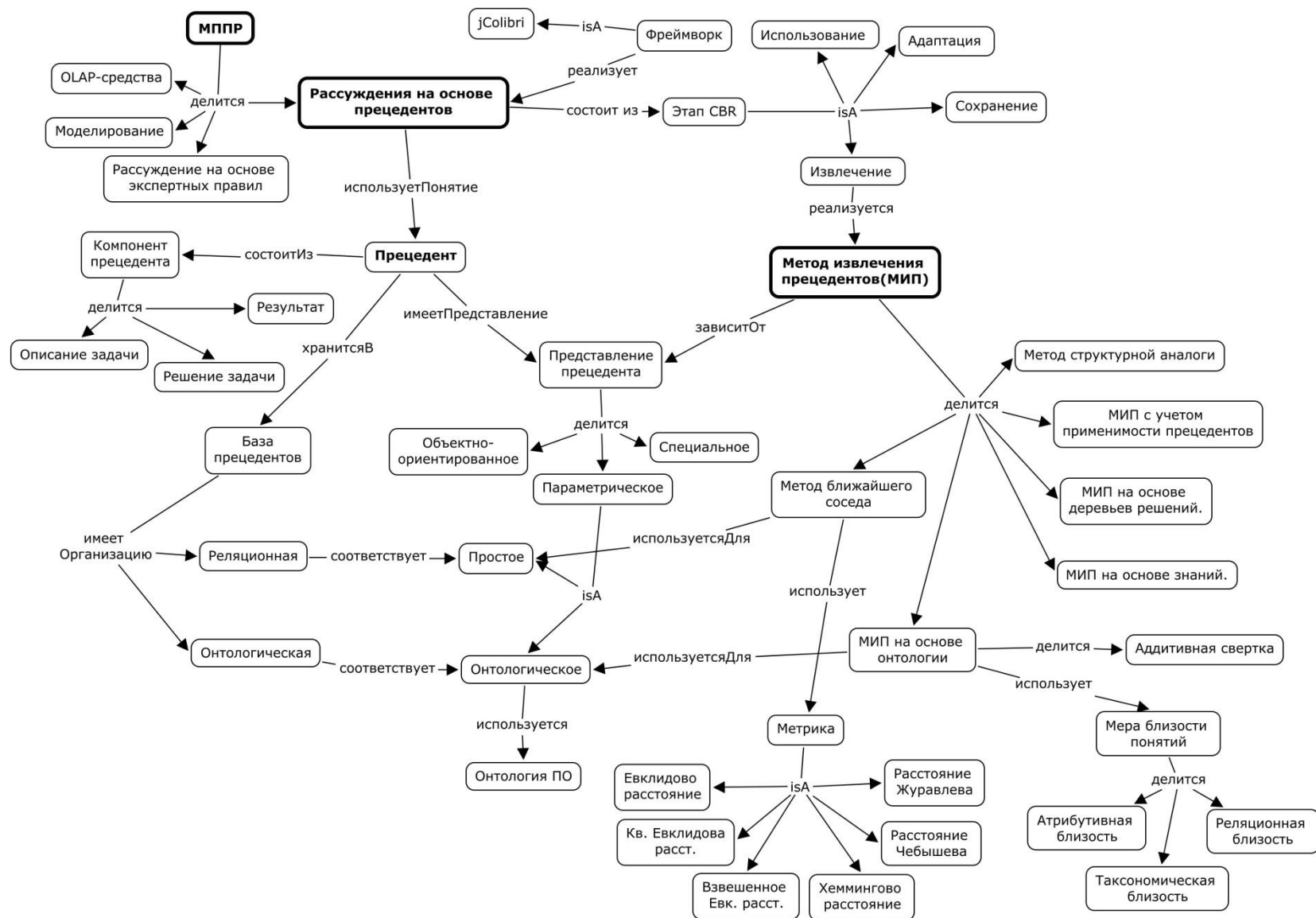


Рисунок В.4. Метод рассуждений на основе прецедентов



Рисунок В.5. Метод когнитивного моделирования

Приложение Г. Программная реализация методов. Фрагменты программного кода

Листинг Г.1. Сервис обработки массивов данных

```
//stack.h
#include <stdio.h>
#include <malloc.h>

#define STACK struct stack
#define N 8

STACK {
    int item;
    STACK *next;
};

void push(STACK **ppStack, STACK * nItem);
int pop(STACK **ppStack);
int peek(STACK *ppStack);
void display(STACK *pStack, double ms[N][N], double sum, int v);
STACK * create (int n);
int nal(STACK *ps, int n, int k);

//stack.c
#include "stack.h"
STACK * create (int n)
{
    STACK *pNewItem;
    pNewItem = (STACK *)malloc(sizeof(STACK));
    pNewItem->item = n;
    pNewItem->next=0;
    return pNewItem;
}

void push1(STACK **ppStack, STACK * nItem) {
    nItem->next = *ppStack;
    *ppStack = nItem;
}

int pop1(STACK **pSt)
{
    STACK *pPrevItem=(*pSt)->next;
    int oldItem=0;
    if (*pSt)
    {
```

```

        oldItem=(*pSt)->item;
        free(*pSt);
        *pSt=pPrevItem;
    }
    return oldItem;
}
int peek1(STACK *pStack)/*прочитать элемент в вершине стека, не извлекая его
оттуда*/
{
    if (pStack)
        return pStack->item;
    else
        return 0;
}
void display(STACK *pStack)
{
    if (pStack)
    {
        printf("%d\n",pStack->item);
        display(pStack->next);
    }
}
//stack.cpp
#include <malloc.h>
#include "stack.h"

STACK * create (int n)
{
    STACK *pNewItem;

    pNewItem = (STACK *)malloc(sizeof(STACK));
    pNewItem->item = n;
    pNewItem->next=0;
    return pNewItem;
}
void push(STACK **ppStack, STACK * nItem) /* добавить в стек ppStack элемент
nItem*/
{
    nItem->next = *ppStack;
    *ppStack = nItem;
}
int pop(STACK **pSt)
{
    STACK *pPrevItem=(*pSt)->next;
    int oldItem=0;

    if (*pSt)
    {
        oldItem=(*pSt)->item;

```

```

        free(*pSt);
