

Ontology of Designing

ISSN 2223-9537 (P)

ISSN 2313-1039 (E)

ОНТОЛОГИЯ ПРОЕКТИРОВАНИЯ



Vol **10**
N **1**
2020

Научный журнал - Scientific journal

Scientific journal

Volume 10

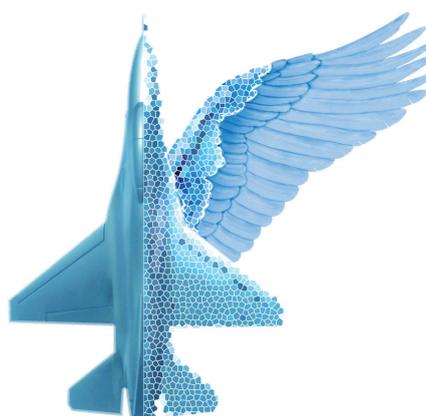
№ 1

ОНТОЛОГИЯ
ПРОЕКТИРОВАНИЯ

Научный журнал

Том 10

№ 1



Editorial Board - Редакционная коллегия

- Anatoly I. **Belousov**, Doctor of Technical Sciences, Professor, Samara University, Samara, Russia
- Nikolay M. **Borgest**, Ph.D., Professor, Samara University, Samara, Russia
- Stanislav N. **Vasiliev**, Doctor of Phys. and Math. Sciences, Professor, Academician of the RAS, ICS of the RAS, Moscow, Russia
- Vladimir G. **Gainutdinov**, Doctor of Technical Sciences, Professor, KNITU-KAI, Kazan, Russia
- Vladimir V. **Golenkov**, Doctor of Technical Sciences, Professor, BSUIR, Minsk, Belarus
- Vladimir I. **Gorodetsky**, Doctor of Technical Sciences, Professor, InfoVings LLC, St. Petersburg, Russia
- Valeriya V. **Gribova**, Doctor of Technical Sciences, Senior Researcher, IAPU FEB RAS, Vladivostok, Russia
- Yury A. **Zagorulkov**, Ph.D., Senior Researcher, ISI SB RAS, Novosibirsk, Russia
- Anton V. **Ivashchenko**, Doctor of Technical Sciences, Professor, Samara State Technical University, Samara, Russia
- Valery A. **Komarov**, Doctor of Technical Sciences, Professor, Samara University, Samara, Russia
- Vladik **Kreinovich**, Ph.D., Professor, University of Texas at El Paso, El Paso, USA
- Victor M. **Kurechik**, Doctor of Technical Sciences, Professor, Southern Federal University, Taganrog, Russia
- Dmitry V. **Lande**, Doctor of Technical Sciences, Senior Researcher, IPRI NAS of Ukraine, Kiev, Ukraine
- Lyudmila V. **Massel**, Doctor of Technical Sciences, Professor, ISEM SB RAS, Irkutsk, Russia
- Aleksandr Yu. **Nesterov**, Doctor of Philosophy, Professor, Samara University, Samara, Russia
- Dmitry A. **Novikov**, Doctor of Technical Sciences, Professor, Corresponding Member of RAS, ICS RAS, Moscow, Russia
- Alexander V. **Palagin**, Doctor of Technical Sciences, Professor, Academician of the NASU, Ins. of Cybernetics, Kiev, Ukraine
- Semyon A. **Piyavsky**, Doctor of Technical Sciences, Professor, Moscow City Pedagogical University, Samara, Russia
- Yury M. **Reznik**, Doctor of Philosophy, Professor, Institute of Philosophy of the RAS, Moscow
- George **Rzevski**, Professor, Open University, London, UK
- Peter O. **Skobelev**, Doctor of Technical Sciences, «Smart solutions» Scientific Production C., Samara, Russia
- Sergey V. **Smirnov**, Doctor of Technical Sciences, ICCS RAS - SamRC RAS, member of IAOA, Samara, Russia
- Dzhavdet S. **Suleymanov**, Doctor of Technical Sciences, Professor, Academician of the Republic of Tatarstan, Kazan, Russia
- Boris E. **Fedunov**, Doctor of Technical Sciences, Professor, State Research Institute of Aviation Systems, Moscow, Russia
- Altynbek **Sharipbay**, Doctor of Technical Sciences, Professor, Institute of Artificial Intelligence, Nur-Sultan, Kazakhstan
- Boris Ya. **Shvedin**, Candidate of Psychological Sciences, Dan Rose LLC, Member of IAOA, Rostov-on-Don, Russia
- Белоусов** Анатолий Иванович, д.т.н., профессор, Самарский университет, Самара, Россия
- Боргест** Николай Михайлович*, к.т.н., профессор, Самарский университет, член IAOA, Самара, Россия
- Васильев** Станислав Николаевич*, д.ф.-м.н., профессор, академик РАН, ИПУ РАН, Москва, Россия
- Гайнутдинов** Владимир Григорьевич, д.т.н., профессор, КНИТУ-КАИ, Казань, Россия
- Голенков** Владимир Васильевич*, д.т.н., профессор, БГУИР, Минск, Беларусь
- Городецкий** Владимир Иванович*, д.т.н., профессор, ООО «ИнфоВингс», Санкт-Петербург, Россия
- Грибова** Валерия Викторовна*, д.т.н., с.н.с., ИАПУ ДВО РАН, Владивосток, Россия
- Загорюлько** Юрий Алексеевич*, к.т.н., с.н.с., ИСИ СО РАН, Новосибирск, Россия
- Иващенко** Антон Владимирович, д.т.н., профессор, СамГТУ, Самара, Россия
- Комаров** Валерий Андреевич, д.т.н., профессор, Самарский университет, Самара, Россия
- Креинович** Владик, профессор, Техасский университет Эль Пасо, Эль Пасо, США
- Курейчик** Виктор Михайлович*, д.т.н., профессор, Южный федеральный университет, Таганрог, Россия
- Ландэ** Дмитрий Владимирович*, д.т.н., с.н.с., ИПРИ НАН Украины, Киев, Украина
- Масель** Людмила Васильевна*, д.т.н., профессор, ИСЭМ СО РАН, Иркутск, Россия
- Нестеров** Александр Юрьевич, д.филос.н., профессор, Самарский университет, Самара, Россия
- Новиков** Дмитрий Александрович, д.т.н., проф., член-корреспондент РАН, ИПУ РАН, Москва, Россия
- Палагин** Александр Васильевич, д.т.н., проф., академик НАН Украины, Ин-т кибернетики, Киев, Украина
- Пиавский** Семён Авраамович, д.т.н., профессор, Московский город. педагог. университет, Самара, Россия
- Резник** Юрий Михайлович, д.филос.н., профессор, Институт философии РАН, Москва, Россия
- Ржевский** Георгий, профессор, Открытый университет, Лондон, Великобритания
- Скобелев** Петр Олегович, д.т.н., НПК «Разумные решения», Самара, Россия
- Смирнов** Сергей Викторович*, д.т.н., ИПУСС РАН – СамНЦ РАН, член IAOA, Самара, Россия
- Сулейманов** Джавдет Шевкетович*, д.т.н., профессор, академик АН РТ, Казань, Россия
- Федунов** Борис Евгеньевич*, д.т.н., профессор, ГосНИИ авиационных систем, Москва, Россия
- Шарипбай** Алтынбек*, д.т.н., профессор, Ин-т искусственного интеллекта, Нур-Султан, Казахстан
- Шведин** Борис Яковлевич, к.психол.н., ООО «Дан Роуз», член IAOA, Ростов-на-Дону, Россия

* - члены Российской ассоциации искусственного интеллекта - http://www.raai.org/about/about.shtml?raai_list

Executive Editorial Board - Исполнительная редакция

Chief Editor	P.O. Skobelev	Samara, Russia	Главный редактор	Скобелев П.О.	директор НПК «Разумные решения»
Deputy Chief Editor	S.V. Smirnov	Samara, Russia	Зам. главного редактора	Смирнов С.В.	зам. директора ИПУСС РАН – СамНЦ РАН
Executive Editor	N.M. Borgest	Samara, Russia	Выпускающий редактор	Боргест Н.М.	директор издательства «Новая техника»
Editor	D.M. Kozlov	Samara, Russia	Редактор	Козлов Д.М.	доцент Самарского университета
Technical Editor	D.N. Borgest	Samara, Russia	Технический редактор	Боргест Д.Н.	специалист Самарского университета
Executive Secretary	S.A. Vlasov	Samara, Russia	Ответственный секретарь	Власов С.А.	аспирант Самарского университета

The journal has entered into an electronic licensing relationship with EBSCO Publishing, the world's leading aggregator of full text journals, magazines and eBooks. The full text of JOURNAL can be found in the EBSCOhost™ databases. The journal has been successfully evaluated in the evaluation procedure for the ICI Journals Master List 2014-2018 and journal received the ICV (Index Copernicus Value) of 67.46 points (2014), 67.64 (2015), 77.98 (2016), 87.78. (2017), 85.38 (2018). Журнал размещен в коллекции «Издания по естественным наукам» на платформе EastView.

Журнал включён в перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание учёных степеней кандидата и доктора наук (Письмо Департамента аттестации научных и научно-педагогических работников Минобрнауки РФ от 01.12.2015 № 13-6518) по научным специальностям 05.13.01, 05.13.17, 05.13.18 и 05.07.02, 05.07.05.

Журнал включён в РИНЦ. Пятилетний импакт-фактор РИНЦ **0,968** (2013), **0,895** (2014), **1,305** (2015), **1,055** (2016), **0,957** (2017), **1,137** (2018).

Журнал зарегистрирован в Федеральной службе по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор). Свидетельство ПИ № ФС 77-70157 от 16.06.2017 г. (ранее выданное свидетельство ПИ № ФС 77-46447 от 07.09.2011 г.)

СОДЕРЖАНИЕ

ОТ РЕДАКЦИИ

ОНТОЛОГИЯ КОНСТИТУЦИИ: проблемы проектирования 5-6

ОБЩИЕ ВОПРОСЫ ФОРМАЛИЗАЦИИ ПРОЕКТИРОВАНИЯ: ОНТОЛОГИЧЕСКИЕ АСПЕКТЫ

Г. Ржевский, С.С. Кожевников, М. Свитек 7-21
Умный город как сложная адаптивная система

А.В. Бухановский, С.В. Иванов, С.В. Ковальчук, Ю.И. Нечаев 22-33
Онтологическая система знаний и вычислительных ресурсов
современных интеллектуальных технологий

ПРИКЛАДНЫЕ ОНТОЛОГИИ ПРОЕКТИРОВАНИЯ

Д.И. Муромцев 34-49
Модели и методы индивидуализации электронного образования
в контексте онтологического подхода

Г.Б. Евгеньев 50-62
Многоагентные системы полуавтоматического проектирования в машиностроении
на базе механизма объект-функции

А.Е. Колоденкова, С.С. Верещагина 63-72
Разработка системы иерархических продукционных правил
для диагностирования электротехнического оборудования

В.Е. Гвоздев, О.Я. Бежаева, Р.А. Насырова 73-86
Модели возникновения ошибок на предпроектной стадии разработки
компонент информационно-вычислительных систем

ИНЖИНИРИНГ ОНТОЛОГИЙ

Т.А. Гаврилова, Э.В. Страхович 87-99
Визуально-аналитическое мышление и интеллект-карты в онтологическом инжиниринге

А.А. Романов, Д.Г. Волчек 100-111
Анализ данных о поведении пользователей в системах электронного обучения

Л.В. Аршинский, А.А. Ермаков, М.С. Нитежук 112-120
Комплексная верификация продукционных баз знаний с использованием V^{TF}-логик

МЕТОДЫ И ТЕХНОЛОГИИ ПРИНЯТИЯ РЕШЕНИЙ

В.В. Антонов, О.В. Бармина, Н.О. Никулина 121-140
Поддержка принятия решений при управлении программными проектами
на основе нечёткой онтологии

Журнал ориентирован на учёных и специалистов, работающих по научным направлениям: онтологические аспекты общих вопросов формализации проектирования, прикладные онтологии проектирования, инжиниринг онтологий, методы и технологии принятия решений.
Правила подготовки рукописей статей размещены на сайте журнала «Онтология проектирования»: http://agora.guru.ru/scientific_journal/.

Контакты учредителей

ФИЦ Самарский научный центр РАН: 443020, Самара, ул. Садовая, 61, тел./факс.: +7 (846) 333 27 70, Смирнов С.В., smimov@iccs.ru.
Самарский университет: 443086, Самара, Московское шоссе 34, корп. 10, тел.: +7 (846) 267 46 47, Боргест Н.М., borgest@yandex.ru.
ООО «Новая техника» (издательство): 443010, Самара, ул. Фрунзе, 145, тел.: +7 (846) 332 67 84, факс: +7 (846) 332 67 81.

Отпечатано в ООО «Новая техника», г. Самара, пр.К.Маркса, 24-76. Дата выхода 06.04.2020. Тираж 300 экз. Свободная цена. (6+).

CONTENT

EDITORIAL

ONTOLOGY OF THE BASIC LAW: problems of design 5-6

GENERAL ISSUES OF FORMALIZATION IN THE DESIGNING: ONTOLOGICAL ASPECTS

G. Rzevski, S. Kozhevnikov, M. Svitek 7-21
Smart city as a complex adaptive system

A.V. Bukhanovsky, S.V. Ivanov, S.V. Kovalchuk, Yu.I. Nechaev 22-33
Ontological system of knowledge and computing resources of modern intellectual technologies

APPLIED ONTOLOGY OF DESIGNING

D.I. Mouromtsev 34-49
Models and methods of e-learning individualization in the context of ontological approach

G.B. Evgenev 50-62
Multi-agent systems of semi-automatic design based on object-functions model in engineering

A.E. Kolodenkova, S.S. Vereshchagina 63-72
Development of a system of hierarchical production rules for electrical equipment diagnosing

V.E. Gvozdev, O.Y. Bezhaeva, R.A. Nasyrova 73-86
Models of errors at the pre-design stage of the development
of information and computing systems components

ONTOLOGY ENGINEERING

T.A. Gavrilova, E.V. Strakhovich 87-99
Visual analytical thinking and mind maps for ontology engineering

A. Romanov, D. Volchek 100-111
Data analysis of user behavior in E-learning systems

L.V. Arshinskiy, A.A. Ermakov, M.S. Nitezhuik 112-120
Complex verification of rule-based knowledge bases using V^{TF} -logic

METHODS AND TECHNOLOGIES OF DECISION MAKING

V.V. Antonov, O.V. Barmina, N.O. Nikulina 121-140
Decision-making support in software project management based on fuzzy ontology

The journal is aimed at scientists and specialists working in the following research areas: ontological aspects of general issues of design formalization, applied design ontologies, ontology engineering, decision-making methods and technologies.

The current version of the Rules for the preparation of manuscripts of articles for the journal «Ontology of Designing» is on the journal website: http://agora.guru.ru/scientific_journal/.

Contacts Founders

FIC Samara Scientific Center of the RAS: 61, Sadovaya St., Samara, 443020, Russia. Tel.: +7 (846) 333 27 70, S.V. Smirnov, smirnov@iccs.ru

Samara University: 34, Moskovskoye shosse, bldg. 10, Samara, 443086, Russia. Tel.: +7 (846) 267 46 47, N.M. Borgest, borgest@yandex.ru

New Engineering LLC (publishing house): 145, Frunze St., Samara, 443010, Russia. Tel.: +7 (846) 332 67 84, fax: +7 (846) 332 67 81



ОТ РЕДАКЦИИ

ОНТОЛОГИЯ КОНСТИТУЦИИ: проблемы проектирования ONTOLOGY OF THE BASIC LAW: problems of design

«Не думаете ли вы, что одержавший победу тиран, народ или другое какое-нибудь правление добровольно установят законы, имеющие в виду что-то иное, кроме их собственной пользы, то есть закрепление за собой власти?»

Платон (Диалоги Платона «Законы», книга 4)

«Нужно, чтобы всё-таки Путин был рядом..»

Валентина Терешкова (10 марта 2020)

Дорогой наш читатель, уважаемые авторы и члены редакционной коллегии!

Жизнь современного общества стремительно набирает обороты за счёт массового освоения новых цифровых технологий. На страницах нашего журнала уже не раз обсуждалась концепция нового цифрового общества и, в частности, японская модель Общества 5.0 как проект «оцифровки» всего общества. основополагающий вопрос: какова проектная онтология этого общества и какова модель правового документа, регламентирующего развитие этого будущего общества?



20 февраля в Самарском университете по приглашению главы региона Дмитрия Азарова с лекцией выступил генеральный менеджер департамента внешних связей корпорации Mitsubishi Electric Уэмура Норицугу¹. Он рассказал о японской стратегии построения суперинтеллектуального общества будущего – Общество 5.0, отметив при этом, что Самарский университет мог бы стать «локомотивом цифровизации Самарского региона».



Ректор Самарского университета Владимир Богатырев видит развитие научно-образовательной деятельности в области искусственного интеллекта как «гринфилд (*greenfield*), как научно-образовательный стартап создания центра компетенций в области искусственного интеллекта... Это принципиальная возможность измениться, создать внутри университета команду, которая станет ориентиром для остальных, а её результаты в науке и коммерциализации научных исследований помогут в реализации планов Правительства Самарской области».

На федеральном уровне в активной фазе идут процессы обсуждения онтологии общественного устройства России и его правовой базы – Конституции Российской Федерации. Скорость, с которой идёт процесс трансформации основного закона страны, поражает взгляд стороннего наблюдателя, а предстоящее пакетное её одобрение вызывает изумление всех, кто знаком с принципами формирования разноплановых, междисциплинарных вопросов и законами принятия консолидированных решений на базе экспертных оценок. Однако это

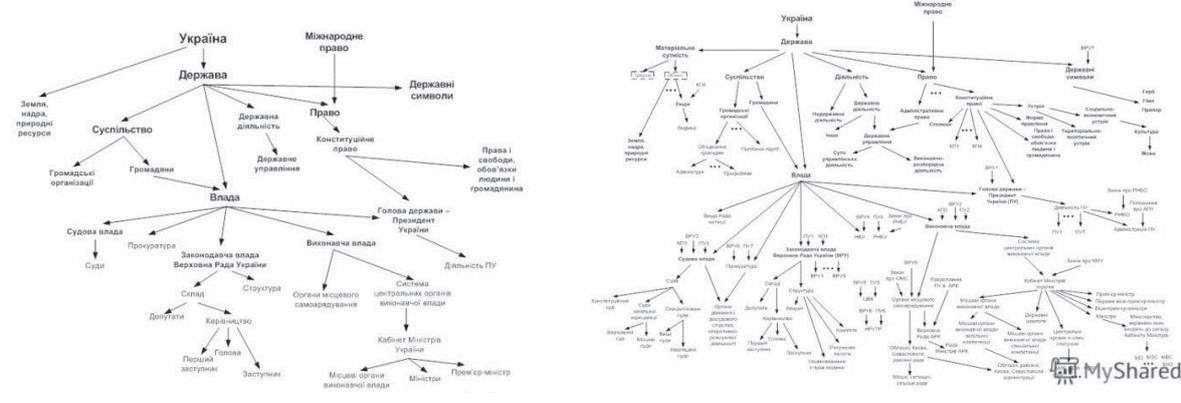
¹ Общество 5.0: японские технологии для цифровой трансформации российской экономики. - <https://www.youtube.com/watch?v=aPg05ZfHPGg>.

жизнь, это реальность, это бытие, наше бытие, в котором мы - часть этого бытия, осознающая его и терзающаяся тем, что собственные модели этого бытия отличаются от действительности. Что говорит о серьёзной проблеме моделирования и соответственно проектирования сложных систем и общественного устройства, в частности. Хотя ещё немецкий социалист Ф. Лассаль различал писаную и действительную конституцию как фактическое соотношение общественных сил в стране, между которыми может быть несовпадение. По его мнению «писаная конституция тогда лишь прочна и имеет значение, когда является точным выражением реальных соотношений общественных сил»². Насколько прочной станет писаная конституция покажет время.

Цифровизация не только ускоряет цивилизационное развитие, даёт гражданам невиданные ранее сервисы, но и является мощным инструментом управления обществом. Элиты всех стран, имея, согласно Платону, вполне определённые целевые установки, активно включились в освоение этого инструмента.

К сожалению, Конституция Российской Федерации рассматривается не только политиками, но и некоторыми учёными лишь как знак онтологии, который максимально раскрывает социально-философское и юридическое содержание³. Попыток построения самой онтологической модели конституции в России пока нет, хотя у наших коллег в Украине такие попытки имели место. Созданная под руководством академика НАНУ Александра Палагина онтология текста конституции Украины представляет собой базу знаний предметной области, которая служит инструментом концептуализации и спецификации, а также их результатом одновременно. Множество терминов, множество отношений, входящих в раздел I текста «Конституция Украины», наглядно представленный в виде семантической сети, способствуют построению адекватной правовой базы молодого государства⁴.

Начальная онтология для текста “Конституція України” **Онтология текста “Конституція України”**



Скорость происходящих событий стала заметна и на примере нашего журнала. Совсем ещё недавно нам казалось, что наш журнал - молодой. Но вот Вы, дорогой наш читатель, держите уже первый номер **десятого** тома. Цвет обложки юбилейного тома символизирует ещё более тесную связь наших знаний с природной средой, с «зелёным полем» (*greenfield*), с нашим бережным отношением к накопленным и формализуемым знаниям. *Dum spiro, spero!*

Ontologists and designers of all countries and subject areas, join us!

² Витрук Н.В. Верность Конституции. 2016. - 490 с. - <https://www.litres.ru/nikolay-vitruk/vernost-konstitucii/chitat-onlayn/>.
³ Юсубов Э.С. Знаки онтологии в конституции Российской Федерации // Вестн. Том. гос. ун-та. Право. 2013. №2 (8). - <https://cyberleninka.ru/article/n/znaki-ontologii-v-konstitucii-rossiyskoy-federatsii-1>.
⁴ Палагин А.В., Кривой С.Л., Петренко Н.Г., Величко В.Ю. Онтологический подход и аспекты обработки естественно-языковых объектов. - <http://www.myshared.ru/slide/224689/>.

ОБЩИЕ ВОПРОСЫ ФОРМАЛИЗАЦИИ ПРОЕКТИРОВАНИЯ: ОНТОЛОГИЧЕСКИЕ АСПЕКТЫ

УДК 004.82:003.6

DOI: 10.18287/2223-9537-2020-10-1-7-21

Умный город как сложная адаптивная система

Г. Ржевский¹, С.С. Кожевников², М. Свитек³

¹ Открытый университет, Милтон-Кинс и Digital Ecosystems Ltd, Лондон, Великобритания

² Чешский институт информатики, робототехники и кибернетики,
Чешский технический университет, Прага, Чехия

³ Факультет транспортных наук, Чешский технический университет, Прага, Чехия

Аннотация

В городской среде стремительное развитие политической, социальной и экономической сфер ставят перед лицом общественности новые задачи, актуальным решением которых является переход от традиционных методов управления городским пространством к Умному городу. При этом, необходимо переосмысление ключевых элементов данной концепции. В работе предложено обновлённое определение Умного города. Дан анализ проблем развития городских агломераций, связанных с повсеместным использованием сложной системы Интернет-экономики, и описано их практическое решение – разработка городских пространств как сложных адаптивных систем, способных предоставлять в реальном времени услуги горожанам и приезжим в зависимости от уровня сложности задачи. Авторы выступают за применение инновационного подхода, который обеспечивает минимальное количество вмешательств в работу городских служб. Концепция синхронной разработки и внедрения мультиагентных систем планирования сервисов в реальном времени и онтологии Умного города, описанные в статье, позволят обеспечить необходимые изменения за счёт параллельной работы географически распределённых проектных групп.

Ключевые слова: Умный город, сложная адаптивная система, городская экосистема, самоорганизация, ко-эволюция.

Цитирование: Ржевский, Г. Умный город как сложная адаптивная система / Г. Ржевский, С.С. Кожевников, М. Свитек // Онтология проектирования. – 2020. – Т. 10, №1(35). – С.7-21. – DOI: 10.18287/2223-9537-2020-10-1-7-21.

Введение в концепцию Умного города

Известно большое количество работ, посвящённых концепции «Умный город». Использование современных технологий для проектирования Умного города в целях достижения синергетического эффекта между различными подсистемами (транспорт, логистика, безопасность, энергетика, административный сектор и др.) с указаниями энергоёмкости и качества жизни жителей описано в [1].

Возможность применения концепции Умного города для узконаправленных задач, таких как оптимизация энергопотребления, улучшение качества воздуха, шумоизоляция, регулирование транспортных систем, и для глобальных проектов, как, например, поддержание самобытности заданного места и его урбанистической структуры, т.е. исторического, культурного, экологического или эстетического аспектов, освещается в [2, 3]; обсуждается также возможность использования подхода в более широком смысле, а именно концепции «Умный поселок» или «Умный регион».

Умный город как сложная система рассматривается в [4, 5], где показано, что для достижения эффективного функционирования подобной системы необходима кооперация между представителями различных профессий: архитекторами, адвокатами, инвесторами, экспертами в области транспорта и энергии, сотрудниками сферы информационных технологий, политиками, социологами и др. [6].

В работе [7] обсуждается роль социальных сетей как нового средства коммуникации для эффективной взаимосвязи между различными группами населения. С использованием информации, полученной от целевой аудитории, становятся возможными корректировка поведенческих привычек и улучшение двусторонней коммуникации между представителями городского управления и жителями города [8]. Это направление особенно важно, так как каждый отдельный горожанин может влиять на жизнь города и должен обладать средствами для выражения собственного мнения.

Для управления городом уже используются различные типы датчиков, сенсоров, различные камеры, а также фотографии со спутников (для прогнозирования погоды, составления карт распределения температуры или выбросов CO - NO_x) [9]. Даже мобильный телефон сейчас, по сути, является «умным» сенсором, предоставляющим важную информацию.

Новые технологии способствуют более эффективному планированию и вовлечению жителей в развитие города. Представление различных решений проблемы может быть осуществлено с помощью современных средств визуализации, таких как виртуальная или дополненная реальность.

С технической точки зрения, Интернет вещей, Интернет людей, Интернет энергии, Промышленный Интернет вещей или Интернет услуг используются для общего обмена информацией [10, 11]. С системной точки зрения применяются киберфизические системы, или, как вариант в случае Умных городов, социокиберфизическая система [12, 13]. Благодаря возрастающему потоку данных управление городскими структурами изменяется, уходя от стандартных предопределённых динамичных планов к адаптивным алгоритмам контроля, которые обеспечивают координацию крупных территориальных единиц [14, 15].

На практике все элементы Умного города могут быть представлены Агентами Потребности (*Demand Agents*) и Агентами Ресурсов (*Resource Agents*), которые могут взаимодействовать между собой и представляют собой все существующие запросы и ресурсы [5]. Подобная холистическая модель играет роль динамичной цифровой площадки с ограниченным и изменяющимся во времени количеством ресурсов. Благодаря этому концепция Умного города со способностью быстро восстанавливаться (*Smart Resilient City 5.0*) [16] является новым взглядом на город как на цифровую платформу и экосистему умных устройств, где агенты людей, вещей, документов, роботов и других подразделений могут напрямую взаимодействовать между собой, базируясь на принципе «заказ-ресурс» и предоставляя наилучшее возможное решение. Такая платформа создаёт умное пространство, и становится основой для самоорганизации индивидуумов, групп или целых систем в условиях устойчивости или, если необходимо, адаптивности [17].

Человеческий фактор является ключевым при принятии решений. По этой причине в рамках концепции Умного города необходимо отметить человеко-машинный интерфейс как связующий элемент между техническими системами и людьми, который должен быть интуитивно понятным и лёгким для использования для всех категорий граждан (дети, пенсионеры, люди с ограниченными возможностями, и т.д.) [18]. Взаимодействие с пользователями может быть оценено с помощью различных видов симуляций для выбранной группы жителей [19].

Для оценки развития Умного города используется множество методологий, например, индекс Умного города [20]. Он включает в себя оценку уровня «оцифрованности» различных

процессов и параметров функциональности некоторых городских подсистем, таких как мобильность, безопасность, управление энергией. Основываясь на разных методологиях, ежегодная экспертиза лучших Умных городов проходит на общемировом уровне.

1 Сложные адаптивные системы

Система считается сложной в тех случаях, если она открыта (обменивается информацией с окружающим миром), состоит из множества частично автономных и связанных между собой компонентов, называемых агентами, и если в этой системе нет централизованного управления. Поведение таких систем неопределённое, но не случайное. Общее поведение складывается из связей между составными агентами. Автономность агентов ограничивается сдерживающими факторами, наложенными на них системой, к которой они принадлежат.

Термин сложная комплексная система (*complex system*) происходит от слова *complexus* (лат. переплетение или взаимосвязь) и этимологически не связан со словами сложный (*complicated*), громоздкий (как бюрократия), неуправляемый, хаотичный (как неорганизованное управление) или сложный для понимания (как документ). Так как ключевым преимуществом сложных комплексных систем является адаптивность принято называть такие системы сложными адаптивными системами [1].

Используя неопределённость в качестве критерия, можно разделить все системы на три класса (таблица 1):

- Детерминированные, где поведение системы предсказуемо (неопределённость = 0). Примеры включают в себя как объекты, созданные человеком, так и закрытые модели физических систем.
- Сложные, поведение которых определяется взаимодействием агентов, детально не предсказуемо и всё же следует некоторым закономерностям ($1 > \text{неопределённость} > 0$). Среди примеров можем выделить: социальная, политическая и экономическая системы, бизнес и рынок.
- Случайные, поведение которых полностью непредсказуемо (неопределённость = 1). Хорошим примером в данном случае может быть движение молекул.

Таблица 1 - Классификация систем

Случайные	Сложные	Детерминированные
Неопределённость = 1	$1 > \text{неопределённость} > 0$	Неопределённость = 0
Полная автономия агентов	Частичная автономия агентов	Отсутствие автономии у агентов
Неорганизованная	Самоорганизация	Организованная
Непредсказуемое поведение	Эмерджентное поведение	Предсказуемое поведение

Ключевыми параметрами, которые отличают комплексные системы от детерминированных, являются: связность, автономность, эмерджентность, нелинейность, неравновесность, самоорганизация, и ко-эволюция.

Связность агента указывает на число тех агентов, с которыми может взаимодействовать первоначальный агент. Сложность системы увеличивается вместе с увеличением связности агентов. Сила связности – важный параметр для оценки сложности системы. Комплексность тем выше, чем слабее связи между агентами и чем проще эти связи разорвать и образовать новые связи с другими агентами.

Автономность агентов указывает на степень свободы поведения агента. Сложность системы увеличивается при увеличении автономности агентов. Тем не менее, в сложных системах всегда имеется ограниченное число степеней свободы. В крайних случаях, когда агентам предоставляется полная свобода, поведение системы становится случайным.

Эмерджентность указывает на наличие у системы особых свойств, не присущих отдельным её элементам. Эти свойства появляются как следствие взаимодействия агентов.

Свойство *нелинейности* присуще системам, в которых связи между агентами не линейны и обладают амплификацией, самоускорением и автокаталитическими свойствами. В результате незначительные возмущения могут приводить к экстремальным последствиям и таким явлениям, как «Эффект бабочки». Не менее важно, что разрастание небольших возмущений во времени может вызвать эффект, известный как «Скольжение к провалу».

Неравновесность – свойство, указывающее на то, что операции подвержены влиянию непредсказуемых разрушающих событий. Если комплексность системы велика, разрушающие события генерируются с высокой частотой, и у системы нет времени, чтобы вернуться к состоянию равновесия до появления следующего события.

Самоорганизация – свойство, которое проявляется в тех случаях, когда случается разрушающее событие (поступление неожиданного запроса, отсрочка, фальсификация, атака и т.д.), и система в срочном порядке определяет поражённые участки и перестраивается для ликвидации или, по крайней мере, для снижения эффекта от воздействия разрушающего события до наступления новой атаки. Самоорганизация – важная особенность сложной системы, которая делает её адаптивной к изменениям окружающей среды и способной быстро восстанавливаться при нападениях извне. Самоорганизация может также вести к саморазвитию, в результате которого агенты самоорганизуются для улучшения работы системы.

Ко-эволюция – свойство сложных систем, которое позволяет им развиваться вместе с окружающей средой. При этом системы поддаются внешнему воздействию со стороны окружающей среды и, в свою очередь, также влияют на неё.

2 Определение понятия Умный город 5.0

Умный город – это адаптивный, способный быстро восстанавливаться и устойчивый в развитии город, который предлагает его гостям и жителям высокий уровень жизни, работы и развлечений при минимальном влиянии на окружающую среду.

Адаптивность, устойчивость в развитии и способность быстро восстанавливаться являются эмерджентными свойствами сложных адаптивных систем [1, 17]. Из этого следует, что для превращения города в умный, ему необходимо пройти через «цифровизацию», т.е. базироваться на таких цифровых технологиях как: мультиагентные системы, искусственный интеллект, Интернет вещей, социальные сети и др., а также набор необходимых сенсоров [21].

3 Проблемы современного города

Суть современных проблем в городском «пространстве» заключается в комплексной природе его политической, социальной и экономической сред, информация и процессы в которых растут экспоненциально, в то время как их административные возможности и технологическая инфраструктура остаются ограниченными. Из чего следует сложность эффективного управления городом в новых динамичных условиях [22, 23]. Результатом подобного развития событий являются недовольство жителей города, увеличение проблем со здоровьем горожан из-за загрязнения окружающей среды и бесполезной трате ресурсов.

Ключевые проблемы, которые требуют решения:

- расточительные, неорганизованные, вредные для окружающей среды услуги, которые оказываются жителям города;

- отсутствие адекватных каналов коммуникации между жителями города, гостями и городской администрацией, что не позволяет горожанам и туристам заявить об их ожиданиях;
- отсутствие мониторинга качества услуг и знаний об эффективном управлении городом в условиях комплексности;
- несовершенные методы стратегического планирования.

В типичном городе предоставляются различные типы услуг, например:

- | | |
|----------------------------|--------------------------------------|
| ▪ образование, | ▪ городское планирование, |
| ▪ медицинские услуги, | ▪ обеспечение безопасности общества, |
| ▪ социальные услуги, | ▪ гигиена окружающей среды, |
| ▪ дороги и транспорт, | ▪ туризм, |
| ▪ поставка продовольствия, | ▪ развлечения, |
| ▪ водоснабжение, | ▪ строительство, |
| ▪ канализация, | ▪ жилищные услуги, |
| ▪ уборка и сбор мусора, | ▪ сбор налогов, |
| ▪ развитие экономики, | ▪ и другие. |

Каждая из этих услуг может быть улучшена при управлении ресурсами в режиме реального времени. Более качественные улучшения могут быть достигнуты за счёт координирования предоставляемых услуг с помощью цифровых технологий в тех случаях, если эти услуги находятся на открытом рынке и конкурируют между собой. На данный момент каждая жизненно важная услуга оказывается отдельно, и координация услуг практически отсутствует.

Подобный метод управления ведёт к значительной трате ресурсов, существенной угрозе состоянию окружающей среды и увеличению числа заболеваний, связанных с её загрязнением. В результате увеличивается число непредсказуемых событий, таких как изменение политики (за счёт политической нестабильности), изменений спросов на услуги (из-за потока иммигрантов, разнообразия потребностей, избыточного туризма), неудачного использования ресурсов и появления задержек (причиной которых являются несчастные случаи, перегруженный трафик, рост спроса на рынке товаров, праздники, туристические сезоны, эпидемии, террористические атаки и т.д.)

В то же время растёт осведомлённость жителей города как о проблемах окружающей среды, так и об угрозах для здоровья, результатом чего выступает фрустрация, которая, в свою очередь, становится причиной демонстраций и требований к улучшению качества жизни.

Важными факторами в секторе городского управления являются жёсткая, разделённая на множество департаментов структура, которая часто мешает эффективной координации услуг, использование традиционных информационных технологий, которые не в состоянии справляться с динамичностью городской жизни.

Проведённое исследование показало, что во многих городах не налажена эффективная система связи общества и власти. В результате представители городской администрации недостаточно хорошо осведомлены о требованиях и ожиданиях тех людей, непосредственно для которых организационные структуры и были созданы.

На данный момент ни в одном городе нет эффективной системы организации жизни, работы и развлечений в условиях непредсказуемых событий в политической, социальной и экономической сферах. Поэтому необходимо распространять знания об управлении сложными комплексными системами максимально широко.

4 Проектирование Умного города 5.0

Основная идея науки о сложных системах (*Complexity Science*) состоит в том, что только сложные адаптивные системы могут эффективно работать в условиях комплексности. Жёстко заданные структуры детерминированных систем имеют тенденцию ломаться, когда помещаются в комплексную среду – это явление в настоящее время наблюдается во многих приходящих в упадок городских хозяйствах по всему миру. В то же время сложные структуры легко приспосабливаются к изменениям, способны быстро восстанавливаться при внешних воздействиях и сосуществуют, и изменяются одновременно с внешними условиями, что делает их более устойчивыми.

Из этого следует необходимость поиска способа превратить современный город в сложную адаптивную городскую систему.

Адаптивные службы способны обнаруживать негативное событие до того, как оно причинит какой-либо ущерб, и быстро перенаправлять ресурсы, затронутые этим событием для нейтрализации или уменьшения последствий. Поскольку непредсказуемые негативные воздействия появляются довольно часто, их обнаружение и последующее перепланирование ресурсов должно выполняться быстро - в режиме реального времени, что может быть достигнуто только с помощью умной технологии принятия решений, такой как мультиагентные системы. Городские службы, как правило, взаимосвязаны, поэтому процесс адаптации в одной службе запустит волну изменений в других. Подразаумевается, что службы в Умном городе будут постоянно самоорганизовываться, чтобы адаптироваться к различным событиям.

От городской системы требуется устойчивое развитие [15]. Устойчивые городские службы способны ко-эволюционировать в своей среде, что представляет собой процесс взаимодействия, в котором службы адаптируются к постоянным изменениям и среда адаптируется к постоянным изменениям в службах. Проиллюстрировать ко-эволюцию служб Умного города и среды можно на следующем примере работы городского транспорта с использованием различных типов энергоносителей.

Увеличение загрязнения от дизельного транспорта, определяемое датчиками загрязнения и специальными службами, например, подходит к критическим значениям. Умная система в таком случае предлагает повышение налогов на использование дизельных транспортных средств в городе, что, в свою очередь, ускоряет процесс перехода с дизельных автомобилей на электромобили. Система адаптивного планирования быстро улавливает тенденцию и инициирует расширение радиуса установки точек для зарядки электромобилей.

Умные города благодаря использованию мультиагентных планировщиков в реальном времени могут обладать такими свойствами, как адаптивность, способность быстро восстанавливаться и устойчивость. При этом мультиагентные планировщики способны кооперировать (или конкурировать) друг с другом, обеспечивая взаимодействие служб с целью максимизации согласованных значений показателей эффективности Умного города.

Архитектура городской службы, работающей в реальном времени, изображена на рисунке 1.

Типовая городская служба состоит из:

- Реального Мира человеческих и физических ресурсов и потребностей;
- Виртуального Мира программных агентов, занимающихся планированием ресурсов Реального Мира в реальном времени;
- базы знаний, содержащих информацию о том, как запустить сервис;
- интерфейсов между двумя мирами, в первую очередь предназначенных для передачи информации о разрушительных событиях из Реального в Виртуальный Мир и инструкции по переносу человеческих и физических ресурсов из Виртуального в Реальный Мир.

5 Трансформация города в Умный город 5.0

В качестве примера рассмотрен типичный город, где есть орган местного самоуправления - городская администрация, ответственная за обслуживание граждан. Данная администрация решила существенно улучшить качество услуг каждой из подведомственных служб и в то же время сократить расходы на их работу.

5.1 Спецификация требований

Существует общепринятое мнение – подходить к решению проблемы «сверху вниз», пытаясь создать спецификацию требований для всего города ещё до начала работ. Такой подход неприемлем при нынешней динамично изменяющейся политической, социальной, экономической и технологической городской среде. Требования будут устаревать ещё до завершения их формирования. Метод эволюционной трансформации, предлагаемый авторами, может обеспечить плавное преобразование.

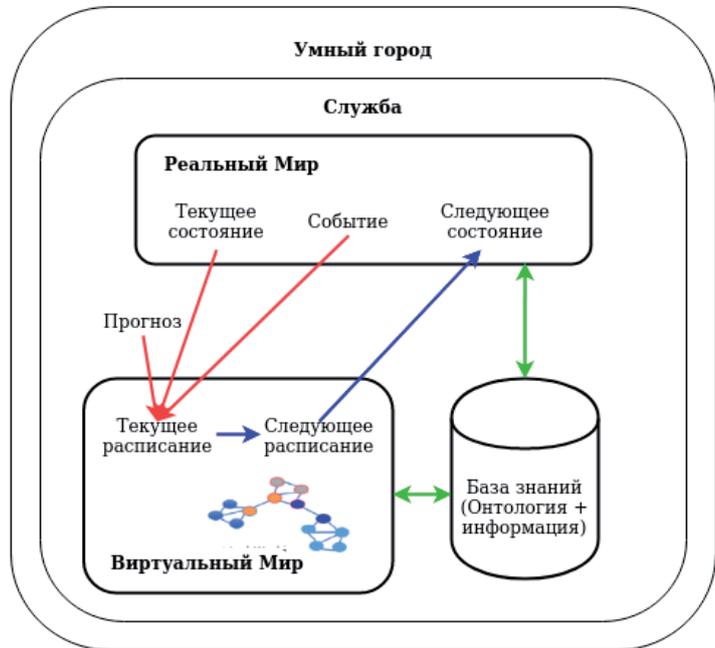


Рисунок 1 - Архитектура сервисов Умного города

5.2 Эволюционный метод цифровой трансформации

В условиях часто изменяющихся требований целесообразно использовать эволюционный метод преобразований в такой последовательности действий:

- 1) рассмотреть все услуги города с точки зрения определения наибольшей ценности для города;
- 2) сформировать спецификацию требований для улучшения выбранной услуги;
- 3) разработать и внедрить адаптивный планировщик в реальном времени только для выбранной услуги;
- 4) оценить и обновить решение, демонстрирующее достигнутый прирост целевых показателей;
- 5) выбрать следующую услугу и повторить шаги 2, 3, 4, удостоверившись, что они кооперируют (или конкурируют) с уже имеющимися службами, которые были изменены ранее.

По мере увеличения количества претерпевших изменения услуг необходимо постоянно контролировать и вносить необходимые изменения непосредственно в Умный город.

Постепенное и непрерывное улучшение качества работы служб города никогда не должно прекращаться. Необходимо обеспечить непрерывный доступ квалифицированных специалистов для постоянного поддержания и улучшения городской экосистемы, что позволит дополнять её новыми услугами, способными к самообслуживанию и самосовершенствованию.

5.3 Режим синхронной трансформации

В дополнение к эволюционному подходу к трансформации работа над спецификациями требований, проектированием и вводом в эксплуатацию более качественно работающих служб должна проводиться синхронно, рассчитывая затрачиваемое время таким образом, чтобы всякий раз, когда:

- разработка проекта улучшения услуги завершена, спецификация требований для улучшения работы следующей услуги была готова к проектированию;
- ввод в эксплуатацию обновлённой услуги был завершён, проект для улучшения работы следующей готов к вводу в эксплуатацию.

Синхронная разработка особенно удобна, если над проектом совместно работают несколько географически распределённых команд.

6 Пример применения концепции услуги в Умном городе

В качестве примера приведена работа скорой помощи - критически важной службы города, функционирование которой при этом необходимо улучшить.

6.1 Разработка онтологии

Один из подходов к формализации знаний в концепции Умный город основан на создании онтологий. Они могут быть представлены семантическими сетями с описанием основных сущностей предметной области в виде классов, атрибутов и отношений. Это помогает отделить базы данных (БД) и код разработчиков от знаний, которые крайне важны для адаптации системы. На данный момент концепция Умного города чаще всего опирается на создание единой БД в городе для обеспечения доступа к услугам [24].

Эта задача требует формализации знаний обо всех аспектах Умного города и обеспечения простого доступа к этим знаниям для различных служб и поддержки взаимодействия на цифровых платформах и в экосистемах [25]. Модель города, основанная на онтологиях, может определять основные объекты, такие как здания, дороги, транспорт, автобусные остановки, светофоры, источники энергии, места отдыха и другие. Эти объекты могут иметь и детальное описание - здание может быть рестораном, бизнес-центром или жилым зданием со многими другими свойствами, сохранёнными в качестве атрибутов (количество этажей, дата строительства и т.д.). Онтология также может описывать людей, их деятельность или запросы, требования или бизнес-процессы [26].

Формализованные таким образом знания могут использоваться различными городскими службами не только в качестве источника необработанной информации (как сейчас), но и для адаптации к текущей ситуации и корректировки работы, обеспечения устойчивости развития города.

Применение концепции начинается с описания онтологии службы скорой помощи, которая включает в себя:

- 1) классы объектов (машина скорой помощи, экипаж скорой помощи, член экипажа, медицинское оборудование, маршрут, дорога, больница, пациент, родственник);
- 2) отношения (экипаж относится к машине скорой помощи, оборудование принадлежит экипажу, машина скорой помощи следует по маршруту);
- 3) свойства классов объектов (для члена экипажа: идентификатор, квалификация, доступность).

На этом этапе целесообразно написать сценарии для агентов, которые будут храниться в онтологии и будут доступны агентам, когда появится необходимость выполнить задачу.

Все политики, правила и положения, регулирующие работу служб, также должны храниться в онтологии. В дополнение к онтологии база знаний также содержит все данные об объектах, которые обычно хранятся в клиентских БД [27].

Вариант онтологии для рассматриваемого примера представлен на рисунках 2 и 3.

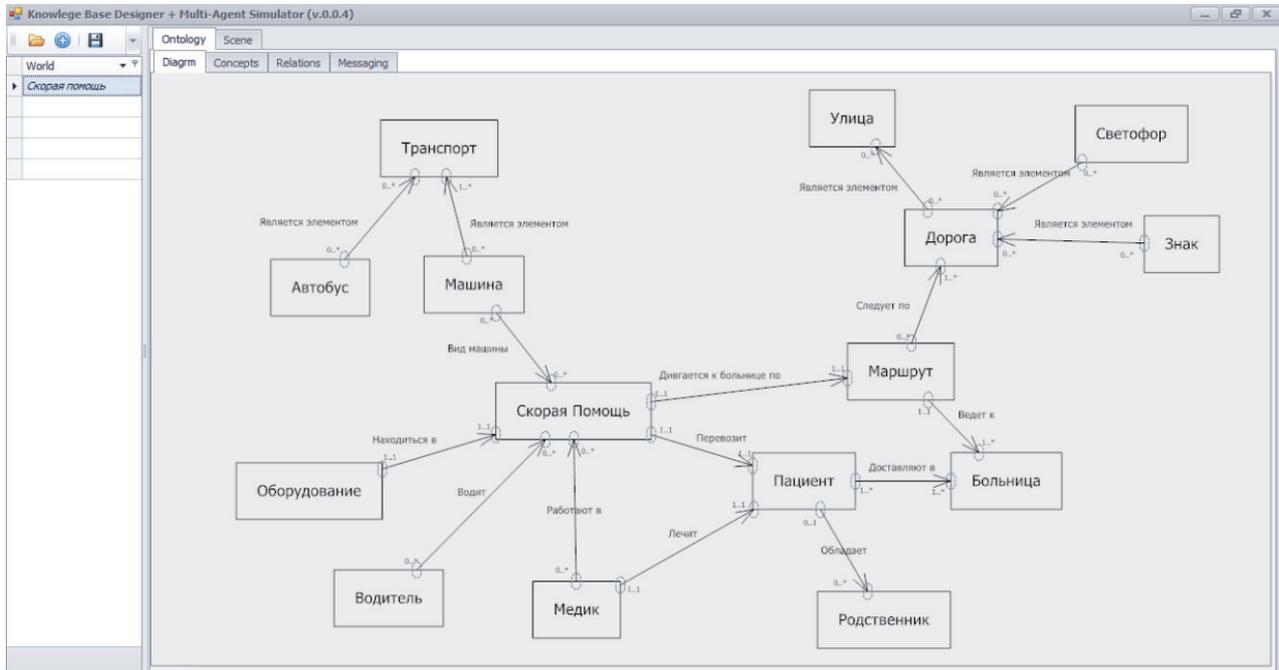


Рисунок 2 – Вариант разработанной онтологии для службы скорой помощи

Name	Type	Source	Target
Является элементом	Association	'-> Source', [0..*], [Знак], [World]	'-> Target', [0..*], [Дорога], [World]
Следует по	Association	'-> Source', [1..*], [Маршрут], [World]	'-> Target', [0..*], [Дорога], [World]
Ведет к	Association	'-> Source', [1..*], [Маршрут], [World]	'-> Target', [1..1], [Больница], [World]
Обладает	Association	'-> Source', [0..*], [Пациент], [World]	'-> Target', [0..1], [Родственник], [World]
Находиться в	Association	'-> Source', [1..1], [Оборудование], [World]	'-> Target', [1..1], [Скорая Помощь], [World]
Лечит	Association	'-> Source', [1..1], [Медик], [World]	'-> Target', [1..1], [Пациент], [World]
Двигается к больнице по	Association	'-> Source', [1..1], [Скорая Помощь], [World]	'-> Target', [1..1], [Маршрут], [World]
Доставляют в	Association	'-> Source', [1..*], [Пациент], [World]	'-> Target', [1..*], [Больница], [World]

Name	Обладает
Type	Association
Source	'-> Source', [0..*], [Пациент], [World]
Concept	Пациент
Alias	
Multiplicity	0..*
Source is [World]	<input checked="" type="checkbox"/>
Target	'-> Target', [0..1], [Родственник], [World]
Concept	Родственник
Alias	
Multiplicity	0..1
Source is [World]	<input checked="" type="checkbox"/>

Рисунок 3 – Концепты и связи разработанной онтологии

6.2 Создание Виртуального Мира

Создание Виртуального Мира означает проектирование цифровой инфраструктуры, которая поддерживает обмен информацией между агентами, число которых может достигать сотен тысяч.

Виртуальный Мир – это место, где агенты «договариваются» между собой о том, как распределять ресурсы по запросам. Этот процесс быстрый и не повторяющийся. Условия, при которых ведутся переговоры между агентами, часто изменяются в процессе.

В Виртуальном Мире службы скорой помощи взаимодействие членов экипажа между собой или с другими экипажами скорой помощи, а также с больницами, взаимодействие дорог с маршрутами, пациентов с больницами и т.д. осуществляется путём обмена сообщениями между соответствующими агентами [28].

Адаптивность достигается путём сопоставления агентами требований к ресурсам и возможностями. Если, например, дорога в выбранную больницу заблокирована из-за интенсивного движения, агент дороги, к ней относящийся, немедленно сообщит о проблеме агентам других дорог, что инициирует волну сообщений между ними для определения нового маршрута в больницу. Если новый маршрут окажется слишком длинным, агенты могут попытаться договориться о том, чтобы конечным пунктом маршрута была другая больница.

Основным преимуществом подобного перепланирования является то, что части экосистемы службы скорой помощи, не затронутые закрытием дороги, продолжают функционировать так, как если бы не было сбоев.

Описанный пример – лишь небольшая часть процесса проектирования, но достаточен для демонстрации преимуществ мультиагентной технологии по сравнению с последовательной оптимизацией. Больше деталей можно найти в [1, 29, 30].

6.3 Проектирование интерфейсов между Реальным и Виртуальным Мирами

Информация о деструктивных событиях, происходящих в Реальном Мире, должна передаваться в Виртуальный Мир, а все решения по планированию, принимаемые агентами в Виртуальном Мире, должны передаваться в Реальный Мир.

Первоначально обмен информацией может осуществляться посредством обмена сообщениями между людьми, выступающими в роли операторов (например, водителями), с использованием их смартфонов или специально разработанных портативных устройств связи, и агентами [23, 31]. Однако наиболее предпочтительным способом является обмен информацией между физическими ресурсами Реального Мира (например, транспортными средствами, роботами, конвейерами) и агентами с использованием технологии Интернета вещей.

6.4 Расширение концепции

Планирование маршрутов машин скорой помощи легко расширяется, например, путём добавления врачей и учреждений, которые требуются пациентам, доставляемым на скорой помощи. Для этого нужны дополнительные агенты. Следующим этапом добавляются агенты для планирования использования других ресурсов больницы, таких как оборудование, операционные и т.д. Проектирование должно продвигаться шаг за шагом и каждый шаг должен проверяться на практике до того, как будет введён в работу следующий.

Заключение

Переход от города в привычном понимании к Умному городу крайне необходим из-за огромного разрыва между возросшей сложностью политической, социальной и экономиче-

ской среды современных населённых пунктов и уже устаревшей административной и технологической инфраструктурой, неспособной эффективно функционировать в новых условиях.

В статье показана необходимость переосмысления ключевых элементов и общей концепции Умного города. Представлено определение Умного города как сложной адаптивной интеллектуальной системы.

Описан метод эволюционной трансформации, который предполагает минимальное вмешательство в работу городских служб, обеспечивает плавные изменения и может выполняться несколькими командами специалистов параллельно. В настоящее время метод используется авторами в проекте, направленном на разработку Умного города 5.0 - цифровой экосистемы служб, позволяющей достичь синергетического эффекта между различными подсистемами (транспорт, экология, энергетика, городское проектирование и др.).

СПИСОК ИСТОЧНИКОВ

- [1] **Rzevski, G.** Managing Complexity / G. Rzevski, P. Skobelev. WIT Press, Southampton, Boston, 2014. ISBN 978-1-84564-936-4.
- [2] **Svítek, M.** Telematic approach into program of smart cities, EATIS 2014, Valparaiso - Chile, 2014, ISBN 978-1-4503-2435-9.
- [3] **Svítek, M.** Smart City 5.0 as an Urban Ecosystem of Smart Services / M. Svítek, P. Skobelev, S. Kozhevnikov // in Borangiu T., Trentesaux D., Leitão P., Giret Boggino A., Botti V. (eds) Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future. SOHOMA 2019. Studies in Computational Intelligence, vol 853. Springer, Cham, 2020, p.426 – 438, ISBN 978-3-030-27476-4.
- [4] **Allam, Z.** Redefining the Smart City: Culture, Metabolism and Governance / Z. Allam, P. Newman // Smart Cities. 1, 4 (2018)
- [5] **Kozhevnikov, S.** Development of Resource-Demand Networks for Smart Cities 5.0 / S. Kozhevnikov, P. Skobelev, O. Pribyl, M. Svítek // in Mařík V., Kadera P., Rzevski G., Zoitl A., Anderst-Kotsis G., A Min Tjoa, Khalil I. (eds) Industrial Applications of Holonic and Multi-Agent Systems. HoloMAS 2019. Lecture Notes in Artificial Intelligence, vol 11710. Springer AG 2019, p.203-217, ISBN 978-3-030-27877-9.
- [6] **Batty, M.** Smart cities of the future / M. Batty, K. Axhausen, F. Giannotti, A. Pozdnoukhov, A. Bazzani, M. Wachowicz, G. Ouzounis, Y. Portugali // The European Physical Journal Special Topics. 214, 481-518 (2012).
- [7] **Postranecky, M.** SynopCity Virtual HUB – A testbed for Smart Cities / M. Postranecky, M. Svítek, E.Z. Carrillo // IEEE Intelligent Transportation Systems Magazin, Vol. 10, Issue 2, 2018, p.50-57, DOI: 10.1109/MITS.2018.2806642.
- [8] **Uribe-Pérez, N.** A novel communication system approach for a Smart City based on the human nervous system / N. Uribe-Pérez, C. Pous // Future Generation Computer Systems. 76 (2017).
- [9] **Trucco, P.** Ontology-based approach to disruption scenario generation for critical infrastructure systems / P. Trucco, V. Petrenj, S. Bouchon, C. Dimauro // International Journal of Critical Infrastructures. 12, 248 (2016).
- [10] **Евгениев, Г.Б.** Индустрия 5.0 как интеграция Интернета знаний и Интернета вещей / Г.Б. Евгениев // Онтология проектирования. – 2019. – Т.9, №1(31). – С.7-23. – DOI: 10.18287/2223-9537-2019-9-1-7-23.
- [11] **Цветков, В.Я.** Метрики сложной детерминированной системы / В.Я. Цветков, А.В. Буравцев // Онтология проектирования. – 2017. – Т. 7, №3(25). – С. 334-346. – DOI: 10.18287/2223-9537-2017-7-3-334-346
- [12] **Микони, С.В.** Обобщённая онтологическая модель управления в концепции социкиберфизической системы / С.В. Микони // Онтология проектирования. – 2019. – Т.9, №2(32). – С.191-202. – DOI: 10.18287/2223-9537-2019-9-2-191-202.
- [13] **Максимов, Н.В.** Семантическое ядро цифровой платформы / Н.В. Максимов, О.Л. Голицына, М.Г. Ганченкова, Д.В. Санатов, А.В. Разумов // Онтология проектирования. -2018. - Т. 8, №3(29). - С.412-426. - DOI: 10.18287/2223-9537-2018-8-3-412-426.
- [14] **Santipantakis, G.** OBDAIR: Ontology-Based Distributed framework for Accessing, Integrating and Reasoning with data in disparate data sources / G. Santipantakis, K. Kotis, G.A. Vouros // Expert Systems with Applications. 90, 464-483 (2017).
- [15] **Costin, A.** Need for Interoperability to Enable Seamless Information Exchanges in Smart and Sustainable Urban Systems / A. Costin, C. Eastman // Journal of Computing in Civil Engineering. 33 (2019).
- [16] **Moraci, F.** Making Less Vulnerable Cities: Resilience as a New Paradigm of Smart Planning / F. Moraci, M. Errigo, C. Fazio, G. Burgio, S. Foresta // Sustainability. 10, 755 (2018).

- [17] **Rzevski, G.** Intelligent Multi-Agent Platform for Designing Digital Ecosystems / G. Rzevski // In Vladimir Marik, Petr Kadera, George Rzevski, Alois Zolti, Gabriele Anderst-Kotsis, A Min Yjoa, Ismail Khalil (eds), Proceedings of the 9th International Conference, HoloMAS 2019, Linz, Austria, August 26 – 29, 2019, p.29-41. LNAI 11710.
- [18] **Ouyang, M.** A three-stage resilience analysis framework for urban infrastructure systems / M. Ouyang // Structural Safety. 36-37, 23-31 (2012).
- [19] **Agudelo-Vera, C.** Harvesting urban resource towards more resilient cities / C. Agudelo-Vera, W.R.W.A. Leduc, A.R. Mels, H. Rijnaarts // Resources, Conservation and Recycling. 64, 3-12 (2012).
- [20] Index of Smart Cities, IMD World Competitiveness Centre, Smart City Observatory, Singapore University of Technology and Design. In: <https://www.imd.org/smart-city-observatory/smart-city-index>.
- [21] **Ganzha, M.** Semantic interoperability in the internet of things: An over-view from the INTER-IoT perspective / M. Ganzha, M. Paprzycki, W. Pawłowski, P. Szmaja, K. Wasielewska // J. Network Comput. 81, 111–124 (2017).
- [22] **Badii, C.** Km4City Smart City API: An integrated support for mobility services / C. Badii, P. Bellini, D. Cenni, G. Martelli, P. Nesi, M. Paolucc // IEEE Int. Conf. on Smart Computing, 1–8 (2016).
- [23] **Городецкий, В.И.** Цифровая платформа киберфизических систем / В.И. Городецкий, П.О. Скобелев. // XIII Всероссийское совещание по проблемам управления ВСПУ-2019, Москва, 17 – 20 июня 2019 г.: Труды. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2019.
- [24] **Seedah, D.P.** Ontology for querying heterogeneous data sources in freight transportation / D.P. Seedah, C. Choubassi, F. Leite // J. Comput. Civ. Eng. 30 (4): 04015069 (2016).
- [25] **Howsawi, A.** An ontology to support the move towards sustainable construction in Saudi Arabia / A. Howsawi, J. Zhang // ASCE Int. Workshop on Computing in Civil Engineering, 296–303 (2017).
- [26] **Bilgin, G.** An ontology-based approach for delay analysis in construction / G. Bilgin, I. Dikmen, M.T. Birgonul // KSCE J. Civ. Eng. 22 (2), 384–398 (2018)
- [27] **Скобелев, П.О.** Применение онтологии в мультиагентной системе адаптивного планирования / П.О. Скобелев, А.А. Жиляев, О.И. Лахин, И.В. Майоров, Е.В. Симонова. // Труды XX Международной конференции «Проблемы управления и моделирования в сложных системах», Самара, 03-06 сентября 2018 г. – Самара: ОФОРТ, 2018. – С.367-375. – ISBN 978-5-473-01200-2.
- [28] **Mustapha, K.** Smart Cities and Resilience Plans: A Multi-Agent Based Simulation for Extreme Event Rescuing /, K. Mustapha, H. Mcheick, S. Mellouli // Dans Smarter as the new urban agenda : a comprehensive view of the 21st century city. (p. 149-170). Public Administration and Information Technology. Texas: Springe (2016).
- [29] **Massei, M.** Simulation of an urban environment by using intelligent agents within asymmetric scenarios for assessing alternative command and control network-centric maturity models / M. Massei, A. Tremori // The Journal of Defense Modelling and Simulation: Applications, Methodology, Technology. 11, 137-153 (2013).
- [30] **Brudermann, T.** Behavioral aspects for agent-based models of resilient urban systems / T. Brudermann, Y. Yamagata // Proceedings of the International Conference on Dependable Systems and Networks. 1-7 (2013).
- [31] **Cavallaro, M.** Assessment of Urban Ecosystem Resilience through Hybrid Social–Physical Complex Networks / M. Cavallaro, D. Asprone, V. Latora, G. Manfredi, V. Nicosia // Computer-Aided Civil and Infrastructure Engineering. (2014).

Сведения об авторах



Георгий Ржевский, профессор. Ученый, предприниматель и консультант (Лондон, Великобритания). Является почётным профессором Центра Сложных Систем, Открытый Университет (Милтон-Кейнс, Великобритания); управляющим директором Digital Ecosystems Ltd, (Лондон, Хельсинки, Москва, Самара, Сингапур, Коломбо). ID автора (Scopus): 6603680750. rzevski@gmail.com.

Сергей Кожевников, научный сотрудник Чешского технического университета (Прага). Окончил Самарский государственный аэрокосмический университет в 2010 году и получил степень магистра по информационным технологиям в авиастроении. В 2015 году окончил аспирантуру в Самарском государственном

техническом университете по специальности «Системный анализ». Научные интересы включают системный анализ, онтологии и базы знаний, интеллектуальный интернет вещей, когнитивные технологии, мультиагентные технологии. Имеет 14-летний опыт работы в IT и AI. ID автора (Scopus): 55575438600. Sergei.Kozhevnikov@cvut.cz.





Мирослав Свитек, доктор технических наук, профессор. Окончил Чешский технический университет в Праге в 1992 году. В 1996 году получил учёную степень по радиозлектронике на электротехническом факультете Чешского технического университета в Праге. С 2005 года был назначен профессором прикладной информатики на факультете естественных наук Университета Матей Бел в Банской Быстрице (Словацкая Республика). С 2008 года является профессором в области инженерной информатики на факультете транспортных наук Чешского технического университета (Прага). С 2018 года является адъюнкт-профессором по теме Умный город Техасского университета (Эль-Пасо, США). Президент чешского кластера умных городов (www.czechsmartcitycluster.cz). ID автора (Scopus): 6602565421. svitek@fd.cvut.cz.

Поступила в редакцию 08.01.2020, после рецензирования 02.03.2020. Принята к публикации 26.03.2020.

Smart City as a Complex Adaptive System

George Rzevski¹, Sergei Kozhevnikov², Miroslav Svitek³

¹ *The Open University, Milton Keynes and Digital Ecosystems Ltd, London*

² *Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University, Prague*

³ *Faculty of Transportation Sciences, Czech Technical University, Prague*

Abstract

As complexity of our political, social and economic environment increases, cities begin to experience new pressures, which makes the transformation from a conventional to a smart city urgent. It also requires re-thinking of what the key elements of the Smart City concept are. The paper does just that, and it provides an updated definition of Smart City. Further, it analyses problems experienced by urban conurbations caused by complexity of the Internet-based global economy and describes a practical solution – to design cities as complex adaptive systems capable of delivering services to citizens and visitors under conditions of complexity. The authors advocate an evolutionary transformation methodology that ensures a minimum of disruptions to city services. The concurrent design of multi-agent real-time schedulers for services and Smart City Ontology, described in this paper, enables the transformation to be carried out by several design teams, each, if necessary, at a different geographical location.

Key words: *Smart City, Complex Adaptive Systems, Urban Ecosystems, Self-organisation, Co-evolution*

Citation: *Rzevski G, Kozhevnikov S, Svitek M. Smart City as a Complex Adaptive System [In Russian]. Ontology of designing. 2020; 10(1): 7-21. DOI: 10.18287/2223-9537-2020-10-1-7-21.*

List of figures and table

Figure 1 – Architecture of the Smart City services

Figure 2 – An example of Ontology for Emergency medical service

Figure 3 – Classes and relations of the ontology

Table 1 – Systems classifications

References

- [1] **Rzevski G, Skobelev P.** Managing Complexity. WIT Press, Southampton, Boston, 2014. ISBN 978-1-84564-936-4.
- [2] **Svitek M.** Telematic approach into program of smart cities, EATIS 2014, Valparaiso - Chile, 2014, ISBN 978-1-4503-2435-9.
- [3] **Svitek M, Skobelev P, Kozhevnikov S.** Smart City 5.0 as an Urban Ecosystem of Smart Services, in Borangiu T., Trentesaux D., Leitão P., Giret Boggino A., Botti V. (eds) Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future. SOHOMA 2019. Studies in Computational Intelligence, vol 853. Springer, Cham, 2020, p.426–438, ISBN 978-3-030-27476-4.
- [4] **Allam Z, Newman P.** Redefining the Smart City: Culture, Metabolism and Governance. Smart Cities. 1, 4 (2018).

- [5] **Kozhevnikov S, Skobelev P, Pribyl O, Svítek M.** Development of Resource-Demand Networks for Smart Cities 5.0. In Mařík V., Kadera P., Rzevski G., Zoitl A., Anderst-Kotsis G., A Min Tjoa, Khalil I. (eds) *Industrial Applications of Holonic and Multi-Agent Systems. HoloMAS 2019. Lecture Notes in Artificial Intelligence*, vol 11710. Springer AG 2019, pp. 203-217, ISBN 978-3-030-27877-9.
- [6] **Batty M, Axhausen K, Giannotti F, Pozdnoukhov A, Bazzani A, Wachowicz M, Ouzounis G, Portugali Y.** Smart cities of the future. *The European Physical Journal Special Topics.* 214, 481-518 (2012).
- [7] **Postranecky M, Svítek M, Carrillo EZ.** SynopCity Virtual HUB – A testbed for Smart Cities, *IEEE Intelligent Transportation Systems Magazin*, Vol. 10, Issue 2, 2018, pp. 50-57, DOI: 10.1109/MITS.2018.2806642.
- [8] **Uribe-Pérez N, Pous C.** A novel communication system approach for a Smart City based on the human nervous system. *Future Generation Computer Systems.* 76 (2017).
- [9] **Trucco P, Petrenj B, Bouchon S, Dimauro C.** Ontology-based approach to disruption scenario generation for critical infrastructure systems. *International Journal of Critical Infrastructures.* 12, 248 (2016).
- [10] **Evgenev GB.** Industry 5.0 as integration of the Internet of Knowledge and the Internet of Things [In Russian]. *Ontology of designing.* 2019; 9(1): 7-23. – DOI: 10.18287/2223-9537-2019-9-1-7-23.
- [11] **Tsvetkov VYA, Buravtsev AV.** Metrics of a complex determinate system. *Ontology of designing.* 2017; 7(3): 334-346. DOI: 10.18287/2223-9537-2017-7-3-334-346.
- [12] **Mikoni SV.** Generalized ontological model of control in the concept of a socio-cyberphysical system [In Russian]. *Ontology of designing.* 2019; 9(2): 191-202. – DOI: 10.18287/2223-9537-2019-9-2-191-202.
- [13] **Maksimov NV, Golitsina OL, Ganchenkova MG, Sanatov DV, Razumov AV.** Semantic core of digital platform [In Russian]. *Ontology of designing.* 2018; 8(3): 412-426. - DOI: 10.18287/2223-9537-2018-8-3-412-426.
- [14] **Santipantakis G, Kotis K, Vouros GA.** OBDAIR: Ontology-Based Distributed framework for Accessing, Integrating and Reasoning with data in disparate data sources, *Expert Systems with Applications.* 90, 464-483 (2017).
- [15] **Costin A, Eastman C.** Need for Interoperability to Enable Seamless Information Exchanges in Smart and Sustainable Urban Systems. *Journal of Computing in Civil Engineering.* 33 (2019).
- [16] **Moraci F, Errigo M, Fazio C, Burgio G, Foresta S.** Making Less Vulnerable Cities: Resilience as a New Paradigm of Smart Planning. *Sustainability.* 10, 755 (2018).
- [17] **Rzevski G.** Intelligent Multi-Agent Platform for Designing Digital Ecosystems. In Vladimir Marik, Petr Kadera, George Rzevski, Alois Zolti, Gabriele Anderst-Kotsis, A Min Yjoa, Ismail Khalil (eds), *Proceedings of the 9th International Conference, HoloMAS 2019, Linz, Austria, August 26–29, 2019*, p.29-41. LNAI 11710.
- [18] **Ouyang M.** A three-stage resilience analysis framework for urban infrastructure systems. *Structural Safety.* 36-37, 23-31 (2012).
- [19] **Agudelo-Vera C, Leduc WRWA, Mels AR, Rijnaarts H.** Harvesting urban resource towards more resilient cities. *Resources, Conservation and Recycling.* 64, 3-12 (2012).
- [20] **Index of Smart Cities**, IMD World Competitiveness Centre, Smart City Observatory, Singapore University of Technology and Design. In: <https://www.imd.org/smart-city-observatory/smart-city-index>.
- [21] **Ganzha M, Paprzycki M, Pawłowski W, Szmeja P, Wasielewska K.** Semantic interoperability in the internet of things: An over-view from the INTER-IoT perspective. *J. Network Comput.* 81, 111–124 (2017).
- [22] **Badii C, Bellini P, Cenni D, Martelli G, Nesi P, Paolucci M.** Km4City Smart City API: An integrated support for mobility services. *IEEE Int. Conf. on Smart Computing*, 1–8 (2016).
- [23] **Gorodetsky VI, Skobelev PO.** Digital platform of cyber-physical systems. XIII All-Russian meeting on management issues of VSPU-2019, Moscow, June 17 - 20, 2019: Transactions. M.: Institute of Management Problems. V.A. Trapeznikova RAS, 2019.
- [24] **Seedah DP, Choubassi C, Leite F.** Ontology for querying heterogeneous data sources in freight transportation. *J. Comput. Civ. Eng.* 30 (4): 04015069 (2016).
- [25] **Howsawi A, Zhang J.** An ontology to support the move towards sustainable construction in Saudi Arabia. *ASCE Int. Workshop on Computing in Civil Engineering*, 296–303 (2017).
- [26] **Bilgin G, Dikmen I, Birgonul MT.** An ontology-based approach for delay analysis in construction. *KSCE J. Civ. Eng.* 22 (2), 384–398 (2018).
- [27] **Skobelev PO, Zhilyaev AA, Lakhin OI, Mayorov IV, Simonova EV.** The use of ontology in a multi-agent adaptive planning system. *Proceedings of the XX International Conference "Problems of Control and Modeling in Complex Systems"*, Samara, September 03-06, 2018 - Samara, 2018. - P.367-375. - ISBN 978-5-473-01200-2.
- [28] **Mustapha K, Mcheick H, Mellouli S.** Smart Cities and Resilience Plans: A Multi-Agent Based Simulation for Extreme Event Rescuing (2016).
- [29] **Massei M, Tremori A.** Simulation of an urban environment by using intelligent agents within asymmetric scenarios for assessing alternative command and control network-centric maturity models. *The Journal of Defense Modelling and Simulation: Applications, Methodology, Technology.* 11, 137-153 (2013).
- [30] **Brudermann T, Yamagata Y.** Behavioral aspects for agent-based models of resilient urban systems. *Proceedings of the International Conference on Dependable Systems and Networks.* 1-7 (2013).

- [31] *Cavallaro M, Asprone D, Latora V, Manfredi G, Nicosia V*. Assessment of Urban Ecosystem Resilience through Hybrid Social–Physical Complex Networks. *Computer-Aided Civil and Infrastructure Engineering*. (2014).
-

About the authors

Prof. George Rzevski, an academic, entrepreneur and consultant, based in London, UK. He is Emeritus Professor, Centre for Complexity and Design, The Open University, Milton Keynes, UK and Managing Director, Digital Ecosystems Ltd, London, Helsinki, Moscow, Samara, Singapore, Colombo. Author ID (Scopus): 6603680750. rzevski@gmail.com.

Sergey Kozhevnikov research scientist at the Czech Technical University in Prague. Graduated from the Samara State Aerospace University in 2010 and has obtained a master degree in Information Technologies in Aircraft construction, specialty “PLM technologies”. In 2015 finished the PhD program in Samara State Technical University, specialty “System Analysis”. Research interests include system optimization and system analysis, ontologies and knowledge bases, smart Internet of Things, cognitive technologies, multi-agent technology for real-time scheduling. 14 years’ experience in the IT and AI. Author ID (Scopus): 55575438600. Sergei.Kozhevnikov@cvut.cz.

Prof. Dr. Ing. Miroslav Svítek, dr.h.c., FEng., EUR ING graduated from the Czech Technical University in Prague, in 1992. In 1996, he received the Ph.D. degree in radioelectronic at Faculty of Electrical Engineering, Czech Technical University in Prague. Since 2005, he has been a professor in applied informatics at Faculty of Natural Sciences, University of Matej Bel in Banská Bystrica, Slovak Republic. Since 2008, he has been full professor in engineering informatics at Faculty of Transportation Sciences, Czech Technical University in Prague. Since 2018, he has been adjunct professor in smart cities at University of Texas at El Paso, USA. Miroslav Svítek is president of Czech Smart City Cluster (www.czechsmartcitycluster.cz). Author ID (Scopus): 6602565421. svitek@fd.cvut.cz.

Received January 8, 2020. Revised March 02, 2020. Accepted March 26, 2020.

Онтологическая система знаний и вычислительных ресурсов современных интеллектуальных технологий

А.В. Бухановский, С.В. Иванов, С.В. Ковальчук, Ю.И. Нечаев

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, Санкт-Петербург, Россия

Аннотация

Обсуждаются концептуальные решения построения онтологической системы знаний в функциональных пространствах современной теории катастроф. Теоретический базис реализации онтологической системы знаний определяет принцип генерации управляющих воздействий в условиях неопределённости на основе иерархической структуры. Анализ альтернатив и выбор решения в среде многофункционального программного комплекса осуществляется на основе ансамблевого прогноза. Интерпретация поведения сложных систем в многофункциональном программном комплексе предсказательных моделей с помощью онтологической системы знаний ведётся в рамках стратегии обработки больших объёмов данных, потока событий и гибридной технологии, а также топологической структуры выявления критических ситуаций с использованием когнитивных и фрактальных структур, нейродинамических и мультиагентных систем, синергетической теории управления. Особое внимание обращается на обоснование и выбор интерпретирующей модели с учётом физических эффектов и закономерностей из условия адекватности.

Ключевые слова: онтологическая система, теория катастроф, система знаний, вычислительные ресурсы, экстренные вычисления.

Цитирование: Бухановский, А.В. Онтологическая система знаний и вычислительных ресурсов современных интеллектуальных технологий / А.В. Бухановский, С.В. Иванов, С.В. Ковальчук, Ю.И. Нечаев // Онтология проектирования. – 2020. – Т.10, №1(35). – С.22-33. – DOI: 10.18287/2223-9537-2020-10-1-22-33.

Введение

Онтологическая система управления сложной системой разнородных знаний интегрирует формализованные знания классической и современной компьютерной математики [1], которые реализуются в нестационарной среде функциональных пространств современной теории катастроф (СТК) [2]. Концептуальный базис построения онтологической системы знаний в среде СТК определяет структурный синтез компонент эволюционирующей среды на базе активной динамической системы [3] (рисунок 1). Одна из особенностей онтологической системы знаний – иерархическая организация, определяющая функции интерпретации и управления в условиях временных задержек, шума и неопределённости.

Стратегическое планирование операций и концептуальных решений на основе онтологической системы в условиях иерархической организации системы знаний отображает фундаментальный результат интеграции компонент динамической модели СТК на базе интеллектуальных технологий и высокопроизводительных вычислений.

Иерархическая модель онтологии позволяет описывать эволюционную динамику сложной системы знаний на различных уровнях абстракции, определяющих функции интерпретации и управления в процессе развития текущей ситуации. При декомпозиции сложной системы реализуется концепция связности, при этом исходная модель знаний представляется совокупностью моделей подуровней, связанных древовидным отношением [4]. Формирование уровней иерархии осуществляется с помощью стандартных оснований декомпозиции.

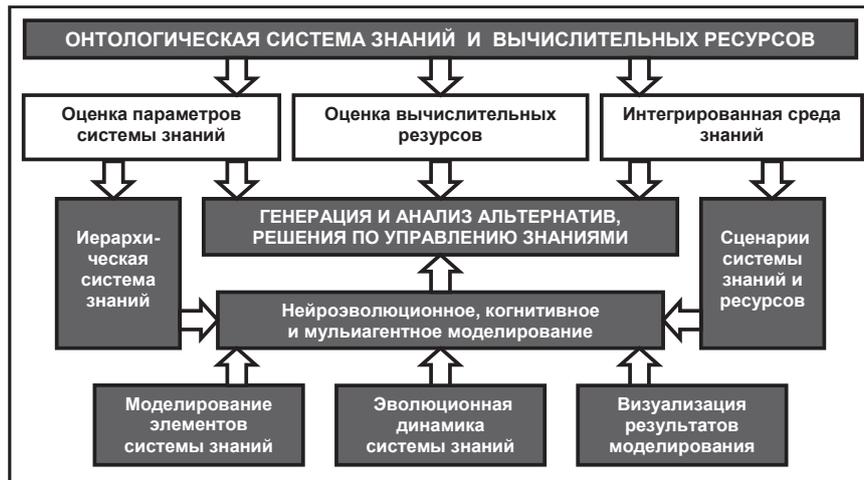


Рисунок 1 - Онтологическая система интегрированной среды интерпретации решений

1 Онтология сложной системы знаний

Концептуальная модель онтологии в функциональном пространстве знаний (рисунок 2) представлена на основе универсальной модели интерпретации сложных систем – динамической модели СТК [2]. Теоретические решения по реализации концептуальной модели онтологии базируются на фундаментальных результатах [1], сформулированных на основе требований минимальной длины описания А.Н. Колмогорова [5] в рамках теории сложности [6], принципа бифуркационного управления Н.Н. Моисеева [7], теории некорректных (обратных) задач А.Н. Тихонова [8].



Рисунок 2 - Функциональное пространство онтологической системы знаний

Онтологическая база инструмента исследования использует методы оценки поведения сложной системы знаний на базе динамической модели СТК [2] – метод функционала действия [9], модифицированные модели Дуффинга и Матъе, нелинейная диаграмма устойчивости Айнса-Стретта, когнитивная парадигма [1], мультиагентное моделирование [10].

Онтология эволюции сложной системы знаний, определяющая фундаментальные принципы и концепцию эволюционирующих систем, может быть представлена в виде определений и утверждений.

Определение 1. Функциональное пространство эволюционной динамики рассматривается как совокупность предельных состояний, определяющих режимы движения к целевому аттрактору и при потере устойчивости:

$$(1) \mathcal{R}^0 \rightarrow \mathcal{R}^n \xrightarrow{\alpha, \beta} \mathcal{R}^r \rightarrow \mathcal{R}(F_R, Attr),$$

где \mathfrak{R}^0 – исходная система состояния модели знаний; $\mathfrak{R}^n, \mathfrak{R}^r$ – системы знаний в пространствах поведения и управления СТК; α и β – ограниченные операторы; $\mathfrak{R}(F_R, Attr)$ – система знаний, определяющая формирование фрактальных и аттракторных структур [1, 2].

Определение 2. Пространство поведения определяет множество состояний сложной системы знаний в виде алгоритмов идентификации, аппроксимации и прогноза в процессе эволюции в нестационарной среде на основе концептуальных решений, критериальных уравнений и нечёткой формальной системы:

$$(2) W(t_0) \rightarrow W^*(t) \xrightarrow{\alpha^*} W(t_i) \dots \xrightarrow{\beta^*} \dots W(t_n),$$

где $W(t_0)$ – исходная система множества состояний; $W^*(t)$ – факторизованная система, реализующая эволюцию знаний в пространстве поведения; $W(t_i)$ и $W(t_n)$ – системы, построенные в результате использования интерпретирующих алгоритмов; α^* – преобразование (гомеоморфизм), определяющее включение системы $W^*(t)$ в $W(t_i)$; β^* – система канонических преобразований (гомоморфизмов) на интервале реализации $[t_0, t_n]$.

Определение 3. Пространство управления формируется на основе процедур построения сценариев развития эволюционной динамики с целью выработки управленческих решений в условиях неопределённости:

$$(3) W^{**}(t) \xrightarrow{\alpha^{**}} W^{**}(t_i) \dots \xrightarrow{\beta^{**}} \dots W^*(t_n),$$

где $W^{**}(t)$ – множество, сформированное по результатам выполнения условий (1), (2); $W^{**}(t_i)$ – множество, определяющее пространство преобразований знаний (генерация альтернатив, построение ансамбля реализаций и выбор решения) при выработке управляющих воздействий; $W^*(t_n)$ – результирующее множество пространства управления в системе интеллектуальной поддержки (ИП); α^{**} – преобразование (гомеоморфизм), определяющее включение системы $W^{**}(t)$ в $W^{**}(t_i)$; β^{**} – система канонических преобразований (гомоморфизмов) на интервале реализации $[t_0, t_n]$.

Критерием целостности функциональной полноты модели знаний служит принцип *неопределённости – дополненности – совместности*, что демонстрирует свойство функциональной отделимости пространства взаимодействия и используется при синтезе функции интерпретации, определяющей выбор решения среди генерируемых альтернатив.

2 Иерархическая среда онтологии сложной системы знаний

Кооперативный характер эволюционной динамики сложной системы знаний порождает способность к нарушению её симметрии. Дифференциация и специализация обеспечивают распределение функций подсистем, а интеграция подсистем приводит к возрастающей способности функциональных возможностей многофункционального программного комплекса (МПК). Иерархическое построение сложной системы ведётся в виде пространственной, временной и функциональной конфигурации (рисунок 3).

Задача построения оптимальной иерархической структуры состоит в построении множества Ω иерархий с заданным функционалом

$$(4) \arg \min G \in \Omega P(G), \quad P: \Omega \rightarrow G [0, +\infty]$$

Множество иерархической структуры задаётся на момент времени t_0 . Динамика сложной системы знаний определяется набором функций f^1, \dots, f^t с известной историей развития ситуации. Управление образует иерархическую систему знаний в момент времени t в виде отображения

$$(5) \psi^t: F \times \dots \times F \times Q(f^{t-1}) \rightarrow Q(f^t),$$

где $Q(f^t)$ представляет собой информацию о внешней среде.

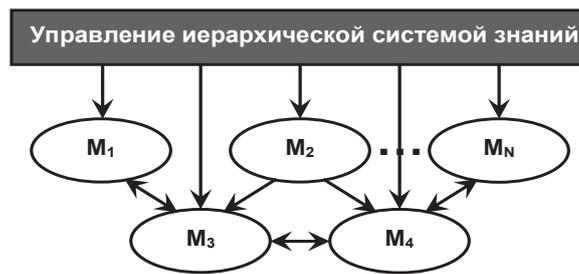


Рисунок 3 - Иерархическая среда сложной системы знаний: M_1, \dots, M_N – модели знаний

Управление в начальный момент времени представляется как отображение.

$$(6) \psi^t : F \rightarrow Q(f^t).$$

Совокупность управлений ψ^t на интервале реализации обозначается $\psi = (\psi^1, \dots, \psi^T)$ и рассматривается как управляющая структура иерархической системы.

На каждом иерархическом уровне вводится описание последовательности операций в пространствах состояний и ситуаций в соответствии со следующими определениями.

Определение 4. Пространство состояний X сложной системы знаний включает в себя тип и число переменных и параметров состояния Ω , функцию отображения $f: X \times \Omega \rightarrow R$, а также описание состояний, соответствующих данному уровню иерархии.

Определение 5. Пространство ситуации сложной системы знаний – множество элементов на данном уровне иерархии, характеризующееся параметрами ситуации и особенностями эволюционной динамики при заданном уровне внешних возмущений.

Определение 6. Математическое описание сложной системы знаний включает множество допустимых входов U и выходов Y , множество состояний Q (пространство ситуаций), функцию перехода $\lambda: Q \times U \rightarrow Q$ и функцию выхода $\gamma: Q \times U \rightarrow Y$.

3 Структурный синтез онтологии сложной системы знаний

Структурный синтез и принципы организации онтологии знаний сложной системы разработаны в рамках концептуальных решений, определяющих функциональные множества иерархической среды [1]. Эти решения основаны на взаимосвязи конфигураций, характеризующих внутреннюю, отображающую и управляющую информацию. Внутренняя информация определяет целенаправленное поведение системы. Отображающая информация содержит сведения об окружающей среде и особенностях текущей ситуации. Управляющая информация – это совокупность управляющих воздействий в заданных внешних условиях.

Модель реального мира, определяющая концептуальное моделирование сложной системы знаний в режиме экстренных вычислений (*Urgent Computing – UC*) [11], обеспечивает реализацию общих принципов обработки информации в рамках динамической модели СТК [2]. Интеграция сложной системы знаний с интеллектуальными технологиями и высокопроизводительными вычислениями представлена на рисунке 4.

Контроль сложной системы знаний на интервале реализации $[t_0, t_k]$ осуществляется с использованием следующих аксиом.

Аксиома 1. Идентификация системы знаний (ситуации) осуществляется из множества альтернативных ситуаций $\{S_j\}, j = 1, \dots, z$, каждой из которых по выбранному критерию R соответствует один из альтернативных алгоритмов $\{A_i\}, i = 1, \dots, q$.

Аксиома 2. Соответствие между исследуемой ситуацией S_i и оптимальной структурой алгоритма $S_i \rightarrow A_i$, реализуется функцией интерпретации $i = \varphi(j), i = 1, \dots, q, j = 1, \dots, z$, с помощью которой решается проблема использования адаптируемой структуры.



Рисунок 4 - Парадигма и онтологические принципы интерпретации знаний в режиме UC

Аксиома 3. Ситуация кодируется вектором $S = (S_1, \dots, S_u)$ в u -мерном пространстве. Каждой точке S_i этого пространства соответствует лучший алгоритм A_i , определяемый как

$$(7) R(A_i, S_i) = \min_{j=1, \dots, q} R(A_j, S).$$

Таким образом, сформулирована q -классовая задача интерпретации сложной системы знаний на основе конечной обучающей выборки $\langle S_j, A_j \rangle$ ($j = 1, \dots, N$), элементы которой определяются решением оптимизационной задачи

$$(8) R(A, S_i) \rightarrow \min_{A_1, \dots, A_q} \Rightarrow A_j,$$

где S_j ($j = 1, \dots, N$) – ситуации, представляющие различные части пространства $\{S\}$.

Практическая реализация аксиом 1 – 3 и концептуальных решений (1) – (8) определяет онтологический принцип использования алгоритмов контроля динамики сложной системы знаний и стратегию управления МПК в режиме UC. Решение задач адаптивного управления ведётся с помощью функции интерпретации (виртуального образа ситуации), осуществляющей генерацию стратегий управления и координацию работы МПК.

4 Онтология эволюционной динамики сложной системы знаний

Онтология пространства знаний сложной системы представлена на рисунке 5.



Рисунок 5 - Онтология пространства знаний сложной системы

Модель онтологии эволюционной динамики сложной системы знаний описывается с помощью переменной $y \in A$ из допустимого множества A . Состояние системы в рассматриваемый момент времени t зависит от управляющих воздействий $u \in U$, $y \in G(u)$. На множестве $U \times A$ задан функционал $\Phi(u, y)$, определяющий эффективность сложной системы знаний. Задача заключается в выборе допустимого управления, которое максимизирует значение эффективности $\Phi(u) \rightarrow \max (u \in U)$.

Динамика сложной системы знаний состоит в переходе из одного состояния в другое под воздействием управления U^* и возмущения W^* :

$$(9) U^* : T \rightarrow U, \quad W^* : T \rightarrow W.$$

В функциональном отношении рассматриваемая система отличается способностью поддержания баланса равновесия между функциями и параметрами системы, управление которой реализуется набором функций:

$$(10) \quad F = F_1 \cap \dots \cap F_k.$$

Пересечение этих множеств описывает процесс принятия решений. Каждый элемент декартова произведения функции F и информационного вектора I_R

$$(11) \quad R \subseteq F \times I_R,$$

определяет уровень интеллектуальности системы.

5 Онтологические принципы прогноза и управления

Онтологические принципы, определяющие стратегию прогноза и управления в сложных системах знаний, представлены на рисунке 6 функциональными блоками концептуальной онтологической модели.



Рисунок 6 - Функциональные блоки прогноза и управления в сложной системе знаний

Концепция интеграции знаний в этом случае рассматривается как активная динамическая система (АДС) [3] управления процессом обработки информации на основе динамической модели СТК [2]. Эволюционирующая структура знаний сложной системы, реализующая стратегию UC , проходит последовательность состояний, определяемых динамической моделью СТК. Эволюционное состояние сложной системы утрачивает устойчивость при возникновении катастрофы. Эволюционные циклы на интервале реализации по своему характеру могут быть качественно различными в зависимости от уровня внешних возмущений и особенностей динамики системы. Физическое понимание качественных явлений в эволюционных процессах связано с *фазовым переходом* эволюционной динамики сложных систем.

Проблемы специализации, дифференциации и распределения функций в АДС знаний представлены в виде модели онтологии функционального пространства.

Утверждение 1. При формировании стратегий ИП используется *механизм информирования* и информационного обеспечения элементов АДС о результатах анализа альтернатив и выбора предпочтительной технологии обработки данных.

Утверждение 2. *Механизм функционирования* АДС определяется целевой функцией, допустимыми множествами решений, интервалом реализации, периодами функционирования в зависимости от стратегий интерпретации и управления при движении контролируемого объекта в процессе эволюции.

Утверждение 3. Механизм управления активной АДС формируется в виде совокупности стратегий ИП, информационного и алгоритмического обеспечения УС.

Задачи прогноза и управления сложной системой знаний в пространствах поведения и управления динамической модели СТК содержат неопределённости, свойственные задачам моделирования на основе данных динамических измерений. Процедуры принятия решений направлены на выбор вектора управляемых переменных в области поиска $Z \in D_Z$ так, чтобы удовлетворить условиям выбора. Область пространства управляемых переменных, в которых выполняются все наложенные условия, называют *областью работоспособности* D_R , а множество $D = D_Z \cap D_R$ – *областью допустимых решений*.

Критерий оптимальности представляется отображением на множестве решений

$$(12) \quad Q := M_1 \rightarrow R^+,$$

где R^+ – множество неотрицательных вещественных чисел.

Функция Q реализует процедуру сравнения вариантов решения $m' \in M'$, если

$$(13) \quad Q(m') = \min Q'(m').$$

Оптимизационная задача $\langle M, D, Q \rangle$ содержит M – пространство решений, D – ограничения в M допустимой области

$$(14) \quad M' \subseteq D, Q: M' \rightarrow R^+ \text{ с критерием оптимальности } Q.$$

6 Методы интерпретации знаний сложных систем

В качестве примеров практической реализации рассмотрены сложные системы знаний при обработке информации на основе нейродинамического, когнитивного и мультиагентного моделирования в пространствах поведения и управления СТК.

6.1 Нейродинамическое моделирование сложной системы знаний

Процедуры синтеза нейросетевых структур [8] основаны на закономерности, определяющей *подобные упорядочения* в среде знаний: отношение \leq для множества X и отношение \leq^* для множества Y . Эти операции обеспечивают условие существования взаимно однозначного отображения f (отображение подобия) множества X на множество Y такое, что

$$(15) \quad (x_1 \leq x_2) \equiv f(x_1) \leq^* f(x_2).$$

Отношение подобия для двух упорядоченных множеств является *отношением эквивалентности*. Отмеченная закономерность выражает фундаментальный результат преобразования информации, основанного на традиционных процедурах классической математики и нейросетевого базиса. Система нейродинамического моделирования при интерпретации сложной системы знаний представлена на рисунке 7. Здесь содержатся различные модели искусственной нейронной сети, реализующие операции преобразования знаний в пространствах поведения и управления СТК.

Нейросетевой ансамбль (рисунок 7а) реализует процедуры идентификации при оценке ситуации и логического вывода по прецеденту [9]. Многослойный перцептрон (рисунок 7б) используется в задачах контроля динамических характеристик и при выявлении «скрытых» знаний в рамках концепции *Data Mining* [1]. Нейросетевой предиктор (рисунок 7в) обеспечивает прогноз эволюционной динамики на интервале реализации.

6.2 Когнитивное моделирование сложной системы знаний

Когнитивная модель сложной системы знаний на основе когнитивной карты представляет собой параметрический векторный функциональный граф [1]. В качестве примера на ри-

сунке 8 рассмотрена когнитивная карта, отображающая поведение сложной системы знаний на основе СТК.

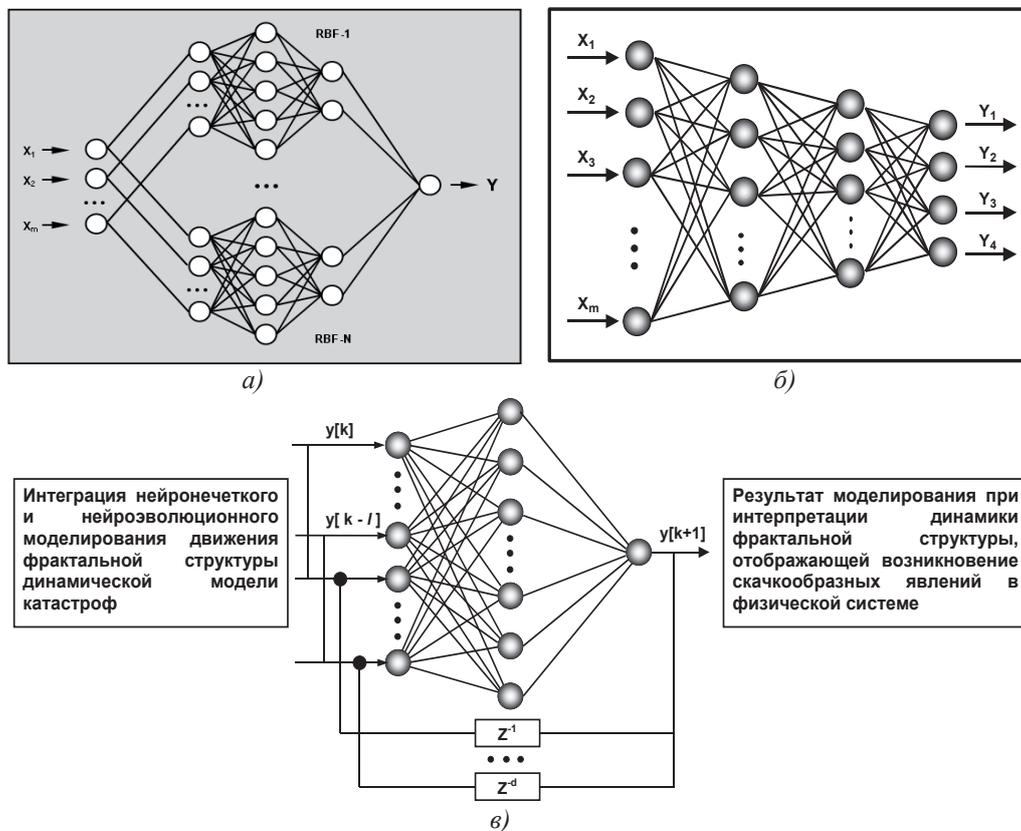


Рисунок 7 - Нейронные сети в задачах идентификации (а), аппроксимации (б) и прогноза (в)

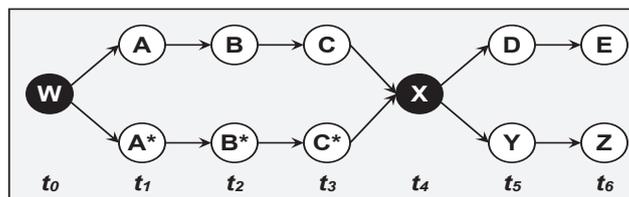


Рисунок 8 - Универсальная когнитивная карта: $A - E$ – этапы эволюции системы; X, D, E – фазы движения к целевому аттрактору; X, Y, Z – фазы, характеризующие потерю устойчивости

Картина поведения интерпретируется следующим образом. В результате контроля развития ситуации под воздействием внешнего возмущения W и априорных данных был реализован прогноз состояния системы на основе принципа конкуренции (альтернативы A и A^*). В течение развития ситуации применены управляющие воздействия (ситуации B и B^*), предсказанные моделью эволюционной динамики, применение которых привело к изменению поведения системы (ситуации C и C^*). Однако в момент времени t_4 (символ X) вследствие интенсивного внешнего возмущения произошло резкое изменение поведения системы. Это событие выходило за рамки предсказанного и вызвало изменение эволюции системы. Контроль вновь возникшей ситуации при эффективности управляющих воздействий (ситуация D) привёл к стабилизации ситуации в области целевого аттрактора (ситуация E). При недостаточной эффективности контроля (ситуация Y) возникает потеря устойчивости движения системы (ситуация Z).

6.3 Мультиагентное моделирование сложной системы знаний

Практическая реализация мультиагентных систем (МАС) в сложной системе знаний рассмотрена применительно к задачам контроля транспортных систем. В качестве примеров выбраны приложения, связанные с контролем движения транспортных потоков (рисунок 9) и эвакуации пассажиров и экипажа с аварийного судна (рисунок 10) [10]. Движение агентов в процессе эвакуации реализовано с учётом угловых перемещений, локальных скоростей и ускорений нерегулярной качки. Пакетный визуализатор обеспечивает анализ ситуаций на основе МАС.

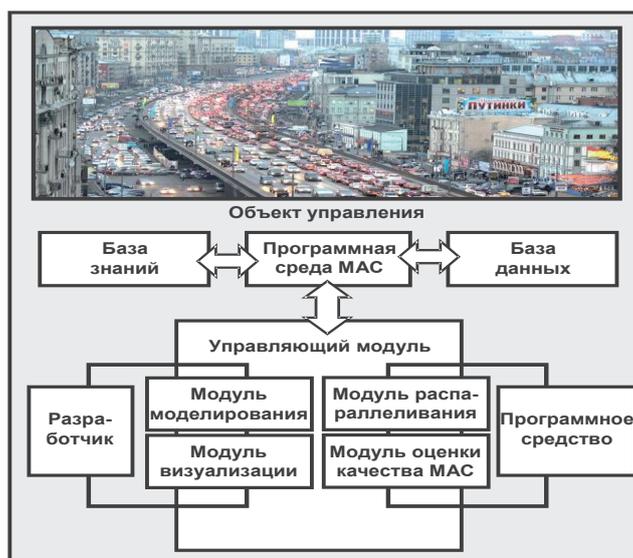


Рисунок 9 - Среда МАС с визуализацией транспортного потока

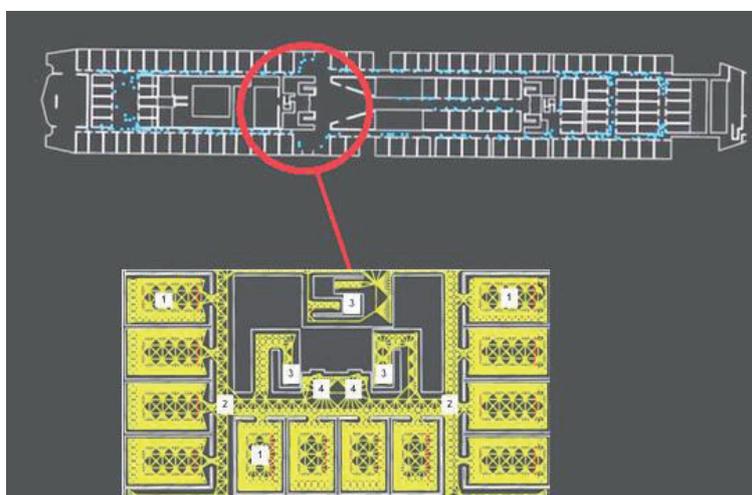


Рисунок 10 - Геометрическая модель судна (одна палуба) и фрагмент графа пешеходных перемещений (включая трапы), построенный растеризацией по квадродеревьям:
1 – каюты; 2 – коридоры; 3 – трапы; 4 – лифты

Заключение

Разработанная онтологическая система, определяющая концептуальные решения и методы интерпретации знаний, обеспечивает построение и использование моделей взаимодействия в рамках парадигмы АДС, предполагающей реализацию теоретических принципов и

механизмов обработки информации в условиях восприятия случайных воздействий внешней среды на основе интеллектуальных технологий и высокопроизводительных вычислений. Онтологическая система знаний реализуется в рамках эволюционной парадигмы СТК при формировании потенциальных свойств интерпретирующих и управляющих моделей на основе искусственных нейронных сетей, когнитивной парадигмы и МАС. Ядром онтологического синтеза сложной системы знаний является МПК, обеспечивающий построение алгоритмов и программного обеспечения прикладных задач обработки знаний в различных областях практических приложений.

Список источников

- [1] **Нечаев, Ю.И.** Современные проблемы информатики и вычислительной техники / Ю.И. Нечаев. – Санкт-Петербург: Арт-Экспресс, 2018. – 315 с.
- [2] **Нечаев, Ю.И.** Теория катастроф: современный подход при принятии решений / Ю.И. Нечаев. – Санкт-Петербург: Арт-Экспресс, 2011.– 392 с.
- [3] **Новиков, Д.А.** Курс теории активных систем / Д.А. Новиков, С.Н. Петраков – М.: СИНЕГ, 1999. – 104 с.
- [4] **Касты, Дж.** Большие системы: связность, сложность и катастрофы / Дж. Касты // Пер. с англ. под ред. д-ра физ.-мат. наук Ю. П. Гупало и канд. физ.-мат. наук А. А. Пионтковского – М.: Мир, 1982. – 216 с.
- [5] **Колмогоров, А.Н.** Избранные труды / А.Н. Колмогоров. – М.: Наука, 2005. – 304 с.
- [6] **Солодовников, В.В.** Теория сложности и проектирование систем управления / В.В. Солодовников, В.И. Тумаркин. - М.: Наука, 1990. – 341 с.
- [7] **Моисеев, Н.Н.** Избранные труды / Н.Н. Моисеев. - М.: Тайрекс Ко. 2003. т.1 – 376 с., т.2 – 264 с.
- [8] **Тихонов, А.Н.** Методы решения некорректных задач / А.Н. Тихонов, В.Я. Арсенин. - М.: Наука, 1986. – 285 с.
- [9] Нейрокомпьютеры в интеллектуальных технологиях XXI века. – М.: Радиотехника, 2011. – 352 с.
- [10] **Нечаев, Ю.И.** Проблемы мультиагентного моделирования на основе нейродинамических систем и современной теории катастроф / Ю.И. Нечаев // XX Международная научно-техническая конференция «Нейроинформатика-2018». - М.: НИЯУ МИФИ. 2018, с.157–199.
- [11] **Siew Hoon Leong, Dieter Kranzlmüller.** Towards a General Definition of Urgent Computing. Proc. Computer Science. Vol.51, 2015, P.2337-2346. <https://doi.org/10.1016/j.procs.2015.05.402>.

Сведения об авторах



Бухановский Александр Валерьевич, доктор технических наук, профессор, директор Национального центра когнитивных разработок Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики (НЦКР Университета ИТМО). AuthorID (РИНЦ): 9142. Author ID (Scopus): 6603474810; Researcher ID (WoS): B-4793-2014. avb_mail@mail.ru.



Иванов Сергей Владимирович, к.т.н., с.н.с. НЦКР Университета ИТМО. AuthorID (РИНЦ): 154943. Author ID (Scopus): 57197019872; Researcher ID (WoS): U-2001-2017. svivanov@mail.ifmo.ru.



Ковальчук Сергей Владимирович, д.т.н., с.н.с. НЦКР Университета ИТМО. AuthorID (РИНЦ): 503924. Author ID (Scopus): 55382199400; Researcher ID (WoS): A-3025-2010. kovalchuk@mail.ifmo.ru.



Нечаев Юрий Иванович, заслуженный деятель науки РФ, академик РАЕН, д.т.н., профессор, ведущий научный сотрудник НЦКР Университета ИТМО, Международный эксперт в области высокопроизводительных вычислений и интеллектуальных систем. AuthorID (РИНЦ): 2770. Author ID (Scopus): 55669900400. nechaev@mail.ifmo.ru.

Поступила в редакцию 30.10.2019, после рецензирования 11.03.2020. Принята к публикации 25.03.2020.

Ontological system of knowledge and computing resources of modern intellectual technologies

A.V. Boukhanovsky, S.V. Ivanov, S.V. Kovalchuk, Yu.I. Nechaev

St. Petersburg National Research University of Information Technologies, Mechanics and Optics,
St. Petersburg, Russia

Abstract

The conceptual solutions of building an ontological knowledge system in the functional spaces of the modern theory of disasters are discussed. The theoretical basis for the implementation of an ontological knowledge system based on a multifunctional software complex determines the principle of generating control actions in conditions of uncertainty based on a hierarchical structure. The analysis of alternatives and the choice of a solution in the multifunctional software complex environment is based on the ensemble forecast. The interpretation of the behavior of complex systems in the multifunctional software complex of predictive models using an ontological knowledge system is carried out as part of a strategy for processing large volumes of Big Data, Work Flow events and hybrid technology, as well as a topological structure for identifying critical situations using cognitive and fractal structures, neurodynamic and multi-agent systems, synergetic control theory. Particular attention is paid to the justification and choice of the interpretive model, taking into account physical effects and patterns from the condition of adequacy.

Key words: ontological system, modern catastrophe theory, complex knowledge system, computing resources, urgent computing.

Citation: Bukhanovsky AV, Ivanov SV, Kovalchuk SV, Nechaev YuI. Ontological system of knowledge and computing resources of modern intellectual technologies [In Russian]. *Ontology of designing*. 2020; 10(1): 22-33. – DOI: 10.18287/2223-9537-2020-10-1-22-33.

List of figures

- Figure 1 - Ontological system of integrated decision interpretation environment
- Figure 2 - Functional space of the ontological knowledge system
- Figure 3 - Hierarchical environment of a complex knowledge system: M1, ..., MN - knowledge models
- Figure 4 - Paradigm and ontological principles of interpretation of knowledge in UC mode
- Figure 5 - Ontology of the space of a complex knowledge system
- Figure 6 - Functional blocks of forecasting and management in a complex knowledge system
- Figure 7 - Neural networks in problems of identification (a), approximation (b) and forecast (c)
- Figure 8 - Universal cognitive map: A - E - stages of system evolution; X, D, E - phases of movement to the target attractor; X, Y, Z - phases characterizing the loss of stability
- Figure 9 - MMS environment with traffic flow visualization
- Figure 10 - Geometrical model of a vessel (one deck) and a fragment of the graph of passenger movements (including ladders) constructed by rasterization along quad trees: 1 – cabins; 2 - corridors; 3 - ladders; 4 - elevators

References

- [1] *Nechaev Yu.I.* Modern problems of computer science and computer technology [In Russian]. St. Petersburg: Art Express. 2018. 315 p.
- [2] *Nechaev Yu.I.* Disaster Theory: A Modern Approach to Decision Making [In Russian]. St. Petersburg: Art Express, 2011. 392 p.
- [3] *Novikov DA, Petrakov SN.* The course of the theory of active systems [In Russian]. Moscow: SINEG, 1999. 104p.
- [4] *John Casti.* Connectivity, Complexity, and Catastrophe in Large-Scale Systems. New York University. A Wiley-Interscience Publication International Institute for Applied Systems Analysis. JOHN WILEY & SONS Chichester - New York - Brisbane – Toronto. 1979.
- [5] *Kolmogorov AN.* Selected works in 6 volumes [In Russian]. - Moscow: Science, 2005. 304 p.
- [6] *Solodovnikov VV, Tumarkin VI.* Complexity theory and design of control systems [In Russian]. Moscow: Science, 1990. 341 p.

- [7] **Moiseev NN.** Selected Works [In Russian]. Moscow: Tireks Co. 2003. V.1. 376 p., V.2. 264 p.
 - [8] **Tikhonov AN, Arsenin VYa.** Methods for solving incorrect tasks [In Russian]. Moscow: Science, 1986. 285 p.
 - [9] Neurocomputers in the intelligent technologies of the 21st century [In Russian]. Moscow: Radio engineering, 2011. 352 p.
 - [10] **Nechaev Yu.I.** Problems of multi-agent modeling based on neurodynamic systems and the modern theory of disasters [In Russian]. XX International Scientific and Technical Conference "Neuroinformatics-2018". Lectures on neuroinformatics. Moscow: NRNU MEPhI. 2018. P.157-199.
 - [11] **Siew Hoon Leong, Dieter Kranzlmüller.** Towards a General Definition of Urgent Computing. Proc. Computer Science. Vol.51, 2015, P.2337-2346. <https://doi.org/10.1016/j.procs.2015.05.402>.
-

About the authors

Alexandr Valerievich Boukhanovsky, D.Sc., professor, director of National Centre of cognitive science of the St.-Petersburg National Research University Information Technologies, Mechanics and Optics. AuthorID (RCI): 9142. Author ID (Scopus): 6603474810; Researcher ID (WoS): B-4793-2014. ORCID 0000-0003-1588-8164. avb_mail@mail.ru.

Sergei Vladimirovich Ivanov, PhD, main scientific employee of National center of cognitive science of the St.-Petersburg National Research University Information Technologies, Mechanics and Optics. AuthorID (RCI): 154943. Author ID (Scopus): 57197019872; Researcher ID (WoS): U-2001-2017. ORCID 0000-0002-4051-8803. svivanov@mail.ifmo.ru.

Sergei Vladimirovich Kovalchuk, D.Sc., main scientific employee of National centre of cognitive science of the St.-Petersburg National Research University Information Technologies, Mechanics and Optics. AuthorID (RCI): 503924. Author ID (Scopus): 55382199400; Researcher ID (WoS): A-3025-2010. ORCID 0000-0001-8828-4615. kovalchuk@mail.ifmo.ru.

Yury Ivanovich Nechaev, Academician of RANS, Russian Federation Science Honoured Figure, the main scientific employee of National centre of cognitive science of the St.-Petersburg National Research University Information Technologies, Mechanics and Optics. International expert in the field of high-performance computing and intelligence systems. AuthorID (RCI): 2770. Author ID (Scopus): 55669900400. nechaev@mail.ifmo.ru.

Received October 30, 2019. Revised March 11, 2020. Accepted March 25, 2020.

ПРИКЛАДНЫЕ ОНТОЛОГИИ ПРОЕКТИРОВАНИЯ

УДК 004.85

DOI: 10.18287/2223-9537-2020-10-1-34-49

Модели и методы индивидуализации электронного обучения в контексте онтологического подхода

Д.И. Муромцев

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, Санкт-Петербург, Россия

Аннотация

Рассматривается индивидуализация электронного обучения (ЭО) как совокупности процессов создания, развития, использования и утилизации цифрового контента и данных ЭО, описаны соответствующие онтологические модели и методы. Описан технологический стек для построения и реализации индивидуальных траекторий обучения, приведены примеры существующих систем, которые полностью или частично используют указанный стек технологий. Для эффективной обработки образовательных материалов и данных, формируемых системами управления обучением, предложена архитектура, позволяющая осуществить семантическое аннотирование и выделение в данных слоёв концептов различной степени абстракции. Эти слои включают: верхнеуровневые абстракции моделирования, общие концепты учебных материалов и образовательного процесса, специфические концепты для доступа и интеграции данных системы ЭО в терминах предметной области. Впервые в качестве формальной основы для индивидуализированного ЭО предложено использовать семантические модели, включающие аппарат векторных представлений графов знаний, который позволяет эффективно обрабатывать большие и сложные структуры данных, а также обладает гибкостью и выразительностью онтологического подхода. Последовательно рассмотрены основные аспекты, связанные с индивидуализацией в системах ЭО, в том числе: существующие технологии и онтологии для ЭО, моделирование индивидуальной траектории, семантическое аннотирование образовательных материалов, способы оценки знаний в индивидуализированном обучении, а также онтологическое моделирование когнитивного профиля обучаемого.

Ключевые слова: электронное обучение, индивидуализация, онтологии, графы знаний, векторное представление, семантические технологии.

Цитирование: Муромцев, Д.И. Модели и методы индивидуализации электронного обучения в контексте онтологического подхода / Д.И. Муромцев // Онтология проектирования. – 2020. – Т. 10, №1(35). – С.34-49. – DOI: 10.18287/2223-9537-2020-10-1-34-49.

Введение

Некоторые результаты исследований проблем индивидуализации электронного обучения (ЭО) изложены в работах [1-6]. Важно отметить, что в процессах ЭО или управления знаниями взаимодействие происходит не между человеком и системой управления, а между цифровым следом и цифровыми артефактами обучения и интеллектуальной системой управления знаниями. Эта особенность порождает целый ряд новых проблем, таких как поиск эффективных способов представления данных индивидуализированного обучения, моделирование цифровых артефактов, порождаемых системами ЭО, интеллектуализация анализа образовательных данных, автоматизация процессов адаптации и актуализации образовательного контента и ряд других. Процессы индивидуализации ЭО являются комплексными, затрагивают формирование индивидуальных компетенций у обучающихся и относятся к системам

управления обучением (*Learning Management System, LMS*), добавляя им требования по синтезу индивидуализированного контента и обеспечению автоматического подбора наиболее релевантных (индивидуализированных) средств оценки знаний.

Понятие индивидуализации имеет несколько корней: это и индивидуализированные методики [1-3], и индивидуализированные технологии [4-6]. В контексте данной работы именно второй аспект имеет наибольший интерес, так как напрямую связан с процессами создания, развития, использования и утилизации цифрового контента и данных ЭО.

С точки зрения конечного результата индивидуализированные процессы ЭО играют огромную роль, так как являются основой для функционирования и устойчивого развития экосистемы цифровой экономики и цифрового общества. Тенденции в промышленном производстве по переходу от массового к персонализированному, интеллектуализация всех сфер человеческой деятельности, роботизация рутинного труда формируют новые профессии, предполагающие наличие специалистов с уникальными наборами знаний.

Создание систем индивидуализированного ЭО и управления знаниями предполагает использование специальных технологий, баз знаний (БЗ) и интеллектуальных алгоритмов анализа данных. В работе анализируется применение к решению поставленных задач таких интеллектуальных технологий, как графы знаний, онтологии и машинное обучение (МО).

1 Технологическая основа для построения БЗ

Сегодня основной тенденцией создания электронного образовательного контента является технология MOOC (массовых открытых онлайн-курсов, *Massive Open Online Courses, MOOC*). Данная технология ориентирована на массовое обучение. Это порождает несоответствие между потребностью обучающихся и возможностями (содержанием) электронных образовательных курсов (и образовательных программ в целом).

Решение задачи индивидуализации как систем ЭО, в частности, так и систем управления знаниями в различных областях может быть достигнуто за счёт превращения баз данных (БД) и хранилищ медиа-контента в системах *LMS* в полноценные БЗ, предоставляющие соответствующие модели репрезентации знаний и методы логического вывода и интеллектуального поиска. Для этого необходима технологическая база для интеллектуализации задач управления контентом и процессами обучения. Современным подходом при построении БЗ является стек семантических технологий. Его основой является язык *The Resource Description Framework (RDF)*, представляющий собой семантическую графовую модель и предназначенный для репрезентации полуструктурированных данных о фактах реального мира или абстракциях. Стандартизованный консорциумом *W3C* [7] *RDF* специфицирует архитектуру, синтаксис и семантику, а также базовый словарь *RDF Schema (RDFS)* для построения моделей предметных областей (ПрО) [8, 9].

Основным элементом языка *RDF* является тройка вида <субъект, предикат, объект>, где субъекты и объекты могут быть уникальными сущностями или наименованными сущностями для представления более сложных конструкций (вложенных подграфов, множеств и др.). Каждая сущность имеет свой универсальный и уникальный идентификатор ресурса — *URI (Uniform Resource Identifier)*. *URI* необходимы для того, чтобы была возможность ссылаться на описываемые сущности. Например, идентификатор Университета ИТМО может содержать *http://en.ifmo.ru/ITMO_University*, где префикс *http://en.ifmo.ru/* — адрес в Интернете. Для удобства полные *URI* можно сокращать в префиксы и использовать запись *prefixName:Entity*. Например, термины *RDF* имеют стандартный префикс *rdf:*, что заменяет <*http://www.w3.org/1999/02/22-rdf-syntax-ns#*>.

Неименованные вершины (*blank nodes*) — анонимно заданные сущности без идентификатора или литерала, могут содержать другие отношения и значения. Они используются для описания сложных предикатов и других конструкций в процессе моделирования. Объектами могут быть также простые строковые литералы, представляющие значения атрибутов субъектов. Предикаты обозначают отношения между субъектами или объектами или свойства (атрибуты) субъектов. Формально тройки можно представить как элементы $x_{ijk} = (e_i, r_k, e_j)$, где $E = \{e_1, \dots, e_{N_e}\}$ — множество всех сущностей (субъектов или объектов), а $R = \{r_1, \dots, r_{N_r}\}$ — множество всех связей (отношений) на графе.

Множество троек формирует *RDF*-граф, который, в свою очередь, можно определить формально [10]. Пусть U, B, L — непересекающиеся бесконечные множества *URI*, неименованных вершин и литералов соответственно. Тогда *RDF*-граф G можно определить как направленный помеченный мультиграф $G = (E, R, \Sigma, L)$, где:

$E \subset (U \cup B \cup L)$ - конечное множество *RDF*-термов, соответствующих узлам графа;

$R \subseteq E \times E$ - конечное множество дуг, связывающих *RDF*-термы;

$\Sigma \subset U$ - множество уникальных меток, определённых с помощью *URI*;

$L: R \rightarrow 2^{\Sigma}$ - отображение дуг на множество меток.

Язык *RDF* используется для описания лишь базовых элементов графа знаний, например, «нечто имеет такой-то тип» или «что-то связано с чем-то», но не позволяет определять классы (группы имеющих сходные атрибуты) или настраивать множества допустимых значений для атрибутов. Расширение *RDFS* для языка *RDF* вводит дополнительные предикаты для построения более сложных моделей, включая иерархии. Это такие предикаты, как *rdfs:Class* для определения классов, *rdfs:Literal* для определения литералов, *rdfs:subClassOf* и *rdfs:subPropertyOf* для определения иерархических отношений. Дополнительно *RDFS* позволяет определять области определения и области значений для отношений между сущностями, а также ряд других возможностей.

Для построения сложных моделей ПрО, использующих в качестве формальной семантики логические выражения, используют язык *Web Ontology Language (OWL)* [11], являющийся расширением языка *RDFS*. Существует достаточно большое количество онтологий, разработанных с помощью языков *RDF*, *RDFS* и *OWL*, которые могут быть полезны при создании индивидуализированных систем ЭО, в том числе следующие.

- *The Academic Institution Internal Structure Ontology (AIIISO)* [12] является онтологией, описывающей внутреннюю организационную структуру образовательного процесса. *AIIISO* предоставляет классы и свойства для описания курсов, модулей, практических и теоретических учебных материалов.
- *The Bibliographic Ontology (BIBO)* [13] является онтологией, описывающей библиографические ресурсы. Словарь *BIBO* может использоваться для описания рекомендованной литературы, научных публикаций, методических пособий и монографий.
- *The Ontology for Media Resources (MA-ONT)* [14] является онтологией, описывающей медиа ресурсы. С помощью классов и свойств *MA-ONT* производится связывание лекций с видеоматериалами.
- Онтология для описания учебных материалов *TEACH (Teaching Core Vocabulary)* [15] является облегчённым словарем, позволяющим преподавателям связывать объекты электронных курсов.
- Онтология *FOAF (Friend of a Friend)* [16] определяет некоторые выражения, используемые в высказываниях о ком-либо, например: имя, пол и другие характеристики.

Существуют также системы ЭО, построенные на основе семантических технологий, например, следующие.