        *pSt=pPrevItem;
    }
    return oldItem;
}
int peek(STACK *pStack)/*прочитать элемент в вершине стека, не извлекая его
оттуда*/
{
    if (pStack)
        return pStack->item;
    else
        return 0;
}
int nal(STACK *ps, int n, int k)
{
    if (ps)
    {
        if (ps->item == n)
        {
            k = 1;
            return k;
        }
        else nal(ps->next, n, k);
    }
    else return k;
}
void display(STACK *pStack, double ms[N][N], double sum, int v)
{
    if (pStack)
    {
        printf("%d\n",pStack->item);
        sum += ms[pStack->item][v];
        v = pStack->item;
        display(pStack->next, ms, sum, v);
    }
    else printf("%f\n", sum);
}
//Grahp.cpp
#include "stack.h"
void path(STACK **ps)
{
    int j;
    int top = peek(*ps);
    if (ps)
    {
        for (j = 0; j < N; j++)
            if ((ms[top][j] != 0) && (!nal(*ps, j, 0)))
            {
                push(ps, create(j));
            }
    }
}

```

```

        if (j == v2)
        {
            display(*ps, ms, 0, v2);
            printf("\n");
        }
        path(ps);
    }
    pop(ps);
}

int main()
{
    STACK *ps = 0;
    push(&ps, create(v1));
    path(&ps);

    return 0;
}

//Xmp2cpp.cpp парсер
#include <iostream>
#include <cstdio>
#include <fstream>
#include <string>
#define N 10

using namespace std;
int max(int a, int b)
{
    if (a > b) return a;
    else return b;
};

int main()
{
    string fname = "map.cxm";
    //getline(cin, fname);

    ifstream fin;
    fin.open(fname);

    if(!fin.is_open()) // если файл не открыт
        cout << "Файл не может быть открыт!\n"; // сообщить об этом
    string Rel = "isem.grmcognitivemap.Relation";
    string Item = "isem.technology.Universal";
    string Un = "Universal";
    string nameV = "Наименование";
    string Val = "value";
    int lenUn=63;// длина строки до номера элемента
    int lenS = 8; // длина стринга
    string names[N], str; // массив имен

```



```

int count = 0;
int mas[N][N]; // матрица смежности
for (int i = 0; i < N; i++)
    for (int j = 0; j < N; j++)
        mas[i][j] = 0;

int from, to, val;
while (!fin.eof())
{
    getline(fin, str);
    if (str.find(Rel) != std::string::npos) //нашли строку со стрелкой
    {
        while (str.find(Un) == std::string::npos)
            getline(fin, str);
        // выяснение номера вершины
        if (str.find(Item) != std::string::npos)
        {
            str.erase(0, lenUn);
            str.erase(str.end() - 2, str.end());
            from = stoi(str);
            while (str.find(nameV) == std::string::npos)
                getline(fin, str);
            getline(fin, str);
            str.erase(0, lenS);
            str.erase(str.end() - 9, str.end());
            names[from] = str;
        }
        else
        {
            str.erase(0, lenUn);
            str.erase(str.end() - 2, str.end());
            from = stoi(str);
        }
        count = max(count, from);
        //значение на дуге
        while (str.find(Val) == std::string::npos)
            getline(fin, str);
        getline(fin, str);
        str.erase(0, lenS);
        str.erase(str.end() - 9, str.end());
        val = stoi(str);
        for (int i=0; i<4; i++)
            getline(fin, str);
        if (str.find("Направление") != std::string::npos)
            val *= -1;
        while (str.find(Un) == std::string::npos)
            getline(fin, str);
        // выяснение номера вершины
        if (str.find(Item) != std::string::npos)
        {

```

```

        str.erase(0, lenUn);
        str.erase(str.end() - 2, str.end());
        to = stoi(str);
        while (str.find(nameV) == std::string::npos)
            getline(fin, str);
        getline(fin, str);
        str.erase(0, lenS);
        str.erase(str.end() - 9, str.end());
        names[to] = str;
    }
    else
    {
        str.erase(0, lenUn);
        str.erase(str.end() - 2, str.end());
        to = stoi(str);
    }
    count = max(count, to);
    mas[from][to] = val;
}
}
for (int i = 0; i <= count; i++)
{
    for (int j = 0; j <= count; j++)
        cout << mas[i][j] << " ";
    cout << "\n";
}
for (int i = 0; i <= count; i++)
{
    cout << names[i] << "\n";
}
fin.close();
return 0;
}

```

Листинг Г.2. Сервис обработки таблиц Excel

```

#include <iostream>
#include "libxl.h"
#include <cstdlib>
#include <math.h>
#include <stdio.h>
#include <locale>
#include <windows.h>
using namespace libxl;
int main()
{
    Book* book = xlCreateBook();
    if(book->load(L"input.xls"))
    {
        Sheet* sheet = book->getSheet(0);
        if(sheet)

```

```

{
    std::wcout.imbue(std::locale("rus_rus.866"));
    const wchar_t* s = sheet->readStr(0, 1);
    std::wcout << s << "\n";
    std::wcout << s[0] << "\n";
    for(int row = sheet->firstRow(); row < sheet->lastRow(); ++row)
    {
        for(int col = sheet->firstCol() + 1 ; col < sheet->lastCol(); ++col)
        {
            CellType cellType = sheet->cellType(row, col);
            if(sheet->isFormula(row, col))
            {
                const wchar_t* s = sheet->readFormula(row, col);
                std::wcout << (s ? s : L"null") << " [formula]";
            }
            else
            {
                switch(cellType)
                {
                    case CELLTYPE_EMPTY: std::wcout << "[empty]";
                    break;
                    case CELLTYPE_NUMBER:
                    {
                        double d = sheet->readNum(row, col);
                        Matrix[col-1] = d;
                        std::wcout << d ;
                        break;
                    }
                    case CELLTYPE_STRING:
                    {
                        const wchar_t* s = sheet->readStr(row, col);
                        double f=_wtof(s) ;
                        std::wcout << f << "\n";
                        Matrix[col-1] = f;
                        break;
                    }
                    case CELLTYPE_BLANK: std::wcout << "[blank]"; break;
                    case CELLTYPE_ERROR: std::wcout << "[error]"; break;
                }
            }
        }
        std::wcout << std::endl; } } }
    }
    book->release();
    return 0;
}

```

Листинг Г.3. Сервис взаимодействия с внешними БД

```

//table.proto
syntax = "proto3";
message ContentForLoad {

```

```

int32 service_id = 1;
string content = 2;
int32 number_of_objects = 3;
message OntologyObject {
  int32 db_id = 1;
  int32 temp_id=2;
  int32 number_of_parameters=3;
  enum ParamType {
    STR=0;
    NUM=1;
  }
  message Description {
    string name=1;
    ParamType type=2;
    string val=3;
  }
  repeated Description parameters=4;
}
repeated OntologyObject data = 4;
}