- Проект *Metacademy* [17], представляющий собой платформу для открытого персонализированного образования. В основе обучения в данной системе лежат концепты ПрО. Пользователь может составить учебный курс или его дорожную карту на основе концептов, которые он хочет изучить. Все учебные материалы в системе хранятся в онтологиях, что позволяет пользователям осуществлять навигацию по теоретическим материалам. В *Metacademy* весь учебный материал, курсы, лекции, книги связаны друг с другом с помощью концептов ПрО.
- Проект *SlideWiki* [18], в рамках которого создана платформа для создания презентаций для учебных курсов. С помощью семантических технологий платформа позволяет повторно использовать уже опубликованные слайды презентаций, аннотировать дополнительной информацией концепты на слайдах и поддерживать множество языков для одного учебного курса [19].

Однако, несмотря на гибкость и выразительных моделей и широкие возможности по интеллектуализации систем ЭО, перевод существующих курсов в семантический формат является необходимым, но недостаточным условием для достижения целей индивидуализации. Одним из главных препятствий выступает грануляция контента, которая чаще всего отражает структуру курса и виды контента, но недостаточна для построения индивидуальных траекторий обучения (ИТО). Кроме того, в системе отсутствуют модели обучаемого и его знаний, приобретаемых в процессе изучения контента. Можно сказать, что без такой модели построить адекватную систему индивидуализации не предоставляется возможным. Она необходима как для выстраивания персонализированных рекомендаций по прохождению курсов, так и для организации адекватной оценки полученных знаний с учётом индивидуальных потребностей обучаемого. При формировании тестовых и проверочных заданий из всего множества оценочных средств необходимо выбирать только те, которые связаны с ИТО.

Как правило, учебный курс организован линейно и состоит из множества модулей и тем внутри модулей (см. рисунок 1).

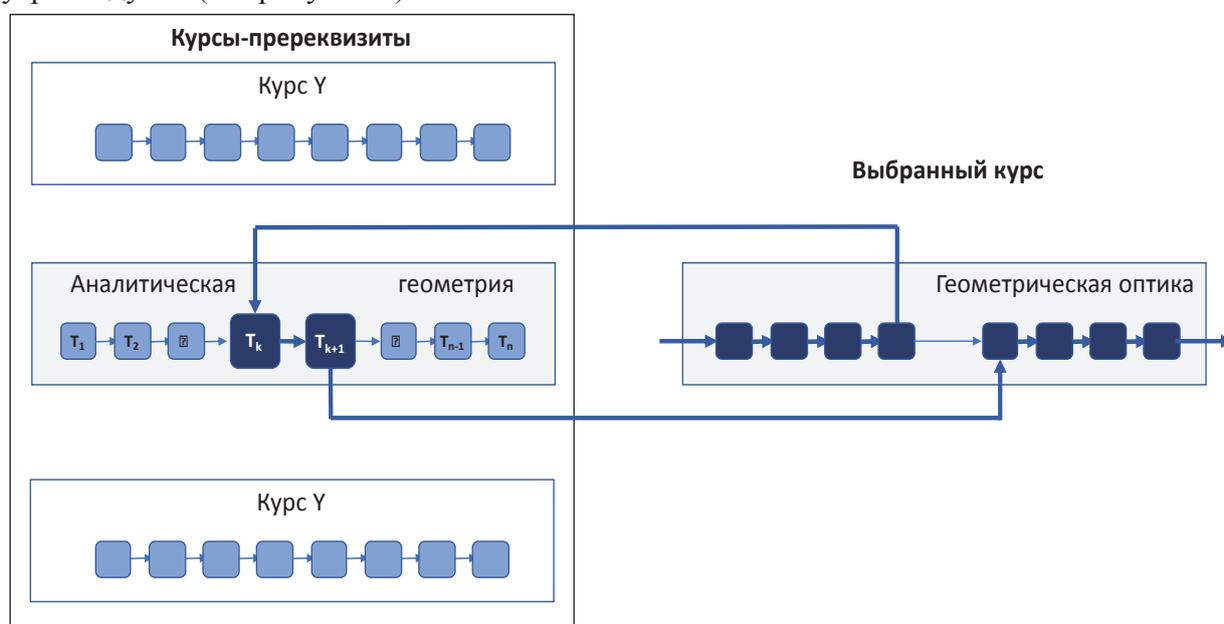


Рисунок 1 - Множество концептов в курсе и позиции точек перехода (изучаемых терминов) от курса к курсу

Выраженные явно или неявно семантические зависимости между модулями задаются в так называемых пререквизитах курса. Т.е. можно выстроить цепочку модулей или курсов, которые необходимо изучить, в зависимости от конкретного выбора обучающегося. Так, ес-

ли обучающийся выбрал вводный курс по геометрической оптике и для понимания, например, явления интерференции ему необходимо также изучить правила сложения векторов, которые вводятся в курсе аналитической геометрии. Такую информацию можно найти в пререквизитах курса по оптике. Но включение в ИТО всего курса по аналитической геометрии или даже модуля по векторной алгебре будет избыточным для данного обучающегося. Достаточно лишь ограничиться необходимыми лекциями из курсов пререквизитов, иначе обучающийся может получить слишком перегруженную ИТО. Аналогичные рассуждения верны и для выбора тестов и заданий из фонда оценочных средств для оценки знаний в процессе прохождения по ИТО. На рисунке 1 показаны гранулы курса и ИТО приведённого примера. Каждая гранула соответствует одному из изучаемых терминов T_i . Из рисунка видно, что точки перехода от курса к курсу могут не совпадать с границами модулей.

Важно отметить, что ИТО не является статической. По ходу освоения элементов курса она может изменяться, дополняясь новыми концептами, если это необходимо для понимания материала и достижения целей обучения. Процесс построения ИТО имеет рекурсивный характер. Так, если в рассмотренном примере при изучении темы «сложение векторов» требуется изучить понятие «вектор», то включение этой гранулы в ИТО может быть выполнено аналогичным способом, как это было проделано с темой «интерференция».

2 Индивидуальные траектории обучения

Многие LMS предлагают средства управления контентом для построения ИТО, позволяя создавать индивидуальную последовательность курсов. Однако есть серьёзное препятствие, связанное с формальным учётом и содержательным анализом данных, необходимых для индивидуализации ЭО.

Управление компетенциями в существующих системах ЭО основано на предположении успешного усвоения курсов, без учёта индивидуальных способностей и интересов обучаемого (и часто без обратной связи). В этом смысле системы являются линейными, а само обучение представляет собой монотонный процесс по достижению целей обучения. Между тем, это грубое приближение к реальным образовательным процессам. Построение ИТО в подавляющем большинстве случаев может быть немонотонным, предполагая возникновение новых точек, через которые должна пройти ИТО. Заранее предположить все возможные точки ИТО сложно по нескольким причинам.

- ИТО возникает в результате проекции друг на друга нескольких онтологических моделей (модели курсов, модели оценки знаний, когнитивная модель обучающегося и др.). Построение полного пространства поиска на основе этих моделей может оказаться вычислительно сложной задачей с множеством противоречий, что требует применения различных эвристик для его оптимизации.
- В процессе обучения производится изменение некоторых из этих моделей. Например, когнитивная модель обучающегося пополняется новыми сущностями по мере продвижения по ИТО, сам обучающийся может вносить коррективы, уточняя свои потребности, возможно изменение моделей курсов, т.к. процесс обновления контента в общем случае может происходить в любое время; модели формирования ИТО также могут меняться по мере накопления данных и изменении образовательного контента, и т.п.
- На процесс обучения могут влиять внешние факторы, например изменяющиеся запросы рынка, различные экономические, социальные и пр. факторы, связанные с образованием.

Индивидуализацию ЭО следует рассматривать не как управление образовательным процессом, а как новую технологию по созданию различных систем интеллектуализации и управления в образовании, которая включала бы такие инструменты и методы как:

- методы онтологического инжиниринга, в том числе автоматическое построение и пополнение онтологий на основе мультимодальных данных;
- методы МО;
- средства семантического анализа и поиска;
- средства выработки рекомендаций и др.

Современные *LMS*, помимо накопленного контента, как правило предоставляют необходимый базовый технологический уровень для построения на их основе индивидуализированных систем обучения, т.к. для хранения данных используются БД, в том числе графовые и *NoSQL* БД, модели репрезентации курсов не являются жёсткими и допускают внесение изменений в их структуру и состав, ведётся достаточно подробное журналирование поведения пользователя в системе, что позволяет выполнять детальный анализ этого поведения. Не представляет труда подключение дополнительных сервисов, что позволяет интегрировать в *LMS* элементы новой технологии и сохранить преемственность в образовательных процессах.

Для удовлетворения описанных требований к системе индивидуализации ЭО необходим стек технологий, обеспечивающий интероперабельность и бесшовную интеграцию различных компонент таких систем. Типовая архитектура, реализующая такой стек технологий, должна включать несколько уровней.

- Интеграционный уровень:
 - провайдеры данных к БД *LMS*;
 - *API* к внешним источникам данных.
- Уровень управления данными:
 - хранилище метаданных;
 - модели МО для семантического анализа логов *LMS* и создания или пополнения онтологий;
 - шаблоны для построения семантических запросов.
- Уровень анализа данных и интеллектуальных сервисов:
 - онтологии курсов, учитывающие индивидуализацию;
 - когнитивные онтологические модели обучающегося;
 - правила порождения ИТО на основе онтологий.
- Уровень приложений и интерфейсов:
 - рекомендательные вопросно-ответные подсистемы для взаимодействия с обучающимся;
 - интерактивная визуализация ИТО.

3 Семантическое аннотирование образовательных материалов и данных результатов обучения

После того, как для подсистемы индивидуализации обеспечен доступ ко всем данным ЭО, необходимо выполнить семантическое аннотирование данных. Этот процесс относится к структурированным данным, например, выгрузкам из БД; полуструктурированным, например, логам системы; неструктурированным, например, текстовому контенту или другой текстовой информации, которая может появиться в системе (эссе и ответы обучающихся на тесты, обсуждения, диалоги и пр.). Для различных видов данных используются различные методы аннотирования. Во всех случаях в качестве результата необходимо получить определённое множество объектов, отражающих ход процесса обучения (пройденные элементы курса, достижения и качества обучающегося и т.п.), а также связи между этими объектами.

Семантические аннотации в общем случае можно определить как некоторые ссылки на метаданные, которые можно выразить через элементы онтологии. В процессе наполнения такой модели экземплярами реальных данных получается граф знаний. Для решения многих сложных задач, в том числе и для задачи индивидуализации ЭО, онтология должна представлять собой развитую концептуальную схему или референтную модель.

Структура референтной модели индивидуализации ЭО представлена на рисунке 2. Такая структура включает слои концептов различной степени абстракции:

- верхнеуровневые абстракции для моделирования ИТО обучающегося;
- общие концепты учебных материалов и образовательного процесса;
- специфические концепты для доступа и интеграции данных системы ЭО в терминах ПрО.

Верхние два уровня практически не зависят от специфики ПрО и конкретных LMS. В то же время нижний уровень может быть адаптирован под специфические требования или даже разбит на несколько подуровней при необходимости. В случае систем индивидуализированного ЭО подобные модификации нижнего уровня онтологии могут выполняться достаточно часто, т.к. сам процесс индивидуализации предполагает построение отдельной модели для каждого обучающегося на основе данных, которые им используются или порождаются в процессе обучения.

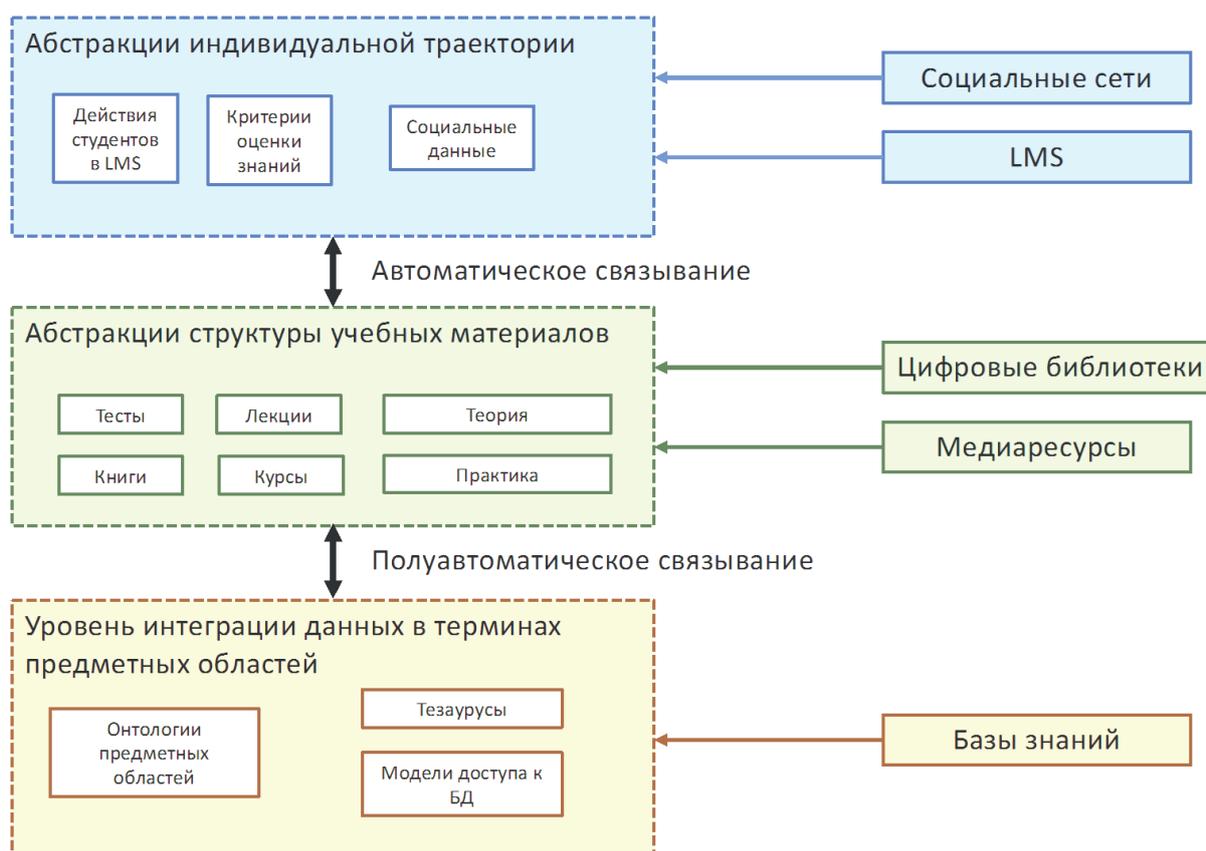


Рисунок 2 - Структура референтной модели индивидуализации электронного обучения

Связывание абстракций индивидуальной траектории и структуры учебных материалов – это часто выполняемая операция над хорошо структурированными данными, т.к. для каждого обучаемого она может выполняться многократно. Также при выполнении этой операции необходимо обеспечить высокий уровень объективности результатов. Эти факторы формируют требования по полной автоматизации операций связывания между этими уровнями,

исключая какое-либо влияние администратора *LMS* или эксперта на результат. Напротив, связывание с уровнем БЗ ПрО предполагает работу со слабоструктурированными и неструктурированными данными и выполняется однократно для каждого курса или его редакции. Точность построения предполагаемых связей будет выше при участии эксперта ПрО в данном процессе, соответственно полная автоматизация представляется нецелесообразной.

Модификация онтологий использует комплекс методов МО и относится к задачам *Information Extraction* [20].

- Распознавание/извлечение именованных сущностей (*Named Entity Recognition/Extraction*) — разграничение позиций упоминаний сущностей во входном тексте. Например, в предложении «Пьер Кюри открыл пьезоэлектричество.» подчёркнутый текст является упоминанием именованных сущностей.
- Связывание/снятие омонимии сущностей или семантическое аннотирование (*Entity Linking/Disambiguation, Semantic Annotation*) - ассоциирование упоминаний сущностей с подходящим и однозначным идентификатором в БЗ. Например, связывание «Пьер Кюри» с сущностью *Q37463* в БЗ *wikidata*.
- Извлечение терминов (*Term Extraction*) — извлечение основных фраз, которые обозначают концепты, релевантные к выбранной ПрО и описанные в корпусе, иногда включая иерархические отношения между концептами. Например, выявление в тексте про МО, что «нейронная сеть» или «к-средних» являются важными концептами в ПрО. Дополнительно можно определить, что оба концепта являются уточнением понятия «искусственный интеллект», а также, что они могут быть связаны с определённым подразделом БЗ.
- Извлечение ключевых слов/фраз (*Keyword/Keyphrase Extraction*) - извлечение основных фраз, которые позволяют категоризировать тематику текста (в отличие от извлечения терминов, задача извлечения ключевых фраз заключается в описании именно текста, а не ПрО). Ключевые фразы также могут быть связаны с БЗ.
- Тематическое моделирование/классификация (*Topic Modeling, Classification*) — кластеризация слов/фраз, которые часто встречаются совместно в сходном контексте. Эти кластеры затем ассоциируются с более абстрактными темами, с которыми связан текст.
- Маркирование/идентификация темы (*Topic Labeling/Identification*) — для кластеров слов, идентифицированных как абстрактные темы, извлечение одиночного термина или фразы, наилучшим образом характеризующей эти темы. Например, определение, что тема, состоящая из {“машинное обучение”, “выборка”, “точность классификации”, “градиентный спуск”} наилучшим образом характеризуется термином «машинное обучение» (которое может быть связано, например, с концептом *Q2539* в *wikidata*).
- Извлечение отношений (*Relation Extraction*) — извлечение потенциальных n-арных отношений из неструктурированных или полуструктурированных (таких как *HTML*-таблицы) источников. Например, из предложения «Пьер Кюри открыл пьезоэлектричество.» можно извлечь открыл (Пьер Кюри, пьезоэлектричество). Бинарные отношения могут быть интерпретированы как *RDF* тройки после связывания предикатов-отношений с соответствующими свойствами в БЗ (таким как *discoverer or inventor (P61)*).

4 Семантическая модель индивидуализированного обучения на основе графа знаний

Модель индивидуализированного обучения представляет собой граф знаний изучаемых дисциплин (ГЗД), дополненный определёнными связями между теми концептами, которые входят в множество полученных знаний обучающегося. По совокупности таких связей для различных обучающихся можно судить о том, насколько ГЗД сбалансирован, какие суще-

ствуют предпочтения и тренды при изучении учебных материалов, становится возможной гармонизация учебных материалов и прогнозирование наиболее релевантных вариантов при построении ИТО.

Граф приобретённых знаний обучающегося (ГЗО), формируется из множества концептов описания Про и является подмножеством общего графа знаний (ГЗ) всех курсов. Изначально этот ГЗ пустой, и в него помещается некое стартовое множество концептов, подтверждённых входным тестом обучающегося, либо полученным в ходе изучения вводного курса. В этом ГЗ пропущены часть концептов и связей, т.к. объём освоенных знаний по дисциплинам изначально неполный. Он содержит «пробелы в знаниях», которые необходимо выявить в процессе сопоставления с общим ГЗ. Для этого необходимо найти проекцию ГЗО на ГЗД, восстановить пропущенные узлы в ГЗО и, как следствие, добавить связи в ГЗД для фиксации части пройденной ИТО, как это показано на рисунке 3. Поскольку пропуски концептов в ГЗО носят случайный характер, решение задачи поиска подграфа на графе может иметь большой процент ошибок.

Для построения алгоритма решения указанной задачи удобно использовать векторное представление ГЗ. Идея векторного представления [21] основана на дистрибутивном представлении латентных свойств сущностей ГЗ. Латентные свойства для каждой сущности (e_i) задаются с помощью вектора $e_i \in R^{H_e}$, где H_e соответствует числу возможных латентных свойств в модели. Например, возможное объяснение того, что библиотека *NumPy* используется для научных расчётов — это возможность быстрого прототипирования сложных вычислений. В данном примере утверждения, что «*NumPy* является эффективной библиотекой для научных расчётов», а «язык *Python* позволяет быстро прототипировать сложные алгоритмы вычислений» можно промоделировать с помощью вектора-столбца, содержащего эвристические значения для двух латентных свойств, которые также могут быть получены в результате последующего обучения модели:

$$e_{NumPy} = \begin{pmatrix} 0.9 \\ 0.2 \end{pmatrix}, e_{PyFastPrototype} = \begin{pmatrix} 0.2 \\ 0.8 \end{pmatrix},$$

где вектор-строка (см. пояснения к рисунку 6) e_{i1} соответствует латентному свойству «эффективная библиотека для научных расчётов», а e_{i2} - латентному свойству «быстрое прототипирование сложных вычислений». Следует отметить, что в отличие от данного примера латентные свойства, выделенные в ходе обучения модели на реальных данных, как правило, сложно интерпретировать.

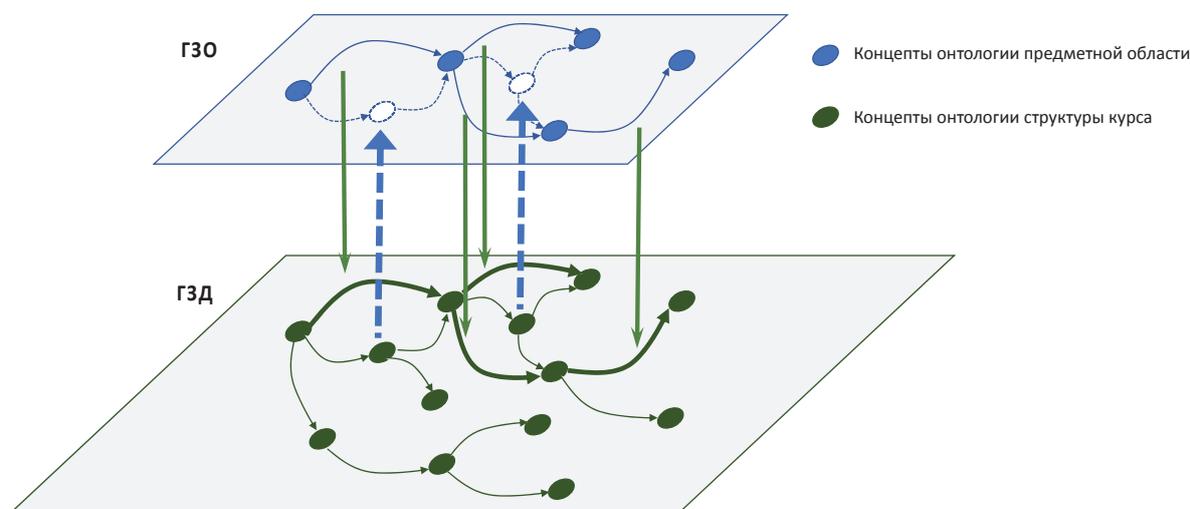


Рисунок 3 - Проекция графа приобретенных знаний обучающегося на граф знаний изучаемых дисциплин

5 Задача обнаружения пропущенных узлов в ГЗО

Для восстановления пропущенных узлов в ГЗО предлагается использовать комбинированный подход, основанный на совместном использовании векторных представлений триплетов из ГЗ и текстового корпуса, основанного на образовательном контенте. Подобные подходы доказали свою эффективность [22] в задаче дополнения ГЗ с использованием предварительно обученных языковых моделей для нейронной сети.

В рамках рассматриваемого подхода узлы и связи ГЗ рассматриваются как текстовые последовательности, состоящие из меток и текстовых описаний соответствующих троек. Например, триплет $\langle \text{PyTorch}, \text{written In}, \text{Python} \rangle$ может быть представлен как «библиотека *PyTorch* написана на языке *Python*». Эта последовательность слов (токенов) содержит три группы, по одной для каждого элемента в триplete $[Tok_{1\dots N_s}^S, Tok_{1\dots N_p}^P, Tok_{1\dots N_o}^O]$. В приведённом примере группам токенов соответствуют $Tok_{1\dots N_s}^S$ — «библиотека *PyTorch*», $Tok_{1\dots N_p}^P$ — «написана на» и $Tok_{1\dots N_o}^O$ — «языке *Python*». Совокупно все эти группы токенов подаются на вход нейронной сети. Для каждой группы рассчитываются отдельные векторные представления. При этом первая и третья группы (субъект и объект) используют общий сегмент векторного представления.

После обнаружения пропущенных концептов ГЗО необходимо установить, какие существуют отношения между этими концептами, что позволит определить последовательность их изучения и перечень необходимых курсов или модулей, которые содержат контент, позволяющий наиболее полно изучить выбранный перечень понятий. Данный аспект особенно важен, так как одни и те же понятия могут излагаться в различных курсах и в различном объёме. Кроме того, содержание курсов периодически обновляется. Эти факторы делают неэффективными любые статические маппинги (проекции) концептов-понятий на концепты-структурные элементы курсов. Более перспективным подходом является метод, основанный на использовании векторного представления для восстановления связей в ГЗ [23]. В различных курсах для связывания терминов может использоваться различный контекст и различное множество связей, а сами связи могут иметь различные области определения и области значений, т.е. одни и те же концепты-термины имеют наборы отличающихся связей.

6 Индивидуализированная оценка результатов ЭО

Тесты и практические задания являются слишком грубыми и случайными для достоверной оценки знаний после прохождения ИТО. А генерация тестов для каждой ИТО потребует большого количества ресурсов и времени преподавателей в условиях массового использования контента. В то же время значительный объём ценной информации о результативности обучения может быть получен за счёт анализа цифрового следа обучающегося путём исследования логов и других цифровых артефактов, которые формируются во время работы в системе ЭО. Похожий подход получил широкое распространение в программной инженерии в задаче автоматизации тестирования программного кода. Применительно к оценке знаний, получаемых в ходе ИТО, цифровые артефакты могут включать следующие виды данных:

- логи поведения пользователя в системе (количество посещений отдельных страниц, время, проведённое на каждой из страниц, действия на страницах и пр.);
- действия пользователя с контентом (динамика просмотра видео, последовательность выполнения заданий, завершённость начатых действий и т.п.);
- активность при взаимодействии с другими пользователями и преподавателем через социальные сервисы (количество вопросов, количество ответов, регулярность отправки сообщений и пр.);

- текстовые данные, которые пишет обучающийся при использовании общего чата или электронной почты;
 - данные модулей проверки знаний (закрытые и открытые тесты, результаты выполнения практических заданий и пр.).
- Основные методы анализа перечисленных данных включают:
- извлечение именованных сущностей и отношений из текстовых данных;
 - статистический анализ логов [24];
 - адаптированные методы юнит-тестирования [25].

7 Средства онтологического моделирования для системы индивидуализированного ЭО

Описанные выше подходы к формированию и динамическому изменению ИТО в системе ЭО, когда курсы изучаются не целиком, а отдельными материалами, могут привести к нежелательной ситуации хаотического перемешивания изучаемых тем, так как фактически авторский замысел преподавателя, который был заложен при создании курса, игнорируется системой. Для предотвращения такой ситуации в системе необходим алгоритм, непрерывно отслеживающий семантическое сходство изученного множества тем и содержания дисциплин. Фактически этот алгоритм выполняет аппроксимацию структуры изученных понятий онтологиями курсов, что позволяет сделать формирование ИТО сфокусированным на предмете изучения за счёт присвоения более высоких весов тем материалам, которые содержатся в курсах, наиболее полно излагающих выбранные для ИТО темы.

Алгоритм оценки семантического сходства *GADES (a Graph-bAseD Entity Similarity)* [25] учитывает три аспекта, связывающих сравниваемые объекты ГЗ: иерархия, соседство и специфичность.

Анализ иерархического сходства основан на выделении на ГЗ G набора иерархических рёбер, для которых применяются методы вычисления подобия. Иерархические рёбра включают те связи ГЗ, имена свойств которых принадлежат иерархическому отношению, например, *rdf:type* или *rdfs:subClassOf*. Каждое такое отношение определяет связанную сущность через операцию обобщения (таксономические отношения или абстракции более высокого порядка) другого объекта. *GADES* использует иерархические методы подобия, такие метрики измерения таксономического расстояния как d_{tax} и d_{ps} [27] для измерения иерархического сходства между двумя объектами. Обе меры основаны на вычислении наименьшего общего предка (*LCA от англ. lowest common ancestor*): узлы сравниваемых сущностей имеют общего предка, наиболее удалённого от корня дерева иерархии и лежащего на обоих путях от этих вершин до корня.

Вычисление близости соседства сравниваемых объектов. Окружение объекта $e \in E$ определяется как множество пар связь-сущность $N_e = \{(r, e_i) | (e, r, e_i) \in R\}$, сущности которых находятся на расстоянии одного шага от e . Это определение окружения позволяет рассматривать вместе сущность-соседа и тип отношения ребра графа. *GADES* использует знания, закодированные в иерархиях отношений и классов диаграммы знаний, для сравнения двух пар.

Специфичность сущности e в ГЗ G вычисляется как величина обратно пропорциональная числу её инцидентных рёбер $Incident(e) = \{(e_i, r, e) \in R\}$. *GADES* вычисляет специфичность наименьшего общего предка e_1 и e_2 . Суть метода в том, что объекты, общий предок которых содержит более общую информацию, менее похожи, чем сущности, общий предок которых содержит более конкретную информацию.

8 Моделирование когнитивного профиля обучаемого

В процессе изучения формируется ГЗО за счёт пополнения ссылками на концепты изученных ПрО. В совокупности множество этих ссылок формирует когнитивный профиль обучаемого. Для каждой ссылки в процессе обучения вычисляется определённый вес, характеризующий то, насколько хорошо была изучена соответствующая тема.

Онтология обучаемого содержит необходимые концепты и связи, позволяющие моделировать (рисунок 4):

- какие темы и понятия были изучены;
- оценка качества изучения;
- характеристики самого обучаемого, получаемые в процессе анализа его действий.

Важным свойством графовых данных является возможность возникновения различных корреляций между множеством взаимосвязанных узлов. Подобные корреляции могут быть вычислены за счёт включения обработки атрибутов, связей и классов связанных сущностей в алгоритм МО. Для моделирования бинарных отношений на графе удобно использовать трёхсторонний тензор \mathbf{Y} , в котором две моды образованы идентично на основе конкатенированных сущностей объектов-узлов, а третья мода содержит отношения между ними [28]. Подобный подход получил название тензорная факторизация.

На рисунке 5 приведена иллюстрация процесса моделирования данным методом. Элемент тензора $y_{ijk} = 1$ обозначает факт, что существует отношение (*i-th entity, k-th predicate, j-th entity*). В противном случае, для несуществующих или неизвестных отношений элемент приравнивается нулю. Каждая из возможных реализаций такого тензора может быть интерпретирована как один из возможных миров. Для получения модели всего ГЗ необходимо оценить совместное распределение $P(\mathbf{Y})$ на множестве $D \in E \times R \times E$ для наблюдаемых троек. Таким образом, строится оценка вероятностного распределения над возможными мирами, которые позволяют предсказать вероятность наличия троек, основываясь на состоянии всего ГЗ.

Сущности ГЗ могут быть эффективно представлены векторами их латентных свойств. Данные свойства называют латентными, т.к. они напрямую не описаны в данных, но могут быть выведены из имеющихся данных в процессе МО. В работе [29] предложена модель графовых латентных свойств *RESICAL*, представляющая тройки посредством парного взаимодействия этих латентных свойств. Вычисление вероятности существования какой-либо тройки в ГЗ осуществляется с помощью специальной оценочной функции. Тензорное представление графа позволяет эффективным образом вычислять подобные оценки через факторизацию срезов тензора $\mathbf{F}_k = \mathbf{E} \mathbf{W}_k \mathbf{E}^T$, где $\mathbf{F}_k \in \mathbb{R}^{N_e \times N_e}$ является матрицей, содержащей все оценки для *k*-й связи (отношения) и *i*-го ряда в матрице $\mathbf{E} \in \mathbb{R}^{N_e \times H_e}$. $\mathbf{W}_k \in \mathbb{R}^{H_e \times H_e}$ является матрицей весов, элементы которой w_{abk} показывают, насколько латентные свойства *a* и *b* взаимосвязаны в *k*-том отношении. Рисунок 6 иллюстрирует описанный принцип вычисления.

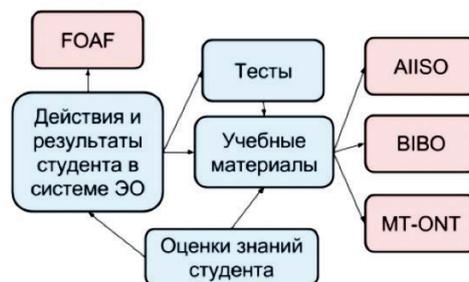


Рисунок 4 - Комплексная модульная онтология в системе ЭО

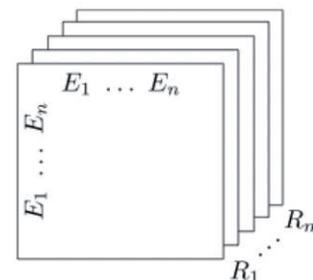


Рисунок 5 - Моделирование отношений с помощью трёхстороннего тензора [27]

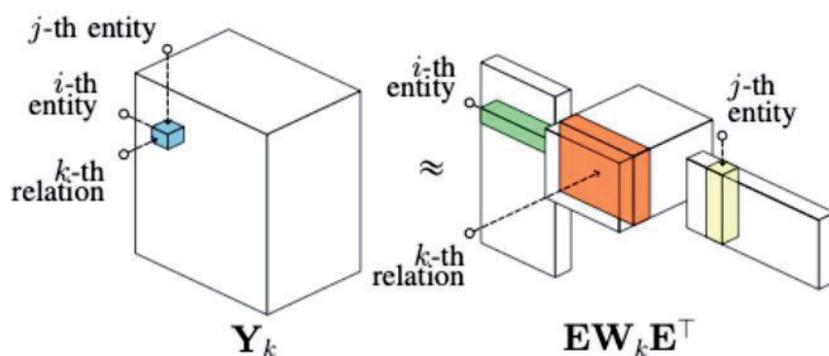


Рисунок 6 - Тензорное представление графа знаний RESCAL [21]

Выводы

Индивидуализация в ЭО является логичным и необходимым этапом эволюции ЭО, которое должно перейти от массовости к персонализации процессов обучения. Этот переход порождает множество методологических, технологических и концептуальных вопросов. Онтологический подход, принятый уже как стандарт де-факто для представления моделей слабоструктурированных данных, отвечает на часть из этих вопросов. Однако вопросы, касающиеся неявных или не декларированных знаний в системах ЭО, не могут быть разрешены непосредственно с помощью онтологий. В статье приведена попытка систематизации подобных вопросов и предложены подходы к их разрешению с помощью аппарата векторных представлений. Проведённый анализ может помочь при создании систем ЭО нового поколения, а также в решении задач обработки и анализа образовательных данных.

Список источников

- [1] *Харабет, Я.К.* Автоматическое выделение количественных конструкций в русскоязычных научно-популярных текстах / Я.К. Харабет // XVIII Объединённая научная конференция «Интернет и современное общество» (IMS-2015). – Санкт-Петербург. – 2015. – С.23-25.
- [2] *Баранова, Ю.Ю.* Индивидуализация обучения: возможности и ресурсы в аспекте введения федеральных государственных образовательных стандартов общего образования / Ю.Ю. Баранова // Научное обеспечение системы повышения квалификации кадров. – 2012. – №. 1. – С.123-129.
- [3] *Anderson, J.Q.* Individualisation of higher education: How technological evolution can revolutionise opportunities for teaching and learning / J.Q. Anderson // *International social science journal*. – 2013. Vol. 64. No. 213-214. – P.305-316.
- [4] *Байдикова, Н.Л.* Индивидуализация обучения студентов магистратуры в условиях накопительно-балльной системы / Н.Л. Байдикова // *Международный научно-исследовательский журнал*. – 2016. – № 11(53) Часть 3. – С.9-12.
- [5] *Иванова, Л.А.* Медиаобразовательное пространство как средство обеспечения индивидуальных учебных траекторий студентов технического вуза / Л.А. Иванова, И.С. Петухова // *Magister Dixit*. – 2011. – №. 4. – С.151-165.
- [6] *Genov, N.* Introduction to “Challenges of individualisation” / N. Genov // *International Social Science Journal*. – 2013. – Vol. 64. No. 213-214. – P.193-196.
- [7] *Huang, C.L.* Generating New Paths for Teacher Professional Development (TPD) through MOOCs / C.L. Huang // *Jiao Yu Yan Jiu Yu Fa Zhan Gi Kan*. – 2018. – Vol. 14. No. 1. – P.35-71.
- [8] *Resource Description Framework (RDF)*. - <https://www.w3.org/RDF/>.
- [9] *RDF Schema 1.1*. - <https://www.w3.org/TR/rdf-schema/>.
- [10] *Deibe, M.A.* Query Processing Over Graph-structured Data on the Web. – IOS Press, 2018. – Vol. 37.
- [11] *Web Ontology Language (OWL)*. - <https://www.w3.org/OWL/>.
- [12] *Academic Institution Internal Structure Ontology*. - <http://vocab.org/aiiso/>.
- [13] *Bibliographic Ontology Specification*. - <http://bibliontology.com>.
- [14] *The Ontology for Media Resources*. - <https://www.w3.org/TR/mediaont-10/>.

- [15] **Kauppinen, T.** Teaching core vocabulary specification / T. Kauppinen, J. Trame, A. Westermann // *LinkedScience.org*, Tech. Rep. — 2012.
- [16] **FOAF** Vocabulary Specification. - <http://xmlns.com/foaf/spec/>.
- [17] **Metacademy**. - <https://metacademy.org>.
- [18] **SlideWiki**: elicitation and sharing of corporate knowledge using presentations / Ali Khalili, S. Auer, D. Tarasowa, I. Ermilov // *Knowledge Engineering and Knowledge Management*. — Springer, 2012. — P.302–316.
- [19] **Crowd-sourcing** (semantically) Structured Multilingual Educational Content (CoS- MEC) / D. Tarasowa, S. Auer, A. Khalili, J. Unbehauen // *Open Praxis*. — 2014. — Vol. 6, No.2. — P.159–170.
- [20] **Martinez-Rodriguez, J.L.** Information extraction meets the semantic web: a survey / J.L. Martinez-Rodriguez, A. Hogan, I. Lopez-Arevalo // *Semantic Web*. Preprint – 2018. – P.1-81.
- [21] **Nickel, M.** et al. A review of relational machine learning for knowledge graphs / M. Nickel et al. // *Proceedings of the IEEE*. – 2015. – Vol.104, No. 1. - P.11-33.
- [22] **Liu, W.** et al. K-BERT: Enabling Language Representation with Knowledge Graph // *arXiv preprint arXiv:1909.07606*. – 2019.
- [23] **Lin, Y.** et al. Learning entity and relation embeddings for knowledge graph completion // *Twenty-ninth AAAI conference on artificial intelligence*. – 2015.
- [24] **Alspaugh, S.** et al. Analyzing log analysis: An empirical study of user log mining // *28th Large Installation System Administration Conference (LISA14)*. – 2014. – P.62-77.
- [25] **Peláez, C.** Unit testing as a teaching tool in higher education // *SHS Web of Conferences*. – EDP Sciences, 2016. – Vol.26. – P.01107.
- [26] **Traverso, I.** et al. GADES: a graph-based semantic similarity measure // *Proceedings of the 12th International Conference on Semantic Systems*. – ACM, 2016. – P.101-104.
- [27] **Paul, C.** et al. Efficient graph-based document similarity // *European Semantic Web Conference*. – Springer, Cham, 2016. – P.334-349.
- [28] **Nickel, M.** A Three-Way Model for Collective Learning on Multi-Relational Data / M. Nickel, V. Tresp, H.P. Kriegel // *ICML*. – 2011. Vol.11. – P.809-816.
- [29] **Nickel, M.** Tensor factorization for multi-relational learning / M. Nickel, V. Tresp // *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. – Springer, Berlin, Heidelberg, 2013. – P.617-621.

Сведения об авторе



Муромцев Дмитрий Ильич, получил степень бакалавра (1997) и магистра (1999) в области проектирования компьютерных систем в Санкт-Петербургском государственном политехническом университете. Получил степень кандидата технических наук в области компьютерных наук в Университете ИТМО в 2003 г. В настоящее время является зав. кафедрой информатики и прикладной математики Университета ИТМО. Дмитрий Муромцев является членом комитетов по техническим программам и редколлегий ряда международных конференций и журналов. Научные интересы: семантические технологии, Интернет вещей, онтологический инжиниринг, представление знаний и искусственный интеллект. Педагогическая деятельность началась в 2001 году на кафедре компьютерных технологий и управления Университета ИТМО.

Дмитрий Муромцев является автором и соавтором более 100 научных и учебно-методических публикаций и 4 книг. AuthorID (РИНЦ): 17726; Author ID (Scopus): 55575780100; ORCID 0000-0002-0644-9242; Researcher ID (WoS): N-6485-2016. *d.muromtsev@gmail.com*.

Поступила в редакцию 02.12.2019, после рецензирования 02.03.2020. Принята к публикации 15.03.2020.

Models and methods of e-learning individualization in the context of ontological approach

D. Mouromtsev

*St. Petersburg National Research University of Information Technologies, Mechanics and Optics,
St. Petersburg, Russia*

Abstract

The article explores the issues of e-learning individualization (EE) as a set of processes for creating, developing, using and utilizing digital content and EE data, describes ontological models and methods for individualizing digital education. The technological stack for building and implementing individual learning paths is considered, as well as examples of existing systems that fully or partially use the specified technology stack. To efficiently process educational materials and data generated by learning management systems, a three-level architecture is proposed that allows semantic annotation and selection of concepts of varying degrees of abstraction in the data layers. These layers include: high-level abstractions of modeling, general concepts of educational materials and the educational process, specific concepts for access and integration of data from the EE system in terms of the subject area. For the first time, it was proposed to use semantic models as the formal basis for an individualized EE, including the vector representations of knowledge graphs, which, on the one hand, allows efficient processing of large and complex data structures, and on the other hand, has the flexibility and expressiveness of the ontological approach. The main aspects related to individualization in EE systems were sequentially considered, including technological aspects and existing ontologies for e-learning, individual trajectory modeling, semantic annotation of educational materials, methods for assessing knowledge in individualized learning, as well as ontological simulation of the cognitive profile of the student.

Key words: *e-learning, individualization, ontologies, knowledge graphs, vector representation, semantic technologies.*

Citation: *Mouromtsev D. Models and methods of e-learning individualization in the context of ontological approach [In Russian]. *Ontology of designing*. 2020; 10(1): 34-49. DOI: 10.18287/2223-9537-2020-10-1-34-49.*

List of figures

- Figure 1 – Levels of the architecture of the individualized system for e-learning
- Figure 2 – Logical levels in the individualized system for e-learning
- Figure 3 – The projection of a domain knowledge graph to a graph of learnt concepts
- Figure 4 – The complex ontology in an e-learning system
- Figure 5 – Modeling of relations between concepts by means of tensors [27]
- Figure 6 – Tensor factorization of a knowledge graph RESCAL [21]

References

- [1] **Kharabet YaK.** Automatic allocation of quantitative constructions in Russian-language popular scientific texts. *XVIII joint scientific conference "Internet and modern society"(IMS-2015)*. Saint-Petersburg. 2015. P.23-25.
- [2] **Baranova JJ.** Individualization of training: opportunities and resources in the aspect of introducing federal state educational standards of general education. *Scientific support of a system for advanced training of personnel*. 2012; 1: 123-129.
- [3] **Anderson JQ.** Individualisation of higher education: How technological evolution can revolutionise opportunities for teaching and learning. *International social science journal*. 2013; 64(213-214): 305-316.
- [4] **Baydikova NL.** Individualization of teaching graduate students in a cumulative-point system // *International Research Journal*. 2016; 11(53) Part 3: 9-12.
- [5] **Ivanova LA, Petukhova IS.** Media-educational space as a means of providing individual educational trajectories for students of a technical university. *Magister Dixit*. 2011; 4: 151-165.
- [6] **Genov N.** Introduction to “Challenges of individualisation”. *International Social Science Journal*. 2013; 64(213-214): 193-196.
- [7] **Huang CL.** Generating New Paths for Teacher Professional Development (TPD) through MOOCs. *Jiao Yu Yan Jiu Yu Fa Zhan Gi Kan*. 2018; 14(1): 35-71.

-
- [8] **Resource Description Framework (RDF)**. <https://www.w3.org/RDF/>.
- [9] **RDF Schema 1.1**. <https://www.w3.org/TR/rdf-schema/>.
- [10] **Deibe MA**. Query Processing Over Graph-structured Data on the Web. IOS Press, 2018. Vol.37.
- [11] **Web Ontology Language (OWL)**. <https://www.w3.org/OWL/>.
- [12] **Academic Institution Internal Structure Ontology**. <http://vocab.org/aiiso/>.
- [13] **Bibliographic Ontology Specification**. <http://bibliontology.com>.
- [14] **The Ontology for Media Resources**. <https://www.w3.org/TR/mediaont-10/>.
- [15] **Kauppinen T, Trame J, Westermann A**. Teaching core vocabulary specification. LinkedScience. org, Tech. Rep. 2012.
- [16] **FOAF Vocabulary Specification**. <http://xmlns.com/foaf/spec/>.
- [17] **Metacademy**. <https://metacademy.org>.
- [18] **SlideWiki**: elicitation and sharing of corporate knowledge using presentations / Ali Khalili, Soeren Auer, Darya Tarasowa, Ivan Ermilov // *Knowledge Engineering and Knowledge Management*. Springer, 2012. P.302–316.
- [19] **Crowd-sourcing (semantically) Structured Multilingual Educational Content (CoS- MEC)** / Darya Tarasowa, Soeren Auer, Ali Khalili, Jourg Unbehauen // *Open Praxis*. 2014; 6(2): 159–170.
- [20] **Martínez-Rodríguez JL, Hogan A, Lopez-Arevalo I**. Information extraction meets the semantic web: a survey. *Semantic Web*. Preprint. 2018. 81 p.
- [21] **Nickel M. et al.** A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*. 2015; 104(1): 11-33.
- [22] **Liu W. et al.** K-BERT: Enabling Language Representation with Knowledge Graph. arXiv preprint arXiv:1909.07606. 2019.
- [23] **Lin Y. et al.** Learning entity and relation embeddings for knowledge graph completion. *Twenty-ninth AAAI conference on artificial intelligence*. 2015.
- [24] **Alspaugh S. et al.** Analyzing log analysis: An empirical study of user log mining. *28th Large Installation System Administration Conference (LISA14)*. 2014: 62-77.
- [25] **Peláez C**. Unit testing as a teaching tool in higher education. *SHS Web of Conferences*. EDP Sciences, 2016; 26: 01107.
- [26] **Traverso I. et al.** GADES: a graph-based semantic similarity measure. *Proceedings of the 12th International Conference on Semantic Systems*. ACM, 2016: 101-104.
- [27] **Paul C. et al.** Efficient graph-based document similarity. *European Semantic Web Conference*. Springer, Cham, 2016: 334-349.
- [28] **Nickel M, Tresp V, Kriegel HP**. A Three-Way Model for Collective Learning on Multi-Relational Data. *ICML*. 2011; 11: 809-816.
- [29] **Nickel M, Tresp V**. Tensor factorization for multi-relational learning. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, Berlin, Heidelberg, 2013: 617-621.
-

About the author

Dmitry Mouromtsev received a bachelor's degree (1997) and a master's degree (1999) in computer system design from St. Petersburg state Polytechnic University (Russia). He received the degree of candidate of technical Sciences in computer science in ITMO University in 2003, is currently the head of departments at the Department of Informatics and Applied mathematics of ITMO University. Dmitry Mouromtsev is a member of the technical program committees and editorial boards of a number of international conferences and journals. Research interests include semantic technology, the Internet of things, Ontological engineering, knowledge representation, and artificial intelligence. Pedagogical activity began in 2001 at the Department of Computer technology and management of ITMO University. Since then, he has taught more than 10 training courses on current topics in knowledge-based systems, computer science and more. Dmitry Mouromtsev is the author and co-author of more than 100 scientific and educational publications and 4 books. AuthorID (RCI): 17726; Author ID (Scopus): 55575780100; ORCID 0000-0002-0644-9242; Researcher ID (WoS): N-6485-2016. d.muromtsev@gmail.com.

Received December 8, 2020. Revised March 2, 2020. Accepted March 15, 2020.

Многоагентные системы полуавтоматического проектирования в машиностроении на базе механизма объект-функции

Г.Б. Евгеньев

Московский государственный технический университет им. Н.Э. Баумана, Москва, Россия

Аннотация

Предложена методология создания систем полуавтоматического проектирования изделий машиностроения с использованием технологий искусственного интеллекта. Методология основана на многоагентных методах создания баз знаний и пригодна для разработки систем проектирования и управления цифровых интеллектуальных производств. В качестве унифицированной модели агента предложена объект-функция с разбиением атрибутов на подмножество входных и выходных атрибутов метода агента. Объект-функция рассматривается как унифицированный модуль создания многоагентных систем. Предложены модели представления агента. Показано, что при таком представлении агент эквивалентен правилу-продукции, являющемуся модулем базы знаний. Наименования переменных модуля знаний должны выбираться из словаря, который может быть составлен с использованием различных естественных языков. Механизмы модулей знаний должны обеспечивать реализацию всех функций, которые могут потребоваться при формировании баз знаний. При использовании технологии экспертного программирования процесс формирования модулей знаний, их трансляции с получением на одном из традиционных языков объектных или исполняемых модулей и тестирования производится как одна операция. После получения необходимого набора модулей производится генерация метода, использующего подмножество модулей знаний. Показано, что кооперативное решение комплексных задач обеспечивается системой агентов.

Ключевые слова: проектирование, изделия машиностроения, интеллектуальные системы, многоагентные системы.

Цитирование: Многоагентные системы полуавтоматического проектирования в машиностроении на базе механизма объект-функции / Г.Б. Евгеньев // Онтология проектирования. – 2020. – Т.10, №1(35). – С.50-62. – DOI: 10.18287/2223-9537-2020-10-1-50-62.

Введение

В настоящее время многоагентные системы (МАС) и технологии (МАС-технологии) рассматриваются как наиболее перспективные информационные технологии ИТ-индустрии [1]. Однако внедрение МАС в практику осуществляется недостаточными темпами [2]. Концепция МАС предложена в середине 1980-годов и была высоко оценена как научными, так и индустриальными кругами. С самого начала она позиционировалась как парадигма создания сложных систем, построенная на биологических принципах, которая предлагает строить системы и решать задачи в том же стиле, в каком они решаются в живой природе и человеческом сообществе [3-5]. Основной принцип создания концептуальной модели МАС-приложений использует разбиение сложной задачи с множеством взаимодействующих сущностей на относительно простые законченные подзадачи, понятные специалисту. Решение этих задач поручается программным агентам, которые разрабатываются и программируются практически автономно, работают асинхронно и параллельно и взаимодействуют с помощью простой техники обмена сообщениями на языке, близком к естественному.

Одной из самых привлекательных сторон МАС-парадигмы является её способность естественно и эффективно решать трудную задачу разработки сложных программ, а именно программирование взаимодействий множества компонент программы. В концепции и техноло-

гии МАС, по существу, эта задача отделяется от программирования агентов и реализуется с помощью диалогов и протоколов их взаимодействия. Важно отметить, что в базовой формулировке парадигма МАС акцентирует внимание на этом факте: она формулируется как *парадигма вычислений на основе взаимодействий* (англ. *Computation as interactions*) [5-7]. Концепция обмена сообщениями с использованием диалогов и протоколов для реализации взаимодействий оказалась продуктивной на практике [8]. Вместе с тем, в работах [9, 10] обозначено отсутствие прогресса в широком промышленном применении МАС в настоящее время. «В то время как имеются убедительные свидетельства важности МАС и технологий как исследовательской области, остаётся неясным, какой практический эффект от неё имеется к настоящему времени» [10].

Агентом принято называть автономную компьютерную программу, которая способна к целенаправленному поведению в динамической, непредсказуемо изменяющейся внешней среде [2]. Автономность агента понимается как его способность функционировать в интересах достижения поставленной цели без вмешательства человека или других систем и при этом осуществлять самоконтроль над своими действиями и внутренним состоянием.

МАС определяется как *сеть слабо связанных решателей частных проблем* (агентов), которые существуют в общей среде и взаимодействуют между собой для достижения тех или иных целей системы. Взаимодействие может осуществляться агентами либо прямым образом – путём обмена сообщениями, либо *косвенным образом*, когда одни агенты воспринимают присутствие других агентов через изменения во внешней среде, с которой они взаимодействуют. МАС может содержать несколько однотипных или разнотипных агентов, которые могут иметь общие и/или различные цели, могут быть распределёнными по компьютерной сети, могут быть написаны на различных языках программирования и работать на различных операционных платформах [2].

Взаимодействие рассматривается в МАС как основной способ вычислений и координации поведения множества автономных программных или физических агентов. Агенты МАС могут взаимодействовать с целью *кооперативного решения* некоторой общей сложной задачи. В этом случае задача разбивается на более простые относительно автономные задачи, которые поручаются разным агентам. В таком варианте взаимодействие агентов имеет целью координацию локальных решений для достижения некоторого требуемого качества решения задачи в целом. Эта координация может достигаться либо в полностью распределённом варианте, либо с помощью управления с той или иной степенью централизации, реализуемого агентом, специально выделенным для этих целей. Качество решения исходной большой задачи обычно оценивается с помощью некоторой глобальной функции полезности, значение которой зависит от локальных решений агентов.

МАС, как одно из направлений в современных информационных технологиях, формирует *область исследований и разработок*, в которой принято различать три основных направления с общей теоретической основой, где агенты и МАС рассматриваются как:

- *метафора концептуального проектирования* (с акцентом на модульность);
- *источник технологий* (с акцентом на взаимодействие как принцип вычислений и принятия согласованных решений);
- *средство имитационного моделирования* (с акцентом на автономное поведение компонент, взаимодействующих на основе протоколов).

К настоящему времени предложены различные теории агентов и МАС, и каждая из них обладает своими выразительными возможностями, определяет сложность концептуальной модели агента и МАС.

1 Агент как объект-функция

Несмотря на более чем 20-летние усилия по разработке агентских технологий, они не достигли достаточной зрелости, например, зрелости объектно-ориентированного подхода (ООП) в программировании, который появился в конце 1960-х годов и в начале 90-х годов стал общепринятым [3].

Объект в программировании представляет собой некоторую сущность в виртуальном пространстве, обладающую определённым состоянием и поведением, имеющую заданные значения свойств (атрибутов) и операций над ними (методов). Значения атрибутов определяют состояние, а методы - поведение объекта.

Технический объект – это вещь, предмет искусственного (технического) происхождения. В процессе своего жизненного цикла объект первоначально возникает как виртуальная сущность, получаемая в процессе проектирования, а затем реализуется в результате применения тех или иных технологических процессов [11]. Т.е. объект соответствует требованию автономности, присущему агенту, и обладает методом, который определяет поведение объекта. Однако в ООП нет требования целесообразности такого поведения. В случае разработки технологии, при которой объект будет вести себя целенаправленно, он превратится в агента. Для решения этой задачи объект необходимо преобразовать в объект-функцию. Для этого атрибуты необходимо разбить на два подмножества: подмножество входных атрибутов метода и соответственно – выходных.

Целесообразность поведения – это свойство МАС, в которой агенты объединяются для решения комплексной проблемы, для чего должна быть разработана соответствующая технология, включающая обмен информацией между отдельными агентами. Для простоты желательно, чтобы агенты были бы однотипными. Следует отметить, что проблемы повышения уровня интеллектуальности интегрированных САПР применительно к проектированию лопаток осевых турбомашин изучались в работах Б.М. Аронова [12]. Применение описываемых в настоящей статье методов является дальнейшим развитием этого направления.

Концептуально искусственный агент представляет собой объект-функцию. Модель функционального блока, связанного с агентом, определена стандартом *IDEF0*. Согласно этой схеме атрибуты объекта разбиты на три категории: входные, выходные и управленческие. Метод объекта реализует механизм функционального блока.

Вместе с тем агент является объектом, в связи с чем его нотацию целесообразно максимально приблизить к принятому в ООП формату *UML (Unified Modeling Language)*. Простым решением, учитывающим все упомянутые факторы, представляется форма, приведённая на рисунке 1. Такая форма содержит все компоненты представления класса, предусмотренные в *UML*: имя, атрибуты, операции и описание. Имя заносится в верхний раздел и состоит из идентификатора, отделённого двоеточием от наименования. Наименование строится по правилам *IDEF0*. Оно содержит описание действия (на рисунке 1 – Расчёт) предмета, на который направлено действие (на рисунке 1 – числа), и дополнительной информации (на рисунке 1 – передаточного). Далее следует описание атрибутов. В отличие от представления классов в *UML* это описание разделено на две части: входные и выходные. Такое разделение диктуется тем, что этот объект является функцией.

Наименования атрибутов, как это принято в интеллектуальных системах, должны выбираться из словаря системы, также как и их имена - идентификаторы, необходимые для написания формул. Условие, как это требует *IDEF0*, представляет собой ограничения, накладываемые на входные и управляющие параметры и определяющие область определения функции, реализуемой данным модулем действий. Условия отделяются от наименования атрибута вертикальной чертой, как это принято в математике. На рисунке 1 переменная «Вид СЕ» (сборочная единица) должна иметь значение «Редуктор», чтобы данный модуль выполнил своё

действие. Это – управляющая переменная. Другим условием выполнения модуля является то, что «Частота вращения на выходе, об/мин» должна быть больше нуля. Тогда выходная переменная «Передаточное число» может быть рассчитана по формуле $Ur_z = nh / nt$.

Под описанием атрибутов, как это принято в *UML* для представления классов, заносится операция, выполняемая данным модулем действий. В данном случае – это расчёт по формуле. С точки зрения *IDEF0* - это описание механизма выполнения данного функционального блока.

В нижнем разделе, как это допускает *UML* при моделировании классов, может записываться дополнительная информация. Она часто включает автора, разработавшего данный модуль, и источник информации, который был использован при разработке.

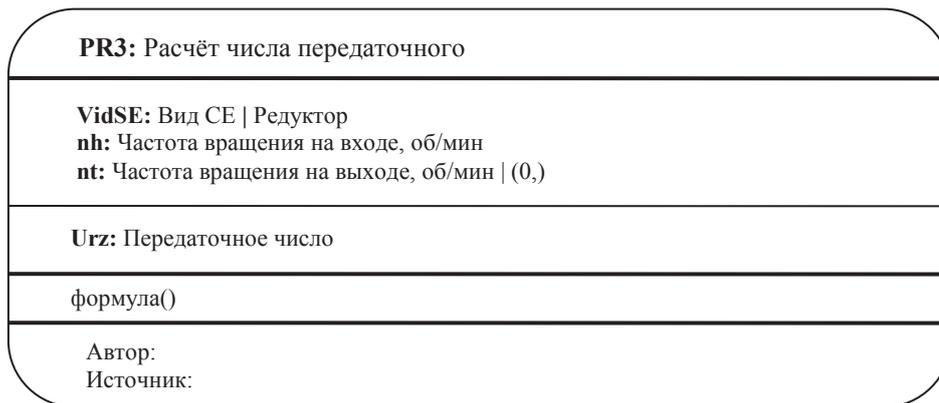


Рисунок 1 - Объект-функция в формате, приближенном к *UML*

При построении диаграмм состояний *UML* целесообразно использовать сокращённое представление, включающее только наименование.

Помимо схемного представления целесообразно иметь эквивалентное ему текстовое представление, которое автоматически генерируется документатором системы экспертного программирования [11]. В интеллектуальных системах проектирования подобные модули действий (с точки зрения *UML*) или функциональные блоки (с точки зрения *IDEF0*), или правила-продукции (с точки зрения искусственного интеллекта, ИИ) принято называть модулями знаний (МЗ).

Имеется возможность с помощью одного МЗ присваивать значения переменным и производить вычисления по набору взаимосвязанных формул. При этом предшествующие выходные переменные могут использоваться для определения последующих выходных переменных. С помощью МЗ типа формул можно формировать текстовые переменные, например, обозначения изделий, тексты содержания технологических операций и переходов и т.д. Пример формирования содержания перехода механической обработки в соответствии с Единой системой технологической документации (ЕСТД) приведён на рисунке 2.

При значениях входных переменных $Per="Точить"$, $ElObr="канавку"$, $NoEl=1$, $DinPer2="кольц."$, $DinPer4="окончательно"$ содержание перехода будет иметь такой вид: "Точить кольц. канавку 1 окончательно". Функция *STR* обеспечивает перевод данных из числовой формы в строковую.