```

```
//Protocol11.php
```

```
<?php
```

```
# Generated by the protocol buffer compiler. DO NOT EDIT!
```

```
# source: protocol11.txt
```

```
namespace GPBMetadata;
```

```
class Protocol11
```

```

{
  public static $is_initialized = false;
  public static function initOnce() {
    $pool = \Google\Protobuf\Internal\DescriptorPool::getGeneratedPool();
    if (static::$is_initialized == true) {
      return;
    }
    $pool->internalAddGeneratedFile(hex2bin(
      "0a89030a0e70726f746663666c31312e74787422ee020a0e436f6e746556" .
      "74466f724c6f6164120f0a07636f6e7465567418012001280912190a116e" .
      "756d6265725f6f665f6f626a65637473180220012805122c0a0464617461" .
      "18032003280b321e2e436f6e74655674466f724c6f61642e4f6e746f6c6f" .
      "67794f626a6563741a81020a0e4f6e746f6c6f67794f626a65637412100a" .
      "0863617465676f7279180120012805121c0a146e756d6265725f6f665f70" .
      "6172616d6574657273180320012805123e0a0a706172616d657465727318" .
      "042003280b322a2e436f6e74655674466f724c6f61642e4f6e746f6c6f67" .
      "794f626a6563742e4465736372697074696f6e1a600a0b44657363726970" .
      "74696f6e120c0a046e616d6518012001280912360a047479706518022001" .
      "280e32282e436f6e74655674466f724c6f61642e4f6e746f6c6f67794f62" .
      "6a6563742e506172616d54797065120b0a0376616c180320012809221d0a" .
      "09506172616d5479706512070a03535452100012070a034e554d10016206" .
      "70726f746f33"
    ));
  }
}

```

```

        static::$is_initialized = true;
    }
}

//ContentForLoad.php
<?php
# Generated by the protocol buffer compiler.  DO NOT EDIT!
# source: protocol11.txt
use Google\Protobuf\Internal\GPBType;
use Google\Protobuf\Internal\RepeatedField;
use Google\Protobuf\Internal\GPBUtil;
/**
 * Protobuf type <code>ContentForLoad</code>
 */
class ContentForLoad extends \Google\Protobuf\Internal\Message
{
    /**
     * <code>string content = 1;</code>
     */
    private $content = "";
    /**
     * <code>int32 number_of_objects = 2;</code>
     */
    private $number_of_objects = 0;
    /**
     * <code>repeated .ContentForLoad.OntologyObject data = 3;</code>
     */
    private $data;

    public function __construct() {
        \GPBMetadata\Protocol11::initOnce();
        parent::__construct();
    }
    /**
     * <code>string content = 1;</code>
     */
    public function getContent()
    {
        return $this->content;
    }
    /**
     * <code>string content = 1;</code>
     */
    public function setContent($var)
    {
        GPBUtil::checkString($var, True);
        $this->content = $var;
    }
}
/**

```

```

* <code>int32 number_of_objects = 2;</code>
*/
public function getNumberOfObjects()
{
    return $this->number_of_objects;
}

/**
* <code>int32 number_of_objects = 2;</code>
*/
public function setNumberOfObjects($var)
{
    GPBUtil::checkInt32($var);
    $this->number_of_objects = $var;
}
/**
* <code>repeated .ContentForLoad.OntologyObject data = 3;</code>
*/
public function getData()
{
    return $this->data;
}

/**
* <code>repeated .ContentForLoad.OntologyObject data = 3;</code>
*/
public function setData(&$var)
{
    GPBUtil::checkRepeatedField($var, \Google\Protobuf\Internal\GPBType::MESSAGE,
\ContentForLoad_OntologyObject::class);
    $this->data = $var;
}
}
//ContentForLoad_OntologyObject.php
<?php
# Generated by the protocol buffer compiler. DO NOT EDIT!
# source: protocol11.txt
use Google\Protobuf\Internal\GPBType;
use Google\Protobuf\Internal\RepeatedField;
use Google\Protobuf\Internal\GPBUtil;
/**
* Protobuf type <code>ContentForLoad.OntologyObject</code>
*/
class ContentForLoad_OntologyObject extends \Google\Protobuf\Internal\Message
{
    /**
    * <code>int32 category = 1;</code>
    */
    private $category = 0;
    /**

```

```

* <code>int32 number_of_parameters = 3;</code>
*/
private $number_of_parameters = 0;
/**
* <code>repeated .ContentForLoad.OntologyObject.Description parameters =
4;</code>
*/
private $parameters;
public function __construct() {
    \GPBMetadata\Protocol11::initOnce();
    parent::__construct();
}
/**
* <code>int32 category = 1;</code>
*/
public function getCategory()
{
    return $this->category;
}
/**
* <code>int32 category = 1;</code>
*/
public function setCategory($var)
{
    GPBUtil::checkInt32($var);
    $this->category = $var;
}
/**
* <code>int32 number_of_parameters = 3;</code>
*/
public function getNumberOfParameters()
{
    return $this->number_of_parameters;
}
/**
* <code>int32 number_of_parameters = 3;</code>
*/
public function setNumberOfParameters($var)
{
    GPBUtil::checkInt32($var);
    $this->number_of_parameters = $var;
}
/**
* <code>repeated .ContentForLoad.OntologyObject.Description parameters =
4;</code>
*/
public function getParameters()
{
    return $this->parameters;
}

```

```

/**
 * <code>repeated .ContentForLoad.OntologyObject.Description parameters =
4;</code>
 */
public function setParameters(&$var)
{
    GPBUtil::checkRepeatedField($var, \Google\Protobuf\Internal\GPBType::MESSAGE,
\ContentForLoad_OntologyObject_Description::class);
    $this->parameters = $var;
}
}

//ContentForLoad_OntologyObject_Description.php
<?php
# Generated by the protocol buffer compiler. DO NOT EDIT!
# source: protocol11.txt
use Google\Protobuf\Internal\GPBType;
use Google\Protobuf\Internal\RepeatedField;
use Google\Protobuf\Internal\GPBUtil;
/**
 * Protobuf type <code>ContentForLoad.OntologyObject.Description</code>
 */
class ContentForLoad_OntologyObject_Description extends
\Google\Protobuf\Internal\Message
{
    /**
     * <code>string name = 1;</code>
     */
    private $name = "";
    /**
     * <code>.ContentForLoad.OntologyObject.ParamType type = 2;</code>
     */
    private $type = 0;
    /**
     * <code>string val = 3;</code>
     */
    private $val = "";
    public function __construct() {
        \GPBMetadata\Protocol11::initOnce();
        parent::__construct();
    }
    /**
     * <code>string name = 1;</code>
     */
    public function getName()
    {
        return $this->name;
    }
    /**
     * <code>string name = 1;</code>

```



```

*/
public function setName($var)
{
    GPBUtil::checkString($var, True);
    $this->name = $var;
}
/**
 * <code>.ContentForLoad.OntologyObject.ParamType type = 2;</code>
 */
public function getType()
{
    return $this->type;
}
/**
 * <code>.ContentForLoad.OntologyObject.ParamType type = 2;</code>
 */
public function setType($var)
{
    GPBUtil::checkEnum($var, \ContentForLoad_OntologyObject_ParamType::class);
    $this->type = $var;
}
/* <code>string val = 3;</code>
 */
public function getVal()
{
    return $this->val;
}
/* <code>string val = 3;</code>
 */
public function setVal($var)
{
    GPBUtil::checkString($var, True);
    $this->val = $var;
}
}
//ContentForLoad_OntologyObject_ParamType.php
<?php
# Generated by the protocol buffer compiler. DO NOT EDIT!
# source: protocol11.txt
/**
 * Protobuf enum <code>ContentForLoad.OntologyObject.ParamType</code>
 */
class ContentForLoad_OntologyObject_ParamType
{
    /**
     * <code>STR = 0;</code>
     */
    const STR = 0;
    /**
     * <code>NUM = 1;</code>