Функциональные зависимости в инженерных документах часто имеют табличную форму представления. Для ввода таких зависимостей в базы знаний (БЗ) используются МЗ с механизмами в виде таблиц. Пример такого модуля для присвоения численных значений приведён на рисунке 3. Прилагаемая к модулю таблица может иметь шапку и боковик. В данном случае шапка содержит значения стандартных диаметров осей, а боковик — диапазоны исходных длин осей. На основе этих двух входов таблица позволяет определить значения вы-

ходной переменной — стандартной длины оси. Таблица может быть недоопределённой, т.е. содержать пустые клетки, как это имеет место в примере. При значениях входных переменных, соответствующих этим клеткам, модуль не даст решения. В таком случае проектировщик должен будет изменить входные данные, например исходную длину оси.

Модуль: ТКР3		
Разработчик: Евгений Г.Б.		
Наименование: Формирование содержания перехода		
Источник информации: ЕСТД		
Наименование	Имя	Ограничение
Переход обработки резанием	<i>Per</i>	[1,)
Элемент обрабатываемый	<i>ElObr</i>	
Номер элемента	<i>NoEl</i>	
Дополнит. информация перехода 2	<i>DinPer2</i>	
Дополнит. информация перехода 4	<i>DinPer4</i>	
Количество элементов	<i>KolEl</i>	
Номер элемента строковый	<i>NoElStr</i>	<i>STR(NoEl:0)</i>
Содержание перехода	<i>SodPer</i>	<i>Per+ " + DinPer2+ " + ElObr+ " + NoElStr+ "</i> <i>+ DinPer4</i>

Рисунок 2 - Пример формирования содержания перехода механической обработки в соответствии с ЕСТД

Разработчик: Евгений Г. Б.		
Наименование: Назначение стандартной длины		
Источник информации: Анурьев В.И. Справочник конструктора, т.2, стр.8		
Наименование	Имя	Ограничение
Диаметр оси стандартный, мм	<i>d</i>	(0, 22]
Длина оси исходная, мм	<i>Lo</i>	(25, 30]
Длина оси стандартная, мм	<i>L</i>	Таблица: TABL1

TABL1

Длина оси	Диаметр оси стандартный, мм							
исходная, мм	5	6	8	10	12	16, 18	20	22
(25, 28]	28	28	28	28	28	28		
(28, 30]	30	30	30	30	30	30	30	

Рисунок 3 - Пример модуля знаний, представляющего функциональную зависимость в табличной форме

Диаграммы компонентов - это один из видов диаграмм, применяемых при моделировании физических аспектов объектно-ориентированной системы. Они показывают организацию наборов компонентов и зависимости между ними. На рисунке 4 представлена диаграмма компонентов применительно к МЗ, приведённому на рисунке 1. Графически такая диаграмма представляется в виде графа с рёбрами и вершинами. Диаграммы компонентов обычно включают в себя: компоненты и интерфейсы, формирующие вершины графа, и отношения между ними, представляющие собой рёбра этого графа.

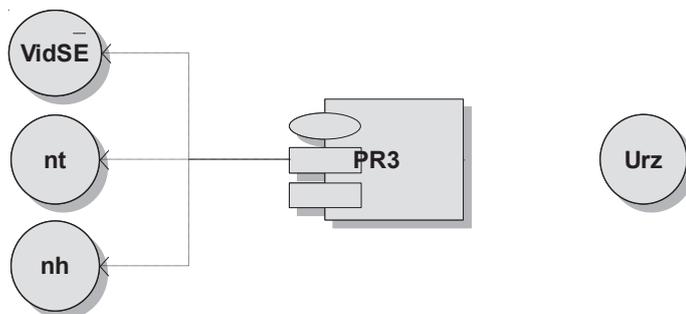


Рисунок 4 - Диаграмма компонентов UML применительно к модулю знаний расчёта числа передаточного

На рисунке 4 приведён компонент *PR3* и четыре интерфейса *VidSE*, *nt*, *nh* и *Urz*. Первые три интерфейса представляют собой управляющую (*VidSE*) и входные (*nt*, *nh*) переменные. Компонент *PR3* зависит от них, что отображается пунктирными стрелками. В свою очередь этот компонент реализует интерфейс (*Urz*), то есть обеспечивает вычисление этой переменной.

Неструктурированная совокупность МЗ в определённой прикладной области представляет собой БЗ этой области, аналогичную БЗ производственной системы.

Наименования и имена входных, управляющих и выходных переменных МЗ должны выбираться из словаря БЗ. Словарь может состояться с использованием различных естественных языков, что обеспечивает соответствие первому базовому требованию многоязычности систем класса *wiki*. Словарь в совокупности со значениями содержащихся в нём переменных выполняет в экспертном программировании функцию рабочей памяти производственной системы [11].

Механизмы МЗ должны обеспечивать реализацию всех функций, которые могут потребоваться при формировании БЗ. В число таких функций входят следующие основные:

- вычисление по формулам (в том числе присвоение значений переменным);
- определение значений по таблицам;
- выбор значений из баз данных;
- обновление значений в базах данных;
- занесение значений в базы данных;
- вычисление значений с использованием подпрограмм;
- вычисление значений с помощью методов, сгенерированных из модулей знаний;
- вычисление значений с помощью исполняемых *exe*-модулей или *dll*-библиотек, сгенерированных другими системами.

2 Механизм объект-функции как система агентов

МЗ, представляющие собой элементарные порождающие системы, объединяют в структурированные порождающие системы, носителем моделей которых они являются. Моделью структурированных порождающих систем с точки зрения ИИ являются семантические сети.

Модель структуры БЗ можно представлять как семантическую сеть взаимосвязанных модулей инженерных знаний. Поименованными узлами этой сети являются сами модули, а ориентированными рёбрами — входные и выходные переменные, наименования которых содержатся в словаре.

Семантическая сеть МЗ представляет собой ациклический ориентированный граф (рисунок 5). Ациклическость необходима для выполнения семантической сетью её функционального назначения — обеспечивать определение значений выходных переменных по заданным входным. Из-за наличия циклов в сети блокируется возможность нахождения значений всех или части искомым переменных.

В связи с ациклическостью всё множество МЗ может быть разбито на строго упорядоченное (ранжированное) множество подмножеств. Ранжирование производят, используя понятие протяжённость, которое существует для всех конечных связных графов.

Протяжённость $l(a, b)$ между вершинами a и b определяется как длина самой длинной связывающей их простой цепи.

Для определения протяжённости целесообразно построить фиктивный МЗ_00, не имеющих входных переменных. В число выходных переменных «нулевого» модуля включают входные переменные сети и выходные переменные модулей, не имеющих входов. Такие модули производят установку значений переменных по умолчанию; к их числу на рисунке 5

принадлежит МЗ_01. Всем МЗ без входов присваивают нулевой ранг. МЗ_00 рассматривают как вершину, от которой рассчитывают протяжённости для всех остальных МЗ, не имеющих нулевого ранга.

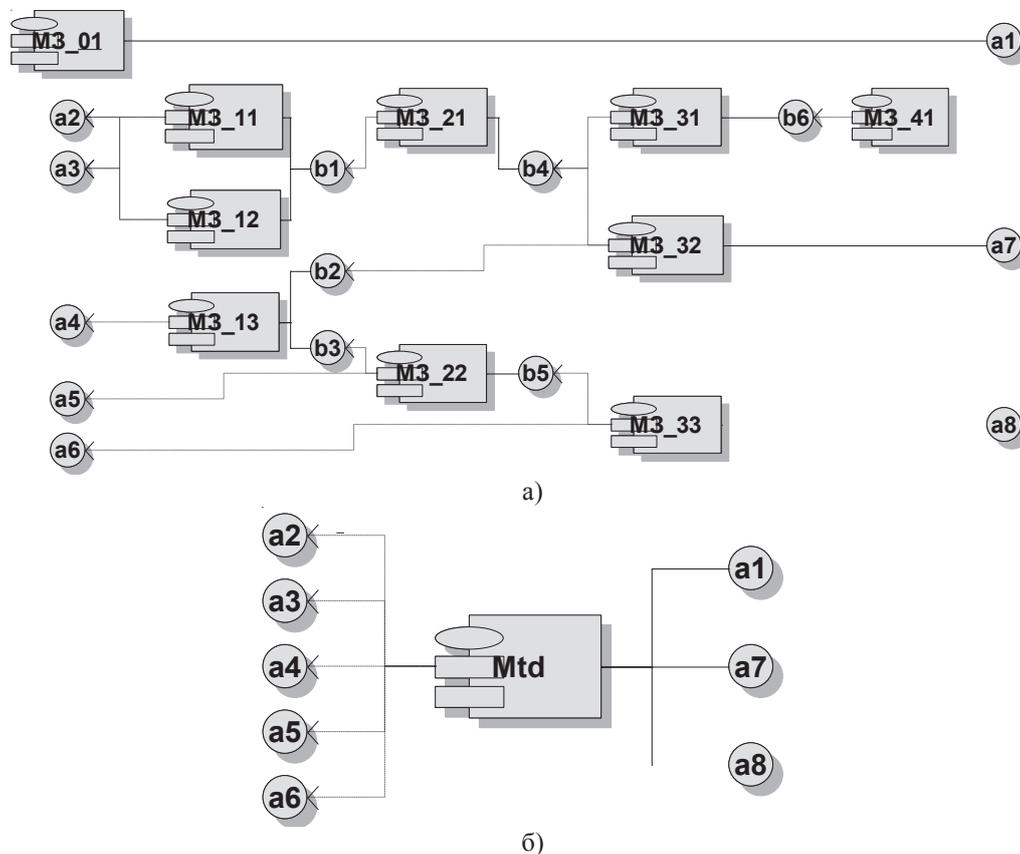


Рисунок 5 - Диаграмма компонентов (а) метода (б)

В результате ранжирования с помощью протяжённости подмножество МЗ ранга 1 составят модули, имеющие в качестве входных переменных только входные переменные метода и выходные МЗ ранга 0. Подмножество ранга 2 образуют МЗ, использующие в качестве входных входные переменные метода и выходные переменные модулей рангов 0 и 1, и т.д.

Ранжирование МЗ производится однократно при формировании сети. В дальнейшем ранги используют как метазнания для логического вывода на сети. Модули запускаются в порядке возрастания их рангов. МЗ нулевого ранга запускаются без проверки предусловий. Далее производится проверка предусловий модулей ранга 1 и запускаются механизмы тех МЗ, предусловия которых удовлетворены.

Альтернативные решения могут возникать у модулей, связанных в узел типа ИЛИ. Такой узел формируют МЗ, принадлежащие к множеству одного ранга и имеющие одинаковые наборы входных и выходных переменных. На рисунке 3 это — МЗ_11 и МЗ_12.

Возникновение альтернативных решений зависит от того, является узел ИЛИ исключающим или неисключающим, т.е. пересекаются или не пересекаются области определения этих модулей, описанные в предусловиях.

Вариант, представленный на рисунке 6а, соответствует исключающему ИЛИ, так как области определения не пересекаются, а вариант на рисунке 6б — неисключающему, поскольку при значениях переменных предусловий, принадлежащих к общей области, могут быть применены оба МЗ. Последний вариант рассматривается как некорректное построение БЗ.

Модули рангов 2 и 3 на рисунке 5 формируют два узла типа И. МЗ, образующие узел типа И, должны принадлежать к подмножеству одного ранга и не иметь общих выходных переменных. Подмножество ранга 1 на рисунке 5 может быть представлено в виде логического выражения $(M3_{11} \vee M3_{12}) \wedge M3_{13}$. Узлы типа И не порождают альтернативных решений. Все модули, входящие в такие узлы, должны быть запущены на исполнение, поскольку они определяют разные переменные.

Подмножества различных рангов связываются логической связкой И, так как для решения задачи они все должны быть выполнены.

Различают два основных типа стратегий управления: безвозвратный и пробный. В безвозвратном режиме управления выбирается применимое правило и используется необратимо без возможности пересмотра в дальнейшем. В экспертном программировании используется такой режим [11].

При использовании технологии экспертного программирования процесс формирования МЗ, их трансляции с получением на одном из традиционных языков объектных или исполняемых модулей и тестирования производятся как одна операция. После получения необходимого набора МЗ производится генерация метода, использующего подмножество сгенерированных МЗ. Метод генерируется с использованием того же языка, что и МЗ, и представляет собой скомпилированную реализацию решателя для данного набора МЗ. В скомпилированном механизме выбора продукционных правил используется схема управления, подобная сети Петри. Процесс планирования и разрешения конфликтов использует условия запуска МЗ. Программный модуль, соответствующий МЗ, исполняется, если имеются значения всех входных и управляющих переменных и выполнено условие его применения.

Сгенерированный метод может быть использован в качестве механизма МЗ, что позволяет использовать иерархию правил.

Чтобы генерируемые методы согласно теореме о структурировании могли решать любые задачи программирования, в них должны быть реализованы три базовые структуры: следование, альтернатива и один из циклов. В экспертном программировании следование МЗ определяется последовательностью вычисления переменных, а альтернатива – условиями, заложенными в МЗ. Для формирования циклов введена выделенная переменная «конец цикла» (*Fincalc*), при появлении которой система генерирует циклический метод, выполняющийся до тех пор, пока *Fincalc*, которому предварительно присвоен 0, не примет значения 1. Один метод может содержать только один цикл. Тело цикла включает модули рангов 1 и более. Модули ранга 0 обеспечивают начальные установки значений переменных и в цикле не исполняются. Возможность использования методов в качестве механизмов МЗ позволяет получать как последовательности циклов, так и программы с вложенными циклами.

На рисунке 5а представлена диаграмма компонентов метода, а на рисунке 5б метод изображён в качестве компонента. Общая архитектура программ, генерируемых при экспертном программировании, соответствует принятой в модульном программировании. При этом модули, изображённые на этом рисунке, соответствуют методам, составленным из МЗ. Методы в качестве механизмов МЗ могут входить в вышерасположенную по иерархии программу.

МЗ, представляющие собой элементарные порождающие системы, объединяют в методы объектов. Из числа средств *UML* для представления методов, построенных на базе объектных функций, наиболее подходят диаграммы состояний.

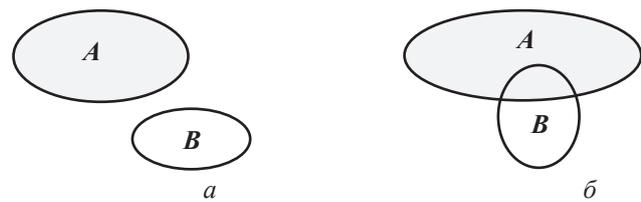


Рисунок 6 - Области определения исключающих (а) и неисключающих (б) ИЛИ

3 Кооперативное решение сложных задач системой агентов

Парадигма МАС, реализуемая на основе взаимодействий, рассмотрена на примере проектирования червячных редукторов.

Имеются два основных способа использования технологий ИИ при автоматизации проектирования: встраивание и надстраивание. Первый подход используется в ряде САД-систем [13]. Недостатком такого подхода является то, что инженеры, не занимающиеся конструированием, например технологи, не могут использовать интеллектуальные возможности таких систем. В статье предлагается использовать подход надстраивания, который позволяет применять методы ИИ на всех этапах жизненного цикла изделий.

МАС обладают статическими и динамическими свойствами. В UML для представления статического и динамического вида системы используются различные диаграммы. Основной статической диаграммой UML является диаграмма классов. Представление классов объектов в ней не обладает функциональностью, необходимой для описания МАС. Во-первых, не предусмотрено деление свойств на входные и выходные, что необходимо для представления объект-функций. Во-вторых, классы объектов не имеют предусловий, необходимых для управления процессом поведения системы. Эти проблемы можно решить, не выходя за рамки средств, принятых для представления диаграмм классов.

Например, в диаграмме классов червячного редуктора коренным классом является классификатор Единой системы конструкторской документации (ЕСКД). Перед началом проектирования целесообразно провести морфологический анализ проблемы. В цифровом машиностроении для проведения этого анализа следует использовать классификатор ЕСКД (рисунок 7). В МАС морфологический анализ производится в форме диалога с инженером. Отвечая на вопросы системы, специалист формирует классификационный код проектируемого изделия, по которому система переходит к БЗ многоагентного синтеза этого изделия. Таким образом, проектирование осуществляется последовательным использованием пары методов «Морфологический анализ» - «Многоагентный синтез».

В результате морфологического анализа сформирован код вида изделия, необходимо произвести многоагентный синтез червячного редуктора, для чего нужно задать параметры технического задания, отмеченные в качестве атрибутов этого изделия: мощность и частота вращения на выходном валу, модель электродвигателя и т.д. В результате может быть задействован многоагентный метод проектирования редуктора.

Редуктор червячный состоит из трёх основных узлов: колеса червячного, червяка и корпуса изделия. В состав узла колеса червячного входят: колесо червячное, вал колеса и подшипник колеса передний. В свою очередь имеются два варианта исполнения колеса: цельно-металлическое и сборное. В случае сборного колеса оно включает детали обода и ступицы. Все перечисленные объекты имеют соответствующие интеллектуальные методы конструирования с получением 3D моделей и чертёжной документации.

Аналогичным образом формируется МАС проектирования других узлов изделия.

Чтобы обеспечить возможность генерации 3D моделей с помощью БЗ необходимо создать МЗ, имеющий механизм, с помощью которого на основе параметризованной в САД-системе модели детали и (или) сборочной единицы можно было бы сгенерировать 3D модель изделия с рассчитанными в других модулях величинами размеров [14].

В качестве САД-системы в примере генерации 3D модели червячного колеса использована программа *Solid Works*. Диаграмма протокола работы МАС включает: выбор проектируемого узла, проектирование узла колеса червячного, проектирование колеса червячного, генерация 3D моделей колеса червячного (рисунок 8).

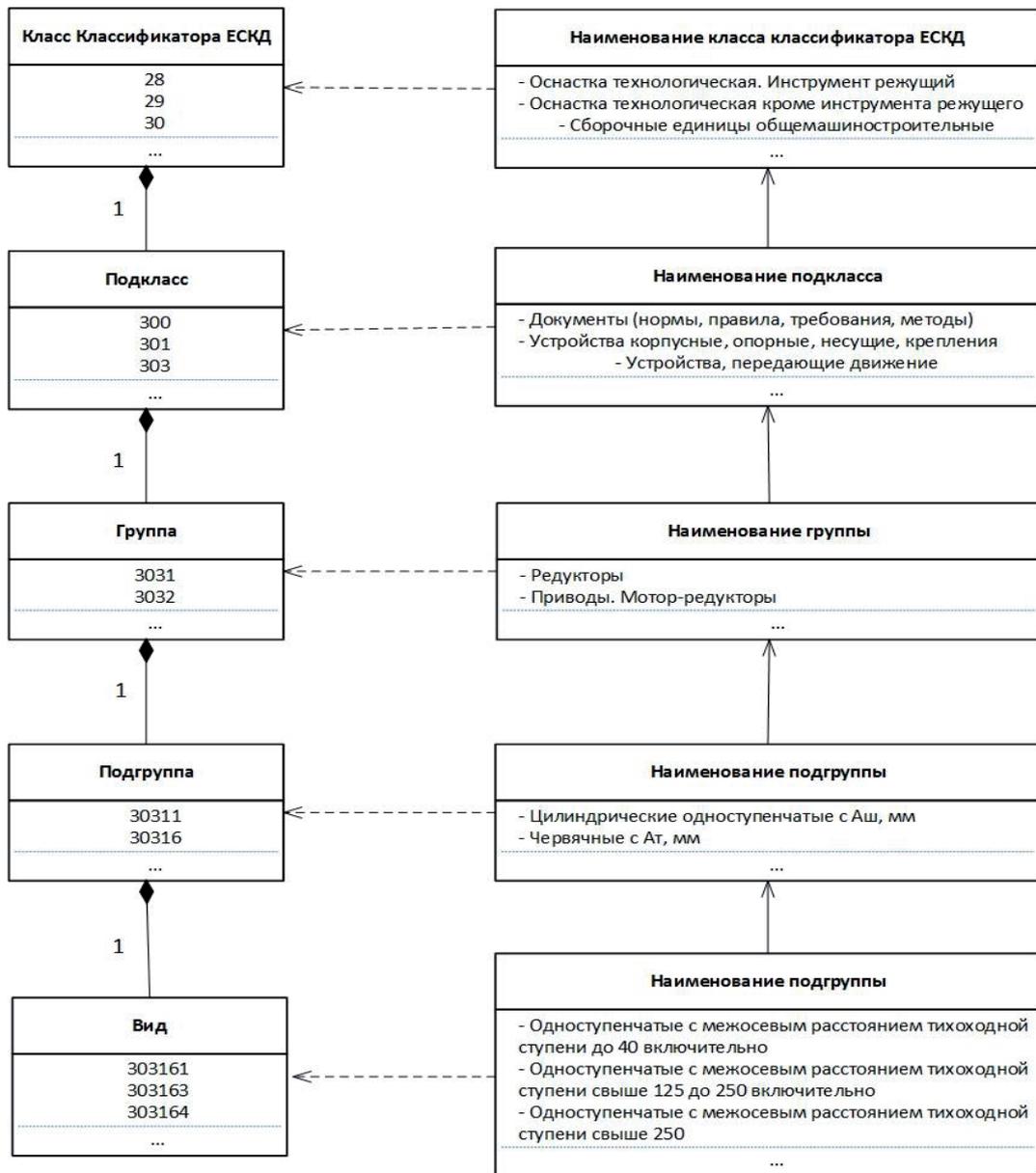


Рисунок 7 - Морфологический анализ проектируемого изделия

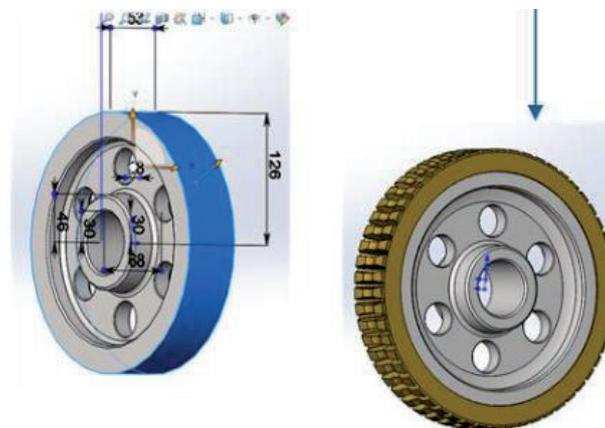


Рисунок 8 - Генерация 3D моделей колеса червячного

После построения 3D моделей всех узлов имеется возможность генерации модели изделия в целом (рисунок 9).

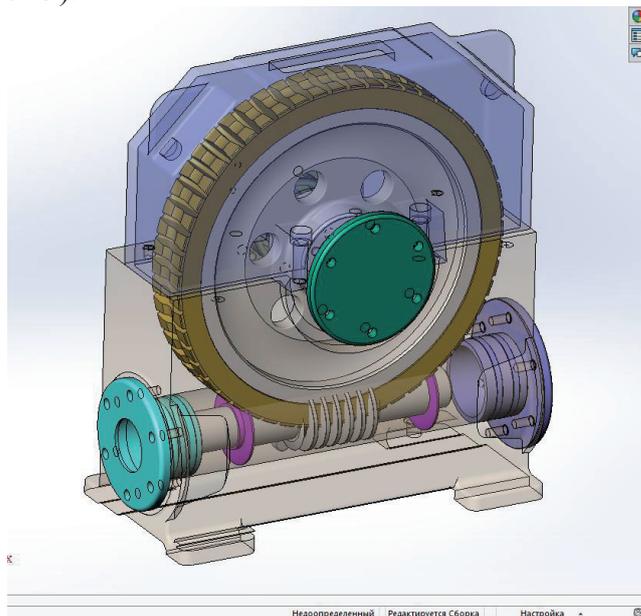


Рисунок 9 - Генерация 3D модели изделия

Заключение

В качестве унифицированной модели агента предложена объект-функция, которая рассматривается как унифицированный модуль создания МАС. При использовании технологии экспертного программирования процесс формирования МЗ, их трансляции с получением на одном из традиционных языков объектных или исполняемых модулей и тестирования производятся как одна операция. После получения необходимого набора модулей производится автоматическая генерация метода, использующего подмножество сгенерированных МЗ.

Предложенная методология позволяет создавать МАС непрограммирующим специалистам. В соответствии с этой методологией разработку таких систем осуществляют магистранты МГТУ им. Н.Э. Баумана.

Список источников

- [1] Gartner. Top Strategic Predictions for 2016 and Beyond: The Future Is a Digital Thing. <https://www.gartner.com>.
- [2] **Городецкий, В.И.** Современное состояние и перспективы промышленных применений многоагентных систем / В.И. Городецкий, О.Л. Бухвалов, П.О. Скобелев, И.В. Майоров // Управление большими системами: сборник трудов, выпуск 66. 2017. - С.94-157 .
- [3] **Alanis, A.Y.** Bio-inspired Algorithms for Engineering / Alma Y. Alanis, Nancy Arana-Daniel and Carlos López-Franco. Elsevier Inc. 2018. - 152 p.
- [4] **Binitha, S.** A Survey of Bio inspired Optimization Algorithms / Binitha S, S Siva Sathya // International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2, May 2012. P.137-151.
- [5] **M. Blowers and A. Sisti.** *Evolutionary and Bio-inspired Computation: Theory and Applications*, Proceedings of Spie (Book 6563), Society of Photo Optical (April 25, 2007). – 174 p.
- [6] **Avi Wigderson.** Mathematics and Computation: A Theory Revolutionizing Technology and Science PRINCETON 2019 – 440 p.
- [7] **Makhatadze, G.I.** Linking computation and experiments to study the role of charge-charge interactions in protein folding and stability. IOP Publishing Phys Biol. 2017 Feb 7. - DOI: 10.1088/1478-3975/14/1/013002.
- [8] **Luck M. et al.** Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing) // AgentLink. – 2005. – <http://www.agentlink.org/roadmap/>.

- [9] *Deloach, S.A.* Moving multi-agent systems from research to practice // Int. J. Agent-Oriented Software Engineering. – 2009. Vol.3, No.4. – P.378–382.
- [10] *Müller J., Fisher K.* Application Impact of Multiagent Systems and Technologies: A Survey // In «Agent-Oriented Software Engineering» book series. – Springer, 2013. – P. 1–26.
- [11] *Евгеньев, Г.Б.* Основы автоматизации технологических процессов и производств. Т.1: Информационные модели. Т.2: Методы проектирования и управления - Москва, Изд-во МГТУ им. Н.Э. Баумана, 2015. - Т.1: - 441 с. - Т.2: - 479 с.
- [12] *Аронов, Б.М.* О путях повышения уровня интеллектуальности интегрированных САПР лопаток осевых турбомашин. Проблемы и перспективы развития двигателестроения / Самар. гос. аэрокосм. ун-т; [Гл. ред. Е. В. Шахматов]. – Самара: СГАУ, 1999. – Вып.3, ч. 2. – С.245-248.
- [13] *Thilmany J.* Putting Artificial Intelligence to Work in CAD Design – Engineers Rule. December 21, 2017. - <https://www.engineersrule.com/solidworks-puts-artificial-intelligence>.
- [14] СПРУТ-Технология. Автоматизация проектирования. - <https://sprut.ru/>.

Сведения об авторе



Евгеньев Георгий Борисович, 1938 г. рождения. Окончил Московский авиационный технологический институт в 1960 г., к.т.н. (1964), д.т.н. (1975). Лауреат премии Совета Министров СССР (1991). Профессор кафедры «Компьютерные системы автоматизации производства» МГТУ им. Н.Э. Баумана. Член Российской ассоциации искусственного интеллекта. В списке научных трудов более 150 работ в области САПР и ИИ. AuthorID (РИНЦ): 17685. g.evgenyev@mail.ru.

Поступила в редакцию 30.01.2020, после рецензирования 17.02.2020. Принята к публикации 15.03.2020.

Multi-agent systems of semi-automatic design based on object-functions model in engineering

G.B. Evgenyev

Moscow State Technical University named after N.E. Bauman, Moscow, Russia

Abstract

A methodology for creating semi-automatic design systems for engineering products using artificial intelligence technologies is proposed. The methodology is based on multi-agent methods for creating knowledge bases and is suitable for the development of design and control systems for digital intelligent industries. As a unified agent model, an object-function mechanism is proposed with a partition of attributes into a subset of the input and output attributes of the agent method. Object-function is considered as a unified module for creating multi-agent systems. Graphic and textual models of agent representation are proposed. It is shown that with this representation, the agent is equivalent to the production rule, which is a knowledge base module. The names of the variables of the knowledge module should be selected from a dictionary that can be compiled using various natural languages. The mechanisms of knowledge modules should ensure the implementation of all the functions that may be required in the formation of knowledge bases. It is shown that the mechanism of an object-function mechanism can be a system of agents. When using expert programming technology, the process of generating knowledge modules, translating them into object or executable modules in one of the traditional languages, and testing is performed as a single operation. After obtaining the necessary set of modules, a method is generated that uses a subset of the generated knowledge modules. The generated method can be used as a mechanism of the knowledge module, which allows the use of a hierarchy of rules. It is shown that a cooperative solution to complex problems is provided by a system of agents.

Key words: *engineering design, products, intelligent systems, multi-agent systems.*

Citation: *Evgenev GB.* Multi-agent systems of semi-automatic design based on object-functions model in engineering [In Russian]. *Ontology of designing.* 2020; 10(1): 50-62. – DOI: 10.18287/2223-9537-2020-10-1-50-62.

List of figures

Figure 1 – Object-function in a format close to UML

Figure 2 – An example of the formation of the content of the transition of machining in accordance with the standard

Figure 3 – An example of a knowledge module representing a functional dependency in tabular form

Figure 4 – UML component diagram as applied to the gear ratio calculation knowledge module

Figure 5 – Diagrams of the components (a) of the method (b)

Figure 6 – Areas of definition of exclusive (a) and non-exclusive (b) OR

Figure 7 – Morphological analysis of the designed product

Figure 8 – Generation of 3D models of a worm wheel

Figure 9 – 3D product model generation

References

- [1] Gartner. Top Strategic Predictions for 2016 and Beyond: The Future Is a Digital Thing. <https://www.gartner.com>.
- [2] **Gorodetsky VI, Bukhvalov OL, Skobelev PO, Mayorov IV.** Current status and prospects of industrial applications of multi-agent systems [In Russian], UBS, 2017; 66: 94-157.
- [3] **Alma Y. Alanis, Nancy Arana-Daniel and Carlos López-Franco.** Bio-inspired Algorithms for Engineering. Elsevier Inc. 2018. 152 p.
- [4] **Binitha S, S Siva Sathya.** A Survey of Bio inspired Optimization Algorithms International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2, May 2012. P.137-151.
- [5] **M. Blowers and A. Sisti,** *Evolutionary and Bio-inspired Computation: Theory and Applications*, SPIE Press, 2007.
- [6] **Avi Wigderson.** Mathematics and Computation: A Theory Revolutionizing Technology and Science PRINCETON 2019 440p.
- [7] **Makhatadze GI.** Linking *computation* and experiments to study the role of charge-charge *interactions* in protein folding and stability. IOP Publishing Phys Biol. 2017 Feb 7. DOI: 10.1088/1478-3975/14/1/013002.
- [8] **Luck M. et al.** Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing) // AgentLink. 2005. <http://www.agentlink.org/roadmap/>.
- [9] **Deloach SA.** Moving multi-agent systems from research to practice // Int. J. Agent-Oriented Software Engineering. 2009. Vol.3, No.4. P.378–382.
- [10] **Müller J., Fisher K.** Application Impact of Multiagent Systems and Technologies: A Survey // In «Agent-Oriented Software Engineering» book series. Springer, 2013. P.1–26.
- [11] **Evgenev GB.** Fundamentals of automation of technological processes and production [In Russian]. V.1: Information models. V.2: Design and control methods - Moscow, Publishing house of MGTU named N.E. Bauman, 2015. V.1: 441 p. V.2: 479 p.
- [12] **Aronov BM.** About the ways to increase the level of intelligence of integrated CAD systems of axial turbomachine blades [In Russian]. Problems and prospects for the development of engine building. Samara state aerospace un-t; [Ch. ed. E.V. Shakhmatov]. Samara: SSAU, 1999. Vol. 3, part 2. P.245-248.
- [13] **Thilmanny J.** Putting Artificial Intelligence to Work in CAD Design – Engineers Rule. December 21, 2017. <https://www.engineersrule.com/solidworks-puts-artificial-intelligence>.
- [14] SPRUT-Technology. Automation design [In Russian]. <https://sprut.ru/>.

About the author

Georgy Borisovich Evgenev (b. 1938) graduated from the Moscow aviation technological Institute in 1960, PhD (1964), Doctor of engineering (1975). He is the Winner of the USSR Council of Ministers Prize (1991). Professor at Moscow State Technical University named after N.E. Bauman. Member of the International Association for Ontology and its Applications. Co-author of about 190 scientific articles and abstracts in the field of CAD and AI. AuthorID (RCI): 17685. g.evgenev@mail.ru.

Received January 30, 2020. Revised February 17, 2020. Accepted March 15, 2020.

Разработка системы иерархических продукционных правил для диагностирования электротехнического оборудования

А.Е. Колоденкова, С.С. Верещагина

Самарский государственный технический университет, Самара, Россия

Аннотация

Процесс диагностирования электротехнического оборудования на этапе эксплуатации является многофакторным, сложным для математического описания и моделирования, поскольку факторы, влияющие на работу и техническое состояние оборудования, могут быть представлены не только количественными, но и лингвистическими переменными. Для повышения производительности функционирования и уровня отказоустойчивости оборудования необходимо разрабатывать методы и модели диагностирования, позволяющие учесть основные факторы, которые влияют на техническое состояние оборудования. В работе решается задача разработки системы иерархических продукционных правил для диагностирования электротехнического оборудования с учётом измерительной и экспертной информации. Новизной является то, что правила привязаны к разработанным неоднородным когнитивным моделям, которые охватывают закономерности получения, передачи, обработки разнотипных данных и в ранжировании правил с учётом длины пути в моделях между вершинами «условие» и «заключение», отражающих обобщённые знания дежурного персонала. Это позволяет уменьшить объём базы знаний, содержащей продукционные правила, сделать процесс поиска более эффективным и принять обоснованные диагностические решения относительно исправности оборудования.

Ключевые слова: диагностика, электрическое оборудование, иерархические продукционные правила, неоднородная когнитивная модель, ранжирование правил.

Цитирование: Колоденкова, А.Е. Разработка системы иерархических продукционных правил для диагностирования электротехнического оборудования / А.Е. Колоденкова, С.С. Верещагина // Онтология проектирования. – 2020. – Т. 10, №1(35). – С.63-72. – DOI: 10.18287/2223-9537-2020-10-1-63-72.

Введение

Диагностирование электротехнического оборудования (ЭО) является важным мероприятием, которое позволяет решать большой спектр различных задач: определять текущее состояние оборудования; выявлять существующие неисправности в работе; определять причины появления неисправностей; прогнозировать техническое состояние; принимать решение о неисправности оборудования и выдавать рекомендации по их устранению. Однако такое диагностирование существенно усложняется наличием следующих проблем: сложностью контролируемого оборудования, большим объёмом диагностических разнородных данных, большим количеством контролируемых параметров оборудования; недостаточностью внедрения современных методов и экспертных систем диагностирования; ограниченностью доступной информации [1-4].

Одним из способов решения данных проблем является применение современных подходов и методов, ориентированных на работу не только с измерительной информацией (количественными переменными), но и с экспертной (лингвистическими, качественными переменными). Реализовать данные подходы и методы с высокой степенью достоверности позволяет аппарат нечёткой логики.

Поскольку при обработке результатов наблюдения в процессе диагностирования ЭО дежурный персонал обычно сталкивается с проблемой обработки больших массивов разнородных данных, то в качестве модели представления знаний целесообразно использовать продукционные правила, которые просты при их разработке и гибки при описании технического состояния ЭО.

В связи с этим, для диагностирования ЭО на этапе эксплуатации необходима система иерархических продукционных правил (ИПП) для диагностирования оборудования, основанная на измерительных данных и знаниях дежурного персонала и позволяющая эффективно формализовать задачу диагностирования технического состояния оборудования и принимать обоснованные диагностические решения относительно исправности оборудования.

1 Этапы разработки системы иерархических продукционных правил

На рисунке 1 представлен фрагмент иерархической функциональной модели диагностирования ЭО, а именно функциональная модель разработки системы ИПП в нотации *IDEFO*, отображающая основные её функции.

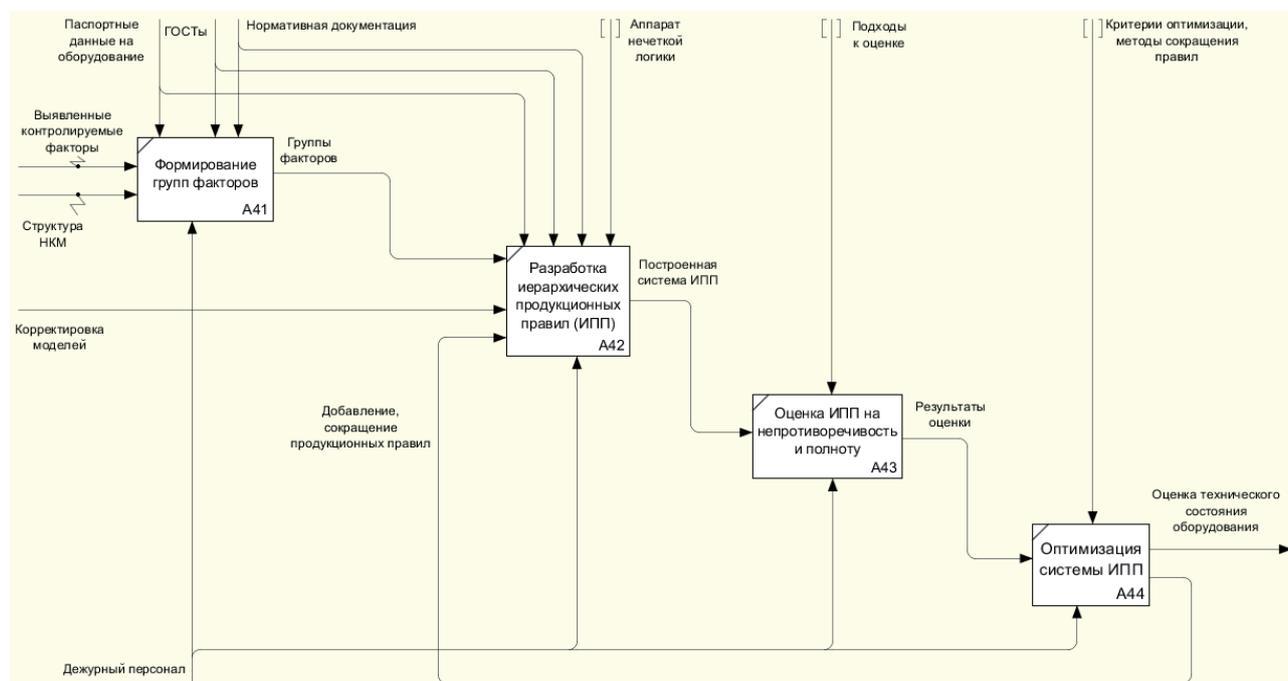


Рисунок 1 – Основные этапы разработки системы иерархических продукционных правил

Основные этапы разработки системы ИПП включают реализацию следующих функций.

1.1 Формирование групп факторов (функция A41)

Данная функция позволяет сформировать группы факторов (например, коэффициент временного перенапряжения, гармоника тока, состояние изоляции, климатические условия, квалификация персонала и др.) на основе документации (паспортные данные на оборудование, ГОСТы, нормативная документация), которая отражает условия эксплуатации ЭО, возможные отклонения параметров. Необходимость привлечения дежурного персонала обусловлена отсутствием или неполнотой объективной информации об оборудовании.

Обобщённая схема формирования групп факторов показана на рисунке 2 на примере не-

однородной когнитивной модели (НКМ), представленной на рисунке 3.

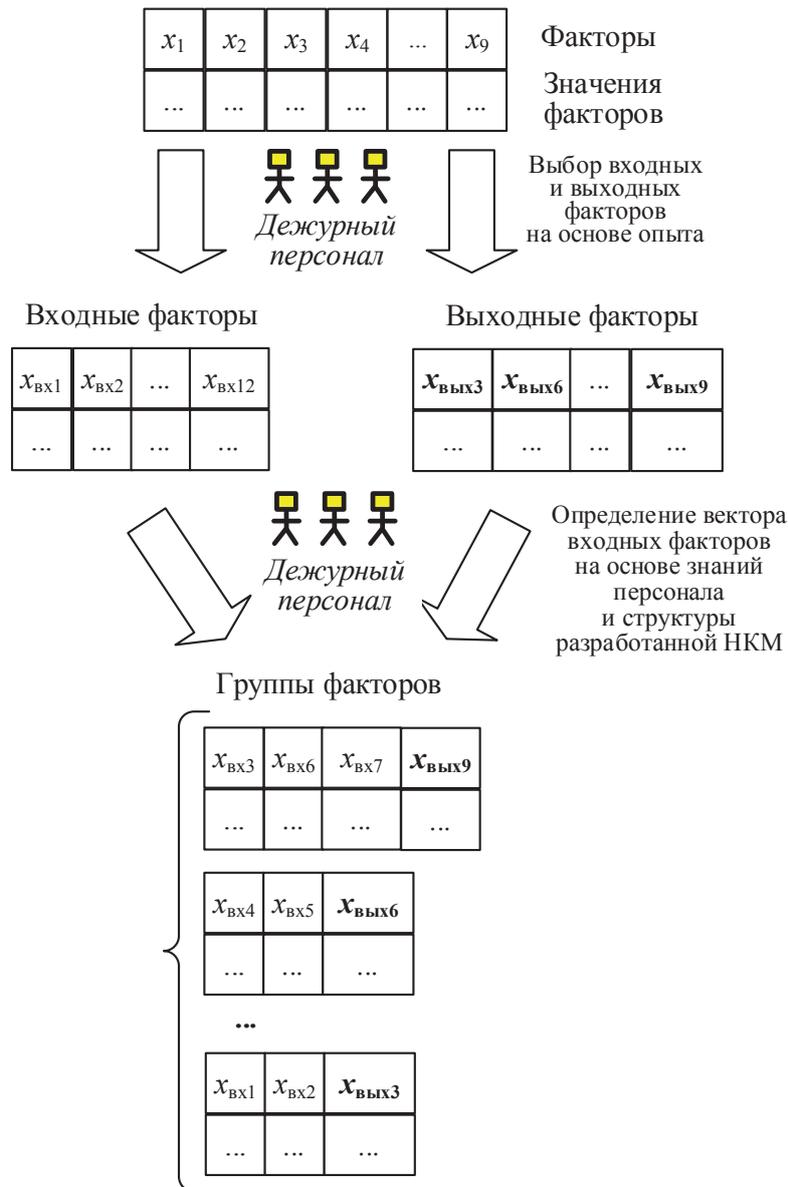


Рисунок 2 – Обобщённая схема формирования групп факторов

Методика формирования групп факторов состоит из четырёх шагов.

Шаг 1. Осуществляется выявление контролируемых факторов, полученных от разнотипных приборов, на основе опыта дежурного персонала, и влияющих на техническое состояние ЭО, $x_i = \{x_1, x_2, \dots, x_h\}$, $i = \overline{1, h}$, h – количество факторов.

Шаг 2. Осуществляется выбор выходных (целевых) факторов $x_{вых_1}, x_{вых_2}, \dots, x_{вых_c}$, $c < h$, которые могут быть получены в перспективе (например, состояние ЭО, аварийное состояние, рекомендации и т.п.), и определяются векторы входных (управляющих) факторов $\vec{x}_{вх_k}$, значимо влияющих на выходные факторы, на которые может быть оказано влияние (квалификация персонала, точность приборов, погодные условия, показания с приборов и т.п.) из всех выявленных контролируемых факторов x_i на основе опыта дежурного персонала и разработанной НКМ, представленной на рисунке 3.

НКМ представляет собой ориентированный взвешенный граф, вершинами которого являются факторы, влияющие на техническое состояние оборудования, а рёбрами – нечёткие причинно-следственные связи между факторами:

$$G_{\text{неод}} = \langle V, F, W \rangle,$$

где $V = \{v_i\}$ – множество вершин, $v_i \in V$, $i = \overline{1, h}$, h – количество вершин; $F: V \rightarrow X$, $X = \{x_{v_i}\}$ – множество значений вершин (каждой вершине ставится один фактор); F – функции, необходимые для вычисления некоторых значений вершин, которые сопоставлены вершинам; W – нечёткие причинно-следственные связи между вершинами v_i и v_j , $i, j = \overline{1, h}$, характеризующие направление и силу влияния между ними [5].

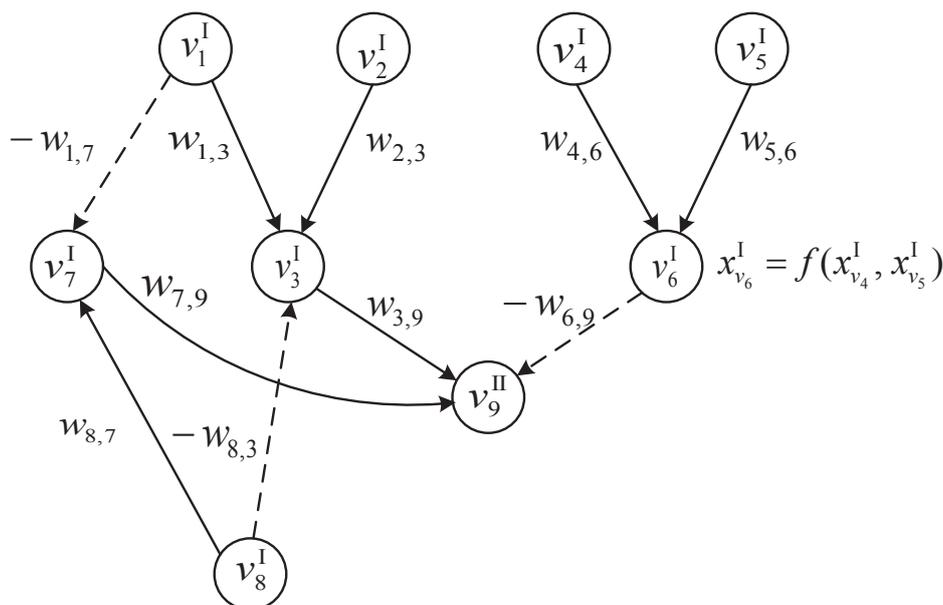


Рисунок 3 – Фрагмент неоднородной когнитивной модели

Неоднородность модели обусловлена наличием вершин двух разных типов.

1) I^I – управляющие факторы. Значения вершин первого типа $x_{v_i}^I$ могут быть представлены в виде: лингвистических переменных (например, опыт персонала – «низкий», «средний», «высокий» и т.п.); интервальных значений, взятых из документации; числовых значений, полученных от измерительных приборов. Значения вершин $x_{v_i}^I$ могут определяться не только на основе опыта, но и вычисляться с помощью функций F , которые могут зависеть от одной или нескольких переменных ($x_{v_6}^I = f(x_{v_4}^I, x_{v_5}^I)$, см. рисунок 3).

2) I^{II} – целевые факторы (v_9^II , см. рисунок 3). Значения вершин второго типа $x_{v_i}^{II}$ могут быть представлены только в виде лингвистических переменных (например, состояние оборудования = «неисправное»).

Отметим, что НКМ являются частным случаем неоднородной семантической сети [6]. Специфика предложенного типа модели заключается в особой процедуре обработки множества вершин и рёбер, которая возникает из постановок задач, связанных с диагностированием ЭО.

Шаг 3. Формируются группы факторов $U_k = \{U_1, U_2, \dots, U_c\}$, $k = \overline{1, c}$, где $c < h$, $U_k \subset x_i$, которые наиболее полно описывают ЭО [7].

Шаг 4. Осуществляется объединение групп факторов $U_k = \{\bar{x}_{\text{вх}_k}, x_{\text{вых}_k}\}$ в иерархию с учётом длины пути в НКМ между вершинами «условия» и «заклучение»

$$R_{x_i} = U_1 \cup U_2 \cup \dots \cup U_c = \{\bar{x}_{\text{вх}_1}, x_{\text{вых}_1}\} \cup \{\bar{x}_{\text{вх}_2}, x_{\text{вых}_2}\} \cup \dots \cup \{\bar{x}_{\text{вх}_k}, x_{\text{вых}_k}\}, k = \overline{1, c}.$$

Следует отметить, что иерархия групп факторов может быть сложной, поскольку она зависит от количества факторов, а также от структуры НКМ.

1.2 Разработка иерархических продукционных правил (функция A42)

Данная функция позволяет разрабатывать иерархические правила на основе групп факторов.

В процессе разработки базы знаний (БЗ) формируется большое количество продукционных правил, при этом БЗ в большинстве случаев оказывается избыточной, что в свою очередь усложняет работу дежурного персонала по интерпретации результатов и приводит к проблеме разрешения конфликта. Для решения данных проблем предлагается производить оптимизацию БЗ (сокращение числа правил в БЗ) за счёт их привязки к НКМ, охватывающей все закономерности получения, передачи, обработки разнотипных данных и ранжировании правил с учётом длины пути в НКМ между вершинами «условие» и «заклучение», отражающими обобщённые знания дежурного персонала.

Поскольку на техническое состояние ЭО влияют группы факторов (много входов) и в качестве выхода выступает единственное решение (один выход), то применяется правило типа много входов-один выход (*MISO, Multi Inputs Single Outputs*).

Модель продукционного правила, основанная на *MISO*-структуре, может быть представлена в виде:

$$Rule_z : \text{Если } x_1 = A_{z1} \text{ и } x_2 = \tilde{A}_{z2} \text{ и } \dots x_h = \tilde{A}_{zh} ([x_{h \min}, x_{h \max}]), \text{ то } y = B_{z1} [Rang_1],$$

...

$$Rule_d : \text{Если } \underbrace{x_1 = A_{d1}}_{\text{чёткое значение}} \text{ и } \underbrace{x_2 = \tilde{A}_{d2}}_{\text{нечёткое значение}} \text{ и } \dots \underbrace{x_h = \tilde{A}_{dh} ([x_{h \min}, x_{h \max}])}_{\text{интервал}}, \text{ то } y = B_{dc} [Rang_c],$$

где x_i – значения параметров оборудования (входные переменные/данные), $i = \overline{1; h}$; A_i – чёткое значение параметра оборудования; $\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$ – нечёткое значение параметра оборудования, $\mu_{\tilde{A}_i}(x_i)$ – функция принадлежности значений параметров оборудования; $\tilde{A}_i([x_{i \min}, x_{i \max}])$ – значение параметра оборудования представлено в виде интервала, где $x_{i \min}, x_{i \max}$ – минимальное и максимальное значение параметра i -го оборудования; y – значение результата по поводу исправности ЭО (выходная переменная, выходные данные); B – чёткое и/или нечёткое значение результата, $Rang$ – значимость правила, $Rang \in \{1, 2, \dots, c\}$, где c – количество продукционных правил.

Значимость продукционных правил устанавливается в соответствии с длиной пути в НКМ и определяется следующим образом:

$$Rang_{(k)} = \max_k \{N_{(k)}\}, k = \overline{1; c},$$

где c – количество продукционных правил; $N^{(k)}$ – количество входных факторов (переменных) в формировании условий правил в k -м продукционном правиле.

Продукционное правило, включающее наибольшее количество входных факторов, которые характеризуют техническое состояние ЭО, получает ранг 1, далее 2 и т.д. Правила, име-

ющие одинаковые ранги (например, $Rule_3$ имеет ранг 5 и $Rule_7$ имеет ранг 5), являются с точки зрения дежурного персонала равноценными. Поэтому они должны иметь ранги 5 и 6.

Особенностью данной модели является то, что она имеет возможность одновременно работать с разнородными данными, как на входе (условия, входные переменные/данные), так и на выходе правил (заклучения, консеквент, выходные переменные/данные); обрабатывать исходные данные, представленные как количественными, так и качественными переменными; учитывать значимость правил на основе их ранжирования.

На основе разработанной группы факторов, строится система ИПП. Данная система представляет собой иерархическую форму правил (отношение включения), т.е. два продукционных правила совместимы с входным вектором и одно правило включено в другое правило.

Пример иерархической формы для трёх продукционных правил может быть представлен в следующем виде:

$Rule_1$: ЕСЛИ x_8 =< норма > И x_{10} =< низкое > И $x_{12} = 233$, ТО x_6 =< исправное оборудование > ,

$Rule_2$: ЕСЛИ x_8 =< норма > И x_{10} =< низкое > , ТО x_6 =< исправное оборудование > ,

$Rule_3$: ЕСЛИ x_{10} =< низкое > , ТО x_6 =< неисправное оборудование > .

Эти же продукционные правила в иерархической форме имеют вид:

$Rule_1$: ЕСЛИ x_8 =< норма > И x_{10} =< низкое > И $x_{12} = 233$, ТО x_6 =< исправное оборудование > [1],

ИНАЧЕ $Rule_2$: ЕСЛИ x_8 =< норма > И x_{10} =< низкое > , ТО x_6 =< исправное оборудование > [2],

ИНАЧЕ $Rule_3$: ЕСЛИ x_{10} =< низкое > , ТО x_6 =< неисправное оборудование > [3].

Здесь x_6 – техническое состояние оборудования; x_7 – тепловизионная съёмка; x_8 – состояние изоляции; x_{10} – точность приборов; x_{11} – коэффициент временного перенапряжения; x_{12} – напряжение; x_{13} – сила тока; x_{14} – действующее значение напряжения обратной последовательности основной частоты трехфазной системы напряжений.

Первое продукционное правило должно иметь больший ранг, чем последнее, поскольку оно включает большое количество входных факторов, которые характеризуют техническое состояние ЭО. Поэтому именно оно является ценным и полезным для принятия решения об исправности ЭО.

То есть, продукционное правило $Rule_d$ входит в правило $Rule_z$ ($Rule_d \subseteq Rule_z$), когда отношение включения $A_{di} \subseteq A_{zi}$, $i = \overline{1;h}$ выполняется для всех входных переменных.

$$\left\{ \begin{array}{l} Rule_z : \text{Если } x_1 = A_{z1} \text{ и } x_2 = \tilde{A}_{z2} \text{ и } \dots x_h = \tilde{A}_{zh} ([x_{h\min}, x_{h\max}]), \text{ то } y = B_{z1} [Rang_1], \\ \dots \\ Rule_d : \text{Если } x_1 = A_{d1} \text{ и } x_2 = \tilde{A}_{d2} \text{ и } \dots x_h = \tilde{A}_{dh} ([x_{h\min}, x_{h\max}]), \text{ то } y = B_{dc} [Rang_c]. \end{array} \right.$$

Чем длиннее продукционное правило (детальное описание условий), тем ценнее правило для принятия диагностических решений. При этом короткие правила, полученные от дежурного персонала, отображают грубые знания и свойственны новичкам.

К особенностям системы ИПП относится то, что выход одного правила может являться входом для другого правила; могут встречаться независимые правила; продукционные правила привязаны к структуре НКМ.

1.3 Оценка ИПП на непротиворечивость и полноту (функция А43)

Данная функция позволяет проверить БЗ, содержащую систему ИПП, на удовлетворение единым формальным требованиям (свойствам), а именно на непротиворечивость и полноту.

Проверка непротиворечивости разработанной системы ИПП является сложной, поскольку в БЗ присутствует большое количество продукционных правил. При этом дежурный пер-

сонал тратит много времени на данную проверку при добавлении новых продукционных правил.

Поскольку для описания диагностирования технического состояния ЭО используется НКМ, отображающая все термы лингвистических переменных и продукционные правила, то НКМ должна обладать следующими свойствами:

- количество нечётких причинно-следственных связей в НКМ равно количеству продукционных правил;
- количество вершин в НКМ не больше, чем количество продукционных правил.

Определение 1 (критерий непротиворечивости). БЗ является непротиворечивой, если в ней отсутствуют продукционные правила, имеющие одинаковые предпосылки (функции принадлежности, одинаковые термы-множества одной и той же входной переменной, интервалы, численные значения), но различные заключения (функции принадлежности, интервалы, численные значения).

Дублирование данных в БЗ повышает риск противоречивости. Для достижения непротиворечивости БЗ необходимо оставлять только одно из всех противоречивых продукционных правил.

Определение 2 (критерий полноты). БЗ является полной, если для любых значений входных переменных в базе существует хотя бы одно правило, которое реализуется в процессе нечёткого логического вывода.

Полнота используется как мера, указывающая на полноту знаний, которые содержатся в БЗ. Под неполной понимают такую БЗ, используя которую невозможно осуществить вывод для ряда определённых ситуаций [8].

В качестве примера для меры полноты БЗ может использоваться критерий, описанный в работе [9]

$$CM(x) = \sum_{k=1}^{N_r} \left\{ \prod_{i=1}^{N_x} \mu_{A_{ik}}(x) \right\},$$

где x – физическая переменная входных данных (условий); $\mu_{A_{ik}}(x)$ – функция принадлежности физической переменной входных данных x к нечёткому терму A_{ik} , которым оценивается i -я входная переменная в k -ом правиле; N_x – число условий в правиле; N_r – число правил в базе правил.

Численные значения, которые принимает критерий $CM(x)$, позволяют классифицировать БЗ, содержащую систему ИПП, по полноте знаний: $CM(x) = 0$ – «неполная» БЗ; $0 < CM(x) < 1$ – БЗ «незначительно полная»; $CM(x) = 1$ – БЗ «точно полная»; $CM(x) > 1$ – БЗ «избыточная».

В работе [10] для определения полноты БЗ предлагается для каждого правила использовать соотношение

$$C^{syn}(X_j)(R_j) = C^{sem}(Y_j)(R_j),$$

где R_j – правило вывода, $j = \overline{1, M}$; X_j, Y_j – условие и заключение j -го правила; C^{syn}, C^{sem} – синтаксическая и семантическая оценки j -го правила (степени принадлежности условий и заключения).

Проверка БЗ на противоречивость и полноту позволяет повысить точность диагностических решений об исправности ЭО, получаемых с применением системы ИПП.

Таким образом, иерархия продукционных правил способствует полноте и непротиворечивости. При этом, чем короче правило, тем больше полнота, а чем длиннее правило, тем лучше непротиворечивость.

1.4 Оптимизация системы ИПП (функция А44)

Данная функция позволяет уменьшить количество ИПП в БЗ и улучшить свойства полноты и непротиворечивости.

Сокращение числа правил может быть реализовано с помощью метода исключения дублирующих правил либо с помощью исключения правил с наименьшими рангами.

Таким способом, решается проблема противоречивых правил, а также существенно уменьшается их общее число. Оставшиеся правила формируют итоговую базу правил.

Заключение

В статье описана разработка системы ИПП для диагностирования ЭО с учётом измерительной и экспертной информации. Новизна состоит в том, что правила привязаны к разработанным НКМ, которые охватывают закономерности получения, передачи, обработки разнотипных данных, и в ранжировании правил с учётом длины пути в моделях между вершинами «условие» и «заклучение», отражающих обобщённые знания дежурного персонала. Разработанная система ИПП позволяет уменьшить объём БЗ, содержащей продукционные правила, сделать процесс поиска более эффективным. Ранжирование правил по значимости позволяет выбрать из исследуемой совокупности правил наиболее существенные, которые полно отражают техническое состояние ЭО для принятия решений о работоспособности ЭО.

Благодарности

Работа выполнена при поддержке РФФИ, проекты № 19-07-00195, 19-08-00152.

Список источников

- [1] *Elyshev, D.K.* On the development of intelligent expert diagnostic system for assessing the conditions of electrical equipment / D.K. Elyshev // Systems. Methods. Technology. - 2017. - No. 3(35). - С.57-63.
- [2] *Pareek, S.* Application of artificial neural networks to monitor thermal condition of electrical equipment / S. Pareek, R. Sharma, R. Maheshwari // 2017 3rd International Conference on Condition Assessment Techniques in Electrical Systems (CATCON). - 2017. - P.183-187.
- [3] *Kolodenkova, A.* Diagnostics of Industrial Electrical Equipment Using Modern Information Technologies / A. Kolodenkova, S. Vereshchagina // 2019 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), Sochi, Russia. - 2019 - P.1-5.
- [4] *Вдовико, В.П.* Методология системы диагностики электрооборудования высокого напряжения / В.П. Вдовико // Электричество. - 2010. - №2. - С.14-20.
- [5] *Силов, В.Б.* Принятие стратегических решений в нечёткой обстановке / В.Б. Силов. - М.: ИНПРО-РЕС, 1995. - 228 с.
- [6] *Осипов, Г.С.* Построение моделей предметных областей. Ч. I. Неоднородные семантические сети / Г.С. Осипов // Известия РАН. Техническая кибернетика. - 1990. - № 5. - С.32-35.
- [7] *Катасев, А.С.* Нечётко-продукционная каскадная модель диагностики состояния сложного объекта / А.С. Катасев, Л.Ю. Емалетдинова // Программные системы и вычислительные методы. - 2013. - № 1(2). - С.69-81.
- [8] *Preece, A.D.* Foundation and application of knowledge base verification / A.D. Preece, R. Shinghal // International Journal of Intelligent Systems. - 1994. - Vol. 9. - P.683-701.
- [9] *Белов, А.А.* Основы теории нечеткости / А.А. Белов, Т.В. Гвоздева. - ГОУ ВПО «Ивановский государственный энергетический университет имени В.И. Ленина», 2001. - 119 с.
- [10] *Солдатова, О.П.* Сравнительный анализ алгоритмов генерации баз нечётких продукционных правил на примере решения задачи классификации/ О.П. Солдатова // Измерение. Мониторинг. Управление. Контроль. - 2014. - № 4 (10). - С.43-48.