```

```

    */
    const NUM = 1;
}
//client.php
<?php
header("Content-Type: text/html; charset=utf-8");
$loader = require( __DIR__ . '/vendor/autoload.php' );
$obj_name= $_GET['obj_name'];
/*Декодируем строку. Пришлось закодировать, поскольку в ней был символ # */
$obj_name= rawurldecode($obj_name);
/*URL загрузчика*/
$url="http://217.79.61.86:80/loader/api";
require_once "D:/Servers/htdocs_aux/chem/vendor/easyrdf/easyrdf/lib/EasyRdf.php";
EasyRdf_Http::setDefaultHttpClient(new EasyRdf_Http_Client(null, array(
    'maxredirects' => 5,
    'useragent'    => 'EasyRdf_Http_Client',
    'timeout'      => 500
)));
$driver = new EasyRdf_Sparql_Client('http://84.237.72.168:3030/chem/sparql');
/*Получаем из онтологии параметры для запроса */
$test = "SELECT ?vo1 ?tl ?to
    WHERE {
        { ?o1 ?oprop1 <\".$obj_name.> .
            ?oprop1
        <http://www.semanticweb.org/ontologies/2018/03/9/ThermalProp#tablelabel> ?tl.
            ?oprop1
        <http://www.semanticweb.org/ontologies/2018/03/9/ThermalProp#tableorder> ?to.
            ?o1 ?dprop1 ?vo1.
            ?dprop1
        <http://www.semanticweb.org/ontologies/2018/03/9/ThermalProp#order>
        \"1\"^^xsd:int
        } union
        {
            <\".$obj_name.> ?dprop1 ?o1.
            ?dprop1
        <http://www.semanticweb.org/ontologies/2018/03/9/ThermalProp#tablelabel> ?tl.
            ?dprop1
        <http://www.semanticweb.org/ontologies/2018/03/9/ThermalProp#tableorder> ?to.
            <\".$obj_name.> ?dprop1 ?vo1
        }
    }
    ORDER BY ASC(?to)";
$answer = $driver->query($test);
/*Формируем данные с помощью классов Protocol Buffer*/
$mes = new ContentForLoad();
$mes->setContent("SqlTemp");
$mes->setNumberOfObjects=1;
$data = array();
$data[1] = new ContentForLoad_OntologyObject();
$data[1]->setCategory(1);

```

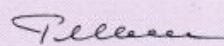
```

$data[1]->setNumberOfParameters(4);
$parameters= array();
foreach ($answer as $row) {
    $i=$row->to->getValue();
    $parameters[$i]=new ContentForLoad_OntologyObject_Description();
    $parameters[$i]->setName($row->tl->getValue());
    $parameters[$i]->setType(0);
    $parameters[$i]->setVal($row->vo1->getValue());
}
$data[1]->setParameters($parameters);
$mes->setData($data);
/*Сериализуем объект в строку для передвачи загрузчику*/
$str=$mes->SerializeToString();
// создаем подключение
$ch = curl_init($url);
// устанавливаем данные для отправки
curl_setopt($ch, CURLOPT_POSTFIELDS, $str);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);
curl_setopt($ch, CURLOPT_VERBOSE, 1);
// отправляем запрос
$response = curl_exec($ch);
/*response - это GUID запроса к внешней БД, хранящегося в загрузчике */
/*Загрузчик отдает GUID, обрамленный кавычками*/
$response = substr_replace($response, "", 0, 1);//отрезаем открывающую кавычку
$response = substr_replace($response, "", -1);//отрезаем закрывающую кавычку
curl_close($ch);
/*Вызываем визуализатор и передаем ему GUID запроса к внешней БД*/
$url_viz="http://217.79.61.189/app/".$response;

header('Location:'.$url_viz);
?>

```

**Приложение Д. Свидетельство о государственной регистрации
программ для ЭВМ. Акты о внедрении результатов
диссертационного исследования**

РОССИЙСКАЯ ФЕДЕРАЦИЯ 	
СВИДЕТЕЛЬСТВО о государственной регистрации программы для ЭВМ № 2019619698	
Программный комплекс для разработки интеллектуальных систем поддержки принятия решений в слабоформализованных предметных областях «СПОРА+»	
Правообладатель: <i>Федеральное государственное бюджетное учреждение науки Институт систем информатики им. А.П. Еришова Сибирского отделения Российской академии наук (RU)</i>	
Авторы: <i>см. на обороте</i>	
	Заявка № 2019618680 Дата поступления 16 июля 2019 г. Дата государственной регистрации в Реестре программ для ЭВМ 23 июля 2019 г. Руководитель Федеральной службы по интеллектуальной собственности  Г.П. Изrael

Министерство науки и высшего образования РФ
 ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
 БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ НАУКИ
ИНСТИТУТ СИСТЕМ ЭНЕРГЕТИКИ
 им. Л.А.МЕЛЕНТЬЕВА
 СИБИРСКОГО ОТДЕЛЕНИЯ
 РОССИЙСКОЙ АКАДЕМИИ НАУК
 (ИСЭМ СО РАН)