Сведения об авторах



Колоденкова Анна Евгеньевна, 1982 г. рождения. Окончила Уфимский государственный авиационный технический университет в 2004 г., д.т.н. (2017). Заведующая кафедрой «Информационные технологии» Самарского государственного технического университета. Член Российской ассоциации искусственного интеллекта, заместитель председателя научного совета Российской ассоциации нечётких систем и мягких вычислений. В списке научных трудов более 150 работ в области атомной энергетики, программной инженерии, системного анализа, интеллектуальных систем, мягких вычислений, экспертной поддержки принятия решений, технической диагностики и мониторинга за состоянием промышленного оборудо-

вания. AuthorID (РИНЦ): 175446. Author ID (Scopus): 57190670136; Researcher ID (WoS): F-1341-2018. anna82_42@mail.ru

Верещagina Светлана Сергеевна, 1983 г. рождения. Окончила Самарский государственный технический университет в 2005 г. Старший преподаватель кафедры «Информационные технологии» Самарского государственного технического университета. Область научных интересов: системный анализ, интеллектуальные системы, мягкие вычисления, экспертная поддержка принятия решений, техническая диагностика и мониторинг за состоянием промышленного оборудования. AuthorID (РИНЦ): 739715. Author ID (Scopus): 57215123417; Researcher ID (WoS): D-9728-2014. werechaginass@mail.ru



Поступила в редакцию 02.02.2020, после рецензирования 10.03.2020. Принята к публикации 16.03.2020.

Development of a system of hierarchical production rules for electrical equipment diagnosing

A.E. Kolodenkova, S.S. Vereshchagina

Samara State Technical University, Samara, Russia

Abstract

The process of electrical equipment diagnosing at the operational stage is a multifactor process, complicated for mathematical description and modeling, since the factors affecting the operation and technical condition of equipment can be represented not only by quantitative variables, but also by linguistic variables. In order to improve the operating performance and the level of equipment fault tolerance, it is necessary to develop diagnostic methods and models using modern information technologies that take into account the main factors that affect the technical condition of the equipment. In this paper, we solve the problem of developing a system of hierarchical production rules for electrical equipment diagnosing including measuring and expert information. The novelty is that the rules are tied to developed heterogeneous cognitive models that cover all the laws of obtaining, transmitting, processing heterogeneous data and ranking rules taking into account the path length in the models between the “condition” and “conclusion” vertices, reflecting the generalized knowledge of the duty staff. This makes it possible to reduce the knowledge base capacity containing production rules, to make the search process more efficient and to take scientifically based diagnostic decisions regarding equipment health.

Key words: *equipment diagnostics, hierarchical production rules, heterogeneous cognitive model, ranking of rules.*

Citation: *Kolodenkova AE, Vereshchagina SS. Development of a system of hierarchical production rules for electrical equipment diagnosing [In Russian]. Ontology of designing. 2020. 1(1): 63-72. – DOI: 10.18287/2223-9537-2020-10-1-63-72.*

List of figures

Figure 1 - Main development stages of a hierarchical production rules system

Figure 2 - A generalized scheme for the formation of groups of factors

Figure 3 - A fragment of a heterogeneous cognitive model

Acknowledgment

The work was supported by RFBR grants No. 19-07-00195, 19-08-00152.

References

- [1] *Eltyshev DK*. On the development of intelligent expert diagnostic system for assessing the conditions of electrical equipment. *Systems. Methods. Technology* 2017; 3: 57-63.
- [2] *Pareek S, Sharma R, Maheshwari R*. Application of artificial neural networks to monitor thermal condition of electrical equipment. 2017 3rd International Conference on Condition Assessment Techniques in Electrical Systems (CATCON) 2017: 183-187.
- [3] *Kolodenkova A, Vereshchagina S*. Diagnostics of Industrial Electrical Equipment Using Modern Information Technologies. 2019 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), Sochi, Russia. 2019: 1-5.
- [4] *Vdoviko VP*. The methodology of the diagnostic system for high voltage electrical equipment [In Russian]. *Electricity* 2010; 2: 14-20.
- [5] *Silov VB*. Strategic decision making in fuzzy environment [In Russian]. - Moscow: INPRO-RES, 1995.
- [6] *Osipov GS*. Building models of subject areas. Part I. Heterogeneous semantic networks [In Russian]. *Izvestiya RAS. Technical cybernetics* 1990; 5: 32-35.
- [7] *Katasev AS, Emaletdinova LY*. Fuzzy-production cascade model for diagnosing the state of a complex object [In Russian]. *Software systems and computational methods* 2013; 1: 69-81.
- [8] *Preece AD, Shinghal R*. Foundation and application of knowledge base verification. *International Journal of Intelligent Systems* 1994; 9: 683-701.
- [9] *Belov AA, Gvozdeva TV*. Basics of the fuzzy theory [In Russian]. Ivanovo State power engineering University, 2001. 119 p.
- [10] *Soldatova OP*. Comparative analysis of algorithms of fuzzy production rules' database generation with an example of solving the classification problem [In Russian]. *Measurement. Monitoring. Management. Control*. 2014; 4: 43-48.

About the authors

Anna Evgenievna Kolodenkova (b. 1982) graduated from the Ufa State Aviation Technical University in 2004, D. Sc. Eng. (2017). She is Associate Professor, Head of «Information technologies» Department at Samara State Technical University. She is a member of Russian Association of Artificial Intelligence and Deputy Chairman of the scientific Council of the Russian Association of fuzzy systems and soft computing. She is a co-author of about 150 scientific articles and abstracts in the field of nuclear energy, software engineering, system analysis, intelligent systems, soft computing, expert decision support and technical diagnostics and monitoring of industrial equipment condition. AuthorID (RCI): 175446. Author ID (Scopus): 57190670136; Researcher ID (WoS): F-1341-2018. anna82_42@mail.ru.

Svetlana Sergeevna Vereshchagina (b. 1983) graduated from the Samara State Technical University in 2005. She is Senior Lecturer of «Information technologies» Department at Samara State Technical University. Area of scientific interest: system analysis, intelligent systems, soft computing, expert decision support and technical diagnostics and monitoring of industrial equipment condition. AuthorID (RCI): 739715. Author ID (Scopus): 57215123417; Researcher ID (WoS): D-9728-2014. werechaginass@mail.ru.

Received February 2, 2020. Revised March 10, 2020. Accepted March 16, 2020.

Модели возникновения ошибок на предпроектной стадии разработки компонент информационно-вычислительных систем

В.Е. Гвоздев, О.Я. Бежаева, Р.А. Насырова

Уфимский государственный авиационный технический университет, Уфа, Россия

Аннотация

В работе дан обзор подходов к исследованию ошибок в спецификациях требований, выделены системообразующие факторы ошибок предпроектной стадии, предложена когнитивная карта ошибок предпроектной стадии, проведён анализ её адекватности. Отмечается важность дальнейшего развития построения формальных моделей процессов, в ходе реализации которых возникают ошибки. Процессам предотвращения дефектов в спецификациях внешнего облика поставлен в соответствие известный архетип «пределы роста» и показано, что существует предел возможностей по предупреждению дефектов, обусловленный как неопределённостью проблемной ситуации, так и компетентностью разработчиков требований. С использованием совокупности структурных моделей процессов предпроектной стадии разработки компонент информационно-вычислительных систем показано, что дефекты в спецификациях внешнего облика обусловлены как особенностями когнитивных моделей субъектов, причастных к формированию требований, так и организацией формирования требований.

Ключевые слова: *предпроектная стадия, спецификация, когнитивная карта, факторы ошибок, модели процессов, функциональный граф.*

Цитирование: *Гвоздев, В.Е. Модели возникновения ошибок на предпроектной стадии разработки компонент информационно-вычислительных систем / В.Е. Гвоздев, О.Я. Бежаева, Р.А. Насырова // Онтология проектирования. – 2020. – Т. 10, №1. - С. 73-86. – DOI: 10.18287/2223-9537-2020-10-1-73-86.*

Введение

Цифровая экосреда является системообразующим фактором системы управления «умным предприятием» [1]. В связи с этим возрастают требования к функциональной безопасности компонент информационно-вычислительных систем (ИВС), обеспечивающих своевременное представление информации субъектам, причастным к управлению «умным предприятием», и имеющим несовпадающие, зачастую нечёткие и до конца не осознаваемые персональные цели и желания [2] (следуя [3, 4], такие субъекты именуются неоднородными акторами, НА). Особенностью «умного предприятия» является ориентация на удовлетворение постоянно изменяющихся персонифицированных потребностей клиентов, что требует постоянного внесения изменений в бизнес-процессы предприятия [1]. Это делает необходимым изменение функциональных и нефункциональных характеристик информационных и инфраструктурных компонент ИВС предприятия в темпе внесения изменений в бизнес-процессы.

Известно много работ, посвященных выявлению и анализу ошибок в спецификациях внешнего облика программных продуктов, спецификациях системных требований пользователей, а также ошибок, возникающих в процессе преобразования этих спецификаций в аппаратную и программную составляющие компонент ИВС. Не получили должного развития методологические основы управления предотвращением ошибок, возникающих на предпроектной стадии разработки компонент ИВС. Несмотря на развитие инструментальных средств,

эффективность реализации проектов ИВС и потребительские свойства получаемых продуктов не всегда достаточны. В качестве критических факторов успеха проектов можно выделить те, которые обусловлены присутствием людей [5].

Целью исследования является разработка формальных моделей возникновения ошибок на предпроектной стадии создания компонент ИВС, которые приводят к дефектам в спецификациях их внешнего облика. Представлены структурные модели процесса, результатом которого является спецификация внешнего облика компонент ИВС; структура системы управления урегулированием проблемной ситуации (ПС) в условиях различия представлений о ценностях НА; когнитивная карта ошибок, возникающих на предпроектной стадии.

1 Обзор подходов к исследованию ошибок в спецификациях требований

Возникновение ошибок на стадии формирования внешнего облика (спецификации требований пользователей) обусловлено следующими факторами:

- размытостью границ фазы формирования требований;
- различным восприятием ПС НА и их разными представлениями о ценностях;
- множественностью и неоднородностью источников требований к потребительским свойствам компонент ИВС;
- различием в требованиях к качеству реализации функциональных и нефункциональных свойств у разных пользователей ИВС;
- неоднозначным толкованием участниками формирования требований содержания разных документов в силу неоднозначности содержания многих терминов естественного языка, на котором составлены документы.

Изначально в управлении качеством спецификаций программных продуктов внимание разработчиков фокусировалось на дефектах. Используются следующие подходы, ориентированные на выявление дефектов: карточки учёта дефектов [6]; выявление дефектов независимыми группами экспертов [7]; анализ спецификаций требований разными целевыми группами пользователей [8]. Эти подходы предназначены для непосредственного выявления дефектов и не ориентированы на выявление причин их возникновения.

К числу распространенных методов выявления причин отклонения поведения компонент ИВС от ожидаемого относится анализ корневых причин (*Root Cause Analysis*, RCA) [9]. RCA используется для выявления дефектов, которые явились причиной возникновения цепочки дефектов, обусловивших наблюдаемые явления. Установление причины возникновения дефекта гораздо важнее, нежели установить дефект [10]. Выявление ошибок как причин возникновения дефектов создаёт основу для поиска ещё не проявившихся дефектов.

Таксономия дефектов в спецификациях требований (*Requirements Error Taxonomy*, RET) содержит методическую основу деятельности инспекторов требований, обеспечивающую сопоставимость результатов анализа ошибок, выявленных разными инспекторами при изучении спецификаций, относящихся к разным программным проектам [11]. В рамках RET имеется возможность идентификации ошибок, совершаемых людьми, физическим проявлением которых являются дефекты в спецификации внешнего облика компонент ИВС. Ограничением RET является то, что этот подход к изучению дефектов непосредственно не включает изучение когнитивной природы ошибок, совершаемых людьми.

Таксономия ошибок, совершаемых людьми (*Human Error Taxonomy*, HET), является результатом научно-обоснованной адаптации подхода к управлению функциональной безопасностью сложных технических систем, концептуальную основу которого составляет модель, предложенная в [12]. Различные системы классификации ошибок, совершаемых людьми, приведены в работах [13, 14]. Эти подходы являются эвристическими. Ошибки когнитивной

природы, совершаемые людьми, являются критическим фактором обеспечения функциональной безопасности [15].

Систематические знания об ошибках, совершаемых людьми, составляют фундамент системы управления предотвращением ошибок [16]. Основу используемых классификаций совершаемых людьми ошибок составляет учёт особенностей получения и обработки информации на разных стадиях процессов организации управления сложной системой: ощущения и осознания ПС; принятия решений; реализации действий.

Выделяют три типа ошибок, обусловленных недостатками когнитивной деятельности, совершаемых людьми: ошибки, обусловленные невнимательностью исполнителей и возникающие при выполнении правильных плановых действий; ошибки, обусловленные забывчивостью исполнителей и возникающие при выполнении правильных плановых действий; ошибки, характеризующие ситуацию, когда разрабатывается неверный план действий вследствие недостатка знаний. Результатом построения таксономии является формирование древовидной структуры ошибок, относящихся к названным типам. В рамках НЕТ ошибки разделяются также на случайные и преднамеренные. Под преднамеренными ошибками понимается сознательное нарушение заранее определённых правил.

На основании приведённого обзора можно заключить следующее.

- Дефекты в спецификациях требований обусловлены как особенностями среды формирования требований (степень неопределённости ПС; особенности коммуникаций между акторами), так и ошибками, обусловленными ментальными особенностями субъектов, причастных к разработке спецификаций.
- Наиболее часто используемыми подходами к таксономии ошибок в спецификациях требований являются RET и НЕТ. Таксономия RET ориентирована на управление ошибками, физическим проявлением которых являются дефекты в организации управления программными проектами. Таксономия НЕТ ориентирована на управление ошибками, физическим проявлением которых являются дефекты в программных продуктах. Целесообразно рассматривать RET и НЕТ как взаимодополняющие подходы к созданию основы управления ошибками, подобно тому, как это сделано в ESA-PSS-05-11 [17].

2 Системообразующие факторы ошибок предпроектной стадии

Основу формирования спецификации внешнего облика компонент ИВС цифровой среды составляет стратегия урегулирования ПС. Формирование стратегии представляет собой многоэтапный процесс, представленный на рисунке 1, реализация которого ориентирована на решение следующих ключевых задач:

- осознание наличия ПС и её описание;
- представление ПС в виде объекта управления;
- формирование НА консолидированного подхода к урегулированию (т.е. целенаправленному изменению) ПС.

Системообразующими факторами возникновения ошибок при выработке стратегии, приводящими впоследствии к возникновению дефектов в спецификации внешнего облика компонент ИВС, являются следующие.

- Недостаток знаний о коренных причинах возникновения ПС (часто разнесённых в пространстве и во времени относительно наблюдаемых симптомов ПС [12]). Омnipотентность и латентность коренных причин, что влечёт за собой ошибки в выборе формальных архетипов для описания ПС [2, 18, 19].

- Неполнота и противоречивость понятийного отображения ПС, вызываемая субъективизмом в восприятии проявлений ПС НА и ограниченностью их персональных онтологий; обусловленная этим неполнота и различие персональных когнитивных моделей [3, 20];
- Ограниченный потенциал моделей, используемых для описания ПС как объекта управления.

Модель источников ошибок формирования стратегии урегулирования ПС представлена на рисунке 2.

Центральной задачей, предшествующей выработке стратегии урегулирования ПС, является получение описания ПС. Онтологическая модель ПС строится на основе персональных онтологических моделей НА, вовлечённых в урегулирование ПС. Основой построения когнитивной модели ПС служат персональные когнитивные модели НА.

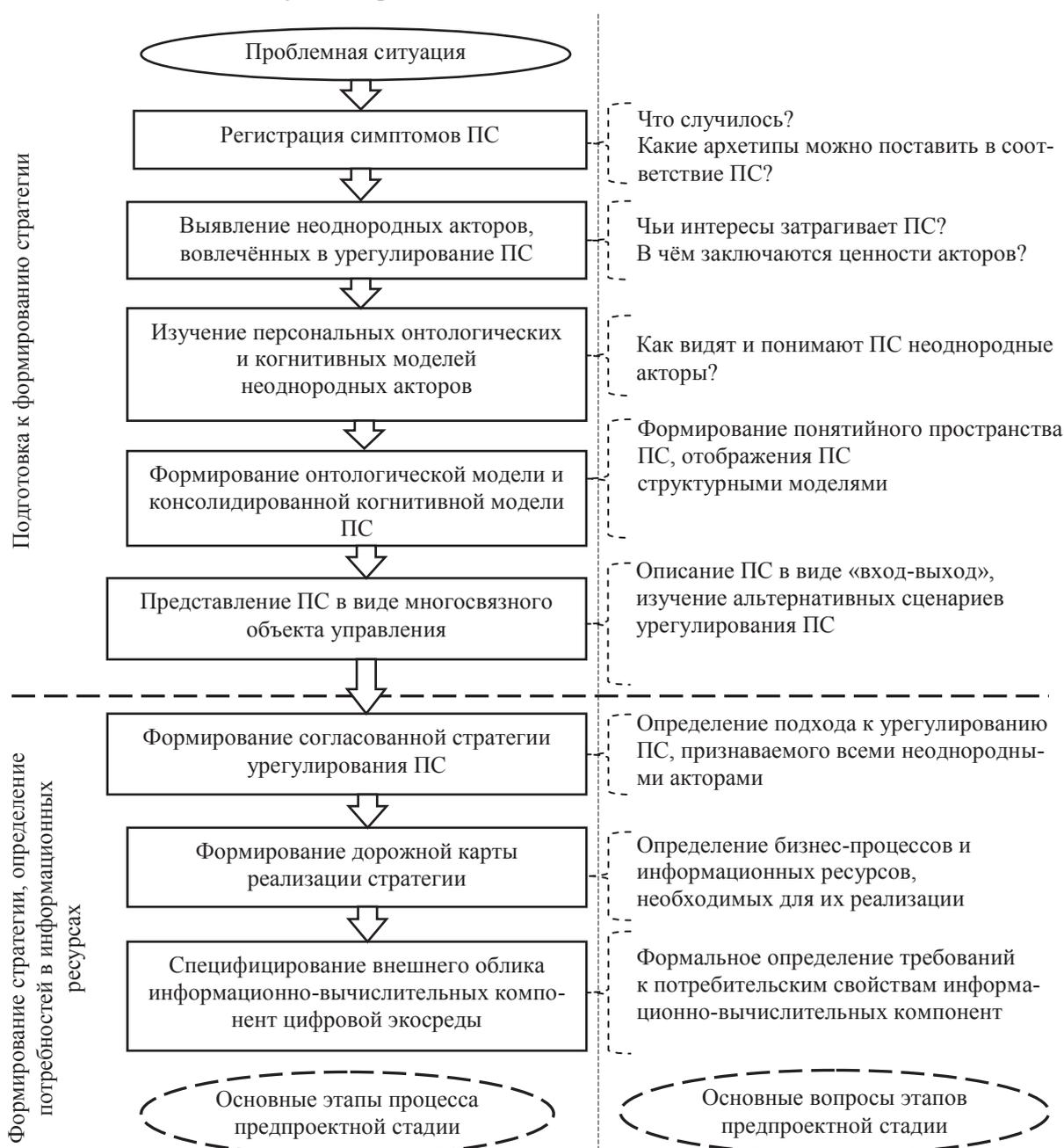


Рисунок 1 - Структура процесса формирования спецификации внешнего облика компонент ИВС

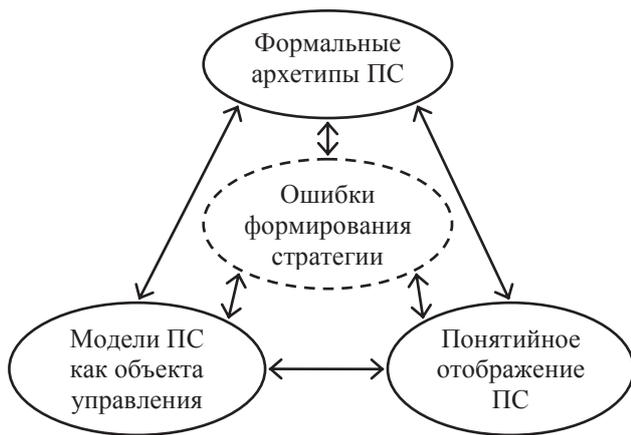


Рисунок 2 - Источники ошибок формирования стратегии урегулирования проблемной ситуации

В силу ограниченности знаний отдельных субъектов ни одна из персональных онтологических и когнитивных моделей не может претендовать на исчерпывающее понятийное отображение ПС. Под исчерпывающим понятийным отображением понимается выделение всех, в том числе латентных, предпосылок к возникновению ПС, знание и понимание причинно-следственных связей между разнесёнными в пространстве и во времени факторами, обуславливающими симптомы ПС.

Получение признаваемого всеми НА описания ПС (её понятийное отображение) предполагает реализацию итерационной

процедуры, основанной на решении двух взаимосвязанных задач [3]:

- формирование на основе персональных онтологических моделей НА онтологической модели ситуации;
- построение на основе персональных когнитивных моделей НА признаваемой всеми заинтересованными сторонами когнитивной модели ситуации.

В ходе реализации итерационной процедуры на основе дискурса НА происходит корректировка персональных онтологических и когнитивных моделей НА [20]. Методическую основу организации дискурса составляют положения теории конвергентного управления [2].

Когнитивная модель ПС служит фундаментом для последующего представления ПС в виде многосвязного объекта управления, что позволяет сформировать спецификации внешнего облика информационно-вычислительных компонент. На рисунке 3 представлена схема формирования понятийного пространства, стратегии и внешнего облика компонент ИВС.

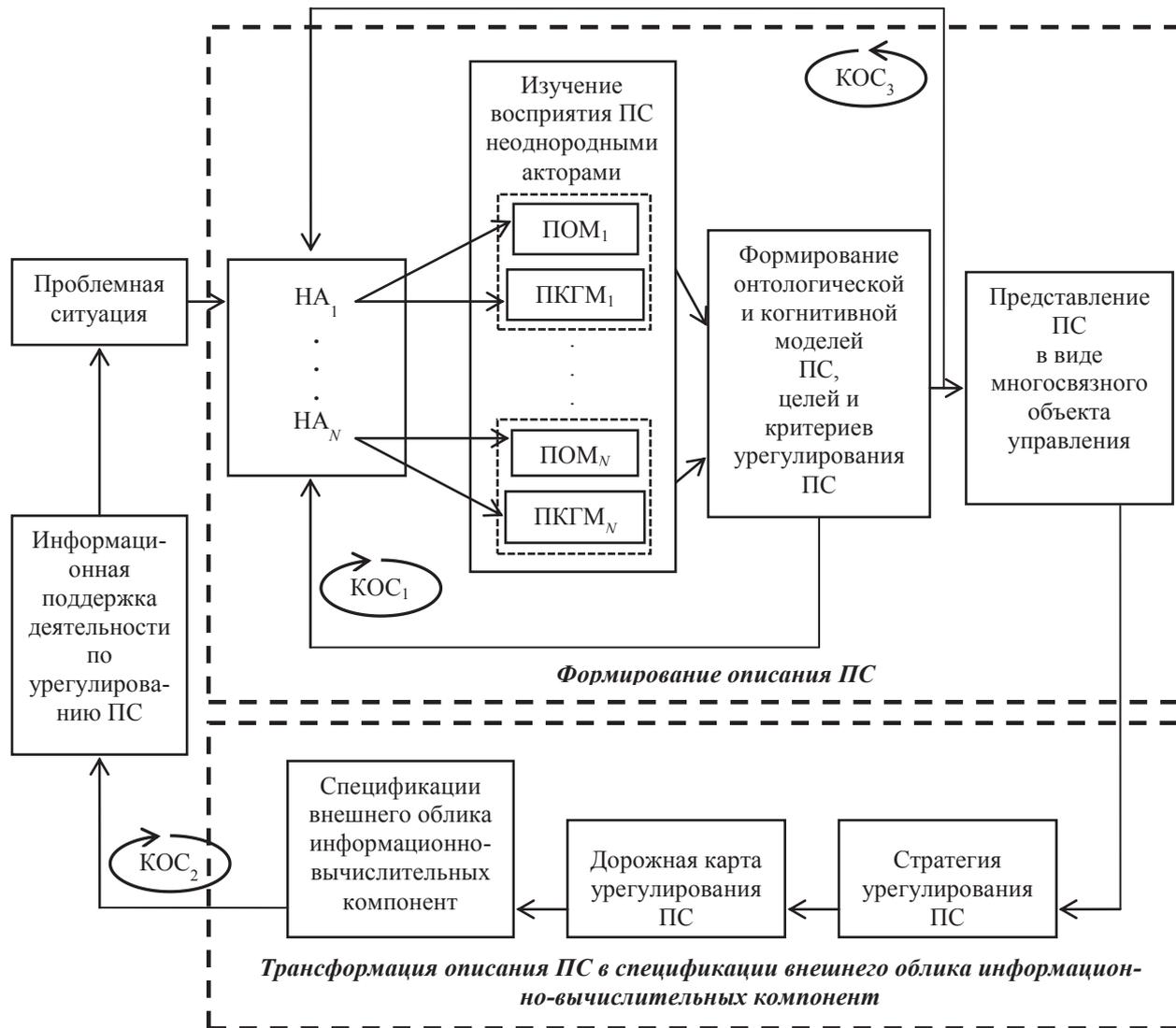
Урегулирование ПС осуществляется на двух уровнях: в пространстве состояний ПС и на уровне формирования нечётких целей и критериев её урегулирования. Контур обратной связи КОС₁ соответствует дискурсу, результатом которого является построение признаваемой всеми НА когнитивной модели ПС. Контур обратной связи КОС₂ соответствует деятельности, направленной на урегулирование ПС в пространстве её состояний. Контур обратной связи КОС₃ соответствует корректировке персональных онтологических и когнитивных моделей НА ПС, целей и критериев её урегулирования с учётом результатов деятельности, связанной с урегулированием ПС.

К числу причин возникновения дефектов в спецификации внешнего облика помимо ошибок, обусловленных мыслительной деятельностью субъектов, относится недостаток знаний об истинных причинах возникновения ПС, ограниченность понятийного отображения ПС, недостаточная потенциальность моделей, используемых для представления ПС в виде многосвязного объекта управления. В совокупности выделенные причины формируют системную составляющую дефектов в спецификации внешнего облика информационно-вычислительных компонент. Помимо этой составляющей дефектов существуют также ментальная и операционная составляющие [11, 12, 15, 16].

3 Когнитивная карта ошибок предпроектной стадии

Основой выработки консолидированного решения по урегулированию ПС является идентификация доступной для формализации части ПС [2]. Совокупность моделей этой ча-

сти ПС создаёт предпосылки для лучшего понимания содержания ПС, в том числе для представления ПС в виде многосвязного объекта управления [21, 22]. Исследование свойств объекта посредством моделирования создаёт основу для формирования консолидированного мнения о подходах к урегулированию ПС в условиях различия представлений о ценностях у НА [3, 4, 20].



НА_{*i*} – *i*-й неоднородный актер; ПОМ_{*i*}, ПКГМ_{*i*} – персональные онтологическая и когнитивная модели *i*-го неоднородного актора соответственно.

Рисунок 3 - Схема формирования понятийного пространства, стратегии и внешнего облика компонент информационно-вычислительной системы

Одним из инструментов для построения формальных моделей ПС является аппарат когнитивного моделирования. Когнитивные модели ориентированы на формализацию слабоструктурированных знаний субъектов (НА), вовлечённых в урегулирование ПС, и используются для оценки правильности восприятия НА содержания ПС. Первым этапом построения когнитивной модели является построение когнитивной карты.

На рисунке 4 приведена когнитивная карта ошибок, соответствующая структуре процесса, представленного на рисунке 1, и схеме формирования понятийного пространства, стратегии и внешнего облика компонент ИВС, представленной на рисунке 3.

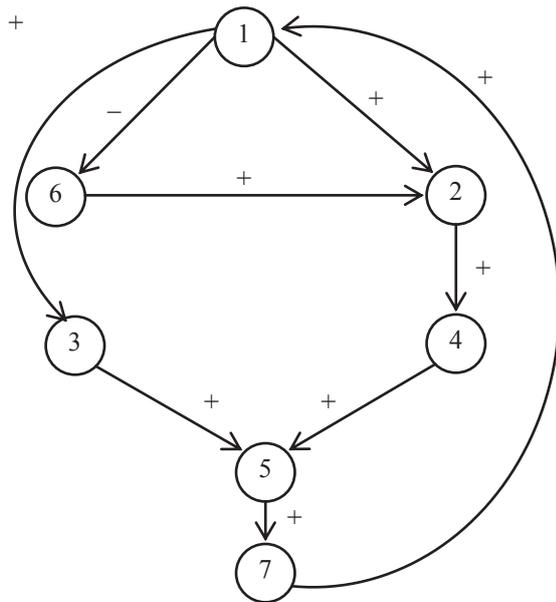


Рисунок 4 – Когнитивная карта ошибок предпроектной стадии

В таблице 1 дано описание концептов, а также соответствие концептов и этапов процесса предпроектной стадии. Цифры в скобках в колонке «ключевой фактор» соответствуют узлам графа, представленного на рисунке 4. Этапы процесса соответствуют основным этапам процесса предпроектной стадии, представленного на рисунке 1. В таблице 2 дано описание содержания отношений между концептами.

Ошибки в оценивании содержания ПС. Источниками этих ошибок являются возмущающие факторы внешней среды и различные персональные онтологические модели НА [3]. Результатом неопределённости ситуации, а также неодинакового восприятия одних и тех же событий разными субъектами является либо гиперболизация, либо недооценка сложности, масштабов и содержания ПС, обусловленных ею угроз и возможностей их парирования [20].

Таблица 1 – Ключевые факторы возникновения ошибок на предпроектной стадии

Ключевой фактор	Основные этапы процесса предпроектной стадии
<i>Ошибка в оценивании содержания ПС (1)</i>	Регистрация симптомов ПС
<i>Ошибки, совершаемые НА (2)</i>	Выявление НА, вовлечённых в урегулирование ПС; Изучение персональных онтологических и когнитивных моделей НА
<i>Недостаточность знаний для урегулирования ПС (3)</i>	Формирование онтологической модели и консолидированной когнитивной модели ПС; Представление ПС в виде многосвязного объекта управления
<i>Ошибки в стратегии урегулирования ПС (4)</i>	Формирование согласованной стратегии урегулирования ПС
<i>Ошибки в целях и планах урегулирования ПС (5)</i>	Формирование дорожной карты реализации стратегии
<i>Ошибки организации системы исследований на предпроектной стадии (6)</i>	Регистрация симптомов ПС; Выявление НА, вовлечённых в урегулирование ПС; Изучение персональных онтологических и когнитивных моделей НА; Формирование онтологической модели и консолидированной когнитивной модели ПС; Представление ПС в виде многосвязного объекта управления
<i>Ошибки в спецификациях внешнего облика и способах их реализации (7)</i>	Специфицирование внешнего облика информационно-вычислительных компонент цифровой экосреды

Ошибки, совершаемые НА, вовлечёнными в урегулирование ПС. Для таксономии ошибок, характерных для информационно-вычислительных компонент, целесообразно следовать классификации, используемой для ошибок, совершаемых людьми при управлении сложными техническими системами критического назначения (летательные аппараты, предприятия атомной энергетики и т.д.). Перечень ошибок, совершаемых при разработке программных продуктов и относящихся к выделенным классам, приведён в [14].

Недостаточность знаний для урегулирования ПС. Неполнота знаний об истоках ПС часто приводит к тому, что объектом управления становятся не её коренные причины, а наблюдаемые симптомы [9]. Перечень ошибок, возникающих при разработке программных продуктов и относящихся к этому классу, приведён в [14].

Таблица 2 – Описание отношений между концептами

Пары концептов	Содержание отношения
{1,2}	Стремление менеджеров переложить вину на исполнителей
{1,6}	Неверное понимание содержания ПС задаёт неверное направление проведения исследований
{1,3}	Неверное понимание ПС приводит к неверным технологиям получения новых знаний
{2,4}	Чем больше ошибок совершают люди, тем больше ошибок в стратегии урегулирования ПС
{3,5}	Недостаточность знаний обуславливает постановку ложных целей урегулирования ПС
{4,5}	Ошибки в стратегии урегулирования ПС приводят к ошибкам в планах
{5,7}	Ошибки в целях и планах урегулирования ПС приводят к дефектам в спецификации внешнего облика информационно-вычислительных компонент, что является источником ошибок и дефектов в организации и выполнении проекта
{7,1}	Низкое качество информационного обеспечения деятельности по урегулированию ПС является источником ошибок в оценивании содержания ПС
{6,2}	Ошибки в организации исследований увеличивают количество ошибок, совершаемых людьми

Ошибки в стратегии урегулирования ПС. Причинами этих ошибок являются:

- сложность, многомерность, многосвязность ПС как объекта управления;
- латентность ряда ключевых факторов, обуславливающих возникновение ПС;
- латентность намерений НА, вовлечённых в урегулирование ПС;
- стремление свести урегулирование уникальной ПС в условиях неопределённости относительно её содержания и недостаточности знаний обо всех ключевых факторах, её обусловивших, к использованию известных, ранее подтвердивших свою эффективность подходов и методов, при урегулировании ПС иного содержания.

Ошибки в целях и планах урегулирования ПС. Свойственная ПС неопределённость является причиной нечёткости целей урегулирования и критериев их достижения. В совокупности с ошибками стратегии это является источником ошибок в планах урегулирования ПС.

Ошибки организации системы исследований на предпроектной стадии. Эффективность решения задач, связанных с урегулированием ПС, в значительной степени зависит от того, насколько глубоко и полно она исследована и насколько полно и достоверно определены условия, в которых предполагается эксплуатация ИВС. Системные ошибки в организации изучения ПС, обусловленные её неопределённостью и ошибками менеджеров, являются источниками латентных дефектов, характерных для разных стадий жизненного цикла компонент ИВС. Ошибки в определении содержания ПС являются одним из критических факторов ошибок в определении требований к свойствам системы защиты от возникновения ошибок. Совершенствование системы защиты препятствует совершению ошибок людьми [17, 23]).

Ошибки в спецификациях требований и способах их реализации. Следствием проявления вышеперечисленных факторов возникновения ошибок является возникновение дефектов в спецификациях требований. Эти дефекты являются причинами ошибок и дефектов, возникающих как на последующих стадиях жизненного цикла программных продуктов, так и в организации процессов реализации программного продукта [16].

На графе, представленном на рисунке 4, можно выделить несколько контуров.

Контур, показанному на рисунке 5а, может быть дано следующее толкование: неверное понимание содержания ПС приводит к неверной организации её исследования (системным ошибкам). Однако низкая эффективность деятельности по урегулированию ПС вынуждает корректировать систему исследований, что приводит к уменьшению системных ошибок. Этот контур соответствует контуру КОС₃, выделенному на рисунке 3.

Контур, приведённому на рисунке 5б, можно дать следующее толкование. Недостаточная эффективность деятельности по урегулированию ПС и стремление менеджеров оправдать

неудачи за счёт ошибок, совершаемых исполнителями, но не просчётами, допущенными при организации системы исследований на предпроектной стадии, способствует повышению качества деятельности исполнителей. Этот контур является развивающим возможности системы управления качеством спецификации внешнего облика компонент ИВС за счёт уменьшения количества технических ошибок и более полного выполнения требований стандартов и руководств. Этот контур соответствует контуру КОС₂, выделенному на рисунке 3.

Развивающему контуру (в), показанному на рисунке 5в, можно дать следующее толкование: недостаточная эффективность деятельности по урегулированию ПС лишь за счёт уменьшения ошибок, обусловленных невнимательностью или забывчивостью исполнителей, вынуждает получать новые знания относительно причин возникновения ПС. Это приводит к уменьшению количества ошибок, обусловленных недостатком знаний. Можно утверждать, что выделенный контур является развивающим возможности системы управления качеством спецификации внешнего облика. Этот контур соответствует контуру КОС₁, выделенному на рисунке 3.

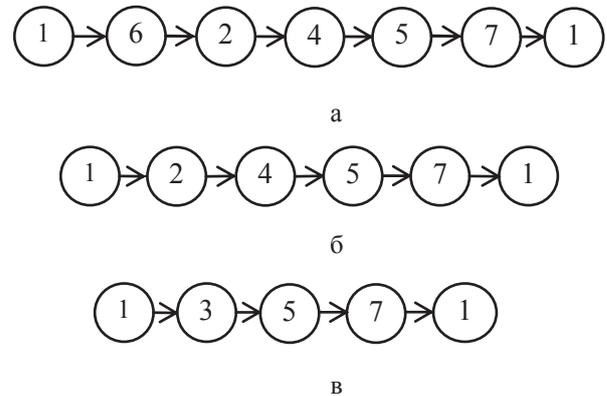


Рисунок 5 – Стабилизирующий (а) и развивающие (б), (в) контуры

Предложенной когнитивной модели может быть поставлен в соответствие архетип «пределы роста» [18, 19]. Возможности предупреждения возникновения ошибок и устранения дефектов определяются особенностями организации исследований ПС, а также менталитетом субъектов, привлекаемых для формирования спецификации требований. Предложенная когнитивная карта позволяет прийти к следующим выводам.

- Должен быть баланс в распределении усилий по борьбе с ошибками разной природы. Невозможно обеспечить высокое качество спецификаций компонент ИВС лишь за счёт предотвращения ошибок, допускаемых людьми без совершенствования организации исследования ПС.
- Дефекты являются фактором, снижающим ценность компонент ИВС как одного из инструментов урегулирования ПС, но также являются ключевым фактором совершенствования системы изучения ПС за счёт получения новых знаний, совершенствования методической и инструментальной базы предотвращения возникновения, локализации и устранения дефектов.

4 Анализ адекватности когнитивной карты

Анализ адекватности является необходимым условием разработки формальных моделей сложных систем [24]. Основу анализа адекватности структурной модели, представленной на рисунке 4, составляет проведение расчётов на основе таких исходных данных, для которых заранее известны результаты.

Технологическую основу расчётов составляет метод, описанный в [25]. Исходный знаково-ориентированный граф представляется в виде квадратной матрицы $\|H\|$, в которой «1» соответствует наличию положительной связи между узлами, «-1» - наличию отрицательной связи, «0» - отсутствию связи. Формируется вектор состояния узлов графа, причём если имеет место положительное изменение состояния узла, компоненту вектора присваивается «1», если состояние не изменяется – «0», если ухудшается – «-1». Основу формирования вектора

состояния на $(k+1)$ -й итерации составляет соотношение: $\vec{C}_{(k+1)} = \vec{C}_{(k)} \cdot \|H\|$, где k – номер итерации, $k = \overline{0;n}$. При формировании значений компонент вектора состояния пользуются следующими правилами [25]:

- Если в ходе расчётов значение компонента вектора состояния превышает значение «+1», компоненту присваивается значение «единица».
- Если в ходе расчётов значение компонента вектора состояния меньше, чем «-1», компоненту присваивается значение «минус единица».
- Если в ходе расчётов значение компонента вектора состояния принимает значение «нуль», компоненту присваивается значение «нуль».
- Значение компонента вектора состояния, на которое на нулевой итерации оказывается воздействие, на всех последующих итерациях остаётся неизменным.
- Если вектор состояния на $(k+1)$ -й итерации совпадает с вектором состояния на k -й итерации, считается, что система перешла в устойчивое состояние и расчёты завершаются. Если вектор состояния на $(k+1)$ -й итерации совпадает с вектором на k -й итерации, но векторы на $(k+m)$ -х итерациях $m = \overline{1;l-1}$ отличаются от вектора на k -й итерации и различаются между собой, считается, что система перешла в колебательный режим с периодом l . На этом расчёты заканчиваются.

Граф, показанный на рисунке 4, описывается матрицей, приведённой на рисунке 6.

$$\|H\| = \begin{array}{|c|c|c|c|c|c|c|c|} \hline 0 & 1 & 1 & 0 & 0 & -1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

Рисунок 6 – Матрица, описывающая граф на рисунке 4

Случаю воздействия с целью улучшения состояния узла 6 (уменьшение ошибок в организации системы исследований на предпроектной стадии) соответствует вектор состояния $\vec{c}_0 : \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0}$. Следуя описанной расчётной схеме, на шестой итерации вектор состояния системы приобретает вид $\vec{c}_6 : \boxed{1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1}$, который на последующей итерации не изменяется. Интерпретация полученного результата: совершенствование организации системы исследований препятствует возникновению ошибок разной природы, что ускоряет урегулирование ПС.

Воздействию с целью улучшения состояния узла 2 (ошибки, совершаемые НА, вовлечёнными в урегулирование ПС) соответствует вектор состояния вида $\vec{c}_0 : \boxed{0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0}$. На пятой итерации вектор состояния \vec{c}_5 приобретает вид $\boxed{1 \ 1 \ 1 \ 1 \ 1 \ 1 \ -1 \ 1}$, который на последующих шагах не изменяется. Полученный результат интерпретируется следующим образом. Уменьшение ошибок, совершаемых людьми, в итоге повышает качество спецификаций внешнего облика компонент ИВС, но ухудшает мотивацию к поиску пробелов в организации системы исследований ПС, приводящих к возникновению ошибок, обусловленных недостатком знаний исполнителя.

Случаю воздействия на узел 7 (устранение дефектов непосредственно в спецификации) соответствует вектор состояния $\vec{c}_0 : \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1}$. На четвёртой итерации вектор состояния \vec{c}_4 принимает вид $\boxed{1 \ 1 \ 1 \ 1 \ 1 \ 1 \ -1 \ 1}$. Интерпретировать полученный результат можно следующим образом. Стремление улучшить качество спецификации стимулирует выявление как непосредственных причин дефектов, так и коренных причин ошибок, допущенных исполнителями, и путей их распространения [9]. Однако при этом вне зоны внимания оказывается организация системы исследований ПС.

Выполненный анализ результатов моделирования позволяет сделать заключение о том, что они не противоречат содержанию ситуаций, имеющих место при разработке спецификаций внешнего облика компонент ИВС.

Заключение

Критически важным фактором обеспечения функциональной безопасности ИВС является создание системы предупреждения возникновения ошибок на разных стадиях жизненного цикла компонент. В рамках известных подходов к обеспечению функциональной безопасности в центре внимания оказывались задачи, связанные с поэтапным преобразованием формального описания внешнего облика компонент ИВС в программные и аппаратные компоненты, обладающие требуемыми потребительскими свойствами. В работе предложена совокупность структурных моделей, концептуальную основу которых составляет постулат, что дефекты в спецификациях внешнего облика обусловлены не только особенностями когнитивных моделей субъектов, причастных к формированию требований, но и организацией формирования требований. Предложена когнитивная карта ошибок предпроектной стадии и выполнена её верификация. Показано, что существует предел возможностей по предупреждению дефектов, обусловленный как неопределённостью ПС, так и компетентностью разработчиков требований.

Аппарат нечётких когнитивных карт, параметрических векторных функциональных графов можно определить как инструментальную основу дальнейших исследований.

Благодарности

Результаты получены при поддержке гранта № 19-08-00177 «Методологические, теоретические и модельные основы управления функциональной безопасностью аппаратно-программных комплексов в составе распределённых сложных технических систем».

Список источников

- [1] *Schuh, G.* Industrie 4.0 Maturity Index Managing the Digital Transformation of Companies / G. Schuh, R. Anderl, J. Gausemeier, M. Hompel, W. Wahlster // Acatech STUDY, 2018. – 60 p.
- [2] *Райков, А.Н.* Конвергентное управление и поддержка решений / А.Н. Райков – М.: Издательство ИКАР, 2009. – 245 с.
- [3] *Виттих, В.А.* Введение в теорию интересубъективного управления / В.А. Виттих. – Самара: Самарский научный центр РАН, 2013. – 64 с.
- [4] *Виттих, В.А.* Принятие решений на основе консенсуса с применением мультиагентных технологий / В.А. Виттих, Т.В. Моисеева, П.О. Скобелев // Онтология проектирования. – 2013. – №2. – С. 20-25.
- [5] CHAOS Report. The Standish Group International, Inc., 2018, 68 p. – <https://www.standishgroup.com/news/37>.
- [6] *Parnas, D.L.* The Role of Inspection in Software Quality Assurance / D.L. Parnas, M. Lawford // IEEE Transactions on Software Engineering, 2003. – Vol. 29. – Issue 8. – P. 674-676.
- [7] *Kantorowitz, E.* The performance of the N-fold requirement inspection method / E. Kantorowitz, A. Guttman, L. Arzi // Requirements Engineering. – 1997. – Vol.2. – №3. – P. 152-164.
- [8] *Shull, F.* How Perspective-Based Reading Can Improve Requirements Inspections / F. Shull, I. Rus, V. Basili // Computer, IEEE, 2000. – 33(7). – P.73-79.
- [9] *Андерсен, Б.* Анализ основной причины. Упрощенные инструменты и методы. / Б. Андерсен, Т. Фагерхруд. // ASQ Quality Press, Милуоки, Висконсин, 1999. – 155 с.
- [10] *Lanubile, F.* Experimenting with error abstraction in requirements documents / F. Lanubile, F. Shull, V. Basili // Proc. of the 5th Int. Symposium on Software Metrics – Bethesda, Maryland, 1998. – P. 114–121.
- [11] *Carver, J.C.* Defect prevention in requirements using human error information: An empirical study. / J.C. Carver, W. Hu, V. Anu, G. Walia, G. Bradshaw // Requirements Engineering: Foundation for Software Quality - 23rd International Working Conference, REFSQ 2017. – P. 61-76. – https://doi.org/10.1007/978-3-319-54045-0_5.

- [12] **Reason, J.** Revising the «Swiss Cheese» Model of Accidents. / J. Reason, E. Hollnagel, J. Paries // European organization for Safety of air navigation, EEC Note. – 2006. – No.13/06. – 25 p.
- [13] **Shappell, S.A.** The Human Factors Analysis and Classification System – HFACS / S.A. Shappell, D.A. Weigmann // Final Report, U.S. Department of Transportation, Federal Aviation Administration, 2000. – 15 p.
- [14] **Rasmussen, J.** Human Errors. A taxonomy for description human malfunction in industrial installations / J. Rasmussen // J. Occup. Accid. 1982. – Vol. 4. – no 2-4. – P. 311-333.
- [15] **Anu, V.K.** Using Human Error Models to Improve the Quality of Software Requirements / V.K. Anu // A Dissertation Submitted to the Graduate Faculty of the North Dakota State University of Agriculture and Applied Science. – Fargo, North Dakota, 2018. – 119 p.
- [16] **Huang, F.** Software defect prevention based on human error theories / F. Huang, B. Liu // Chinese Journal of Aeronautics, 2017. – №30 (3). – P. 1054-1070.
- [17] ESA PSS-05-11. Guide to software quality assurance, 1995. – 55 p.
- [18] **Медоуз, Д.** Азбука системного мышления / Д. Медоуз. – М.: Бином. Лаборатория знаний, 2010. – 343 с.
- [19] **Сенге, П.** Пятая дисциплина. Искусство и практика обучающейся организации / П. Сенге. – М.: Манн, Иванов и Фербер, 2018. – 496 с.
- [20] **Виттих, В.А.** Неоднородный актор и повседневность как ключевые понятия эвергетики / В.А. Виттих. – Самара: Самарский научный центр РАН, 2014. – 12 с.
- [21] **Case, D.** Fuzzy Cognitive Map to Model Project Management Problems. / D. Case, C. Stylios // Proceedings of 35th Annual Conference of the North American Fuzzy Information Processing Society NAFIPS'2016, October 31-November 4, 2016, El Paso, USA. - <https://ieeexplore.ieee.org/document/7851612> (дата обращения 29.09.2019)
- [22] **Kleppe, A.G.** MDA Explained: The Model Driven Architecture: Practice and Promise / A. G. Kleppe, J. Warmer, W. Bast // Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2003. – 170 p.
- [23] TRIPOD BETA. Guidance on using Tripod Beta in the investigation and analysis of incidents, accidents and business losses. – London, Energy Institute, 2015. – 19 p.
- [24] **Трусов, П.В.** Введение в математическое моделирование / П.В. Трусов, В.Н. Ашихмин, М.Б. Гитман, И.Э. Келлер, О.Б. Наймарк, В.Ю. Столбов, П.Г. Фрик // Учебное пособие. – М.: Логос, 2005. 440 с.
- [25] **Pelaez, C.E.** Using fuzzy cognitive maps as a system model for failure modes and effects analysis / C.E. Pelaez, J.B. Bowles // Information Sciences, 1996. – P. 177-199.

Сведения об авторах



Гвоздев Владимир Ефимович, 1956 г. рождения. Окончил Уфимский авиационный институт им. Орджоникидзе в 1973 г., д.т.н. (2000). Заведующий кафедрой технической кибернетики Уфимского государственного авиационного технического университета (УГАТУ). В списке научных трудов более 300 работ в области прикладного статистического анализа, информационной поддержки управления программными системами, информационной поддержки управления состоянием территориальных систем. AuthorID (РИНЦ): 174520. Author ID (Scopus): 7101700484. wega55@mail.ru



Бежаева Оксана Яковлевна, 1977 г. рождения. Окончила УГАТУ в 2000 г., к.т.н. (2004). Доцент кафедры технической кибернетики УГАТУ. В списке научных трудов более 100 работ в области разработки моделей и программного обеспечения сложных производственных и социально-экономических систем. AuthorID (РИНЦ): 271220. obezhaeva@gmail.com



Насырова Рима Айратовна, 1995 г. рождения. Магистрант кафедры технической кибернетики УГАТУ. В списке научных трудов более 20 работ в области надежности функционирования программно-аппаратных комплексов. AuthorID (РИНЦ): 979980. nasyrova.rima@yandex.ru

Поступила в редакцию 15.11.2019. Принята к публикации 11.12.2019.

Models of errors at the pre-design stage of the development of information and computing systems components

V.E. Gvozdev, O.Y. Bezhaeva, R.A. Nasyrova

Ufa State Aviation Technical University, Ufa, Russia

Abstract

The paper gives an overview of approaches to the study of errors in specifications to requirements, identifies the system-forming factors of errors of the pre-design stage, proposes a cognitive map of the errors of the pre-design stage, and analyzes its adequacy. The importance of the further development of the construction of formal models of processes during the implementation of which errors occur is noted. It was established that the known archetype “growth limits” can be associated with the processes of preventing defects in the specifications of the external appearance, which implies that there is a limit to the ability to prevent defects, due to both the uncertainty of the problem situation and the competence of the developers of the requirements. Using a set of structural models of the pre-design stage of processes of the development of information and computing systems components, it was shown that defects in the specifications of the external appearance are caused by both the features of cognitive models of the subjects involved in the formation of requirements and the organization of the formation of requirements.

Key words: *pre-design stage, specification of the external appearance, cognitive error map, error factors, structural models of processes, functional graph.*

Citation: *Gvozdev VE, Bezhaeva OY, Nasyrova RA. Models of errors at the pre-design stage of the development of information and computing systems components [In Russian]. Ontology of designing. 2020; 10(1): 73-86. DOI: 10.18287/2223-9537-2020-10-1-73-86.*

Acknowledgment: The work supported by grant of Russian Foundation for basic research (project N 19-08-00177).

List of figures and tables

- Figure 1 - Structure of the computational components external shape formation process
- Figure 2 - Error sources in strategy formation of a problem situation settlement
- Figure 3 - Schema of conceptual space, strategy and external shape formation of computational components
- Figure 4 – Cognitive map of errors at the pre-design stage
- Figure 5 – Stabilizing (a) and Developing (б, в) contours
- Figure 6 - Matrix that describes the graph in figure 4
- Table 1 - Error occurrence key factors at a pre-design stage
- Table 2 - Description of relations between the concepts

References

- [1] *Schuh G, Anderl R, Gausemeier J, Hompel M, Wahlster W.* Industrie 4.0 Maturity Index Managing the Digital Transformation of Companies. Acatech STUDY; 2018.
- [2] *Raikov AN.* Converged Management and Solution Support [In Russian]. – Moscow: IKAR publ.; 2009.
- [3] *Vittikh VA.* Introduction to Intersubjective Management Theory [In Russian]. – Samara: Samara Scientific Center of RAS; 2013.
- [4] *Vittikh VA, Moiseeva TV, Skobelev PO.* Consensus decision making using multi-agent technology [In Russian]. *Ontology of designing.* 2013; №2: 20-25.
- [5] CHAOS Report. The Standish Group International, Inc., 2018. <https://www.standishgroup.com/news/37>.
- [6] *Parnas DL, Lawford M.* The Role of Inspection in Software Quality Assurance. *IEEE Transactions on Software Engineering.* 2003; 29(8): 674-676.
- [7] *Kantorowitz E, Guttman A, Arzi L.* The performance of the N-fold requirement inspection method. *Requirements Engineering.* 1997; 2(3): 152-164.
- [8] *Shull F, Rus I, Basili V.* How Perspective-Based Reading Can Improve Requirements Inspections. *Computer, IEEE;* 2000: 73-79.

- [9] **Andersen B, Fagerhoud T.** Root Cause Analysis - Simplified Tools and Techniques. Second Edition. ASQ Quality Press, Milwaukee Wisconsin, 2006. 224 p.
- [10] **Lanubile F, Shull F, Basili V.** Experimenting with error abstraction in requirements documents: Proc. of the 5th Int. Symposium on Software Metrics (Bethesda, Maryland, 1998); 1998: 114–121.
- [11] **Carver JC, Hu W, Anu V, Walia G, Bradshaw G.** Defect prevention in requirements using human error information: An empirical study. Requirements Engineering: Foundation for Software Quality - 23rd International Working Conference, REFSQ; 2017: P. 61-76. https://doi.org/10.1007/978-3-319-54045-0_5
- [12] **Reason J, Hollnagel E, Paries J.** Revising the «Swiss Cheese» Model of Accidents. European organization for Safety of air navigation, EEC Note, 2006; 13/06.
- [13] **Shappell SA, Weigmann DA.** The Human Factors Analysis and Classification System – HFACS. Final Report, U.S. Department of Transportation, Federal Aviation Administration; 2000.
- [14] **Rasmussen J.** Human Errors. A taxonomy for description human malfunction in industrial installations. Occup. Accid. 1982; 4(2-4): 311-333.
- [15] **Anu VK.** Using Human Error Models to Improve the Quality of Software Requirements. A Dissertation Submitted to the Graduate Faculty of the North Dakota State University of Agriculture and Applied Science. Fargo, North Dakota; 2018.
- [16] **Huang F, Liu B.** Software defect prevention based on human error theories. Chinese Journal of Aeronautics, 2017; 30 (3): 1054-1070.
- [17] ESA PSS-05-11. Guide to software quality assurance, 1995.
- [18] **Meadows D.** The ABC of System Thinking [In Russian]. – Moscow: Binom publ.; 2010.
- [19] **Senge P.** The Fifth Discipline: The Art and Practice of the Learning Organization [In Russian]. – Moscow: MIF publ.; 2006.
- [20] **Vittikh VA.** Heterogeneous actor and everyday life as key concepts of evergetics [In Russian]. – Samara: Samara Scientific Center of RAS; 2014.
- [21] **Case D, Stylios C.** Fuzzy Cognitive Map to Model Project Management Problems. Proceedings of 35th Annual Conference of the North American Fuzzy Information Processing Society NAFIPS'2016 (October 31-November 4, 2016, El Paso, USA). <https://ieeexplore.ieee.org/document/7851612>
- [22] **Kleppe AG, Warmer J, Bast W.** MDA Explained: The Model Driven Architecture: Practice and Promise. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2003.
- [23] TRIPOD BETA. Guidance on using Tripod Beta in the investigation and analysis of incidents, accidents and business losses. London, Energy Institute, 2015.
- [24] **Trusov PV, Ashihmin VN, Gitman MB, Keller IE, Naimark OB, Stolbov VU, Frik PG.** Introduction to Mathematical Modeling [In Russian]. – Moscow: Logos publ.; 2005.
- [25] **Pelaez CE, Bowles JB.** Using fuzzy cognitive maps as a system model for failure modes and effects analysis. Information Sciences; 1996: 177-199.

About the authors

Vladimir Efimovich Gvozdev (b. 1956) graduated from the Ufa Aviation Institute (Ufa-city) in 1973, D. Sc. Eng. (2000). He is Head of Department at Ufa State Aviation Technical University (Department of Technical Cybernetics). He is co-author of more than 300 scientific publications in the field of applied statistical analysis, information support for managing software systems, and information support for managing the state of territorial systems. AuthorID (RCI): 174520. Author ID (Scopus): 7101700484. wega55@mail.ru

Oksana Yakovlevna Bezhaeva (b. 1977) graduated from the Ufa State Aviation Technical University in 2000, PhD. (2004). She is Assistant professor at Ufa State Aviation Technical University (Department of Technical Cybernetics). She is co-author of more than 100 scientific publications in the field of development of models and software for complex production and socio-economic systems. AuthorID (RCI): 271220. obezhaeva@gmail.com

Rima Airatovna Nasyrova (b. 1995). She is undergraduate student at Ufa State Aviation Technical University (Department of Technical Cybernetics). She is co-author of more than 20 scientific publications in the field of reliability of functioning of hardware and software systems. AuthorID (RCI): 979980. nasyrova.rima@yandex.ru

Received November 15, 2019. Accepted December 11, 2019.

ИНЖИНИРИНГ ОНТОЛОГИЙ

УДК 004.82:003.6

DOI: 10.18287/2223-9537-2020-10-1-87-99

Визуально-аналитическое мышление и интеллект-карты в онтологическом инжиниринге**Т.А. Гаврилова, Э.В. Страхович***Высшая школа менеджмента Санкт-Петербургского государственного университета,
Санкт-Петербург, Россия***Аннотация**

Статья посвящена вопросам практического применения принципов визуально-аналитического мышления в задачах структурирования знаний при разработке онтологий. Под визуально-аналитическим мышлением понимается применение методологий, использующих различные виды диаграмм для представления идей, концептов, отношений и процессов. Из известных практически используемых видов диаграмм выбраны интеллект-карты как наиболее распространенный, удобный и простой метод корректного формирования и проектирования онтологий сложных предметных областей. Интеллект-карты отражают иерархические связи между понятиями и позволяют достаточно глубоко отображать особенности и закономерности предметных областей с их спецификой отношений. Бьюзен сформулировал идею интеллект-карт в качестве компактного средства организации конспектов, которое впоследствии было доведено до программной реализации и получило широкое распространение в различных областях образования, научных исследований и бизнеса. В статье рассматриваются основные принципы формирования таких карт и анализируются типичные ошибки разработчиков. Впервые предлагается классификация ошибок с учётом синтаксических, семантических и прагматических аспектов. Приводится разбор наиболее частых ошибок, связанных с нарушением правил «хорошего обобщения» и «разумного минимализма». Статья обобщает десятилетний опыт обучения и тренинга навыков визуально-аналитического мышления на программах Executive MBA и на корпоративных тренингах и может быть интересна разработчикам интеллектуальных систем и систем управления знаниями.

Ключевые слова: онтологический инжиниринг, интеллект-карты, визуально-аналитическое мышление, визуальные модели, управление знаниями.

Цитирование: Гаврилова, Т.А. Визуально-аналитическое мышление и интеллект-карты в онтологическом инжиниринге / Т.А. Гаврилова, Э.В. Страхович // Онтология проектирования. – 2020. – Т. 10, №1(35). – С.87-99. – DOI: 10.18287/2223-9537-2020-10-1-87-99.

«Целью визуализации являются не картинки,
а проникновение в суть»
Бен Шнайдерман, 1999

Введение

Интерес к разработке онтологий плавно нарастает как со стороны разработчиков интеллектуальных систем, так и со стороны бизнес-аналитиков. В первое десятилетие истории онтологий после пионерских работ Грубера и Гуарино усилия исследователей в основном были направлены на разработку технологических инструментов и примеров [1, 2]. Последние 5 лет набирает силу формирование современной теории онтологического инжиниринга [3–7]. Однако разработка практических онтологий в производстве, проектировании и менеджменте,

особенно ИТ-менеджменте, остаётся скорее на уровне «искусства». Проектирование онтологий довольно слабо освещено в литературе по онтологическому инжинирингу, большинство авторов сосредоточили свои усилия на формализации и моделировании как таковом [8–11]. Известные методологии и технологии (например, NEON и METHONTOLY [12, 13]) также ориентированы на организационные и технологические аспекты, не касаясь проблемы формирования понятий, их уровня абстракции, баланса отношений и других вопросов семантики. В то время как онтологии стали стандартом де-факто в области разработки баз знаний, процессы извлечения и особенно структурирования знаний по-прежнему остаются некоторым «белым пятном» в современной литературе по инженерии знаний. Можно сказать, что в семиотическом треугольнике «синтаксис-семантика-прагматика» пока главенствуют синтаксис и прагматика.

Специалистам-разработчикам необходим простой и наглядный инструмент для осмысления и понимания особенностей понятий предметной области (ПрО), глубинных отношений между концептами и классами ПрО. Таким инструментом может послужить поле знаний (в отечественной литературе) или ментальная модель (в зарубежной литературе).

Под полем знаний [14] понимается условное неформализованное описание основных объектов ПрО, их атрибутов и отношений между ними, выявленных из некоторого источника знаний, в виде графа, диаграммы, таблицы или текста. Фактически это «скелет» или прообраз базы знаний. Обычно поле знаний носит иерархическую структуру, что обусловлено иерархичностью понятийной структуры знаний человека [15, 16].

Термин «ментальная модель» был введён в психологии познания для отображения процессов осмысления действительности в виде некоторых упрощённых моделей. Впервые понятие было предложено в работах Джонсон-Лэйрда [17] и Бирн [18].

Визуальный подход к созданию ментальных моделей применялся с самого начала, поскольку с древних времён визуализация считалась мощным инструментами познания, т.е. средством, предназначенным для организации и облегчения процесса познания. Инструментом визуализации ментальных моделей служили различные виды концептуальных карт и графов, семантические сети и когнитивные карты, и, конечно, *интеллект-карты* (и-карты, от англ. *mind maps*). Особенно широко и-карты применяются в менеджменте и в управлении знаниями как один из наиболее привлекательных и простых способов отображения профессиональных знаний, бизнес-идей, проектов, функций или любых других сложных понятийных структур.

Однако и-карты отражают в основном иерархические связи и не позволяют глубоко анализировать закономерности ПрО с их спецификой отношений. Для отображения разнородных отношений чаще применяются концептуальные графы (к-графы, семантические сети, концептуальные карты). К-графы были предложены Новаком в начале 70-х годов [19] при изучении детского мышления и формирования первых научных понятий. В этих исследованиях были использованы идеи Дэвида Асубеля [20] о формировании понятийного мышления. К-графы оказались эффективным инструментом отображения понятийной системы человека. Они обычно состоят из узлов и направленных поименованных отношений или связей, соединяющих узлы. Связи могут быть различного типа, например, «является», «имеет свойство» и т.п. Узлы и связи имеют универсальный характер для некоторого класса понятий ПрО. Поэтому любая разработка к-графа подразумевает анализ структурных взаимодействий между отдельными понятиями ПрО.

И-карты и концептуальные карты можно использовать для прототипирования и разработки онтологий на начальном этапе. Статья посвящена проблеме создания таких карт с позиции последующей разработки онтологических моделей.

1 О визуально-аналитическом мышлении

В настоящее время рост информационных потоков привел к появлению новых направлений исследований, связанных с визуализацией (инфографика, когнитивная графика, виртуальная реальность, компьютерная графика и т.д.) [21]. Основная задача этих новых видов графической подачи информации – сделать сложные и громоздкие понятия более простыми и наглядными для восприятия и обучения, фактически – это сжатие и упрощение информации. Так, инфографика показывает взаимосвязь между единицей информации, количественным значением и общим положением данной информации в системе. Фактически, это графическая интерпретация информации (т.е. данных и знаний), позволяющая сжимать информацию, освобождая её от словесного «балласта», и взаимосвязывать различные типы информации, выявляя закономерности.

Визуализация информации «вынуждена отвечать» на растущие объёмы общедоступных данных, как когнитивный фильтр, как «увеличительная линза» понимания, и она, по возможности, не должна добавлять «шума» к информационному потоку. Не всякую визуализацию следует считать положительным шагом вперёд. В контексте визуализации информации простая передача данных в визуальной форме, не проливающая свет на изображаемый предмет, или — что даже хуже — усложняющая его, может считаться только неудачной [22].

Вся история науки и образования подчёркивает важность визуализации и основана на использовании диаграмм, рисунков, схем и эскизов, без которых некоторые науки были бы невозможны (астрономия, геометрия, география и др.). В любой деятельности и обучении человеческое сознание использует несколько механизмов мышления [23]. Исторически первым в процессе эволюции являлся наглядно-действенный механизм. Несколько позже появился наглядно-образный механизм мышления, обеспечивающий работу с изображениями, образами и представлениями об этих образах. Его называют также геометрическим, интуитивным и т.п. Затем появился механизм, позволяющий работать с абстрактными цепочками символов, с текстами и т.п. Этот механизм мышления обычно называют абстрактным, логическим или символическим¹. Следует отметить терминологическое разнообразие, противоречивость и нечёткость определений в данной области исследований.

Под визуально-аналитическим мышлением в статье понимается набор методологий, использующих различные виды диаграмм для представления идей, понятий, отношений и процессов.

Традиционно в визуальном мышлении выделяют три фазы:

- 1) формирование набросков идей (понятий, процессов);
- 2) их осмысление;
- 3) создание картинки, образа, диаграммы [24].

Эти фазы подробно рассмотрены на примере разработки и-карт как эскизов онтологических моделей.

2 Формирование и-карт

И-карты — это иерархические диаграммы, используемые для представления идей, проектов, заданий, которые связаны с центральным ключевым понятием и радиально организованы вокруг него [25, 26]. В некоторых источниках и-карту часто называют графическим выражением процесса радиантного или параллельного мышления в отличие от последователь-

¹ Авторы не вторгаются в вопросы когнитивной психологии, а лишь пытаются конструктивно применять имеющиеся научные результаты. *Прим. авт.*

ного. Правила формирования и-карт по Бьюзену представлены на примере абстрактной и-карты на рисунке 1 (подробнее в [25, 27]):

- главное понятие представлено в центральном образе (A);
- основные понятия и/или процессы, связанные с главным, расходятся от центрального образа в виде ветвей, которые именуются ключевыми словами и образами (B, C, D, E);
- ветви формируют связную иерархическую структуру.

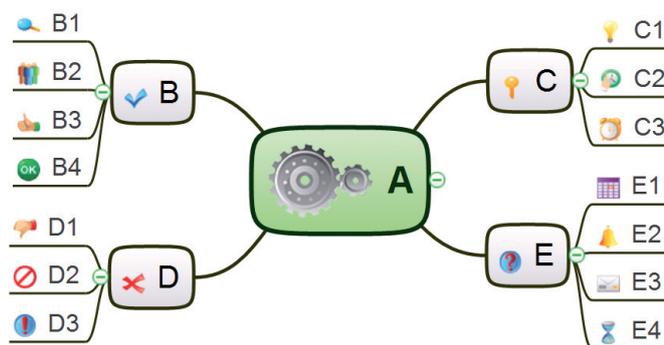


Рисунок 1 — Пример абстрактной и-карты

Идея и-карт (по Бьюзену) заключа-

ется в использовании и совмещении функции логического и образного мышления для достижения целостного и наглядного представления рассматриваемого понятия. Фактически, это переход от последовательного (текстового) изложения к параллельному (сетевому). Бьюзен задекларировал три правила для и-карт – использование разных шрифтов (уменьшение их от центра), разных цветов и графических образов для увеличения наглядности. Предложенный в качестве компактного средства организации конспектов метод и-карт стал использоваться как мощное орудие мышления применительно к научной работе, инновациям, бизнес-идеям, политическим дискуссиям, педагогике.

2.1 Нейробазис

Идея и-карт имеет некоторый нейрофизиологический базис. Человеческий мозг обладает выраженной сетевой структурой [28]. Естественные нейронные сети мозга содержат миллиарды нейронов, каждый из которых может иметь десятки тысяч связей. И хотя природа их взаимодействия исследована далеко не полностью, так называемый «радиантный» характер передачи возбуждения от центра на периферию доказан.

И-карты применяются, с одной стороны, для генерации, визуализации, структурирования и классификации идей, с другой стороны, для облегчения (ускорения) процесса обучения [29], процесса разрешения проблемы или принятия решения [30–32], а также при составлении и написании различных документов [33]. Их можно использовать в обучении как для объяснения, так и для проверки усвоенного материала. В этом случае структура и-карты служит критерием понимания изучаемого предмета. Применение и-карт может быть полезно при выполнении групповых проектов, анализе кейсов и т.д. Благодаря упрощённому и ясному изображению идей в графическом виде и-карты иногда используют для активизации «мозгового штурма», при решении организационных задач и осуществлении планирования [27, 34]. В и-карте можно подразумевать любой тип связи между понятиями, что может породить хаос и избыточность полученных диаграмм.

2.2 Когнитивные проблемы

Далее рассмотрены когнитивные проблемы разработки и-карт как с позиции разработчика, так и с позиции пользователя карты. Для разработчика важно справиться с проблемой материализации собственных представлений, их адекватной и однозначной репрезентации в карте. Для пользователя важно правильно прочесть и интерпретировать карту, даже если она вызывает когнитивный диссонанс.

О стратегии. Существуют две стратегии в разработке и-карт – дедуктивная (сверху вниз) и индуктивная (снизу вверх). Первая связана с анализом и декомпозицией, вторая – с синтезом и конструированием понятий.

О критериях общности. Серьезная методологическая проблема связана с выбором критерия декомпозиции или синтеза.

Для дедуктивной стратегии центральное понятие разбивается последовательно на каждом из уровней ветвления карты по выбранному критерию или признаку. Например, можно при построении карты современного программного обеспечения использовать на первом уровне критерий операционной системы (Windows или Mac OS), а можно страны производителя. Соответственно, карты получатся различными. Можно предложить первое правило хорошей декомпозиции: «Принцип (признак) декомпозиции должен быть понятным и прозрачным».

Для индуктивной стратегии можно также выбирать разные признаки. Так, набор концептов (Париж, Рим, Токио, Москва, Санкт-Петербург, Бомбей) можно собирать в категории десятком вариантов группировки в зависимости от выбранных атрибутов. Например, разбиение $X = \{(Париж, Москва, Петербург, Рим), (Токио, Бомбей)\}$ соответствует делению по понятному критерию «Европа – Азия», а разбиение $Y = \{(Париж, Рим, Петербург), (Москва, Дели, Бомбей)\}$ – не совсем. При этом разбиение Y соответствует делению по принципу «наличие величайших коллекций мирового искусства».

Другая проблема связана с системностью мышления разработчика, позволяющей ему выделять понятия одного уровня абстракции. Эта характеристика связана с индивидуальным когнитивным стилем аналитика, чему посвящены работы [35, 36].

Особенно это важно тогда, когда и-карты используются для проектирования онтологий. В этом случае следует накладывать ряд системных ограничений на процесс формирования понятий более низкого уровня иерархии, главное из которых правило однородности (хорошего обобщения): «Понятия на одном уровне иерархии должны быть одной природы и связаны с «предком» одним типом отношения».

О восприятии и-карт. Правильное восприятие и понимание и-карты связано с рядом эргономических проблем. Важными элементами и-карт являются графические образы, цвет и различная величина шрифтов на разных уровнях иерархии, — это позволяет четче и яснее отразить структуру и-карты, ускорить и упростить «чтение» и-карты. На рисунке 2 представлен пример и-карты для понятия «Окно». И-карта описывает виды окон, их конструктивные элементы, определение, предназначение и характеристики окон. Карта создана при помощи инструмента Mind Manager (<https://www.mindjet.com/>).

2.3 Системные проблемы разработки и-карт

Основная сложность создания и-карты – это системное формирование понятий первого уровня, описывающих центральную идею. Ясную и понятную и-карту обычно может нарисовать эксперт ПрО. У новичков часто получаются запутанные и невнятно организованные карты, вследствие нарушения правила однородности. Это правило лежит в основе любой таксономии, так как деление на классы происходит обычно по ОДНОМУ признаку.

Это означает, что понятия одного уровня требуют принадлежности одному уровню обобщения. Понятие «уровень обобщения» требует понимания методов системного анализа (класс-подкласс-элемент класса) и глубокого знания особенностей ПрО.

Так, при декомпозиции понятия ФРУКТЫ один из правильных рядов понятий первого уровня будет: «яблоки, груши, мандарины». Все эти объекты принадлежат к одному уровню грануляции или обобщения. И неправильным будет ряд «яблоки, груши, цитрусовые». «Цитрусовые» – это более высокий уровень обобщения.

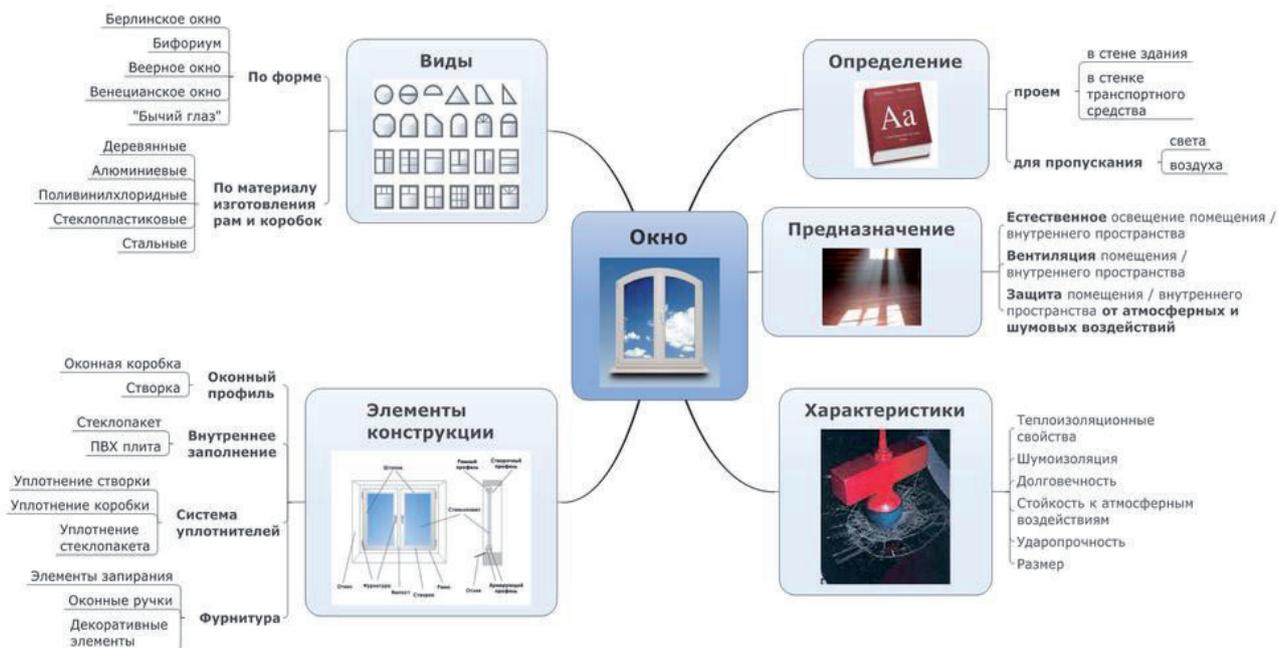


Рисунок 2 – Пример и-карты (автор Д. Кудрявцев [36])

Элементы одного уровня обобщения связаны с вышестоящим понятием одним типом отношения, они однородны, т.е. все признаки или атрибуты понятия, или все его компоненты, или ассоциации. Так, на рисунке 3 объекты нижнего уровня не однородны. Объект «карманы» не продолжает ряд атрибутов (таких, как производитель, цена и вес). Он связан с центральным понятием «рюкзак» отношением «часть-целое».



Рисунок 3 – Пример ошибки разнородности

На рисунке 4 также можно увидеть эту ошибку, но приблизительную структуру знаний о малом предприятии карта отражает.



Рисунок 4 – И-карта понятия «малое предприятие»

3 Практические рекомендации по построению и-карт

3.1 Правила построения хорошей и-карты

Обобщая опыт разработки и-карт и обучения аналитиков, можно сформулировать ряд эвристических правил, облегчающих построение и-карты:

- *Правила, способствующие глубине понимания*
 - А. Правило системного мышления: «Охватывайте все элементы, а не выборочные фрагменты».
 - Б. Правило хорошего обобщения (см.п.2.2)
Эти правила требуют значительных когнитивных усилий и навыков системно-аналитического мышления.
- *Правила, увеличивающие наглядность*
 - В. Правила Бьюзена – об использовании цвета, рисунков и уменьшения шрифтов при удалении от центра [25].
 - Г. Бритва Оккама: «Не стоит множить сущности без необходимости». В этом ошибка карты на рисунке 5.

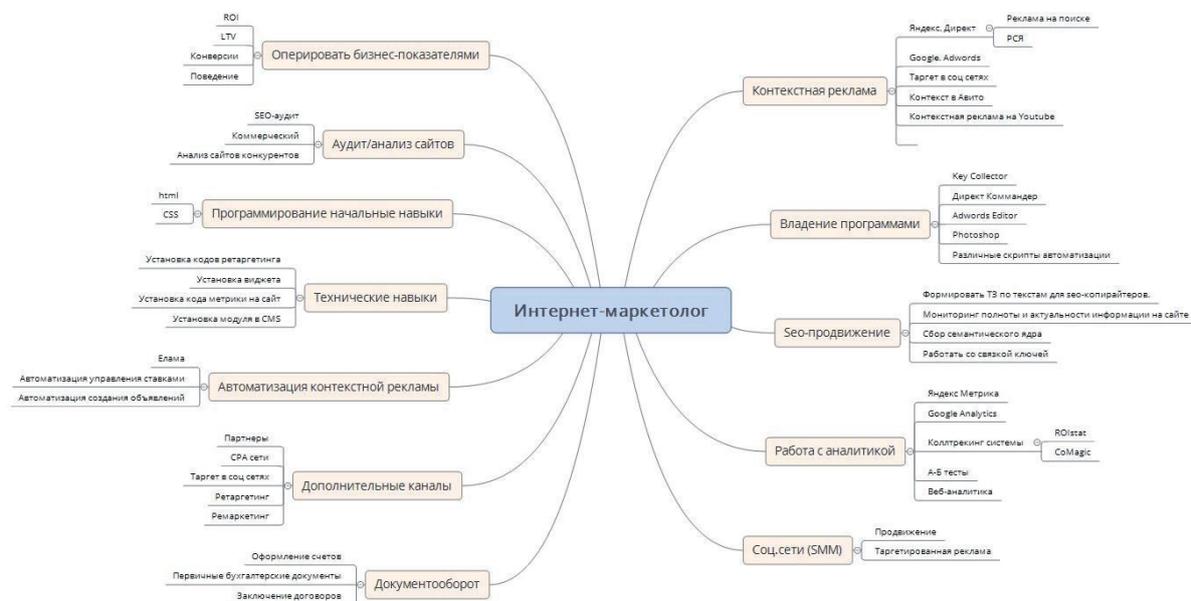


Рисунок 5 – Пример и-карты, нарушающей правило Оккама и другие правила²

Д. «Не стоит использовать шрифты с засечками». Например, Times New Roman.

Е. «В узле и-карты должны быть словосочетания в именительном падеже, не более трёх слов». Например, «железная дорога» или «входные финансовые потоки».

3.2 Анализ традиционных ошибок

В анализе обобщён 10-летний педагогический и практический опыт авторов по разработке онтологий на основе и-карт.

На рисунке 6 впервые предложена типология основных ошибок при разработке и-карт, основанная на анализе более 750 и-карт как студентов старших курсов Санкт-Петербургского государственного университета, так и слушателей программ Executive MBA.

² <https://kwork.ru/keywords/94461/intellekt-karta-mind-map-dlya-vashego-proekta>



Рисунок 6 – Типология ошибок разработки и-карт

К ошибкам I рода отнесены существенные смысловые ошибки разработки, в частности, семантические ошибки, включающие ошибки понимания (нет понимания, как устроена ПрО, отсюда неадекватные категории, неверная атрибуция к категориям) и фрагментарность (карта отражает только часть проблемы). Ошибки I рода включают также ошибки в систематизации элементов ПрО. Часто это излишнее дробление концептов и их избыточная детализация. К этой ошибке обычно ведёт нарушение правила «Бритвы Оккама». К системным также можно отнести самую «тяжёлую» и плохо поддающуюся исправлению ошибку разнородности понятий одного уровня (см. правило Б подраздела 3.1). Ошибка возникает при сочетании двух первых системных ошибок и недостатке знаний разработчика карты.

Ошибки II рода не столь принципиальны, так как касаются нарушений правил «хорошего» стиля, влияющих на восприятие общего образа и-карты. К ним относятся синтаксические и эргономические ошибки.

Синтаксические ошибки обусловлены нарушением правил Бьюзена о шрифте, цвете и пиктограммах, а также пренебрежением правил использования в и-карте минимума слов и имён существительных в именительном падеже в единственном числе.

Эргономические ошибки касаются непосредственно дизайна и стиля карты, что также важно в восприятии онтологии. Гангеми и его последователи разработали специальные метрики оценки эргономичности онтологий [37]. Список нарушений такого рода рекомендаций включает использование более трёх цветов (пестрота), наличие слишком длинных или коротких ветвей (дисбаланс), употребление мелкого шрифта (нечитабельность), перегруженность концептами и картинками и такой неформализуемый критерий, как отсутствие гармонии. Вопросы гармонии в дизайне бизнес-диаграмм и онтологий подробно обсуждены в работах [36, 38].

В целом ошибки первого рода вызваны отсутствием навыков системного и структурного анализа, в части категоризации и обобщения понятий. Помимо отсутствия навыков эти ошибки можно объяснить особенностями характеристик когнитивного стиля аналитика [35]. В частности, существует характеристика «способность к обобщению», которая не может быть исправлена ни тренингом, ни опытом. Анализ ошибок проектирования носит диагностический характер, то есть карта обнажает пробелы знаний и особенности мышления.

Отдельной дискуссии заслуживает вопрос «правильности и пользы». Можно допустить, что любая карта лучше её отсутствия. Однако тиражирование искажённых или неадекватных представлений о структуре ПрО также нежелательно.

И-карта, отражающая мнение одного эксперта, субъективна. Карта, полученная в результате мозгового штурма или совместного обсуждения, может приблизиться к объективной картине.

Заключение

Интерес к инфографике и визуализации, помогающим быстрому пониманию, не случаен в настоящее время. Визуализации позволяют подняться на более высокий уровень концептуального обобщения больших и сверхбольших объёмов данных, облегчить коммуникативные разрывы между различными группами исследователей и экспертов, между студентами и преподавателями [39].

Для онтологического инжиниринга тема пока не стала «горячей» в силу «незрелости» и учебного характера многих онтологий. При этом и-карты уже превратились в рабочий инструмент стратегического менеджмента, а онтологии завоёвывают всё новые области компьютерного моделирования, и здесь важно не допускать ошибок в их проектировании.

Десятилетний опыт по обучению и тренингу навыкам визуально-аналитического мышления на программах Executive MBA и на корпоративных тренингах показывает, что не все обучаемые способны научиться разрабатывать «полезные и правильные» и-карты.

Главной особенностью онтологического моделирования является системный взгляд и корректное обобщение при формировании категорий и классов. Разработка онтологии на первоначальном этапе должна базироваться на системном хорошо структурированном каркасе ПрО. И-карты делают этот каркас зримым.

Благодарности

Статья посвящается Дмитрию Поспелову - основателю междисциплинарного подхода к исследованиям по искусственному интеллекту в России.

Работа выполнена при финансовой поддержке гранта РФФИ (проект N 20-07-00854).

Список источников

- [1] *Lambrix, P.* Evaluation of ontology development tools for bioinformatics / Lambrix, Patrick, Manal Habbouche, and Marta Perez // Bioinformatics. – 2003. – 19(12). – P.1564-1571. – DOI: 10.1093/bioinformatics/btg194.
- [2] *Corcho, O.* Methodologies, tools and languages for building ontologies. Where is their meeting point? / Corcho, Oscar, Mariano Fernández-López, and Asunción Gómez-Pérez // Data & knowledge engineering. – 2003. – 46(1). – P.41-64. – DOI: 10.1016/S0169-023X(02)00195-7.
- [3] *Guizzardi, G.* Taking It to the Next Level: Nicola Guarino, Formal Ontology and Conceptual Modeling / Guizzardi, G., & Mylopoulos, J. // Ontology Makes Sense. – 2019. – P.223-241. – DOI: 10.3233/978-1-61499-955-3-223
- [4] *Hitzler, P.* Ontology Engineering with Ontology Design Patterns: Foundations and Applications / Hitzler, Pascal, Aldo Gangemi, and Krzysztof Janowicz, eds. // Studies on the Semantic Web. – IOS Press, 2016. – 25. – 388 p. – DOI: 10.3233/978-1-61499-676-7.
- [5] *Боргест, Н.М.* Онтологии проектирования от Витрувия до Виттиха / Н.М. Боргест // Онтология проектирования. – 2018. – Т. 8, № 4. – С. 487-522. – DOI: 10.18287/2223-9537-2018-8-4-487-522.
- [6] *Петрова, Г.Г.* Онтология FIBO и её применение для решения прикладных задач финансовой сферы / Г.Г. Петрова, А.Ф. Тузовский // Знания-Онтологии-Теории (ЗОНТ-2015) - 2015. – С.96-99.
- [7] *Ахмадеева, И.Р.* Сбор онтологической информации для интеллектуальных научных Интернет-ресурсов / И.Р. Ахмадеева, О.И. Боровикова, Ю.А. Загорулько, Е.А. Сидорова // Системная информатика. — 2014. — № 3. — С.13-23. – DOI: 10.31144/si.2307-6410.2014.n3.p13-23.
- [8] *Добров, Б.В.* Онтологии и тезаурусы: модели, инструменты, приложения / Б.В. Добров, В.В. Иванов, В.Д. Соловьев, Н.В. Лукашевич. – М.: Интернет-Университет ИТ; БИНОМ. Лаборатория знаний, 2009. – 173 с.

- [9] **Гришин, М. В.** Онтологии проектирования шаблонной оснастки в авиационном производстве / М. В. Гришин, С.Н. Ларин, П.И. Соснин // Онтология проектирования. - 2016. - Том 6, 1(19). - С.7-28. - DOI: 10.18287/2223-9537-2016-6-1-7-28.
- [10] **Suárez-Figueroa, M. C.** Scheduling ontology development projects / M.C. Suárez-Figueroa, A. Gómez-Pérez, M. Fernández-López // Data & Knowledge Engineering. - 2016. - 102. - P.1-21. - DOI: 10.1016/j.datak.2015.11.004.
- [11] **Otero-Cerdeira, L.** Ontology matching: A literature review / Otero-Cerdeira, Lorena, Francisco J. Rodríguez-Martínez, Alma Gómez-Rodríguez. // Expert Systems with Applications. - 2015. - 42(2). - P.949-971. - DOI: 10.1016/j.eswa.2014.08.032.
- [12] **Suárez-Figueroa, M. C.** The NeOn methodology for ontology engineering / M.C. Suárez-Figueroa, A. Gómez-Pérez, M. Fernández-López // Ontology engineering in a networked world. Springer, Berlin, Heidelberg. - 2012. - P. 9-34. - DOI: 10.1007/978-3-642-24794-1_2.
- [13] **Fernandez, M.** METHONTOLOGY: From Ontological Art Towards Ontological Engineering / M. Fernandez, A. Gomez-Perez, N. Juristo // AAAI Technical Report SS-97-06. - 1997. - http://oa.upm.es/5484/1/METHONTOLOGY_.pdf.
- [14] **Гаврилова, Т.А.** Инженерия знаний. Модели и методы / Т.А. Гаврилова, Д.В. Кудрявцев, Д.И. Муромцев. - СПб, Лань, 2016. - 324 с.
- [15] **Брунер, Дж.** Психология познания / Дж. Брунер. - М.: Прогресс, 1977. - 413 с.:
- [16] **Веккер, Л.М.** Психические процессы (в 3-х томах) / Л.М. Веккер. - Л. ЛГУ, 1974. - т. 2. - 344 с.
- [17] **Johnson-Laird, Ph. N.** Mental models: Towards a cognitive science of language, inference, and consciousness / Philip Nicholas Johnson-Laird. - Harvard University Press, 1983. - 513 p.
- [18] **Johnson-Laird, Ph. N.** Mental models or formal rules? / Philip N. Johnson-Laird, Ruth MJ Byrne // Behavioral and Brain Sciences. - 1993. - 16. - P.368-380. - DOI: 10.1017/S0140525X0003065X.
- [19] **Novak, J.** Concept maps and vee diagrams: Two metacognitive tools for science and mathematics education / J. Novak // Instructional Science. - 1990. - 19. - P.29-52. - DOI: 10.1007/BF00377984.
- [20] **Ausubel, D. P.** Educational psychology: A cognitive view / D. P. Ausubel. - New York, Holt, Rinehart and Winston, 1968. - 733 p.
- [21] **Рапуто, А.Г.** Deskриптивное моделирование образного мышления при репрезентации дидактических объектов / А.Г. Рапуто // Институт информатизации образования, Сборник «Ученые записки». - М.: ИИО РАО. 2011. - Выпуск 34. - С.114-116.
- [22] **Shneiderman, B.** Information Visualization Manifesto / Ben Shneiderman // VC blog. - 1999. - <http://www.visualcomplexity.com/vc/blog/?p=644>.
- [23] **Зинченко, В.П.** Большой психологический словарь / В.П. Зинченко, Б.Г. Мещеряков. - М.: АСТ, 2008. - 632 с.
- [24] **McKim, R.H.** Experiences in visual thinking / R. H. McKim. - Brooks/Cole, 1972. - 210 p.
- [25] **Бьюзен, Т.** Интеллект-карты. Практическое руководство / Т. Бьюзен, Б. Бьюзен.- Минск: Попурри, 2010. - 368 с.
- [26] **Черниговская, Т.В.** Сенсо-моторный и когнитивный латеральный профиль: тестирование и интерпретация / Т.В. Черниговская, Т.А. Гаврилова, А.В. Воинов, К.Н. Стрельников // Физиология человека. - 2005. - Том 31, N2. - С.35-44.
- [27] **Koznov, D.** Mind maps merging in collaborative work / D. Koznov, E. Larchik, M. Pliskin, N. Artamonov // Programming and Computer Software. - 2011. - Vol. 37, Issue 6. - P.315-321. - DOI: 10.1134/S036176881106003X.
- [28] **Дубровский, Д.И.** Психические явления и мозг / Д.И. Дубровский. - М.: Рипол Классик, 2013. - 394 с.
- [29] **Chei Chang, C.** The effect of concept mapping on students' learning achievements and interests / C. Chei Chang // Innovations in Education and Teaching International. - 2008. - Vol. 45, № 4. - P.375-387. - DOI: 10.1080/14703290802377240.
- [30] **Роэм, Д.** Визуальное мышление Решение проблем и продажа идей при помощи картинок на салфетке / Д. Роэм. - М.: Эксмо, 2009. - 296 с.
- [31] **Hverle, D.** Visual Tools for Transforming Information Into Knowledge / D. Hverle. - Corwin, 2009. - 192 p.
- [32] **Jeffery, A.B.** Improving team decision-making performance with collaborative modeling / A.B. Jeffery, J.D. Maes, M.F. Bratton-Jeffery // Team Performance Management. - 2005. - Vol. 11, № 1-2. - P.40-50. - DOI: 10.1108/13527590510584311.
- [33] **Margulies, N.** Visual Thinking Tools for Mapping Your Ideas / N. Margulies, C. Valenza. - Crown House Publishing, 2005. - 160 p.
- [34] **Lim, B.C.** Team mental models and team performance: A field study of the effects of team mental model similarity and accuracy / B.C. Lim, K.J. Klein // Journal of Organizational Behavior. - 2006. - 27(4). - P.403-418. - DOI: 10.1002/job.387.

- [35] **Холодная, М.А.** Когнитивные стили. О природе индивидуального ума / М.А. Холодная. – ПЕР СЭ, 2002. – 304 с.
- [36] **Григорьев, Л.Ю.** Технология наполнения баз знаний онтологического типа / Л.Ю. Григорьев, А.А. Заблоцкий, Д.В. Кудрявцев // Научно-технические ведомости СПбГПУ. Серия «Информатика. Телекоммуникации. Управление». – 2012. – 3. – С.27-36.
- [37] **Gangemi, A.** Ontology design patterns / Aldo Gangemi, Valentina Presutti // Handbook on ontologies. - Springer, Berlin, Heidelberg, 2009. – P.221-243. – DOI: 10.1007/978-3-540-92673-3_10.
- [38] **Effinger, Ph.** On a study of layout aesthetics for business process models using BPMN / Philip Effinger, Nicole Jogsch, Sandra Seiz. // International Workshop on Business Process Modeling Notation. Springer, Berlin, Heidelberg. – 2010. – P.31-45. – DOI: 10.1007/978-3-642-16298-5_5.
- [39] **Henderson, M.K.** Flexible Sketches and Inflexible Data Bases: Visual Communication, Conscriptioin Devices, and Boundary Objects in Design Engineering / М. К. Henderson // Science Technology Human Values. – 1991. – 16(4). – P.448-473. – DOI: 10.1177/016224399101600402.

Сведения об авторах



Гаврилова Татьяна Альбертовна. Проф, д.т.н. (с 1996 г.), зав. кафедрой информационных технологий в менеджменте Высшей школы менеджмента Санкт-Петербургского государственного университета. Автор более 200 публикаций, число цитирований в РИНЦ 5163 (на 28 января 2020), индекс Хирша 27, председатель Северо-западного отделения Российской ассоциации искусственного интеллекта. AuthorID (РИНЦ): 7060. Author ID (Scopus): 56447180000; Researcher ID (WoS): E-9154-2010. gavrilova@gsom.spbu.ru.

Страхович Эльвира Витаутасовна. Окончила математико-механический факультет Ленинградского государственного университета,

к.ф.-м.н. (1992). Доцент кафедры информационных технологий в менеджменте Высшей школы менеджмента Санкт-Петербургского государственного университета. До 2010 г. работала в ИТ-компании. Область научных интересов - информационно-коммуникационные технологии в образовании. AuthorID (РИНЦ): 443895; Author ID (Scopus): 55304837700; ORCID 0000-0003-1374-9670; Researcher ID (WoS): I-6632-2013. strakhovich@gsom.spbu.ru.



Поступила в редакцию 08.01.2020, после рецензирования 10.02.2020. Принята к публикации 10.03.2020.

Visual analytical thinking and mind maps for ontology engineering

T.A. Gavrilova, E.V. Strakhovich

Graduate School of Management, St. Petersburg University, Saint-Petersburg, Russia

Abstract

The article is devoted to the practical application of the principles of visual analytical thinking for the problems of knowledge structuring for the ontology design and development. Visual analytical thinking refers to implementation of methodologies that use different types of diagrams to represent ideas, concepts, relationships and processes. From the well-known practically used types of diagrams, mind maps were selected as the most common, convenient and simple method for the proper formation and design of ontologies of complex domains. Mind maps reflect hierarchical relationships among concepts and allow the analyst to sufficiently deeply reflect the features and patterns of the domain with their specific relationships. Buzan formulated the idea of mind maps in the 1970s as a compact means of organizing abstracts, he later deepened and enhanced this idea, which was brought to software implementation and was widely used in various fields of education, research and business. The paper discusses the basic principles of the formation of such maps and analyzes the typical mistakes of analysts. For the first time, a classification of main errors and mistakes is proposed taking into account syntactic, semantic and pragmatic aspects. The analysis of the most common errors associated with the violation of the rules of “good generalization” and reasonable minimalism is given. The article may be of interest to both intelligent systems’ developers and knowledge management systems’ creators; it summarizes ten years of experience in teaching and training visual analytical thinking skills in executive MBA programs and corporate trainings.

Keywords: *ontology engineering, mind maps, visual analytical thinking, visual models, knowledge management.*

Citation: *Gavrilova TA, Strakhovich EV.* Visual analytical thinking and mind maps for ontology engineering [In Russian]. *Ontology of designing.* 2020; 10(1): 87-99. DOI: 10.18287/2223-9537-2020-10-1-87-99.

Acknowledgment: The article is dedicated to Dmitry Pospelov, the founder of an interdisciplinary approach to research on artificial intelligence in Russia. The work supported by grant of Russian Foundation for basic research (project N 17-18-30048).

List of figures

Figure 1 - An example of an abstract mind map

Figure 2 - An example of a mind map “Window” (by D. Kudryavtsev)

Figure 3 - An example of a heterogeneity error

Figure 4 - Mind map of the concept of “small business”

Figure 5 - An example of a mind map that violates Occam's rule and other basic rules

Figure 6 - Typology of errors

References

- [1] **Lambrix P, Habbouche M, Perez M.** Evaluation of ontology development tools for bioinformatics. *Bioinformatics.* 2003; 19(12): 1564-1571. – DOI: 10.1093/bioinformatics/btg194.
- [2] **Corcho O, Fernández-López M, Gómez-Pérez A.** Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data & knowledge engineering.* 2003; 46(1): 41-64. – DOI: 10.1016/S0169-023X(02)00195-7.
- [3] **Guizzardi G, Mylopoulos J.** Taking It to the Next Level: Nicola Guarino, Formal Ontology and Conceptual Modeling. *Ontology Makes Sense.* 2019: 223-241. – DOI: 10.3233/978-1-61499-955-3-223.
- [4] **Hitzler, Pascal.** Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Hitzler, Pascal, Aldo Gangemi, and Krzysztof Janowicz, eds. *Studies on the Semantic Web.* IOS Press, 2016; 25. 388 p. DOI: 10.3233/978-1-61499-676-7.
- [5] **Borgest NM.** The ontologies of designing from Vitruvius to Wittich [In Russian]. *Ontology of designing.* 2018; 8(4): 487-522. DOI: 10.18287/2223-9537-2018-8-4-487-522.
- [6] **Petrova GG, Tuzovsky AP.** Fibo ontology and its application for solving applied problems in the financial sphere [In Russian]. *Knowledge-Ontologies-Theories (ZONT-2015)* 2015: 96-99.
- [7] **Akhmadeeva IR, Borovikova OI, Zagorulko YuA, Sidorova EA.** Collection of ontological information for intellectual scientific Internet resources [In Russian]. *System Informatics.* 2014; 3: 13-23. DOI: 10.31144/si.2307-6410.2014.n3.p13-23.
- [8] **Dobrov BV, Ivanov VV, Soloviev VD, Lukashevich NB.** Ontologies and thesauruses: models, tools, applications: handbook [In Russian]. Moscow: Internet-University of Information Technology, BINOM. Knowledge Laboratory, 2009. 173 p.
- [9] **Grishin MV, Larin SN, Sosnin PI.** Ontologies of design of template equipment in aircraft manufacturing [In Russian]. *Ontology of designing.* 2016; 6(1): 7-28. DOI: 10.18287/2223-9537-2016-6-1-7-28.
- [10] **Suárez-Figueroa MC, Gómez-Pérez A, Fernández-López M.** Scheduling ontology development projects. *Data & Knowledge Engineering.* 2016; 102: 1-21. DOI: 10.1016/j.datak.2015.11.004.
- [11] **Otero-Cerdeira L, Rodríguez-Martínez FJ, Gómez-Rodríguez A.** Ontology matching: A literature review. *Expert Systems with Applications.* 2015; 42(2): 949-971. DOI: 10.1016/j.eswa.2014.08.032.
- [12] **Suárez-Figueroa, M. C.** The NeOn methodology for ontology engineering / M.C. Suárez-Figueroa, A. Gómez-Pérez, M. Fernández-López // *Ontology engineering in a networked world.* Springer, Berlin, Heidelberg. – 2012. – P. 9-34. DOI: 10.1007/978-3-642-24794-1_2.
- [13] **Fernandez, M.** METHONTOLOGY: From Ontological Art Towards Ontological Engineering / M. Fernandez, A. Gomez-Perez, N. Juristo // AAAI Technical Report SS-97-06. – 1997. – http://oa.upm.es/5484/1/METHONTOLOGY_.pdf.
- [14] **Gavrilova TA, Kudryavtsev DV, Muromtsev DI.** Knowledge Engineering. Models and methods [In Russian]. SPb, Lan, 2016. 324 p.
- [15] **Bruner G.** Cognitive Psychology [In Russian]. Moscow: Progress, 1977. 413 p.
- [16] **Vekker LM.** Psychological processes (in 3-e volumes) [In Russian]. L. LGU, 1974; vol.2. 344 p.
- [17] **Johnson-Laird PhN.** Mental models: Towards a cognitive science of language, inference, and consciousness. Harvard University Press, 1983. 513 p.

- [18] **Johnson-Laird PhN, Byrne RMJ.** Mental models or formal rules? *Behavioral and Brain Sciences*. 1993; 16: 368-380. DOI: 10.1017/S0140525X0003065X.
- [19] **Novak J.** Concept maps and vee diagrams: Two metacognitive tools for science and mathematics education. *Instructional Science*. 1990; 19: 29-52. DOI: 10.1007/BF00377984.
- [20] **Ausubel DP.** Educational psychology: A cognitive view. New York, Holt, Rinehart and Winston, 1968. 733 p.
- [21] **Raputo AG.** Descriptive modeling of imaginative thinking in the representation of didactic objects [In Russian]. Institute of Informatization of education, *Scientific notes*. 2011; 34: 114-116.
- [22] **Shneiderman B.** Information Visualization Manifesto. Visual complexity. VC blog. 1999. <http://www.visualcomplexity.com/vc/blog/?p=644>.
- [23] **Zinchenko VP, Mescheriakov BG.** Big psychological dictionary [In Russian]. Moscow: AST, 2008. 632 p.
- [24] **McKim RH.** Experiences in visual thinking. Brooks/Cole, 1972. 210 p.
- [25] **Buzan T, Buzan B.** The Mind Map Book [In Russian]. Minsk: Popurry, 2003. 280 p.
- [26] **Chernigovskaya TB, Gavrilova TA, Voinov AV, Strelnikov KN.** Sensorimotor and cognitive lateral profile: testing and interpretation [In Russian]. *Human physiology*. 2005; 31(2): 35-44.
- [27] **Koznov D, Larchik E, Pliskin M, Artamonov N.** Mind maps merging in collaborative work. *Programming and Computer Software*. 2011; 37(6): 315-321. DOI: 10.1134/S036176881106003X.
- [28] **Dubrovsky DI.** Mental phenomena and the brain [In Russian]. Moscow: Ripol Classic, 2013. 394 p.
- [29] **Chei Chang C.** The effect of concept mapping on students' learning achievements and interests. *Innovations in Education and Teaching International*. 2008; 45(4): 375-387. DOI: 10.1080/14703290802377240.
- [30] **Roem D.** Visual thinking. Solving problems and selling ideas with pictures on a napkin [In Russian]. Moscow: Eksmo, 2009. 296 p.
- [31] **Hverle D.** Visual Tools for Transforming Information Into Knowledge. Corwin, 2009. 192 p.
- [32] **Jeffery AB, Maes JD, Bratton-Jeffery MF.** Improving team decision-making performance with collaborative modeling. *Team Performance Management*. 2005; 11(1-2): 40-50. DOI: 10.1108/13527590510584311.
- [33] **Margulies N, Valenza C.** Visual Thinking Tools for Mapping Your Ideas. Crown House Publishing, 2005. 160 p.
- [34] **Lim BC, Klein KJ.** Team mental models and team performance: A field study of the effects of team mental model similarity and accuracy. *Journal of Organizational Behavior*. 2006; 27(4): 403-418. DOI: 10.1002/job.387.
- [35] **Kholodnaya MA.** Cognitive style. On the nature of the individual mind [In Russian]. PER SE, 2002. 304 p.
- [36] **Grigoriev LYu, Zablotzky AA, Kudryavtsev DV.** Ontological-type knowledge base filling technology [In Russian]. *Scientific and technical statements of SPbSPU. Series "Informatics. Telecommunications. Control"*. 2012; 3: 27-36.
- [37] **Gangemi A, Presutti V.** Ontology design patterns. *Handbook on ontologies*. Springer, Berlin, Heidelberg, 2009. P. 221-243. DOI: 10.1007/978-3-540-92673-3_10.
- [38] **Effinger Ph, Jogsch N, Seiz S.** On a study of layout aesthetics for business process models using BPMN. International Workshop on Business Process Modeling Notation. Springer, Berlin, Heidelberg. 2010. P.31-45. DOI: 10.1007/978-3-642-16298-5_5.
- [39] **Henderson MK.** Flexible Sketches and Inflexible Data Bases: Visual Communication, Conscriptioin Devices, and Boundary Objects in Design Engineering. *Science Technology Human Values*. 1991; 16(4): 448-473. DOI: 10.1177/016224399101600402.

About the authors

Tatiana Albertovna Gavrilova, prof., DSc. (since 1996), head of Information Technologies in Management Department at Graduate School of Management (GSOM) in St. Petersburg University, author of more than 200 publications, citation number in Russian citation index 5163 (as of January 28, 2020), Hirsh index 27, Head of North-West chapter of Russian Association for Artificial Intelligence. AuthorID (RCI): 7060. Author ID (Scopus): 56447180000; Researcher ID (WoS): E-9154-2010. gavrilova@gsom.spbu.ru.

Elvira Vitautasovna Strakhovich graduated from Leningrad State University, mathematical and mechanical department, Ph.D. (1992). She is Associate Professor in Information Technology in Management department in the Graduate School of Management, Saint-Petersburg State University. Before 2010 E.V. Strakhovich worked for IT company. Her research interests – information and communication technologies in education. AuthorID (RCI): 443895; Author ID (Scopus): 55304837700; ORCID 0000-0003-1374-9670; Researcher ID (WoS): I-6632-2013. strakhovich@gsom.spbu.ru.

Received January 8, 2020. Revised February 10, 2020. Accepted March 10, 2020.

Анализ данных о поведении пользователей в системах электронного обучения

А.А. Романов, Д.Г. Волчек

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, Санкт-Петербург, Россия

Аннотация

Рассматриваются современные системы электронного обучения, имеющие возможность записывать действия пользователей, такие как передвижения, использование интерактивных материалов, регистрация на курсы, их завершение и др. Анализ действий пользователей в системах управления процессом обучения предоставляет возможности для персонализации образовательных траекторий. На основе изучения поведения пользователей становится возможным формирование рекомендаций для разработчиков курсов по улучшению контента и структуры, а также рекомендаций по прохождению курса обучающимся. Исследуются данные, содержащиеся в логах активности, для получения информации, поиска зависимостей путём фильтрации релевантных логов, структурирования информации из них и предоставления данных в удобном для анализа и получения выводов виде. Рассматриваются данные основных типов событий, генерируемых в результате записи действий пользователя в системе управления обучением, и сценарии использования результатов анализа этих данных.

Ключевые слова: образование, поведение пользователей, электронное обучение, веб-программирование, базы данных.

Цитирование: Романов, А.А. Анализ данных о поведении пользователей в системах электронного обучения / А.А. Романов, Д.Г. Волчек // Онтология проектирования. – 2020. – Т. 10, №1(35). – С.100-111. – DOI: 10.18287/2223-9537-2020-10-1-100-111.

Введение

Прибегая к использованию MOOC (массовых открытых онлайн-курсов, *Massive Open Online Courses, MOOC*), университеты сталкиваются с различными проблемами, например, выбор качественных онлайн-курсов и оценка эффективности онлайн-обучения. Первая задача активно решается в последние годы в рамках проекта «Современная цифровая образовательная среда в России» [1]. Но по-прежнему остаются открытыми вопросы оценки эффективности обучения. Аналитика учебной деятельности обучающихся при освоении ими онлайн-курсов является одним из инструментов повышения качества обучения [2]. Понимание поведения пользователей в системах онлайн-обучения может сильно улучшить качество и облегчить обучение в дальнейшем. Анализ поведения позволит ускорить изучение материала, предоставляя контент, который точно будет соответствовать потребностям учащихся. Обычно при изучении поведения в онлайн-образовании пользуются системой управления обучением (*Learning Management System, LMS*). Из неё можно узнать, когда вошёл каждый учащийся, историю его посещений, а также запись пройденных курсов. Однако эта информация не является достаточной для построения моделей слушателя и необходимо дополнять её за счёт поведенческих логов.

Основная цель исследования – найти эффективные способы работы с большими объёмами данных, создаваемых веб-средами обучения. Для получения этих данных необходимо

структурировать логи активности пользователей. Структурированные данные необходимо предоставить в удобно читаемом виде, доступном для анализа, извлечения выводов или различных метрик, а также построения предсказательных моделей.

1 Принципы анализа электронного обучения

Если бы можно было анализировать поведение учащихся при работе с онлайн-учебными материалами, это облегчило и ускорило бы процесс обучения, предоставляя учащимся контент, который точно соответствует их потребностям. Обычно система управления обучением позволяет отслеживать «прогресс» обучающихся, однако информация зачастую не так подробна, как хотелось бы. При анализе результатов обучения могут быть полезны следующие основные принципы.

Персонализация обучения. Концепция персонализации обучения в рамках MOOC получила широкое распространение и поддержку. Актуальной является задача создания адаптирующихся персонализированных массовых онлайн курсов, поддерживающих наличие нового контента и возможности построения индивидуальных траекторий [3]. Модель курса должна быть гибкой и уметь адаптироваться под действия пользователей, а также обеспечивать наличие некоторого «базового» пути, который должен использоваться в случае, если обучающийся мало знаком с предметной областью и не готов самостоятельно строить свою образовательную траекторию. Для решения этой проблемы целесообразно выполнять кластеризацию пользователей на основе их поведения: насколько успешно происходит усвоение материала, какие разделы курса просмотрел пользователь, переходил ли он по каким-либо внешним ссылкам, представленным в рамках курса и т.д. [4, 5]. Пользователи, которые менее заинтересованы в обучении или испытывают сложности, могут получать лишь базовый контент для достижения общих компетенций. Пользователи, которые показывают успехи, могут получать дополнительные материалы [6].

Понимание поведения учащихся. Для обеспечения возможности персонализации образовательного процесса необходимо точное и подробное отслеживание действий пользователей. Это позволит определять, как пользователи взаимодействуют с системой и учебным материалом. Без такой информации нет возможности определить, что конкретно не знает тот или иной человек, следовательно, нет возможности персонализировать подаваемый материал для пользователя. Современные системы обучения позволяют детально записывать каждый шаг изучения материала. Анализируя эту информацию, можно заметить, что пользователи, например, возвращаются к одному и тому же фрагменту курса. Этот фрагмент может быть информативным и полезным, но недостаточно понятным, и учащиеся предпринимают несколько попыток изучить данный материал. Возможно, что эта часть курса является обязательной для выполнения контроля (тест, упражнения, экзамен и т.д.) и дальнейшего освоения материала, и пользователи несколько раз пытаются её изучить [7, 8].

Возможность различать такие события с помощью информации о поведении пользователей является отправной точкой для создания персонализированного обучения и понимания того, что конкретно нужно пользователям и как следует адаптировать контент курса в дальнейшем.

Отслеживание действий и рекомендации. Существует множество примеров анализа поведения пользователей в сети. В том числе генерация индивидуальных предложений, формирование поисковой выдачи на основе семантики ранее произведённых запросов и т.д. В результате анализа поведения пользователя в системе обучения можно формировать сведения о регулярности участия пользователя в образовательном процессе, производить оценку его успеваемости и многое другое [9]. Исследование поведения обучающегося позволяет произ-

водить анализ потребляемого контента конкретными пользователями и создавать рекомендации по улучшению процесса обучения.

2 Модель представления логов

Развитие технологий обучения порождает множество учебных курсов, предлагаемых новыми системами управления обучением. Это не только даёт доступ к обучению, но и помогает преподавателям оценивать успеваемость учащихся. *LMS* является инструментом, который позволяет преподавателям автоматизировать отслеживание и запись действий учащихся. Такие записи называют логами (англ. *log*) — это файлы с записями о событиях в хронологическом порядке, обеспечивающие журналирование всех действий пользователя. Возможные полезные параметры, которые могут быть получены по логам [10, 11]:

- *активность* (можно отслеживать, когда пользователь в последний раз был в *LMS*);
- *частота* (отслеживание частоты посещений *LMS*);
- *длительность* (как долго пользователи остаются в *LMS* во время посещений);
- *обратная связь* (отзывы пользователей о содержании курса);
- *оценки* (отслеживание средних баллов и сравнение относительно проходных и средних по пользователям);
- *попытки* (установление количества попыток в том или ином задании или курсе).

Для отслеживания действий могут быть использованы различные методы. Например, это может быть генерация и отправка отчётов об активности конкретного пользователя целевой аудитории. Метод ручного отслеживания наиболее прост, но трудозатратен и малоэффективен. Современным является наличие выделенного веб-сервиса для мониторинга действий пользователя, результатов обучения и конечного вывода.

Из логов активности в *LMS* можно получить следующие характеристики поведения пользователя [11].

- Время, затрачиваемое на обучение (это может помочь спрогнозировать требуемое время для прохождения того или иного курса на практике).
- Время, затрачиваемое на освоение материала (отличается от предыдущего активностью, так как некто может, например, открыть курс и отвлекаться на что-то другое. Для этого учитывается активность в рамках окна браузера, переходы на другие страницы, вкладки и т.д.).
- Время, затраченное на каждый модуль (если пользователь тратит много (или слишком мало) времени на конкретный модуль, то это может указать на проблему в содержании курса: возможно материал сложный (простой/неинформативный) для освоения).
- Оставление курса (если курс обладает высоким уровнем отсева, это может свидетельствовать как о сложности материала, так и его некорректности или несогласованности упражнений с лекциями).

2.1 Типы событий

Рассматриваемые логи состоят из событий — действий пользователя в рамках *LMS*. Существует множество событий, которые в соответствии с активностью студентов описываются логами с использованием той или иной платформы. В работе в качестве источника данных использовалась платформа *Open edX* Университета ИТМО. Все события в логе содержат данные об источнике событий. Так, значение «браузер» подразумевает некоторое действие пользователя, а значение «сервер» — действие *LMS*, например проверка ответа пользователя. Лог-записи также содержат поля, в которых хранится служебная информация о конкретном событии [12]. Используются описания событий следующих типов.

События передвижения по обучающей системе.

Когда обучающийся переходит по ссылке или объекту, например, для передвижения по курсу, осуществляется соответствующая лог-запись.

Источник события: браузер.

Поля в лог-записях (все типы событий передвижения имеют одинаковые поля, представленные в таблице 1):

- *seq_goto* инициируется, когда пользователь переходит между страницами в последовательности;
- *seq_next* инициируется, когда пользователь переходит к следующей странице в последовательности;
- *seq_prev* инициируется, когда пользователь переходит к предыдущей странице в последовательности.

Таблица 1 – Подробности полей событий передвижений

Поле	Тип	Подробности
old	integer	Для <i>seq_goto</i> . Индекс страницы, с которой был произведён переход.
new	integer	Для <i>seq_goto</i> . Индекс страницы, на которую произведён переход.
id	integer	edX ID последовательности

События взаимодействия с видео.

Лог-запись об этом событии осуществляется, когда студент воспроизводит видео в обучающей системе или как-то иначе взаимодействует с видео.

Источник события: браузер.

События, которые могут происходить:

- *pause_video* инициируется, когда пользователь ставит видео на паузу;
- *play_video* инициируется, когда пользователь запускает (продолжает) просмотр видео;
- *seek_video* инициируется, когда пользователь проматывает видео;
- *speed_change_video* инициируется, когда пользователь изменяет скорость воспроизведения видео.

Перечисленные события ссылаются в поля, указанные в таблице 2.

Таблица 2 – Подробности полей событий взаимодействия с видео

Поле	Подробности
current_time	Время в видео, которое пользователь выбрал для изменения скорости воспроизведения.
old_speed	Скорость видео, с которой оно воспроизводилось ранее.
now_seed	Скорость, которую выбрал пользователь.

События взаимодействия с PDF-документами.

LMS содержит информацию интерактивных учебников в виде *pdf*-документов, для которых доступны поля, указанные в таблице 3.

Источник события: браузер.

Таблица 3 – Подробности полей событий взаимодействия с *PDF* документами

Поле	Тип	Подробности
type	string	'gotopage', 'prevpage', 'nextpage'
old	integer	Номер страницы, с которой произведен переход.
new	integer	Номер страницы, на которую перешли.

События перехода по внешним ссылкам.

Эти события содержат информацию, представленную в таблице 4.

Источник события: браузер.

Таблица 4 – Подробности полей события перехода по ссылкам

Поле	Тип	Подробности
current_url	string	Страница, на которой находился пользователь, где была нажата ссылка.
target_url	string	Адрес, на который перешел пользователь.

Событие вызова подсказки.

Вызов подсказок доступен в упражнениях, заданиях и опросах курса. Лог содержит идентификатор на тот или иной вид контроля, где пользователь запросил подсказку (таблица 5).

Источник события: сервер.

Таблица 5 – Подробности полей события вызова подсказки

Поле	Тип	Подробности
problem_id	string	Идентификатор вида контроля

Событие проверки правильности ответа.

Серверное событие проверки ответа пользователя. Поля указаны в таблице 6.

Источник события: сервер.

Таблица 6 – Подробности полей события проверки правильности ответа

Поле	Тип	Подробности
answer	object	Идентификатор вида контроля и ответ на него в виде пары имя/значение. Для компонента с несколькими вопросами перечислены все пары.
attempts	number	Количество попыток пользователя ответить на вопрос
problem_id	string	Идентификатор вида контроля
success	string	Правильно/неправильно

Рассмотренные события были получены в виде *JSON (JavaScript Object Notation)* структуры данных из образовательной платформы *Open edX* Университета ИТМО. Их описания содержат несколько полей в соответствии с тем действием, которое было совершено пользователем, и его типом. Пример пользовательского лога активности показан на рисунке 1.

Он получен для события ответ на вопрос при прохождении курса. В логе содержится служебная информация: о браузере пользователя, к какой странице он обратился, какое задание выполнял, сколько попыток было затрачено и т.д.:

username – имя пользователя, которым было инициировано событие;

event_type – тип события;

ip – IP-адрес пользователя, совершившего действие;

agent – браузер пользователя, который вызвал событие;

event – это поле содержит информацию о каждом из событий, вызванных пользователем. Каждое событие описано полями: *attempts* – количество попыток, которое потратил пользователь для ответа на вопрос, *problem_id* – идентификатор вида контроля, *success* – успешность ответа на вопрос.

Такой формат логов понятен разработчикам и тем, кто детально изучил документацию о платформе *LMS*.

2.2 Структурирование информации из логов

Для релевантных типов событий можно выделить две основные части данных, содержащихся в логах: идентификационная информация и детали события. Идентификационная часть включает:

- идентификатор пользователя;
- имя пользователя;
- время события;
- идентификатор курса.

```
{
  "username": "staff",
  "event_type": "problem_check",
  "ip": "10.0.2.2",
  "agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.81 Safari/537.36",
  "host": "vagrant",
  "referer": "http://localhost:8000/courses/course-v1:ITMO+1+1337/courseware/bdd79acb14354b379df387c142d92f70/05b50d3648da4c
b99f823bd9d07972ae/?child=last",
  "accept_language": "en;q=0.6, en;q=0.4",
  "event": {
    "attempts": 67,
    "problem_id": "block-v1:ITMO+1+1337+type@problem+block@7e266df0eb6e4b408c1943c093e955f7",
    "success": "correct"
  },
  "event_source": "server",
  "hint": false,
  "context": {
    "user_id": 5,
    "org_id": "ITMO",
    "module": {
      "display_name": "checkboxes"
    }
  },
  "course_id": "course-v1:ITMO+1+1337",
  "path": "/courses/course-v1:ITMO+1+1337/xblock/block-v1:ITMO+1+1337+type@problem+block@7e266df0eb6e4b408c1943c093e955f7/
handler/xmodule_handler/problem_check"
},
  "time": "2017-04-26T18:54:42.803083+00:00",
  "page": "x_module"
}
```

Рисунок 1 – Пример лога активности

Детали события специфичны для каждого типа события. События можно разделить на группы:

- навигация пользователей (переходы на внешние ссылки и навигация в секциях курса);
- взаимодействие с видеоматериалами (воспроизведения, паузы, перемотки, изменения скорости);
- взаимодействие с документами (открытие документа, перелистывание, навигация);
- ответы на вопросы (подсказки, показ ответов, проверка правильности);
- работа с курсами (регистрация на курсе, отмена курса, получение сертификата).