664033, г. Иркутск, ул. Лермонтова, 130

Тел. (3952) 42-47-00

Факс (3952) 42-67-96

E-mail: info@isem.irk.ru

<http://isem.irk.ru>

от 07.10.2009 № 1534-34-2317

на № _____ от _____

630090, г. Новосибирск,
 пр. Академика Лаврентьева, 6в
 Диссертационный совет
 Д 003.061.02
 на базе ФГБУН Институт
 вычислительной математики и
 математической геофизики
 Сибирского отделения Российской
 академии наук
 (ИВМиМГ СО РАН)

АКТ О ВНЕДРЕНИИ

Настоящий акт подтверждает, что результаты диссертационного исследования Загорулько Галины Борисовны по теме: «Модель, методы и средства комплексной поддержки разработки СППР в слабоформализованных предметных областях» обладают актуальностью и представляют практический интерес.

Полученные ею результаты были применены при совместном выполнении проектов ИСИ СО РАН и ИСЭМ СО РАН, поддержанных грантами РФФИ: №16-07-00569 (2016-2018) «Методы и средства комплексной поддержки процесса разработки интеллектуальных СППР в слабоформализованных предметных областях на основе сервис-ориентированного подхода и технологий Semantic Web» и №13-07-00422 (2013-2015) «Методы и технологии создания, анализа и обработки интеллектуальных научных Интернет-ресурсов на основе онтологий и семантических сетей».

Предложенные Загорулько Г.Б. онтологии и репозиторий методов поддержки принятия решений использовались при создании информационно-аналитического ресурса (онтологического пространства знаний в области энергетики) в составе разработанной в ИСЭМ СО РАН интеллектуальной инструментальной среды, обеспечивающей поддержку коллективной деятельности экспертов и исследователей по выработке решений по развитию энергетики. Рекомендовано дальнейшее применение разработанных ею методик и инструментальных средств для интеллектуальной поддержки исследований энергетики.

Директор ИСЭМ СО РАН,
 член-корр. РАН



В.А. Стенников

МИНОБРНАУКИ РОССИИ

Для представления в диссертационный совет

Федеральное государственное
автономное образовательное
учреждение высшего образования
«Новосибирский национальный
исследовательский государственный
университет»
(Новосибирский государственный
университет, НГУ)

**МЕХАНИКО-МАТЕМАТИЧЕСКИЙ
ФАКУЛЬТЕТ**

ул. Пирогова, д. 1, Новосибирск, 630090.
Тел. (383) 363-40-20
Адрес в интернете: mmf.nsu.ru

13.11.19г. № 291

АКТ

о внедрении результатов кандидатской диссертации
Загорулько Галины Борисовны
в образовательный процесс ММФ НГУ

Настоящий акт подтверждает, что результаты диссертационного исследования Загорулько Галины Борисовны по теме: «Модель, методы и средства комплексной поддержки разработки СППР в слабоформализованных предметных областях» используются в образовательном процессе при подготовке студентов, специализирующихся на кафедре программирования ММФ НГУ.

Информационно-аналитический интернет-ресурс, предоставляющий содержательный доступ к систематизированной на основе онтологии информации об области знаний «Поддержка принятия решений» (ППР), способствует более быстрому и качественному освоению студентами данной области знаний. Ресурс доступен по адресу <https://uniserv.iis.nsk.su/rdms/>.

Репозиторий методов ППР позволяет получить доступ к реализациям методов ППР, найти метод, необходимый для решения конкретной задачи, опробовать его, получить инструкцию по его применению и встраиванию в программные системы. Использование репозитория, а также методики разработки СППР в слабоформализованных предметных областях упрощает и ускоряет разработку прикладных систем такого класса.

Средства комплексной поддержки разработки СППР (онтология,

информационно-аналитический интернет-ресурс, репозиторий методов) задают область приложения результатов выпускных квалификационных работ бакалавров и магистров НГУ. Развитие и пополнение этих средств входило в перечень задач восьми работ, выполненных и успешно защищенных под руководством и соруководством Загорулько Г.Б. Две такие работы готовятся к защите в настоящее время.

Помимо этого, средства комплексной поддержки разработки СППР были использованы в ряде выпускных квалификационных работ для создания прикладных систем.

Декан ММФ
д.ф.-м.н., профессор РАН

И. В. Марчук

Зав. кафедрой программирования
ММФ НГУ
д.ф.-м.н., профессор

А.Г. Марчук