Для извлечения и структурирования данных логов активности написан программный модуль на языке *Python* с использованием библиотеки *psycogp2*. К каждому из типов событий написан индивидуальный модуль, так как данные в них зачастую разнородны. Для хранения данных была выбрана СУБД *PostgreSQL*. Схема базы данных (БД) для хранения данных из логов пользователей приведена на рисунке 2. Типы событий разделены на группы, и для каждой группы событий создана соответствующая таблица. Внутри таблиц есть поле типа события для идентификации типа. Поля пользователей и курсов связаны с соответствующими таблицами. Всего было обработано более 60000 логов. Обработка выполнялась с помощью разработанных модулей на языке *Python*. События, которые не подходят ни под один тип, были записаны в таблицу *others* для отслеживания активности того или иного пользователя в целом.

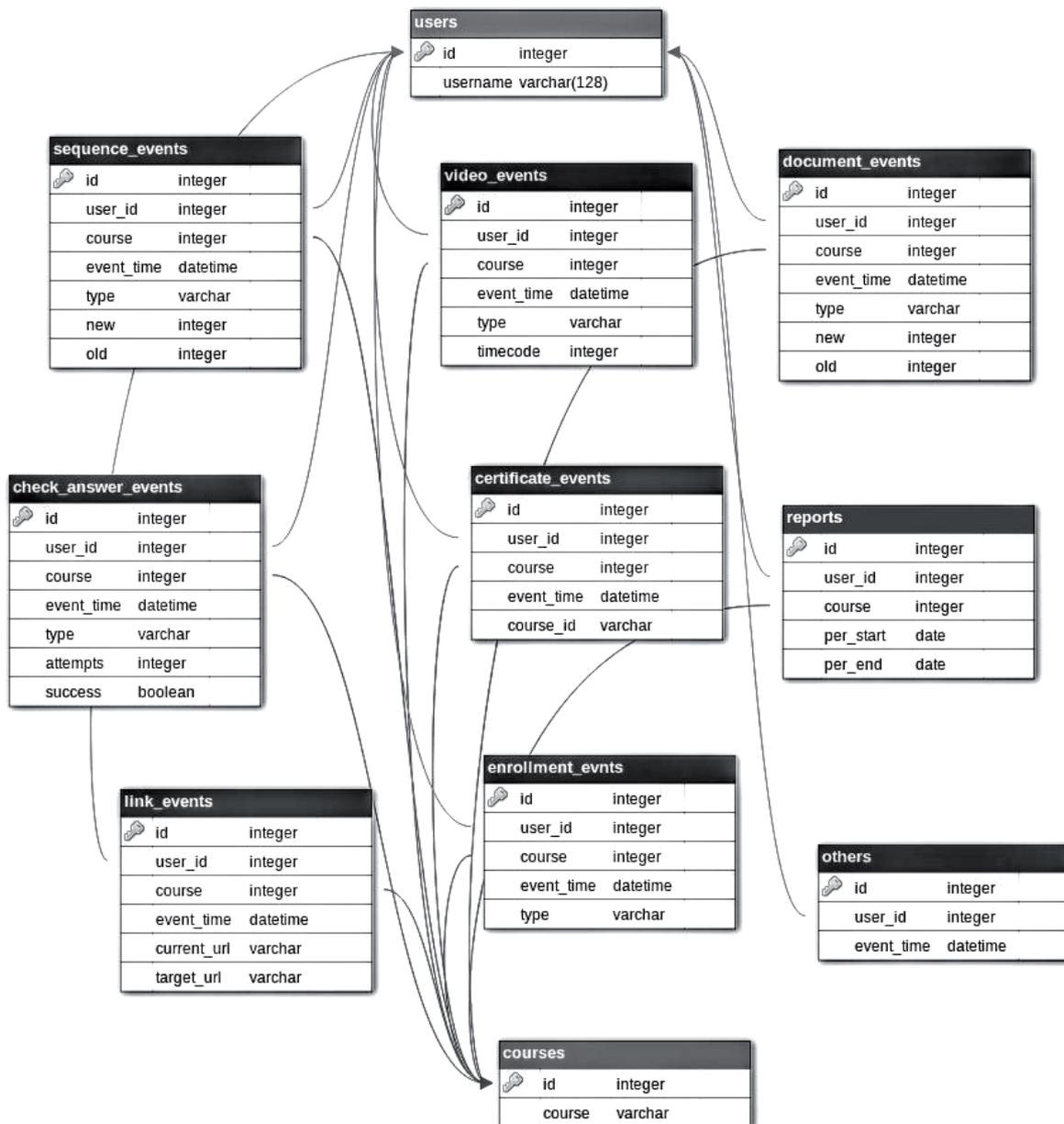


Рисунок 2 – Схема БД для хранения данных из логов

3 Полученные результаты

После автоматизированного наполнения БД для хранения логов с помощью разработанных модулей преобразования *JSON* логов было создано веб-приложение для отображения данных (рисунок 3). Приложение написано на языке *Python* с использованием веб-фреймворка *flask*, в качестве *ORM* (Object-Relational Mapping) использовалась программная библиотека *SQLAlchemy*. В классах моделей *ORM* были описаны соответствующие таблицы в БД. Все дальнейшие взаимодействия с данными происходили посредством интерфейса, предоставляемого пакетом *SQLAlchemy*. Приложение позволяет просмотреть пользователей, курсы и связанные с ними события. Реализованы фильтры по типам событий, пользователям, специфичные фильтры для групп, которые доступны для экспорта.

	Имя пользователя	Идентификатор	Дата и время	Тип события	Количество попыток	Успешность выполнения
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 13:12:46	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 13:11:56	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 13:11:14	проверка ответа	11	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:55:05	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:54:07	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:53:29	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:52:41	проверка ответа	2	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:42:18	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:42:03	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:19:39	проверка ответа	3	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:18:52	проверка ответа	1	👁
👁	12830403	course-v1:ITMO+x1011.00+spring_2016	17.02.2018 12:18:29	проверка ответа	1	👁

Рисунок 3 – Веб-приложение для просмотра активности пользователей

Приложение позволяет получить сводную информацию о том или ином пользователе через интерфейс пользователя. Отчёты формируются в виде *pdf*-документа и содержат информацию о том, когда пользователь был в последний раз в системе, какие курсы он проходит и прошел, сколько это заняло времени и насколько правильно он ответил на вопросы курса, отследить его активность во времени и какими материалами курса он пользовался.

В итоге взаимодействие приложений в подготовке отчётов можно отобразить на схеме, показанной на рисунке 4. Из рисунка видно, что информация, генерируемая системой управления обучением, записывается на сервере в текстовые файлы. Это информация о действиях пользователей в *LMS*. После этого данные обрабатываются с помощью соответствующего модуля парсинга логов и заполняется БД. Для использования данных из БД создаются модели с помощью пакета *SQLAlchemy* для дальнейшей работы с объектами классов языка *Python*. На этой основе строятся веб-приложение для отображения полученных данных и модуль составления отчётов для получения более детальной и структурированной информации с выводами о данных и успехах пользователей.

Для реализации формирования сводных *pdf*-документов (рисунок 5) использован пакет *appu.pod*. Он позволяет создавать динамические документы в форматах *pdf*, *.doc*, *odt*, *.rtf*. Для использования этого пакета заранее создаётся документ-шаблон со всеми заголовками и расстановкой мест для данных, которые должны быть в выходном документе. Отчёты позволяют вывести заранее определённую информацию, например графики активности во времени, соотношения типов событий, собранных в системе, и предпочитаемые материалы.

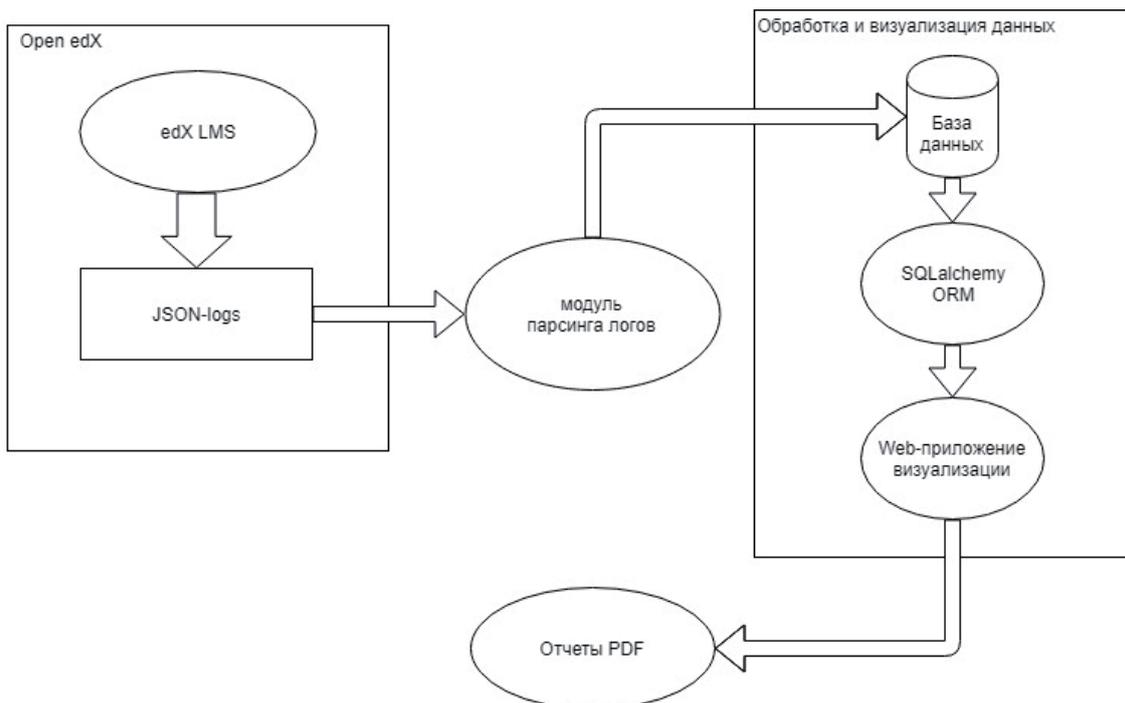


Рисунок 4 – Общая схема взаимодействия приложений в подготовке отчётов

Отчет по пользователю: 12875446

Последняя активность в системе: 18.01.2018 10:48:58

Пользователь зарегистрирован на курсах:

№	Курс	Время на регистрации	Время отмены курса	Время получения сертификата	Правильно/не правильно
1	course-v1:ITMO+X1052.00+spring_2018	17.01.2018 11:13:39	---	---	0/0
2	course-v1:ITMO+X1053.00+spring_2018	17.01.2018 11:14:14	---	---	12/38

График активности:

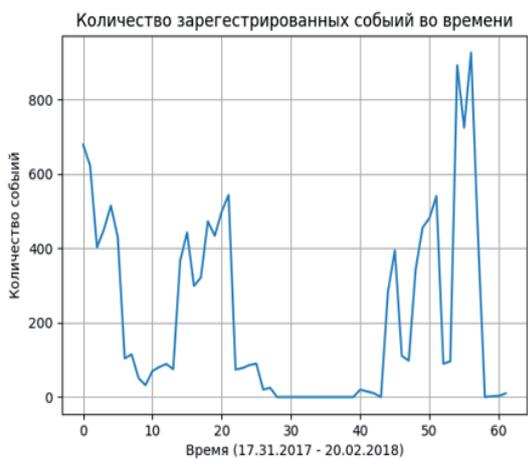


График по типам событий:

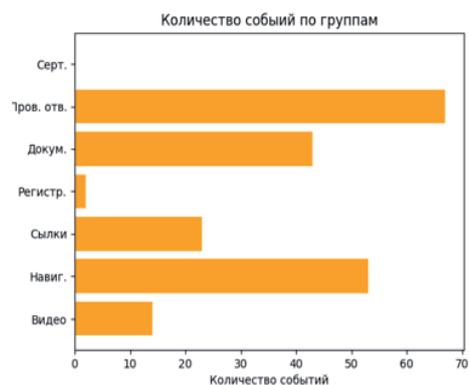


График по предпочитаемым материалам:

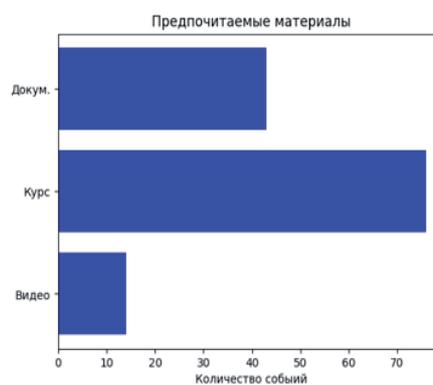


Рисунок 5 – Итоговый отчёт со сводной информацией по пользователю

Заключение

В работе выполнен анализ средств обработки логов пользователей в системах онлайн обучения, предложены и реализованы дополнительные инструменты оценивания поведения пользователей:

- выявление аномального поведения пользователей в системе;
- выявление «проблемных» материалов, на которые пользователи тратили много времени или к которым возвращались несколько раз;
- выявление «пустых» материалов, которые пользователи в большинстве случаев пропускали.

Использование таких инструментов позволяет улучшать качество образовательного контента и вырабатывать рекомендации для создателей курсов, а также создаёт информационную базу для разработки методов построения индивидуальных образовательных траекторий.

Для эксперимента были использованы данные, полученные из платформы *edX* Университета ИТМО. Платформа имеет свою систему логирования событий как со стороны сервера, так и со стороны клиента. Данные были обработаны с помощью написанного программного модуля для их структурирования и визуализации в веб-приложении с целью получения подробной информации о ходе прохождения курсов пользователями. В будущем возможно объединение полученной БД с данными системы обучения о курсах и пользователях для предоставления более детализированной информации. Результаты применимы к другим системам обучения, так как принципы их работы похожи, а для анализа поведения пользователей используется единый метод.

Список источников

- [1] Оценка качества онлайн-курсов. – <http://neorusedu.ru/activity/otsenka-kachestva-onlayn-kursov>.
- [2] **O'Farrell L.** Using Learning Analytics to Support the Enhancement of Teaching and Learning in Higher Education // National Forum for the Enhancement of Teaching and Learning in Higher Education. 2017. – https://www.teachingandlearning.ie/wp-content/uploads/2018/01/Final_LA-Briefing-Paper_Web-with-doi.pdf.
- [3] **Стародубцев, В.А.** Персонализированные MOOK в смешанном обучении // Высшее образование в России. – 2015. – №. 10. – С.133-144.
- [4] **Barnes, L.** Why understanding Learner Behavior Benefits You / L. Barnes. September 2017. - <https://elearningindustry.com/why-understanding-learner-behaviour-benefits-you>.
- [5] **Rosalina Rebucas Estacio, Rodolfo Callanta Raga Jr.** Analyzing students online learning behavior in blended courses using Moodle. June, 2018. – <https://www.emeraldinsight.com/doi/full/10.1108/AAOUJ-01-2017-0016#>.
- [6] **Быстрова, Т.Ю.** Учебная аналитика MOOK как инструмент прогнозирования успешности обучающихся / Т.Ю. Быстрова, В.А. Ларионова, Е.В. Сеницын, А.В. Толмачев // Вопросы образования. – 2018. – №. 4. - С.139–166.
- [7] **Morrison, K.** 8 Essential metrics to use for tracking employee training / Kim Morrison // January 3, 2019. – <https://elearningindustry.com/tracking-employee-training-8-essential-metrics-use>.
- [8] **Ragel, R.G.** Students Behavioural Analysis in an Online Learning Environment Using Data Mining / R.G. Ragel, S. Deegalla. - University of Peradeniya, SriLanka. 2014.
- [9] **Bienkowski, M.** Enhancing Teaching and Learning Through Educational Data Mining and Learning Analytics: An Issue Brief / M. Bienkowski, M. Feng, B. Means. U.S. Department of Education. October, 2012.
- [10] **Yadav, R.K.** Understanding Logs in edX for Monitoring Student Progress / Ravi Kumar Yadav // Department of Computer Science and Engineering Indian Institute of Technology, Bombay. May, 2014.
- [11] How to track e-learning in a LMS. March 30, 2018. – <https://www.getmagicbox.com/blog/how-to-track-elearning-in-lms/>.
- [12] Event tracking logs. EdX research Guide. – https://edx.readthedocs.io/projects/devdata/en/stable/internal_data_formats/tracking_logs.html

Сведения об авторах



Романов Алексей Андреевич, 1989 г. рождения. Окончил Университет ИТМО в 2012 году. Преподаватель Высшей школы цифровой культуры и ассистент факультета программной инженерии и компьютерной техники Университета ИТМО. AuthorID (РИНЦ): 763669; Author ID (Scopus): 57197729875; ORCID 0000-0002-6991-464X; Researcher ID (WoS): K-2908-2015. romanov@itmo.ru.

Волчек Дмитрий Геннадьевич, 1989 г. рождения. Окончил Университет ИТМО в 2012 году. Преподаватель Высшей школы цифровой культуры и ассистент факультета программной инженерии и компьютерной техники

Университета ИТМО. AuthorID (РИНЦ): 766386; Author ID (Scopus): 57197732532; ORCID 0000-0002-0310-1654; Researcher ID (WoS): I-1688-2016. dvolchek@yandex.ru



Поступила в редакцию 02.12.2019, после рецензирования 13.03.2020. Принята к публикации 25.03.2020.

Data Analysis of User Behavior in E-Learning Systems

A. Romanov, D. Volchek

*St. Petersburg National Research University of Information Technologies, Mechanics and Optics,
St. Petersburg, Russia*

Abstract

This article discusses modern e-learning systems that have the ability to record user actions, such as movement, use of interactive materials, registration for courses, their completion, and more. Analysis of user actions in learning management systems provides opportunities for personalization of educational trajectories. Based on the study of user behavior, it becomes possible to form recommendations for course developers to improve the content and structure, as well as recommendations for passing the course for students. This work examines the data contained in activity logs to obtain information, search for dependencies by filtering relevant logs, structuring information from them, and providing data in a form that is convenient for analysis and drawing conclusions. The paper considers data of the main types of events generated as a result of recording user actions in the training management system, and scenarios for using the results of analyzing this data.

Key words: *education, user behavior, e-learning, web programming, databases.*

Citation: *Romanov A, Volchek D. Data Analysis of User Behavior in E-Learning Systems [In Russian]. *Ontology of designing*. 2020; 10(1): 100-111. DOI: 10.18287/2223-9537-2020-10-1-100-111.*

List of figures and tables

- Figure 1 – Activity log example
- Figure 2 – Database schema for storing data from logs
- Figure 3 – Web application for viewing user activity
- Figure 4 – The general scheme of application interaction in the preparation of reports
- Figure 5 – Final report with user summary
- Table 1 – Movement event fields details
- Table 2 – Details of the video interaction event fields
- Table 3 – Details of event fields for interaction with PDF documents
- Table 4 – Following a link event fields details
- Table 5 – Call tips event fields details
- Table 6 – Details of response validation event fields

References

- [1] Evaluating the quality of online courses [In Russian]. <http://neorusedu.ru/activity/otsenka-kachestva-onlayn-kursov>.
- [2] **O'Farrell L.** Using Learning Analytics to Support the Enhancement of Teaching and Learning in Higher Education // National Forum for the Enhancement of Teaching and Learning in Higher Education. 2017. https://www.teachingandlearning.ie/wp-content/uploads/2018/01/Final_LA-Briefing-Paper_Web-with-doi.pdf.
- [3] **Starodubtsev VA.** Personalized MOOCs in mixed training [In Russian]. *Higher education in Russia*. 2015; 10: 133-144.
- [4] **Lorna Barnes.** Why understanding Learner Behavior Benefits You. September 2017. <https://elearningindustry.com/why-understanding-learner-behaviour-benefits-you>.
- [5] **Rosalina Rebus Estacio, Rodolfo Callanta Raga Jr.** Analyzing students online learning behavior in blended courses using Moodle. June, 2018. <https://www.emeraldinsight.com/doi/full/10.1108/AAOUJ-01-2017-0016#>.
- [6] **Bystrova TYu, Larionova VA, Sinitsyn EV, Tolmachev AV.** Educational Analytics of MOOC as a tool for predicting the success of students [In Russian]. *Questions of education*. 2018; 4: 139-166.
- [7] **Kim Morrison.** 8 Essential metrics to use for tracking employee training. January 3, 2019. <https://elearningindustry.com/tracking-employee-training-8-essential-metrics-use>.
- [8] **RG Ragel, S Deegalla.** Students Behavioural Analysis in an Online Learning Environment Using Data Mining. University of Peradeniya, SriLanka. 2014.
- [9] **Marie Bienkowski, Mingyu Feng, Barbara Means.** Enhancing Teaching and Learning Through Educational Data Mining and Learning Analytics: An Issue Brief. U.S. Department of Education. October, 2012.
- [10] **Ravi Kumar Yadav.** Understanding Logs in edX for Monitoring Student Progress. Department of Computer Science and Engineering Indian Institute of Technology, Bombay. May, 2014.
- [11] How to track e-learning in a LMS. March 30, 2018. <https://www.getmagicbox.com/blog/how-to-track-elearning-in-lms/>.
- [12] Event tracking logs. EdX research Guide. – https://edx.readthedocs.io/projects/devdata/en/stable/internal_data_formats/tracking_logs.html.

About the authors

Aleksei Romanov (b. 1989) He graduated from ITMO University in 2012. Lecturer at the Higher school of digital culture and an assistant at the faculty of software engineering and computer engineering at ITMO University. AuthorID (RCI): 763669; Author ID (Scopus): 57197729875; ORCID 0000-0002-6991-464X; Researcher ID (WoS): K-2908-2015. romanov@itmo.ru.

Dmitry Volchek (b. 1989) He graduated from ITMO University in 2012. Lecturer at the Higher school of digital culture and an assistant at the faculty of software engineering and computer engineering at ITMO University. AuthorID (RCI): 766386; Author ID (Scopus): 57197732532; ORCID 0000-0002-0310-1654; Researcher ID (WoS): I-1688-2016. dvolchek@yandex.ru.

Received December 2, 2019. Revised March 13, 2020. Accepted March 25, 2020.

Комплексная верификация продукционных баз знаний с использованием V^{TF} -логик

Л.В. Аршинский, А.А. Ермаков, М.С. Нитежук

Иркутский государственный университет путей сообщения, Иркутск, Россия

Аннотация

Рассматривается комплексная процедура верификации продукционных баз знаний с использованием логик с векторной семантикой в варианте V^{TF} -логик при специальном представлении фактов и правил. Описанная техника позволяет решать такие задачи верификации, как выявление несвязанных фактов, выявление незавершённых продукций, выявление логических кругов, контроль соответствия между множеством гипотез и множеством терминальных фактов, выявление противоречий, выявление молчащих продукций, выявление нештатных обрывов цепочек вывода. Продукции в базе знаний упорядочиваются причинно-следственным образом так, что если один и тот же факт входит в правую часть одной продукция и левую часть другой, первая продукция всегда выполняется раньше. В результате процедура верификации имеет линейную сложность по числу правил и экспоненциальную по числу стартовых фактов. Объём вычислений можно существенно уменьшить, выделяя группы фактов, относящихся к конкретной гипотезе. Новым является применение для верификации аппарата логик с векторной семантикой, которые сохраняют способность к выводу при аномальных значениях истинности. Это позволяет, в частности, использовать машину вывода для динамической верификации знаний. В результате не требуется вводить в систему дополнительные архитектурные элементы (например, таблицы решений), создавать внешние верифицирующие программы и т.п. Получение решения обеспечивается штатными средствами экспертной системы. Статическая верификация обеспечивается специальным представлением фактов и правил.

Ключевые слова: экспертные системы, продукционная модель знаний, верификация, логики с векторной семантикой.

Цитирование: Аршинский, Л.В. Комплексная верификация продукционных баз знаний с использованием V^{TF} -логик / Л.В. Аршинский, А.А. Ермаков, М.С. Нитежук // Онтология проектирования. – 2020. – Т. 10, №1. – С.112-120. – DOI: 10.18287/2223-9537-2020-10-1-112-120.

Введение

Проблема верификации баз знаний (БЗ) экспертных систем (ЭС) продукционного типа известна давно. В [1] отмечается, что некоторые из первых результатов в этом направлении были представлены в [2]. В [3, 4] была подчеркнута важность этой проблемы. Уже на ранних этапах сложился терминологический аппарат, и были сформулированы такие основные требования к качеству БЗ, как полнота (способность работать во всех допустимых входных ситуациях) и отсутствие противоречий в ходе вывода [5-7]. В [7] рассмотрен ряд известных на тот момент систем и методов верификации БЗ и отмечено, что нет универсального подхода к верификации, и полноценной она становится, когда эти методы применяются в комплексе или для специальных БЗ. Сходное мнение выражено в [8]. Это делает, в частности, актуальными работы по верификации БЗ для отдельных предметных областей (см. напр. [9]). Для обнаружения и удаления артефактов было предложено генерировать соответствующие метазнания, тесты [7, 10], использовать подходы, аналогичные доказательству правильности компьютерных программ, разработаны языки спецификаций [11-13].

В частности, в работе [14] рассмотрены вопросы применения логик с векторной семантикой для обнаружения дефектов продукционных БЗ. Основное внимание было уделено артефактам, ведущим к появлению противоречий, обрыву цепочек вывода. Показано, что при использовании машины вывода, работающей на основе логик с векторной семантикой в варианте V^{TF} -логик, в которых истинность формализуется вектором $\langle a^+; a^- \rangle$ (где $a^+, a^- \in [0,1]$; a^+ – степень уверенности, что суждение a истинно, а a^- , – что оно ложно), противоречивость, неопределённость или ложность посылок в общем случае не обрывает вывод, но порождает проходящие через всю последующую цепочку аномальные значения истинности, которые можно отследить. Это позволяет предложить метод поиска артефактов: организация прямого вывода для различных допустимых сочетаний истинности входных фактов с последующей регистрацией аномального результата. Если допустимый вход порождает «неправильный» выход, это считается следствием артефакта. Далее организуется обратная трассировка вывода для обнаружения причин аномалии.

Одной из проблем верификации является экспоненциальный рост вычислений с увеличением объёма БЗ. Верификация во многом реализуется с помощью дополнительных компонентов, выходящих за рамки «штатных» средств ЭС. В предлагаемой работе описан комплекс процедур и алгоритмов, развивающих подход к проблеме верификации, представленный в [14]. При этом верификация может выполняться без включения в состав архитектуры ЭС дополнительных составляющих.

1 Подготовка и статическая верификация БЗ

Подготовка БЗ начинается с формирования множества атомарных утверждений (базы фактов). Каждое утверждение представляется четвёркой:

$$(1) \quad A = (a, C_1, C_2, \|a\|).$$

Здесь a – содержательная часть утверждения, например, «Температура воздуха высокая» или «Скорость объекта больше 30 м/сек», «У пациента наблюдается сыпь» и т.п.; C_1 и C_2 – счётчики числа вхождений A , соответственно, в левую и правую часть правил;

$$(2) \quad \|a\| = \langle [a^+_1; a^+_2]; [a^-_1; a^-_2] \rangle,$$

– вектор истинности a в интервальном представлении. Если a строго истинно, $\|a\| = \langle [1; 1]; [0; 0] \rangle$; если строго ложно – $\langle [0; 0]; [1; 1] \rangle$. Когда свидетельства «за» или «против» a отсутствуют (т.е. имеет место неопределённость), $\|a\| = \langle [0; 0]; [0; 0] \rangle$. Наконец для полностью противоречивого случая $\|a\| = \langle [1; 1]; [1; 1] \rangle$. Первоначальное значение истинности для всех фактов выставляется в неопределённость. Далее она задаётся пользователем или вычисляется в ходе присоединённого вывода.

В конкретной реализации (1) может дополняться чем-либо ещё, что необходимо с точки зрения разработчика, но эти четыре составляющих рассматриваются как необходимые.

Каждое вхождение факта A в ту или иную продукцию сопровождается увеличением соответствующего счётчика C_1 или C_2 на единицу, а его удаление – уменьшением. Это позволяет разбить множество фактов на подмножества:

- стартовые факты ($C_1 > 0, C_2 = 0$);
- промежуточные факты ($C_1 > 0, C_2 > 0$);
- терминальные факты (гипотезы) ($C_1 = 0, C_2 > 0$);
- неиспользуемые факты ($C_1 = 0, C_2 = 0$).

Представление (1) решает проблему выявления неиспользуемых фактов, а также позволяет разработчикам сопоставить фактический набор гипотез с тем, что был запланирован

изначально. Это, в частности, снимает один из вопросов обнаружения недостижимых гипотез [1].

Далее факты связываются в правила R , которые представляются пятёркой:

$$(3) \quad R = (i, K_1, K_2, \|i\|, L).$$

Здесь i – импликация (ядро продукции), K_1 и K_2 – счётчики числа фактов в левой и правой частях правила, $\|i\|$ – истинность ядра, также представляющая собой вектор с интервальными значениями:

$$\|i\| = \langle [i^+_1; i^+_2]; [\bar{i}^-_1; \bar{i}^-_2] \rangle.$$

Значение $\|i\|$ задаётся разработчиками. L – положение (уровень) продукции в причинно-следственном порядке.

Счётчики K_1 и K_2 в (3) позволяют автоматизировать выявление незавершённых правил, в которых отсутствует антецедент или консеквент (также один из возможных артефактов [1]).

Серьёзной проблемой разработки ЭС является рост объёма вычислений с увеличением БЗ, если не пользоваться специальными методами и алгоритмами. Классическим примером такого метода служит алгоритм Rete [15]. Для решения задачи верификации на завершающем этапе формирования БЗ предлагается использовать причинно-следственное упорядочение продукций. Под причинно-следственным упорядочением понимается такая расстановка правил, что если две продукции $R1$ и $R2$ содержат общее утверждение a , причём в одном правиле, например $R1$, оно входит в консеквент, а в другом ($R2$) в антецедент, правило $R1$ при выводе всегда предшествует $R2$. В результате для любой продукции R все логически предшествующие ей продукции выполняются прежде R [16]. Логический вывод при этом выполняется за один проход БЗ, а объём вычислений с увеличением числа правил растёт не экспоненциально, а линейно. Если использовать машину вывода, в том числе для задач верификации [14], это существенно упрощает дело.

Подобное упорядочение при неизменной БЗ выполняется однократно и до начала работы с системой (редактирование БЗ требует переупорядочения).

Упорядочение может выполняться по следующему принципу:

- начальные значения уровней всех правил, антецедент которых состоит только из стартовых фактов, выставляется в 0 (т.е. $L = 0$);
- если в антецедент правила входят факты, входящие в консеквенты правил с уровнями L_1, \dots, L_n , текущему правилу приписывается уровень $L = \max(L_1, \dots, L_n) + 1$;
- процедура выполняется за несколько проходов БЗ, пока не прекратится изменение уровней.

При следовании такому принципу правила одного уровня никогда не ссылаются друг на друга, а правила уровня L никогда не ссылаются на правила с уровнями, предшествующими L . Обеспечивается однопроходность вывода и линейный рост объёма вычислений с увеличением БЗ. Часть вывода, требующая больших затрат времени, переносится на этап настройки БЗ и не затрагивает пользователя ЭС.

Процедура может войти в бесконечный цикл, если в БЗ имеются причинно-следственные петли (выход правила попадает на вход одного из предшествующих правил соответствующей цепочки вывода), что требует ограничения на число возможных уровней, которое при отсутствии петель равно длине максимальной цепочки. Если задать достаточно большое ограничивающее число, его достижение для какой-то группы продукций сигнализирует о наличии причинно-следственной петли в этой группе – потенциальном логическом круге. Это позволяет заранее выявлять подобные аномалии.

Таким образом, на этапе формирования БЗ контролируются такие статические аномалии, как несвязанные факты (включая гипотезы), незавершённые продукции, логические круги,

выявляется принадлежность гипотез множеству терминальных фактов (т.е. действительно ли они гипотезы для машины вывода). Дополнительный контроль соответствия между терминальными фактами и гипотезами обеспечивается также причинно-следственным упорядочением правил.

2 Динамическая верификация БЗ

Под динамической верификацией понимается выявление артефактов, способных проявиться только при работе машины вывода. К таким артефактам относятся получение противоречивого заключения, незапланированный обрыв цепочки вывода, «молчание» (никогда не запускающиеся) правила.

В [14] эти ситуации обсуждались с точки зрения их выявления средствами V^{TF} -логик. Основная идея сводится к тому, что ситуации, порождающие эти артефакты, как то:

- совместное получение утверждения и отрицания какого-либо факта (противоречие),
- постоянно ложный антецедент или (в векторном случае) постоянно неопределённый антецедент (все компоненты вектора истинности близки к нулю) - не препятствуют процедуре логического вывода, но порождают заключения с аномальными значениями истинности соответствующего типа. Этот факт устанавливается на завершающем этапе вывода выявлением гипотез с такими же аномальными значениями истинности. Обратной трассировкой цепочки вывода можно установить причину аномалии. Важной составляющей такого вывода служит процедура объединения свидетельств, которая в V^{TF} -случае опирается на одну из следующих операций [16].

- «10»-композицию, которой называется суждение (первая форма дизъюнкции), образованное из a и b с вектором истинности:

$$\|a \vee b\| = \langle a^+ \oplus b^+; a^- \bullet b^- \rangle.$$

- «01»-композицию с вектором истинности (первая форма конъюнкции):

$$\|a \& b\| = \langle a^+ \bullet b^+; a^- \oplus b^- \rangle.$$

- «11»-композицию с вектором истинности (вторая форма дизъюнкции):

$$\|a \vee_2 b\| = \langle a^+ \oplus b^+; a^- \oplus b^- \rangle.$$

- «00»-композицию с вектором истинности (вторая форма конъюнкции):

$$\|a \&_2 b\| = \langle a^+ \bullet b^+; a^- \bullet b^- \rangle.$$

Первые формы дизъюнкции и конъюнкции аналогичны дизъюнкции и конъюнкции в классическом и нечётком случаях, вторые формы – специфика V^{TF} -логик. Здесь \bullet и \oplus – триангулированная (треугольные – *triangle*) норма и ко-норма в инфиксной записи с дополнительным свойством [17]:

$$x \bullet y = 1 - (1 - x) \oplus (1 - y);$$

либо:

$$x \oplus y = 1 - (1 - x) \bullet (1 - y).$$

Это свойство связывает векторное и нечёткое представления истинности в случае, когда $a^- + a^+ = 1$ [18].

В [14] перечислены возможные стратегии объединения свидетельств, часть которых способствует выявлению артефактов. С точки зрения выявления противоречий подходят следующие две.

- 1) объединение по схеме «10»/«11»-композиции – стратегия *нарастающего доверия*. Свидетельства в пользу истинности b , как и свидетельства в пользу истинности $\neg b$, объединя-

ются отдельно по правилу «10»-композиции (дизъюнкции). Полученные после этого векторы $\|b\|$ и $\|\neg b\|$ «складываются» в единый вектор по правилу «11»-композиции. При таком объединении доверие к b возрастает по мере получения утвердительных доказательств факта, доверие к $\neg b$ – по мере вывода его отрицаний. Итоговый вектор объединяет свидетельства как за, так и против.

- 2) объединение по схеме «11»-композиции – стратегия *полного доверия*. Доверяем и подтверждающим и опровергающим свидетельствам, это упрощённый вариант стратегии нарастающего доверия. Здесь быстрее растёт степень противоречивости промежуточных фактов и гипотез, что проще с алгоритмической точки зрения.

Оба подхода при получении взаимоисключающих утверждений b и $\neg b$ генерируют вектор истинности, фиксирующий противоречие. Т.е., если по некоторой ветви вывода получено заключение b с истинностью $\|b\| = \langle 1; 0 \rangle$, а по другой $\neg b$ также с истинностью $\|\neg b\| = \langle 1; 0 \rangle$, то объединение на основе «11»-композиции даёт (с учетом, что $\|\neg b\| = \langle b^-; b^+ \rangle$):

$$\|b\| = \langle 1 \oplus 0; 0 \oplus 1 \rangle = \langle 1; 1 \rangle - \text{полное противоречие.}$$

Возможна ситуация, когда противоречие имеет неявную форму: из одних и тех же посылок генерируются логически несовместимые заключения b и c [1, 19]. Введение дополнительной продукции с ядром $b \rightarrow \neg c$ (или $c \rightarrow \neg b$) позволит выявить и этот случай.

Верификация выполняется в ходе прямого вывода и предполагает перебор всех допустимых значений истинности входных фактов. Поскольку предлагаемая модель истинности непрерывна, следует ограничиться конечным набором проверочных комбинаций. Чтобы избежать маскирующего влияния нестрогих значений истинности, истинность малых посылок целесообразно задавать в виде допустимых комбинаций строгой истины $\langle [1; 1]; [0; 0] \rangle$ и/или строгой лжи $\langle [0; 0]; [1; 1] \rangle$ при строгой истинности импликации.

В результате на этом этапе возможно выявление таких артефактов, как противоречие и неустранимая ложность или неопределённость антецедентов продукций [14]. Последнее в классическом случае приводит к обязательному обрыву цепочки вывода, а в рассматриваемой технике порождает аномальную истинность гипотезы, сигнализирующую об артефакте.

Если дополнить (3) счётчиком числа обращений к продукции, легко установить «молчащие» продукции, к которым не было ни одного обращения. Однако при использовании причинно-следственного порядка этот артефакт можно обнаружить уже на этапе статического анализа.

3 Снижение вычислительной сложности

Остаётся прояснить вопрос о процедуре выставления значений истинности стартовых фактов – посылок. В предельном случае N посылок порождают 2^N комбинаций истинности. Даже для небольшой БЗ, содержащей несколько десятков входных фактов, это приводит к взрывообразному росту числа проверок. Простой перебор может оказаться невозможным. Количество проверок необходимо уменьшить без потери полноты перебора. Для решения этой задачи предлагается использовать то обстоятельство, что в основе каждой гипотезы лежит относительно небольшое количество стартовых посылок. Тогда для каждой гипотезы можно выделять только относящиеся к ней факты. Поскольку в реальных задачах обычно n существенно меньше N , где n – число стартовых фактов гипотезы, получаем кардинальное снижение вычислительной нагрузки. Далее приводится дополненная схема алгоритма, представленного в [14].

- 1) объявить все продукции строго истинными (запомнив, при необходимости, значения истинности, выставленные экспертами).

- 2) выделить терминальный факт (гипотезу).
- 3) установить все связанные с ним стартовые факты.
- 4) выставить значения истинности стартовых фактов в одну из допустимых комбинаций - $\langle 1; 0 \rangle$ или $\langle 0; 1 \rangle$, - а истинность промежуточных и терминальных - в $\langle 0; 0 \rangle$.
- 5) выполнить прямой логический вывод, получив истинность гипотезы.
- 6) проверить истинность заключения на допустимость.
- 7) если заключение недопустимо:
 - 7.1) выполнить обратную трассировку вывода для выяснения источника проблемы – артефакта БЗ;
 - 7.2) устранить артефакт (или внести его в соответствующий список);
 - 7.3) вернуться к шагу 4.
- 8) если не все возможные входные значения посылок проверены, вернуться к шагу 4.
- 9) если не все возможные гипотезы проверены, вернуться к шагу 2.
- 10) вернуть исходные значения истинности продукций.
- 11) стоп.

Такая верификация не требует генерирования дополнительных структур данных, например таблиц принятия решений или их аналогов.

Заключение

В работе представлена комплексная процедура верификации продукционных БЗ на основе V^{TF} -логик с введением в факты и правила дополнительных элементов, а также специальным упорядочением правил. Решаются следующие вопросы:

- выявление несвязанных фактов;
- выявление незавершённых продукций;
- выявление логических кругов;
- контроль соответствия между множеством гипотез и множеством терминальных фактов;
- выявление противоречий;
- выявление «молчащих» продукций;
- выявление нештатных обрывов цепочек вывода.

Верификация имеет линейную сложность по числу правил и экспоненциальную по числу стартовых фактов. Объём вычислений можно существенно уменьшить, выделяя факты, относящиеся к конкретной гипотезе.

Представленная процедура не исчерпывает всех проблем верификации (например, дублирования правил) и применима к заранее спроектированным под неё БЗ. Однако при этом не требуется введение в состав ЭС дополнительных архитектурных элементов в виде таблиц решений и т.п. Полученные решения задачи обеспечиваются штатными средствами ЭС.

Список источников

- [1] *Nguyen, T.A.* Knowledge Base Verification / T.A. Nguyen, W.A. Perkins, J.T. Laffey, D. Pecora // AI Magazine. - 1987. - V. 8, № 2. – P. 69-75.
- [2] *Davis, R.* Applications of Meta-Level Knowledge to the Construction, Maintenance, and Use of Large Knowledge Bases / R. Davis // Ph D. diss, Dept of Computer Science, Stanford Univ., 1976.
- [3] *O'Leary, D.* Design, development and validation of expert systems: A survey of developers / D. O'Leary // In M. Ayel and J-P. Laurent (Eds.) Validation, Verification and Test of Knowledge-Based Systems. 1991. – P. 3-20.
- [4] *Hamilton, D.* State-of-the-practice in knowledge-based system verification and validation / D. Hamilton, K. Kelley, C. Culbert // Expert Systems with Applications. - 1991. – 3. - P. 403–410.
- [5] *Benbasat, I.* A framework for the validation of knowledge acquisition / I. Benbasat, J.S. Dhaliwal // Knowledge Acquisition. - 1989. – 1. - P. 215–233.

- [6] **Laurent, J.-P.** Proposals for a valid terminology in KBS validation / J.-P. Laurent // In B. Neuman (Ed.) Proceedings of the 10th European Conference on Artificial Intelligence (ECAI-92), 1992. - P. 829–834.
- [7] **Preece, A.D.** Evaluation of Verification Tools for Knowledge-Based Systems / A.D. Preece, S. Talbot, L. Vignollet // Int. J. Hum.-Comput. Stud. - 1997. - V. 47. – P. 629-658.
- [8] **Логунова, Е.А.** Обзор подходов к разрешению недостатков продукционной базы знаний системы логического вывода / Е.А. Логунова // Современные наукоемкие технологии. - 2015. - № 9. – С. 46-48.
- [9] **Racunas, S.A.** A case study in pathway knowledge base verification / S.A. Racunas, N.H. Shan, N.V. Fedoroff // BMC Bioinformatic. – 2006. 7: 196. (Published online 2006 Apr 8). - DOI: 10.1186/1471-2105-7-196.
- [10] **Bindilatti, A. de A.** Verification and validation of knowledge bases using test cases generated by restriction rules / A. de A. Bindilatti, A.E.A da Silva // International Journal of Artificial Intelligence and Expert Systems. - 2012. - V. 3, Issue 4. – P. 117-125.
- [11] **Antonioni, G.** Verification and Validation of Knowledge-Based Systems. Report on Two 1997 Events / G. Antonioni, F. van Harmelen, R. Plant, J. Vanthienen // AI Magazine: Workshop Report. - 1998. - V.19, № 3. – P. 123-126.
- [12] **Бобина, В.А.** Верификация и подтверждение правильности (V&V) системы управления базой знаний с помощью формальных спецификаций / В.А. Бобина // Международный научный журнал «Символ науки». - 2015. - №5. – С.17-20.
- [13] **Терновой, М.Ю.** Формальная спецификация свойств баз нечетких знаний Мамдани на основе метаграфа / М.Ю. Терновой, Е.С. Штогрин // Вісник Харківського національного університету імені В.Н. Каразіна. Серія «Мат. моделювання. Інформаційні технології. Автоматизовані системи управління», 2015. - Вып. 27. – С. 157-171.
- [14] **Аршинский, Л.В.** Логика с векторной семантикой как средство верификации баз знаний / Л.В. Аршинский, А.А. Ермаков, М.С. Нитежук // Онтология проектирования. - 2019. - Т.9, № 4(34). – С. 510-521. - DOI: 10.18287/2223-9537-2019-9-4-510-521.
- [15] **Forgy, Ch.L.** Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem / Ch.L. Forgy // Artificial Intelligence. - 1982. - V. 19, № 1. - P 17-37.
- [16] **Аршинский, Л.В.** Особенности работы машины вывода системы моделирования правдоподобных рассуждений «Гераклит» / Л.В. Аршинский // Информационные и математические технологии в науке и управлении. - 2016. - №2. – С.18-29.
- [17] **Gottwald, S.** Treatise on Many-Valued Logics / S. Gottwald. - Leipzig, 2000. - 604 p.
- [18] **Аршинский, Л.В.** Содержательный и формальный выводы в логиках с векторной семантикой / Л.В. Аршинский // Автоматика и телемеханика. - 2007. - № 1. - С. 153-162.
- [19] **Проскуряков, Д.П.** Управление разрешением конфликтов в продукционных экспертных системах / Д.П. Проскуряков // Вестник Иркутского государственного технического университета. - 2015. - № 8. - С. 47-51.

Сведения об авторах



Аршинский Леонид Вадимович, 1957 г. рождения. Окончил Иркутский государственный университет в 1979 г., д.т.н. (2008). Заведующий кафедрой «Информационные системы и защита информации» Иркутского государственного университета путей сообщения. В списке научных трудов более 200 работ в области околоэкранный аэродинамики, распознавания образов, моделирования правдоподобных рассуждений, агрегированного оценивания систем и др. AuthorID (РИНЦ): 520252; Author ID (Scopus): 57193195356; Researcher ID (WoS): C-3869-2013; ORCID 0000-0001-5135-7921. larsh@mail.ru.

Ермаков Анатолий Анатольевич, 1951 г. рождения. Окончил Киевское высшее военное авиационное инженерное училище в 1976 г., к.т.н. (1982). Профессор кафедры «Информационные системы и защита информации» Иркутского государственного университета путей сообщения. В списке научных трудов около 70 работ в области технической диагностики, надежности и эксплуатации сложных систем, распознавания образов и др. AuthorID (РИНЦ): 861476; Researcher ID (WoS): AАН-4024-2020; ORCID 0000-0002-9312-9916. anatolij.ermackow2011@yandex.ru.



Нитежук Марина Сергеевна, 1984 г. рождения. Окончила Иркутский государственный университет путей сообщения в 2010 г. Старший преподаватель и аспирант кафедры «Информационные системы и защита информации» Иркутского государственного университета путей сообщения. В списке научных трудов 9 работ в области информационных систем и технологий. AuthorID (РИНЦ): 1007958; Author ID (Scopus): 57214999113; Researcher ID (WoS): AАН-4012-2020; ORCID 0000-0003-3420-8312. *marino_@mail.ru*.



Поступила в редакцию 10.02.2020, после рецензирования 10.03.2020. Принята к публикации 15.03.2020.

Complex verification of rule-based knowledge bases using V^{TF} -logic

L.V. Arshinskiy, A.A. Ermakov, M.S. Nitezjuk

Irkutsk State Transport University, Irkutsk, Russia

Abstract

In this paper, we consider a complex procedure for verifying rule-based knowledge bases using logics with vector semantics in the form of V^{TF} -logic with a special representation of facts and rules. The described technique allows solving verification tasks such as identifying unrelated facts, identifying incomplete rules, identifying logical circles, checking the correspondence between a set of hypotheses and a set of terminal facts, identifying contradictions, identifying silent rules, and identifying abnormal breaks in the inference chains. Rules in the knowledge base are arranged in a causal way. It means if the same fact is included in the right part of one rule and the left part of another, the first rule is always performed earlier. As a result, the verification procedure is linear in terms of the number of rules and exponential in terms of the number of starting facts. However, the amount of calculations can be significantly reduced by selecting groups of facts related to a particular hypothesis. The novelty of the study is the use of logics with vector semantics for rules verification. This retains the ability to inference in case of anomalous values of truth. This logic provides to use the inference engine for dynamic verification of knowledge. As a result, you do not need to include additional architectural elements into the system (for example, rule set dependency chart), create external verification programs, etc. In fact everything is provided by regular means of the expert system. Static verification is provided by a special representation of facts and rules.

Keywords: *expert systems, rule-based systems, verification, logic with vector semantics*

Citation: *Arshinskiy LV, Ermakov AA, Nitezjuk MS. Complex verification of rule-based knowledge bases using V^{TF} -logic [In Russian]. *Ontology of designing*. 2020; 10(1): 112-120. DOI: 10.18287/2223-9537-2020-10-1-112-120.*

References

- [1] *Nguyen TA, Perkins WA, Laffey JT, Pecora D.* Knowledge Base Verification. *AI Magazine*; 1987; 8(2): 69-75.
- [2] *Davis R.* Applications of Meta-Level Knowledge to the Construction, Maintenance, and Use of Large Knowledge Bases. Ph D. diss, Dept of Computer Science, Stanford Univ.; 1976.
- [3] *O'Leary D.* Design, development and validation of expert systems: A survey of developers. In M. Ayel & J-P. Laurent, Eds. *Validation, Verification and Test of Knowledge-Based Systems*; 1991: 3-20.
- [4] *Hamilton D, Kelley K, Culbert C.* State-of-the-practice in knowledge-based system verification and validation. *Expert Systems with Applications*; 1991; 3: 403-410.
- [5] *Benbasat I, Dhaliwal JS.* A framework for the validation of knowledge acquisition. *Knowledge Acquisition*; 1989; 1: 215-233.
- [6] *Laurent J-P.* Proposals for a valid terminology in KBS validation. In B. Neuman, Ed. *Proceedings of the 10th European Conference on Artificial Intelligence (ECAI-92)*; 1992; 829-834.
- [7] *Preece AD, Talbot S, Vignollet L.* Evaluation of Verification Tools for Knowledge-Based Systems. *Int. J. Hum.-Comput. Stud*; 1997; 47: 629-658.
- [8] *Logunova EA* Approaches to resolving the deficiencies of production knowledge base of inference system [in Russian]. *Modern High Technologies*; 2015; 9: 46-48.

- [9] **Racunas SA, Shan NH, Fedoroff NV.** A case study in pathway knowledge base verification. *BMC Bioinformatic*; 2006, 7: 196. DOI: 10.1186/1471-2105-7-196.
 - [10] **Bindilatti AA, da Silva AEA.** Verification and validation of knowledge bases using test cases generated by restriction rules. *International Journal of Artificial Intelligence and Expert Systems (IJAE)*; 2012; 3(4): 117-125.
 - [11] **Antonioni G, van Harmelen F, Plant R, Vanthienen J.** Verification and Validation of Knowledge-Based Systems. Report on Two 1997 Events. *AI Magazine: Workshop Report*, 1998; 19(3): 123-126.
 - [12] **Bobina VA.** Verification and validation (V&V) of the knowledge base management system using formal specifications [in Russian]. *International Scientific Journal "Symbol of Science"*; 2015; 5: 17-20.
 - [13] **Ternovoy MYu, Shtogrina ES.** Formal specification of properties of Mamdani fuzzy knowledge bases based on metagraph [in Russian]. *Kharkov University bulletin*; 2015; 27: 157-171.
 - [14] **Arshinskiy LV, Ermakov AA, Nitezuk MS.** Logic with vector semantic as a means of knowledge bases verification [In Russian]. *Ontology of Designing*; 2019; 9(4): 510-521. DOI: 10.18287/2223-9537-2019-9-4-510-521.
 - [15] **Forgy ChL.** Rete: A Fast Algorithm for the Many Pattern / Many Object Pattern Match Problem. *Artificial Intelligence*; 1982; 19(1): 17-37.
 - [16] **Arshinskiy LV** Features of the inference engine of the plausible reasoning modeling system "Heraclitus" [in Russian]. *Information and Mathematical Technologies in Science and Management*; 2016; 2: 18-29.
 - [17] **Gottwald S.** Treatise on Many-Valued Logics. Research Studies Press; 2001.
 - [18] **Arshinskii LV.** Substantial and formal deductions in logics with vector semantics. *Automation and remote control*; 2007; 68(1): 139-148.
 - [19] **Proskuriakov DP.** Conflict resolution management in rule-based expert systems [in Russian]. *Proceedings of Irkutsk State Technical University*; 2015; 8: 47-51.
-

About the authors

Leonid Vadimovich Arshinskiy (b. 1957) graduated from the Irkutsk State University (Irkutsk-city) in 1979, Dr of Tech. Sc. (2008). He is Head of the Department "Information Systems and Information Security" Irkutsk State Transport University. Corresponding Member of the Russian Academy of Natural Sciences and Russian Engineering Academy. He is author of more than 200 scientific articles and abstracts in the field of aircraft with ground effect wings, image recognizing, plausible inference, aggregate evaluation of systems etc. AuthorID (RCI): 520252; Author ID (Scopus): 57193195356; Researcher ID (WoS): C-3869-2013; ORCID 0000-0001-5135-7921. larsh@mail.ru.

Ermakov Anatoly Anatolyevich (b. 1951). He graduated from the Kiev Higher Military Aviation Engineering School in 1976, Ph.D. (1982). Professor, Department "Information Systems and Information Security", Irkutsk State Transport University. The list of scientific papers includes about 70 works in the field of technical diagnostics, reliability and operation of complex systems, pattern recognition, etc. AuthorID (RCI): 861476; Researcher ID (WoS): AAH-4024-2020; ORCID 0000-0002-9312-9916. anatolij.ermackow2011@yandex.ru.

Nitezuk Marina Sergeevna (b. 1984) graduated from the Irkutsk State Transport University in 2010. Senior lecturer and graduate student of the department "Information Systems and Information Security" of Irkutsk State Transport University. The list of scientific works includes 10 works in the field of information systems and technologies. AuthorID (RCI): 1007958; Author ID (Scopus): 57214999113; Researcher ID (WoS): AAH-4012-2020; ORCID 0000-0003-3420-8312. marino_@mail.ru.

Received February 10, 2020. Revised March 10, 2020. Accepted March 15, 2020.

МЕТОДЫ И ТЕХНОЛОГИИ ПРИНЯТИЯ РЕШЕНИЙ

УДК 004.89

DOI: 10.18287/2223-9537-2020-10-1-121-140

Поддержка принятия решений при управлении программными проектами на основе нечёткой онтологии

В.В. Антонов¹, О.В. Бармина², Н.О. Никулина¹

¹Уфимский государственный авиационный технический университет, Уфа, Россия

²ООО Нордиджи, Санкт-Петербург, Россия

Аннотация

Рассмотрены вопросы управления знаниями при выполнении программных проектов и организации поддержки принятия решений для их участников. Актуальность исследований обоснована необходимостью снизить степень неопределённости при принятии решений и подтверждена статистикой выполнения проектов. Поддержку принятия решений предложено организовать на основе анализа прецедентов ранее происходивших проблемных ситуаций и поиска оптимального решения в текущей ситуации, для чего предложено использовать методы инженерии знаний, в частности, онтологического анализа предметной области. Разработана методика формирования нечёткой онтологии поддержки принятия решений при управлении программными проектами. Новизна модели поиска прецедентов заключается в сочетании различных механизмов логического вывода при принятии решений на базе всего комплекса знаний о предметной области. Большой объём информации не позволяет использовать простые поисковые запросы для оперативного поиска необходимой информации, созданной в процессе выполнения проекта. Предложен алгоритм формирования поискового запроса, учитывающего нечёткие свойства, нечёткие таксономические и ассоциативные отношения между классами объектов в онтологии. Эффективность предложенных решений подтверждена экспериментом на прототипе интеллектуальной системы поддержки принятия решений.

Ключевые слова: управление проектами, нечёткая онтология, управление знаниями, поисковый запрос, поддержка принятия решений, прецедент проблемной ситуации.

Цитирование: Антонов, В.В. Поддержка принятия решений при управлении программными проектами на основе нечёткой онтологии / В.В. Антонов, О.В. Бармина, Н.О. Никулина // Онтология проектирования. – 2020. – Т.10, №1(35). – С.121-140. – DOI: 10.18287/2223-9537-2020-10-1-121-140.

Введение

Одним из направлений развития проектно-ориентированных ИТ-компаний является сосредоточение на извлечении уроков из успешных отраслевых проектов [1, 2]. Так, в Руководстве к своду знаний по управлению проектами активами организации, использование которых оказывает благоприятное влияние на достижение целей проекта, считаются: политики, правила и процедуры, а также корпоративная база знаний (БЗ) [3]. При этом «репозитории знаний организации, предназначенные для хранения и извлечения информации, включают в себя, среди прочего, репозитории данных по управлению проблемами, содержащие сведения о статусе проблем, информацию о контроле, данные о разрешении проблем, а также результаты предпринятых действий» [3, с.41]. Совокупность этих данных можно назвать прецеден-

тами проблемных ситуаций (ПС) [4]. Такая информация должна быть зафиксирована и доступна всем участникам проекта в рамках их компетенции и полномочий.

Программные проекты (ПП) обладают рядом особенностей, среди которых – высокая степень уникальности конечного продукта, многообразие методов и средств, применяемых при разработке программного обеспечения (ПО), затрудняющее их выбор для реализации конкретного проекта, плохо формализуемые взаимодействия между участниками проекта, большая доля творческой составляющей в деятельности человека. Всё это диктует необходимость применения инструментов, позволяющих не только управлять разработкой ПО (своевременно отслеживать статус решаемых задач, процент их выполнения и объём затраченных ресурсов, вести журнал выявленных и исправленных ошибок), но и накапливать информацию для устранения ПС. Возникновение ПС в ПП часто связывают с [5, 6]:

- неточным определением сроков и стоимости проекта – 62%;
- нарушением взаимодействия между различными участниками проекта – 47%;
- отсутствием интеграции между различными системами, используемыми при управлении ПП (системами календарно-ресурсного планирования, системами отслеживания ошибок, системами бюджетирования) – 38%;
- отсутствием единого централизованного источника информации о проекте – 35%;
- неудовлетворительным управлением ресурсами – 31%;
- отсутствием видимости результатов в ещё не завершённых задачах – 21%.

Возможность использовать знания и опыт, накопленные при управлении ПП в различных ситуациях, позволяет быстрее и более качественно решать возникающие проблемы. Однако всего лишь 14% компаний применяют этот подход на практике, что связано со сложностью представления и формализации знаний в этой области, высокой неопределённостью в принятии решений [1, 5, 7].

1 Нечёткая онтология как модель представления знаний

Неопределённость и неточность могут рассматриваться как две противоположные точки зрения на неполноту информации. Информация, как правило, выражается в виде логического высказывания, содержащего предикаты, квантификаторы. Под БЗ понимается множество сведений, имеющих у субъекта (группы субъектов) или содержащихся в информационной системе и относящихся к одной и той же предметной области (ПрО). Любое высказывание может рассматриваться как утверждение, относящееся к появлению некоторого события. Известны три эквивалентных способа анализа множества данных в зависимости от того, делается ли акцент на структуре (логическая точка зрения), содержании (теоретико-множественная точка зрения) этой информации или на её отношении к действительным фактам (событийная точка зрения) [8].

Информационную единицу знания можно представить в виде четвёрки:

$$(1) \quad K = \langle O, A, V, C \rangle,$$

где K – информационная единица знания (*knowledge*), O – объект (*object*), A – признак (*attribute*), V – значение (*value*), C – уверенность (*certainty*).

Под признаком понимается функция, задающая значение (множество значений) объекта или предмета, название которого упоминается в информационной единице знания. Значение соответствует некоторому предикату. Уверенность есть показатель надёжности информационной единицы знания. Принятие решения в большинстве случаев заключается в генерации возможных альтернативных решений, их оценке и выборе наилучшего варианта, иногда с привлечением экспертов или других способов информационной поддержки.

При выборе варианта приходится учитывать большое число неопределённых и противоречивых факторов [9, 10]. Неопределённость является неотъемлемой частью процессов принятия решений и может быть связана с:

- неполнотой знаний о проблеме, по которой должно быть принято решение;
- невозможностью полного учёта реакции окружающей среды на принимаемые решения;
- непониманием лицом, принимающим решение (ЛПР), своих целей.

Противоречивость возникает из-за неоднозначности оценки ситуаций, ошибки в выборе приоритетов, что осложняет принятие решений. Исследования показывают, что ЛПР без дополнительной аналитической поддержки, как правило, использует упрощённые, а иногда и противоречивые правила выбора решения [9]. Кроме того, наличие разнородных источников информации также затрудняет процесс принятия решений. Основная проблема, возникающая при попытках повторного использования знаний и передового опыта, заключается в сложности выбора успешных решений, опыт которых применим к конкретной решаемой задаче. Эта проблема может быть решена путём систематизации знаний и моделирования ПрО с позиций методов, учитывающих нечёткость описаний модели исследуемого объекта [11-13]. Снижению неопределённости способствует систематизация знаний о деятельности ЛПР, повышение интеграции источников знаний и создание единого информационного пространства, использование онтологий [14].

Онтологию можно представить в виде неструктурированного описания на естественном языке (*not formal ontology*), в виде полуструктурированного описания как графической схемы (*semi-formal ontology*), в виде структурированного описания на языке *OWL*, *RDF* (*formal ontology*). Онтология предназначена для решения проблем семантического различия информации, поступающей из различных источников, предоставления возможностей автоматического вывода знаний и уменьшения неопределённости знаний в ПрО. Примеры применения онтологий в различных ПрО приведены в [14-16]. Онтологическая модель описания прецедентов ПС, встречающихся в ПП, подробно описана в [17]. Чаще всего онтологии используют для уменьшения неопределённости легко структурируемых знаний, примерами могут служить таксономии с чёткими концептами, а также знания, основанные на правилах. В случае трудно структурируемых знаний применяются онтологии, основанные на нечёткой логике, теории вероятностей и теории возможностей [8, 18].

Нечёткие знания характерны для многих ПрО, в том числе таких как интеллектуальный анализ информации, машинное обучение, которые оперируют неточными или неполными знаниями. Нечёткая онтология может быть представлена в виде:

$$(2) \quad O = \langle C, I, P, T, N, A \rangle,$$

где C – множество нечётких понятий (классов), каждое понятие является нечётким множеством в области экземпляров, такое, что $C : I \rightarrow [0, 1]$; I – экземпляры (элементы классов). Множество объектов нечёткой онтологии определяется как $E = CUI$; P – множество нечётких свойств класса C ; T – множество нечётких таксономических отношений, такое что $T : E \rightarrow [0, 1]$; N – множество нечётких не таксономических ассоциативных отношений $N : E \rightarrow [0, 1]$; A – множество нечётких аксиом.

Нечёткая онтология может быть создана в редакторе онтологий *Protege 5.2.0* [19] с использованием плагина *FuzzyOWL*, основные элементы которого представлены в таблице 1.

Синтаксис *FuzzyOWL* предполагает три формата алфавита: для представления нечётких классов (*fuzzy concepts*), нечётких отношений (*fuzzy roles*) и нечётких объектов классов онтологии (*fuzzy individuals*), которые определяются свойством аннотации *fuzzyLabel*, хранящей параметры и значения функции принадлежности, соотносимые с конкретным классом, объектом, отношением внутри заданного нечёткого множества данных [18, 20].

Таблица 1– Основные элементы *FuzzyOWL*

Элемент <i>FuzzyOWL</i>	Формат	Пример представления в нечёткой онтологии <i>FuzzyOWL</i> в виде <i>OWL/XML</i> -разметки
<i>Fuzzy ontologies</i> - множество операторов нечётких множеств	<pre><FUZZY_LOGIC> := "zadeh" <FUZZY_LOGIC> := "lukasiewicz"</pre>	<pre><Annotation> <AnnotationProperty IRI="#fuzzyLabel"/> <Literal datatypeIRI="http://www.w3.org/1999/02/22- rdf-syntax-ns#PlainLiteral"><fuzzyOwl2 fuzzyType="ontology";>; <FuzzyLogic logic="zadeh"; />; </fuzzyOwl2>; </Literal> </Annotation></pre>
<i>Fuzzy datatypes</i> - множество функций степени принадлежности	<pre><DATATYPE>:= <Datatype type="leftshoulder" a="<DOUBLE>" b="<DOUBLE>" /> <Datatype type="rightshoulder" a="<DOUBLE>" b="<DOUBLE>" /> <Datatype type="triangular" a="<DOUBLE>" b="<DOUBLE>" c="<DOUBLE>" /> <Datatype type="trapezoidal" a="<DOUBLE>" b="<DOUBLE>" c="<DOUBLE>" d="<DOUBLE>" /></pre>	<pre><AnnotationAssertion > <AnnotationProperty IRI=' #fuzzyLabel' /> <IRI>#YoungAge</IRI> <Literal datatypeIRI=' &rdf;PlainLiteral' > <fuzzyOwl2 fuzzyType="datatype"> <Datatype type="leftshoulder" a="10" b="30" /> </fuzzyOwl2> </Literal> </AnnotationAssertion></pre>
<i>Fuzzy concepts</i> - множество нечётких классов	<pre><WEIGHTED_CONCEPT> := <Concept type="weighted" val- ue="<DOUBLE>" base="<STRING>" /></pre>	<pre><AnnotationAssertion > <AnnotationProperty IRI=' #fuzzyLabel' /> <IRI>#Weight0.8C</IRI> <Literal datatypeIRI=' &rdf;PlainLiteral' > <fuzzyOwl2 fuzzyType="concept"> <Concept type="weighted" value="0.8" base="C" /> </fuzzyOwl2> </Literal> </AnnotationAssertion></pre>
<i>Fuzzy axioms</i> - множество нечётких аксиом	<pre><FUZZY AXIOM> := <fuzzyType="axiom"> <Degree value="<DOUBLE>" /></pre>	<pre><ClassAssertion > <Class IRI=' #Tall' /> <NamedIndividual IRI=' #paul' /> <Annotation> <AnnotationProperty IRI=' #fuzzyLabel' /> <Literal datatypeIRI=' &rdf;PlainLiteral' > <fuzzyOwl2 fuzzyType="axiom"> <Degree value="0.5" /> </fuzzyOwl2> </Literal> </Annotation> </ClassAssertion></pre>
<i>Fuzzy roles</i> - множество нечётких отношений	<pre><FUZZY ROLE> := <fuzzyType="role"> <Role type="modified" modifier = "<modifier>" base="<STRING>" /></pre>	<pre><AnnotationAssertion> <AnnotationProperty IRI=" #fuzzyLabel" /> <AbbreviatedRI> generations:hasProject </AbbreviatedIRI> <Literal datatypeIRI="&rdf;PlainLiteral"> <fuzzyOwl2 fuzzyType="role"> <Role type="modified" modifier="low"; base="topObjectProperty" /> </fuzzyOwl2> </Literal> </AnnotationAssertion></pre>
<i>Fuzzy individuals</i> - множество нечётких экземпляров	<pre><FUZZY INDIVIDUALS> := <Concept type="nominal" val- ue="<DOUBLE>" individual="<STRING>" /></pre>	<pre><AnnotationAssertion> <AnnotationProperty IRI=" #fuzzyLabel" /> <IRI>#Project</IRI> <Literal datatypeIRI="&rdf;PlainLiteral"> <fuzzyOwl2 fuzzyType="concept"> <Concept type="nominal" value="0.1" individ- ual="Внедрение ИС" /> </fuzzyOwl2></Literal> </AnnotationAssertion></pre>

Онтологические модели, в том числе нечёткие, позволяют выявлять взаимосвязи в процессах управления проектом. Нечёткая онтология ПрО разработана таким образом, чтобы отобразить множество классов ПС, связанных с решением различных задач, возникающих во время выполнения основных процессов управления ПП, а также структурировать БЗ. Разра-

ботка нечёткой онтологии соответствует принципам *FuzzyOWL* и включает последовательность шагов, представленных на рисунке 1.

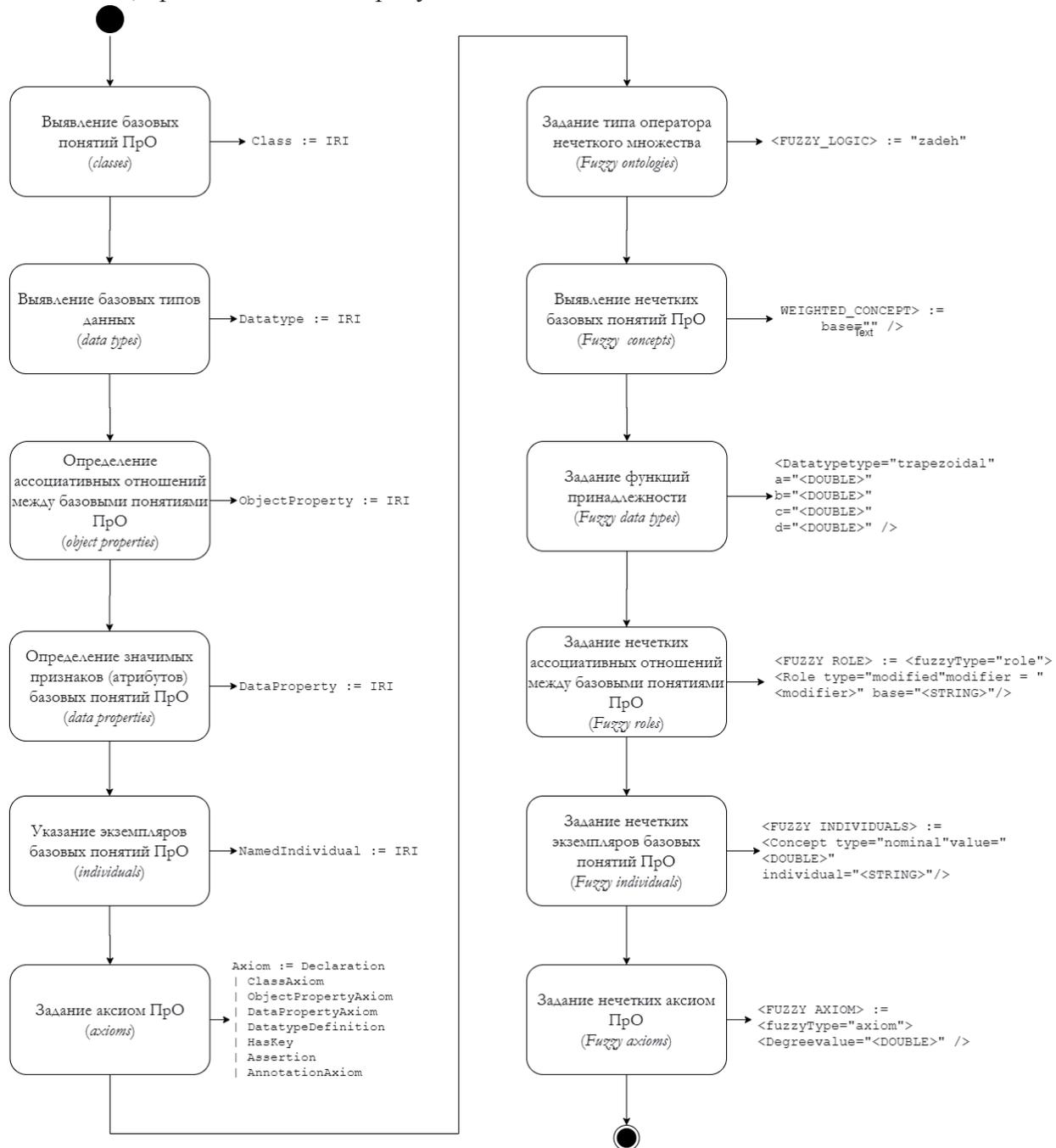


Рисунок 1 – Последовательность построения нечёткой онтологии

После разработки онтологии для обмена и использования знаниями необходимо обеспечить поддержание в актуальном состоянии базы (репозитория) прецедентов. Данная деятельность представляет собой замкнутый цикл, состоящий из шести этапов:

- 1) поиск прецедентов под возникшую проблему;
- 2) ранжирование похожих прецедентов;
- 3) выбор наиболее похожих прецедентов;

- 4) повторное использование информации и знаний из найденного прецедента для решения возникшей проблемы;
- 5) оценка эффективности предложенного решения проблемы;
- 6) сохранение вновь возникшего успешного прецедента в БЗ для решения подобных проблем в будущем.

2 Алгоритм поиска ближайших прецедентов в нечёткой онтологии

Под прецедентом понимается описание ПС в совокупности с подробным указанием действий, предпринимаемых в данной ситуации для решения данной проблемы. Обобщённая модель прецедента имеет следующий вид [4, 21]:

$$(3) \quad \textit{Case: Problem} \rightarrow \textit{Solution; Result},$$

где *Case* – прецедент; *Problem* – ПС, описывающая состояние процесса управления ПП, когда произошел прецедент; *Solution* – решение этой проблемы; *Result* – результат, который описывает состояние процесса управления ПП после произошедшей ПС.

Подробное описание признаков прецедента ПС, возникающих при управлении ПП, приведено в [17]. Одной из наиболее часто встречающихся проблем в проектно-ориентированных компаниях является распределение ресурсов (в том числе трудовых) между параллельно выполняющимися ПП, а успешность выполнения отдельного проекта может зависеть от того, насколько удачным окажется выбор участников команды проекта. Поэтому процесс поиска прецедента может быть ориентирован на достижение различных целей, например:

- найти такой прецедент, чтобы распределение ресурсов между проектами оказалось наиболее надёжным, обеспечивающим достижение конечной цели проекта;
- найти прецедент с таким распределением ресурсов в проекте, которое обеспечивает минимальное время исполнения;
- найти прецедент, для которого время получения нового решения окажется минимальным (минимум модификаций);
- найти прецедент, отражающий наиболее современный (поздний) опыт.

Онтологическая модель ПрО может быть представлена в виде графа. Такое представление семантической модели определяет способ построения запросов к ней, реализованный в языке *SPARQL*. Существующие алгоритмы поиска по прецедентам [22] не поддерживают нечёткие онтологии, поэтому разработан алгоритм поиска, позволяющий учитывать нечёткие свойства, нечёткие таксономические и ассоциативные отношения.

Поисковый запрос Q_i определяется как совокупность множества типов атрибутов t_i , наименований атрибутов n_i , значений атрибутов v_i и типов операций сравнения o_i :

$$(4) \quad Q_i = \langle t_i, n_i, v_i, o_i \rangle$$

Так как в онтологии задана модель данных прецедента ПС, то поисковый запрос Q_i содержит конечное множество кортежей, соответствующих модели данных (рисунок 2).

Поисковый запрос пользователя к БЗ, как правило, не полностью отражает ПС, так как пользователь может не знать всех терминов и структуры данных. Использование меры близости для оценки найденных прецедентов в БЗ позволяет расширять запросы и ранжировать результаты поиска. При осуществлении поиска выполняется сравнение заданных параметров в поисковом запросе со значениями прецедентов – экземпляров в онтологии. На основании таксономической близости поисковый запрос расширяется схожими классами ПС, так что предварительная кластеризация прецедентов становится необязательной.

Схема алгоритма поиска прецедента в БЗ в случае использования нечёткой онтологии приведена на рисунке 3.

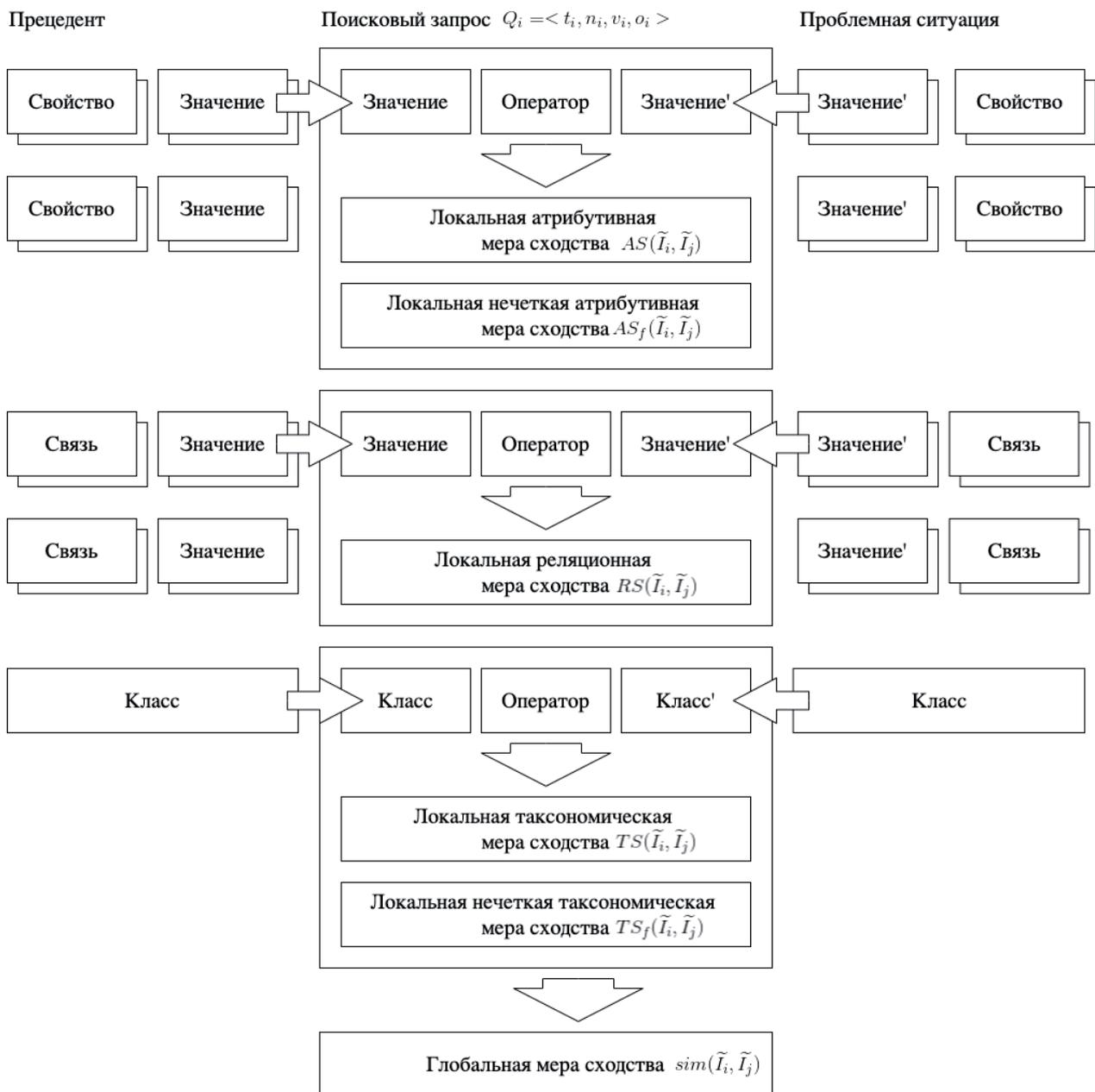


Рисунок 2 – Модель данных поискового запроса Q_i

Определение меры сходства ПС и прецедента в БЗ состоит из следующих шагов.

Шаг 1. Парное сравнение атрибутов прецедента и ПС на основе выбранного оператора сравнения для определения атрибутивной близости $AS(\tilde{I}_i, \tilde{I}_j)$. Атрибутивной мерой сходства $AS(\tilde{I}_i, \tilde{I}_j)$, где $\mu_{\tilde{I}_i}(x) = 1$ и $\mu_{\tilde{I}_j}(x) = 1$, считается определение подобия признаков описания ПС, где, в зависимости от типа данных атрибута, применяется соответствующий оператор сравнения. Так, для строковых типов может использоваться мера сходства – расстояние Левенштейна [23] или строгое посимвольное сравнение.

Шаг 2. Парное сравнение прецедента и ПС для определения нечёткой атрибутивной близости $AS_f(\tilde{I}_i, \tilde{I}_j)$. В качестве нечёткой меры сходства используется мера сходства Заде для нечётких признаков $\tilde{I}_i(x_k)$ и $\tilde{I}_j(x_k)$, $x \in E$ [24]:

$$(5) \quad AS_f(\tilde{I}_i, \tilde{I}_j) = \min \left(\max \left(1 - \mu_{\tilde{I}_i}(x_k), \mu_{\tilde{I}_j}(x_k) \right), \max \left(\mu_{\tilde{I}_i}(x_k), 1 - \mu_{\tilde{I}_j}(x_k) \right) \right), k = \overline{1, n}$$

где $\mu_{\tilde{I}_i}(x_k)$ – значение функции принадлежности нечёткого признака $\tilde{I}_i(x_k)$,
 $\mu_{\tilde{I}_j}(x_k)$ – значение функции принадлежности нечёткого признака $\tilde{I}_j(x_k)$

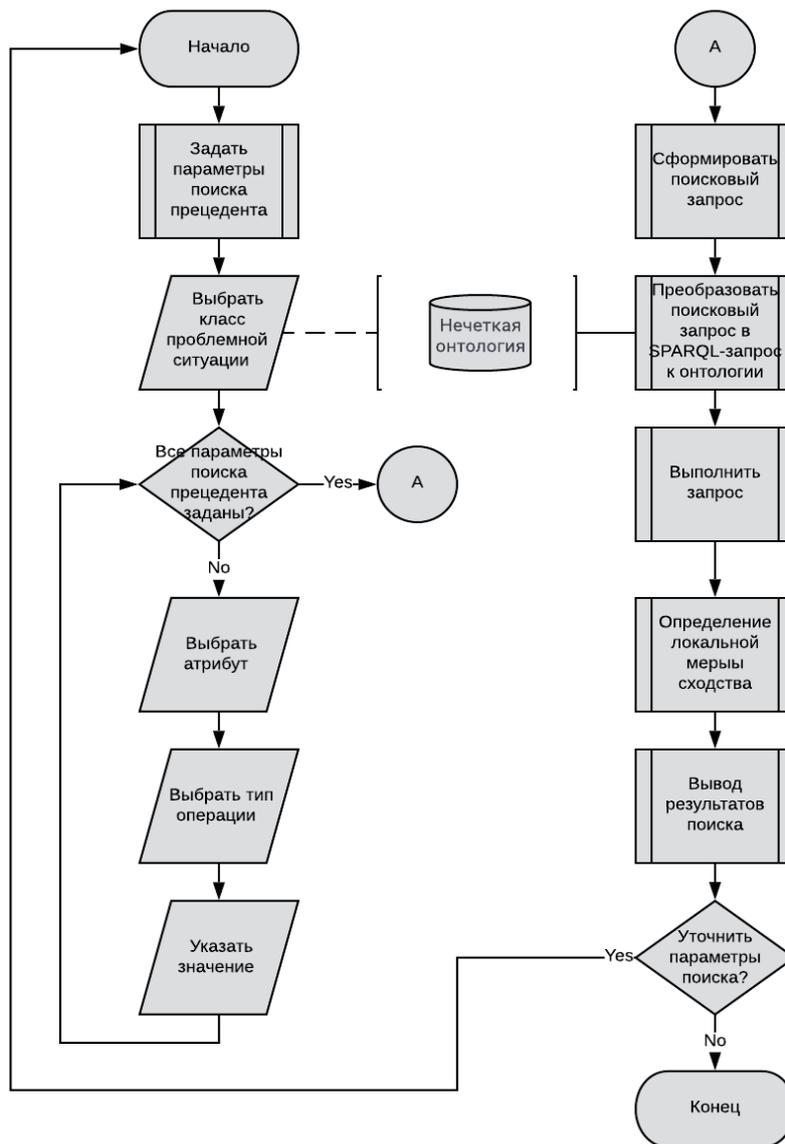


Рисунок 3 – Схема алгоритма поиска прецедента в нечёткой онтологии поддержки принятия решений при управлении программными проектами

Термы лингвистической переменной и тип функции принадлежности задаются в виде аннотации *fuzzyLabel* к пользовательскому типу данных в онтологии (рисунок 4). Описание работы с редактором онтологий *Protege 5.2.0* приведено в [19]. Определение нечёткого типа данных *fuzzy data type* в редакторе онтологий *Protege 5.2.0* приведено на рисунке 4, а определение типа данных для свойства *data property* на рисунке 5. *Domain range* свойства *data property* содержит значения термов лингвистической переменной (5).

В таблице 2 приведён пример определения нечёткой атрибутивной близости для двух проектов C_1 и C_2 к проекту P , в котором обнаружена ПС.

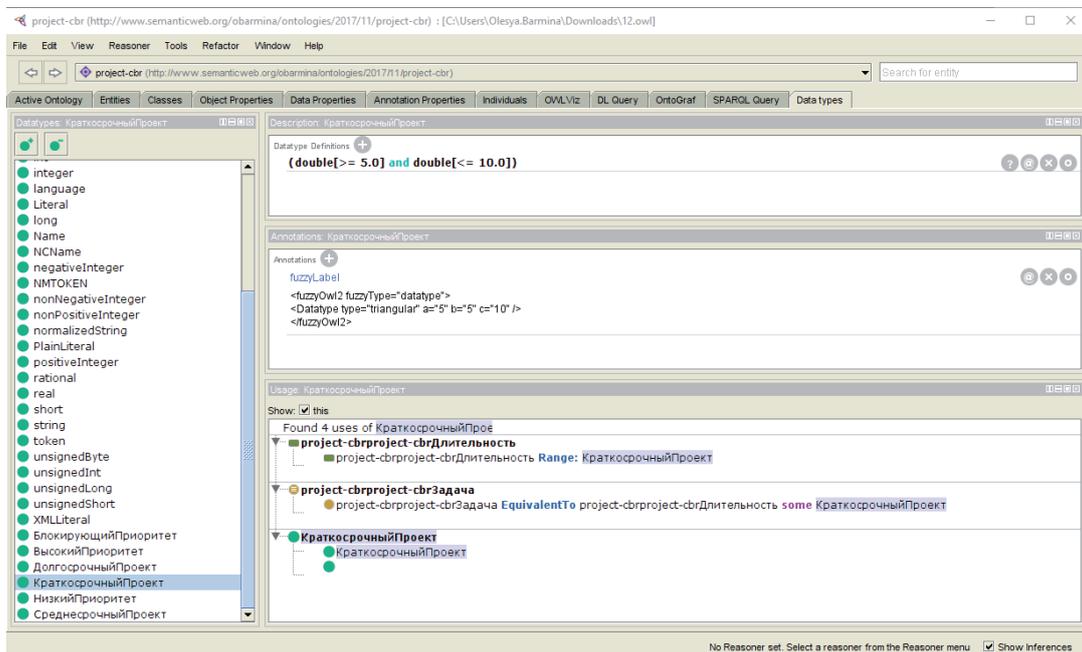


Рисунок 4 – Определение нечёткого типа данных *fuzzy data type* в редакторе онтологий *Protege 5.2.0*

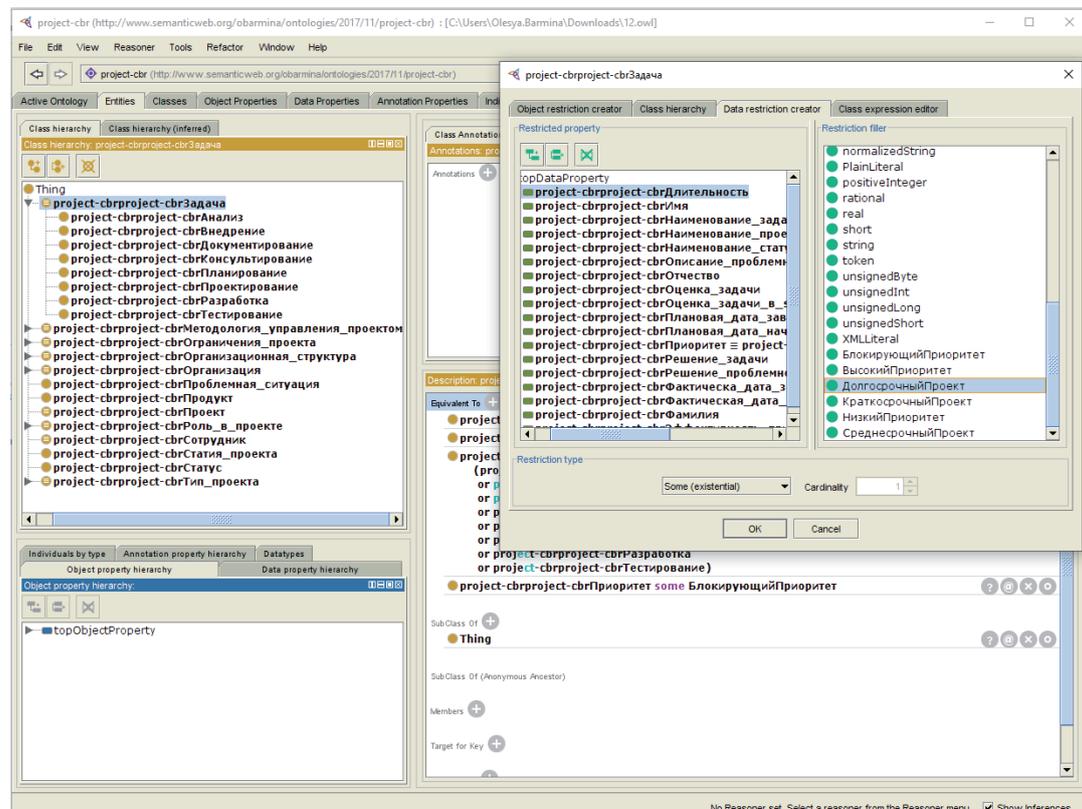


Рисунок 5 – Определение типа данных для свойства *data property* в редакторе онтологий *Protege 5.2.0*

Для определения нечёткой атрибутивной близости требуется провести фаззификацию (введение нечёткости), поскольку в онтологии все значения атрибутов хранятся в виде тер-

мов лингвистической переменной. Фаззификация заключается в установке соответствия между численным значением входной переменной, представленной в виде критериев поискового запроса, и значением функции принадлежности соответствующего ей термина лингвистической переменной. На этапе фаззификации критериям поискового запроса ставятся в соответствие конкретные значения функций принадлежности соответствующих лингвистических термов (рисунок 6).

Таблица 2 – Пример определения нечёткой атрибутивной близости

Признак I , выраженный в виде функции принадлежности $data\ type$	Проект P	Проект C_1	Проект C_2
Чёткое значение длительности проекта	6 лет	1 год	3 года
Нечёткое значение длительности проекта	Долгосрочный, $\mu = 1.0$	Краткосрочный $\mu = 0.1$	Среднесрочный $\mu = 0.5$
$AS_f(\tilde{I}_i, \tilde{I}_j)$ определяется по формуле (5)		$AS_f(\tilde{I}_i, \tilde{I}_j)=0.1$	$AS_f(\tilde{I}_i, \tilde{I}_j)=0.5$

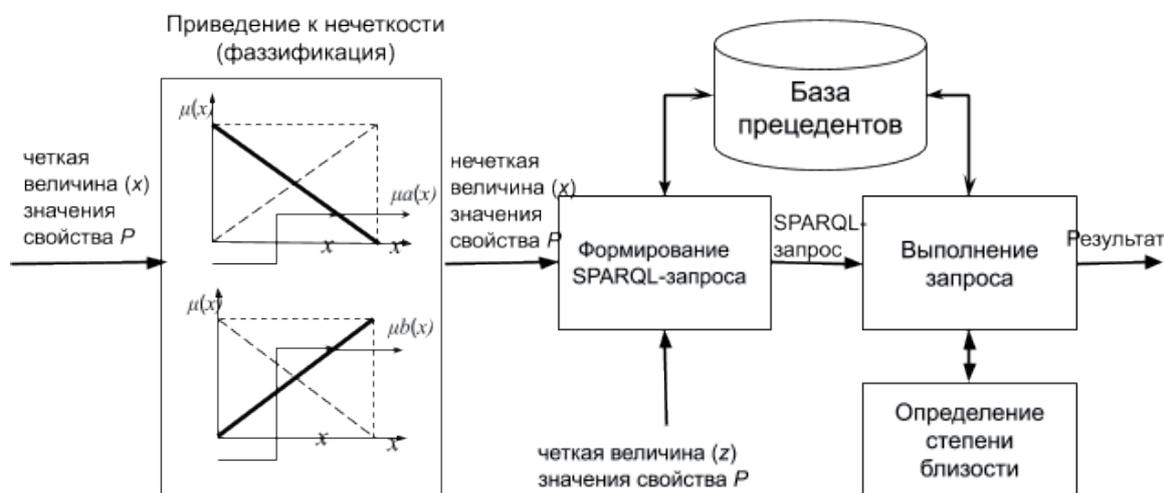


Рисунок 6 – Архитектурная схема поиска прецедента

Пусть переменная *<Длительность проекта>* может принимать любое значение из диапазона от нуля до бесконечности. Каждому значению длительности из указанного диапазона может быть поставлено в соответствие некоторое число от 0 до 1, которое определяет степень принадлежности данной длительности проекта к тому или иному терму лингвистической переменной *<Длительность проекта>*. Степень принадлежности определяется так называемой функцией принадлежности $\mu(x)$. Например, длительности в 1 год (рисунок 7, таблица 2) можно задать степень принадлежности к терму «краткосрочный» равную 0.1, а к терму «среднесрочный» – 0.0.

Шаг 3. Парное сравнение прецедента и ПС для определения реляционной близости $RS(\tilde{I}_i, \tilde{I}_j)$. Реляционная мера сходства $RS(\tilde{I}_i, \tilde{I}_j)$, где $\mu_{\tilde{I}_i}(x) = 1$ и $\mu_{\tilde{I}_j}(x) = 1$, основана на определении подобия ассоциативных отношений ПС.

Шаг 4. Парное сравнение прецедента и ПС для определения таксономической близости $TS(\tilde{I}_i, \tilde{I}_j)$, где $\mu_{\tilde{I}_i}(x) = 1$ и $\mu_{\tilde{I}_j}(x) = 1$. Таксономическая близость между экземплярами \tilde{I}_i и \tilde{I}_j вычисляется с учётом положения соответствующих им понятий C_i и C_j в таксономии

$$(6) \quad H^c: UC(C_i, H^c) = \{H^c(C_i, C_j) \vee C_j = C_i\},$$

где H^c – таксономическая иерархия; UC – вершина, так называемая «верхняя котопия» (*upwards cotopy* – UC) [25]. Таким образом, можно определить таксономическую близость $TS(\tilde{I}_i, \tilde{I}_j)$:

$$(7) \quad TS(\tilde{I}_i, \tilde{I}_j) = \frac{UC(C_i, H^c) \cap UC(C_j, H^c)}{UC(C_i, H^c) \cup UC(C_j, H^c)}$$

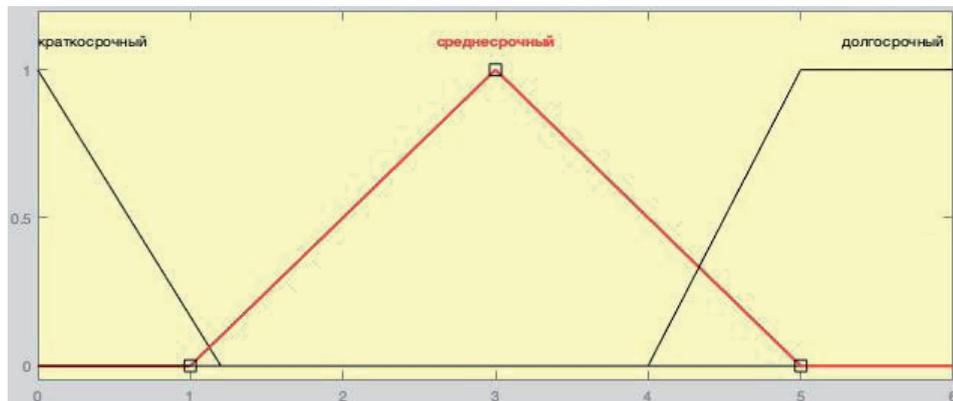


Рисунок 7 – Пример описания лингвистической переменной <Длительность проекта>

Шаг 5. Парное сравнение прецедента и ПС для определения нечёткой таксономической близости $TS_f(\tilde{I}_i, \tilde{I}_j)$

В таблице 3 приведён пример определения нечёткой таксономической близости для двух проектов C_1 и C_2 к проекту P , в котором обнаружена ПС.

Таблица 3 – Пример определения нечёткой таксономической близости

Класс проекта I	Проект P	Проект C_1	Проект C_2	$TS_f(\tilde{I}_i, \tilde{I}_j)$	$TS_f(\tilde{I}_i, \tilde{I}_j)$
Вид проекта	Региональный	Международный	Региональный	<code><fuzzyOwl2 fuzzyType="concept"> <Concept type="weighted" value="0.7" base="C" /></code>	<code><fuzzyOwl2 fuzzyType="concept"> <Concept type="weighted" value="0.77" base="C" /></code>
				$TS_f(\tilde{I}_i, \tilde{I}_j) = 0.7$	$TS_f(\tilde{I}_i, \tilde{I}_j) = 0.77$

Нечёткая таксономическая мера сходства $TS_f(\tilde{I}_i, \tilde{I}_j)$ определяется с использованием меры сходства Заде, где значение функции принадлежности нечёткого класса задаётся в аннотации *fuzzyLabel* в виде параметра *fuzzy classes* (рисунок 8).

Шаг 6. Вычисление глобальной метрики сходства после расчёта локальных контекстно-зависимых метрик сходства.

Общая величина подобия $sim(\tilde{I}_i, \tilde{I}_j)$ экземпляров \tilde{I}_i и \tilde{I}_j определяется формулой:

$$(8) \quad sim(\tilde{I}_i, \tilde{I}_j) = \frac{t \times TS(\tilde{I}_i, \tilde{I}_j) + r \times RS(\tilde{I}_i, \tilde{I}_j) + a \times AS(\tilde{I}_i, \tilde{I}_j) + t_f \times TS_f(\tilde{I}_i, \tilde{I}_j) + a_f \times AS_f(\tilde{I}_i, \tilde{I}_j)}{t + r + a + t_f + a_f}$$

где t, r, a, t_f, a_f – веса различных измерений сходства, которые могут быть подобраны с привлечением экспертных оценок в зависимости от важности учёта различных измерений.

Решение извлекается из найденного похожего прецедента, при необходимости адаптируется к текущей ПС по правилам, разрабатываемым специально для проектно-ориентированной компании в зависимости от условий ее функционирования или для определенных типов программных проектов. Онтологическая модель описания прецедентов ПС в ПП приведена в [17]. В случае, если похожий прецедент не найден, описание ПС сохраняется в онтологии. ЛППР находит решение самостоятельно или с помощью экспертов и вносит принятое решение в онтологию, формируя новый прецедент.

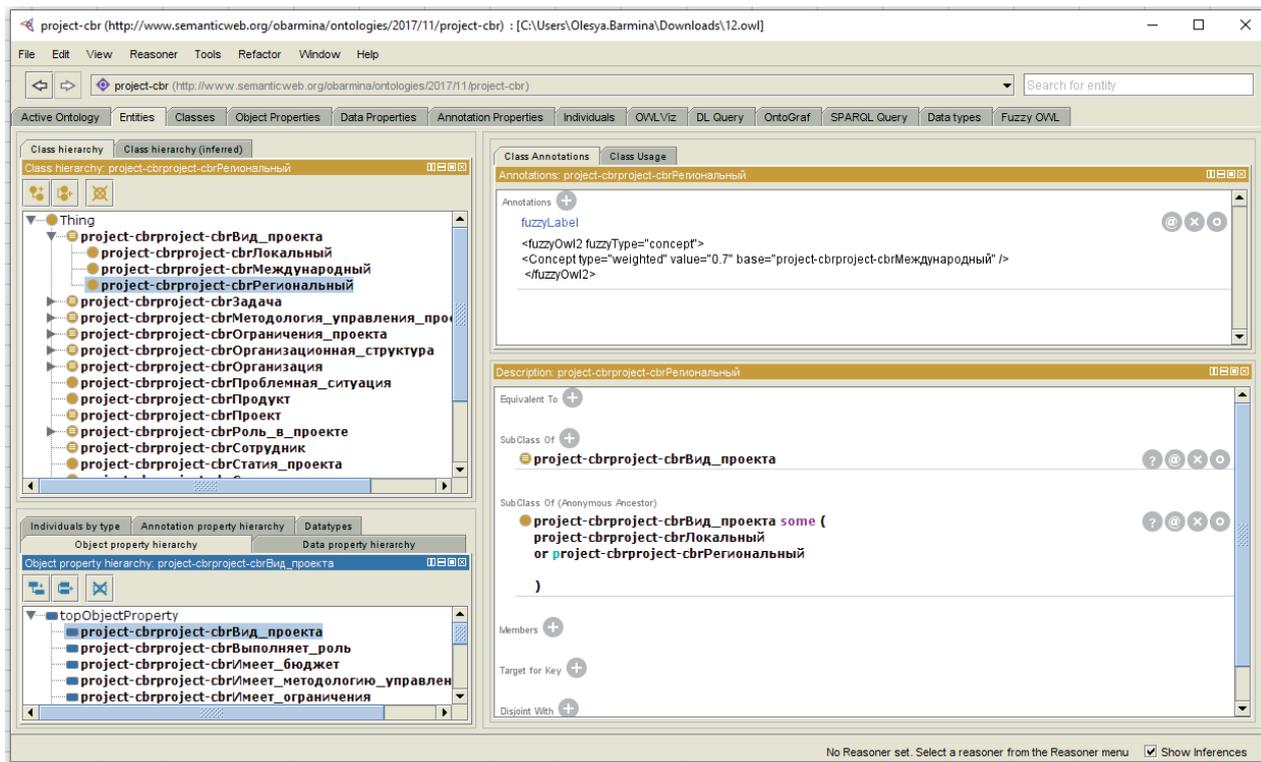


Рисунок 8 – Пример класса с указанной аннотацией *fuzzyLabel*

3 Программная реализация

Для интеллектуальной системы поддержки принятия решений в соответствии с предложенными моделями было разработано расширение к редактору онтологий *Protege 5.2.0 Search-plugin* [26]. Все расширения к редактору онтологий *Protege 5.2.0* типа *TabWidget* реализуются с использованием *OSGi*-спецификации динамической модульной системы и сервисной платформы для разработки *Java*-приложений (рисунок 9).

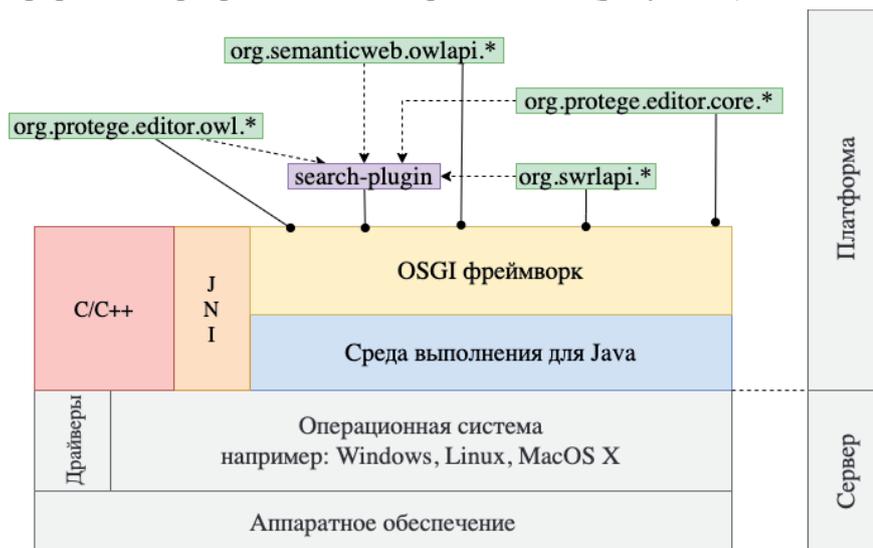


Рисунок 9 – Программная реализация плагина *Search-plugin* к *Protege 5.2.0* для поиска прецедентов проблемных ситуаций

Особенность использования *OSGi* заключается в возможности динамически переинсталлировать компоненты и составные части приложения без необходимости его останова и перезапуска. Круг применений данной спецификации довольно широк: она изначально разрабатывалась для создания встроенных систем (в частности, для автомобилей *BMW*, также в разработке спецификации активно участвует *Siemens*), но сейчас на базе *OSGi* строят многофункциональные автономные настольные приложения (например, *Eclipse SDK*) и корпоративные системы. Модули (плагины, бандлы (*bundle*)) взаимодействуют между собой посредством сервисов: объектов системы с заявленными реализованными интерфейсами. Модули регистрируют сервисы для предоставления определённой функциональности, механизма создания и обработки событий. Модуль *OSGi* (*OSGi bundle*) содержит *java*-классы и другие ресурсы, которые реализуют необходимые функции, а также предоставляют сервисы и пакеты другим модулям. На рисунке 10 представлена процедура запуска разработанного плагина.

```

*****
**                                     Protege
*****

----- Initialising and Starting the OSGi Framework -----
FrameworkFactory Class: org.apache.felix.framework.FrameworkFactory

The OSGi framework has been initialised
----- Starting Bundles -----
Starting bundle org.protege.common
Starting bundle org.eclipse.equinox.common
Starting bundle org.eclipse.equinox.supplement
Starting bundle org.protege.editor.core.application
Starting bundle org.eclipse.equinox.registry
Starting bundle log4j.over.slf4j
Starting bundle com.google.inject
Starting bundle jul.to.slf4j
Starting bundle ch.qos.logback.core
Starting bundle slf4j.api
Starting bundle org.apache.commons.io
Starting bundle search-plugin

```

Рисунок 10 – Запуск плагина *Search-plugin* в *Protege 5.2.0*

Под плагином понимается *Java™ Archive (JAR)* – самодостаточный и самоопределяемый модуль. Самодостаточность означает, что модуль содержит в себе код и ресурсы, необходимые ему для работы. Самоопределяемость означает наличие информации о своей сущности, описание требований модуля к внешней среде и описание собственных возможностей для внешней среды. Плагин (модуль *Search-plugin*) предназначен для поиска похожих прецедентов ПС, возникающих при разработке ПО, путём формирования *SPARQL*-запросов к нечёткой онтологии в диалоговом режиме [26]. Пример формирования запроса на поиск похожих прецедентов в нечёткой онтологии с использованием интерфейса разработанного плагина к редактору онтологии *Protege 5.2.0* и листинг запроса представлены на рисунках 11 и 12 соответственно.

На запрос поиска прецедентов по ПС, возникшей в программном проекте при выполнении задачи со следующими параметрами: затронуты процессы <Разработка документации>, влияет на работу специалиста <Аналитик>, возможная причина проблемной ситуации состоит в <Неправильная оценка проекта> найдено 13 прецедентов (рисунок 11).

4 Оценка эффективности поиска ближайших прецедентов

Для оценки адекватности разработанных моделей, алгоритмов и ПО были проведены вычислительные эксперименты на основании данных о выполнении ПП (разработка системы

электронного документооборота, системы управления финансами, аналитической системы) в различных ПрО для различных заказчиков. Программные проекты выполнялись в одной и той же ИТ-компании, данные для эксперимента собирались в течение 6 месяцев. План экспериментов приведён в таблице 4.

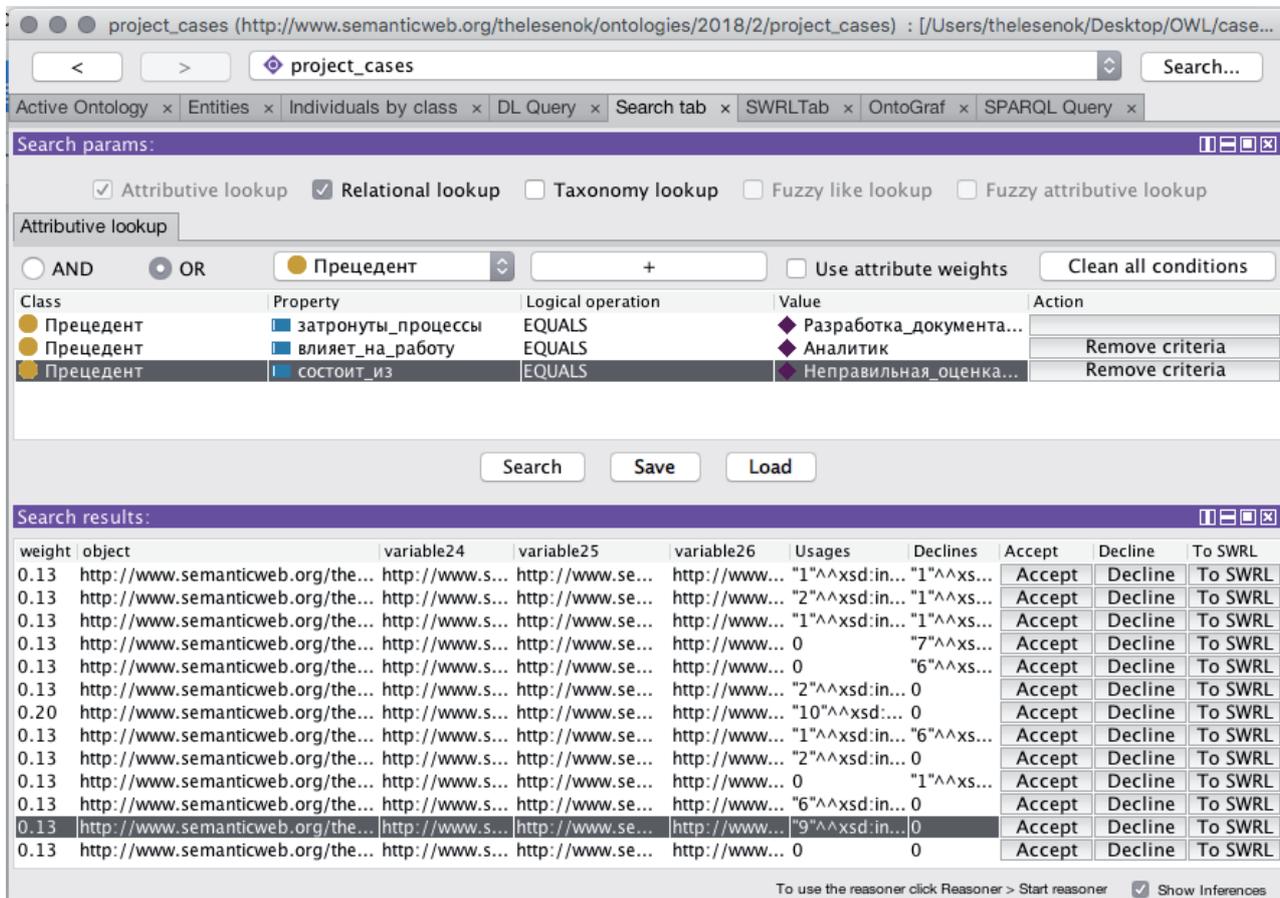


Рисунок 11 – Плагин Search-plugin в редактору онтологий Protege 5.2.0

```
SELECT * WHERE {
?object a <http://www.semanticweb.org/ontologies/2018/2/project_cases#Прецедент>.
?object prfx3:влияет_на_работу ?variable1.
?object prfx3:затронуты_процессы ?variable2.
?object prfx3:состоит_из ?variable2
FILTER ((STR(?variable1) = "http://www.semanticweb.org/ontologies/2018/2/project_cases#Аналитик")
&& (STR(?variable2) =
"http://www.semanticweb.org/ontologies/2018/2/project_cases#Разработка_документации") &&
(STR(?variable3) =
"http://www.semanticweb.org/ontologies/2018/2/project_cases#Неправильная_оценка_проекта"))}
```

Рисунок 12 – Листинг SPARQL-запроса на поиск прецедентов

Таблица 4 – План проведения экспериментов

№ п/п	Наименование этапа
1	Анализ инструментов поиска прецедентов ПС при выполнении ПП
2	Выявление часто встречающихся поисковых запросов
3	Создание и наполнение БЗ (разработка онтологий <i>FuzzyOWL</i> , <i>Jcolibri</i> , разработка базы данных <i>MySQL</i>)
4	Поиск прецедентов в БЗ, которая представлена в виде онтологии <i>FuzzyOWL</i> в <i>Protege 5.2.0</i>
5	Поиск прецедентов в БЗ, которая представлена в виде базы данных <i>MySQL</i>
6	Поиск прецедентов в БЗ, которая представлена в виде онтологии <i>OWL</i> в <i>JColibri</i>
7	Анализ выполненных экспериментов

При проведении исследований оценивались: инструмент собственной разработки (*Search-plugin*), редактор построения SQL-запросов и готовое решение *JColibri* [27] – платформа для построения решений по прецедентам, основывающаяся на онтологиях с дескрипционной логикой. Результаты сравнения инструментов поиска ближайших прецедентов сведены в таблицу 5.

Таблица 5 – Результаты сравнения инструментов поиска прецедентов

№ п/п	Параметр	<i>Search-plugin</i>	<i>SQL</i>	<i>JColibri</i>
Используемые меры сходства				
1	атрибутивная мера сходства	+	+	-
2	реляционная мера сходства	+	+	+
3	таксономическая мера сходства	+	-	+
4	нечёткая атрибутивная мера сходства	+	-	-
5	нечёткая таксономическая мера сходства	+	-	-
6	атрибутивная мера сходства нечёткого значения атрибута	+	-	-
База знаний				
7	<i>OWL</i>	+	-	+
8	<i>FuzzyOWL</i>	+	-	-
9	<i>MySQL</i>	-	+	+
Принятие решения в использовании прецедента				
10	Учёт количества принятия/отказа в использовании прецедента	+	-	-

Эксперимент заключается в последовательном выполнении поисковых запросов, ранжированных по частоте использования (выявленные на шаге 2 плана проведения эксперимента).

Для оценки эффективности выполнения поиска прецедентов использовались параметры: точность поиска, полнота поиска, *F*-мера Ван Ризбергена [28]. Показатель полноты поиска (*R*) характеризует способность системы находить нужные прецеденты ПС для ЛПП, но не учитывает количество нерелевантных прецедентов, выданных в результатах поиска. Показатель точности поиска (*P*) характеризует способность системы находить только релевантные прецеденты ПС. *F*-мера является интегральной метрикой, объединяющей полноту и точность поиска. Результаты проведённого эксперимента по сравнению эффективности поиска прецедентов приведены в таблице 6 и на рисунке 13. Использование предложенного алгоритма повышает интегральный показатель более чем на 40%.

Таблица 6 – Результаты проведения эксперимента

Параметр	<i>Search-plugin</i>	<i>SQL</i>	<i>JColibri</i>
Точность <i>P</i>	0,78	0,33	0,51
Полнота <i>R</i>	0,77	0,45	0,33
<i>F</i> -мера	0,78	0,38	0,40

Для проведения экспериментов были выбраны типовые запросы (всего 10 видов), что позволило рассчитать средние значения величин, приведённых в таблице 6. Номер эксперимента – это факт выполнения типового запроса вида: «ПС произошла в ПП <Наименование проекта> при выполнении задачи типа <Разработка новой функциональности>, связанной с доработкой модуля <Название модуля> исполнителем <Разработчик>, что повлияло на процесс <Проведение приемочного тестирования> и затронуло изменение сроков проекта <Да>». Количество запросов в каждом ПП, которое позволило считать их типовыми, варьировалось от 20 до 224.

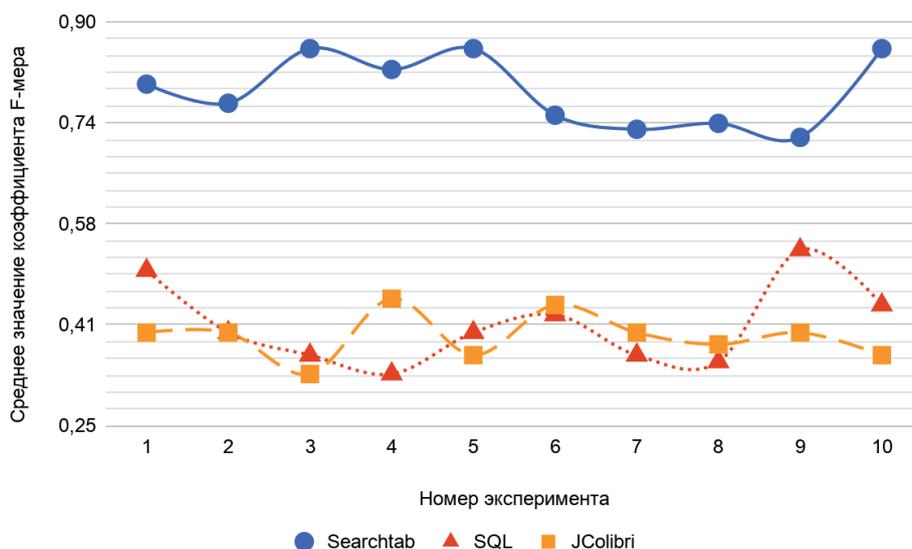


Рисунок 13 – Результаты оценки F -меры поиска похожих прецедентов ПС (фрагмент эксперимента)

Применение предложенных методов и алгоритмов для управления ресурсами проекта в ПС позволило осуществить при выполнении вышеуказанных ПП оперативный поиск прецедентов ПС и снабжение ЛПР необходимой информацией. Использование нечётких свойств в онтологии в качестве критериев позволяет значительно повысить точность и полноту поиска за счёт последовательного сужения области поиска заранее заданными параметрами.

Заключение

Обоснована необходимость разработки нечёткой онтологии для описания процедур поддержки принятия решений в ходе управления ПП. Применение прецедентного подхода в процессе принятия решений позволяет использовать накопленный опыт решения схожих проблем. Разработка новых методов, повышающих эффективность повторного использования прецедентов, остаётся актуальной, особенно для проектов по созданию ПО, характеризующихся высокой степенью уникальности. Предложенный метод использования нечёткой онтологии основан на принципах *FuzzyOWL* и позволяет построить онтологию, отображающую множество классов ПС, требующих решения в ходе выполнения ПП, и структурировать БЗ.

Разработан алгоритм поиска, позволяющий учитывать нечёткие свойства, нечёткие таксономические и ассоциативные отношения. Эффективность алгоритма поиска ближайшего прецедента в нечёткой онтологии подтверждена на основании экспериментальных данных, полученных при тестировании прототипа интеллектуальной системы поддержки принятия решений на проектах по разработке ПО.

Благодарности

Исследования выполнены при финансовой поддержке Российского фонда фундаментальных исследований (грант № 19-08-00937 «Методы и модели поддержки принятия решений при управлении программными проектами в среде производственных предприятий»).

СПИСОК ИСТОЧНИКОВ

- [1] Тренды проектного управления // GANTBPM Управление проектами. - <https://gantbpm.ru/trendy-proektnogo-upravleniya/>.
- [2] **Черняховская, Л.Р.** Разработка моделей и методов интеллектуальной поддержки принятия решений на основе онтологии организационного управления программными проектами / Л.Р. Черняховская, А.И. Малахова // *Онтология проектирования*. 2013. №4 (10). - С.42-52.
- [3] A guide to the project management body of knowledge (PMBOK), Sixth ed. / Project Management Institute, Inc., 14 Campus Boulevard, Newton Square, Pennsylvania 19073-3299 USA, 2017, 756 p.
- [4] **Aamodt, A.** Case-based reasoning: foundational issues, methodological variations, and system approaches / A. Aamodt, E. Plaza // *AI Communications*. IOS Press. Vol. 7: 1. 1994. – P.39-59.
- [5] The Access Group: Inbox Insight Survey. - http://www.theaccessgroup.com/media/250643/5_project_lessons_to_take_into_2014_whitepaper_v2.pdf.
- [6] Complete collection project management statistics. - <https://www.wrike.com/blog/complete-collection-project-management-statistics-2015/>.
- [7] Top project management trends. - http://www.esi-intl.co.uk/resource_centre/white_papers/top-project-management-trends-2016.pdf.
- [8] **Дюбуа, Д.** Теория возможностей. Приложения к представлению знаний в информатике / Д. Дюбуа, А. Прад // Пер. с фр. - М.: Радио и связь, 1990. - 288 с.
- [9] **Ларичев, О. И.** Теория и методы принятия решений / О.И. Ларичев. - М.: Логос, 2006. - 392 с.
- [10] **Мадера, А.Г.** Бизнес-процессы и процессное управление в условиях неопределенности: количественное моделирование и оптимизация / А.Г. Мадера. - М.: ЛЕНАНД, 2019. 160 с.
- [11] **Антонов, В.В.** Формальная модель предметной области на основе нечётких отношений / В.В. Антонов, Г.Г. Куликов // *Программные продукты и системы*. 2011. №2(94). - С.48-51.
- [12] **Антонов, В.В.** Семантико-математический язык описания структуры интеллектуальной системы на основе нечёткой логики / В.В. Антонов, Г.Г. Куликов // *Программные продукты и системы*. 2011. №3(95). - С.33-35.
- [13] **Антонов, В.В.** Построение формальной модели предметной области с применением нечёткой кластеризации / В.В. Антонов, Г.Г. Куликов, Д.В. Антонов // *Вестник УГАТУ: науч. журн. Уфимск. гос. авиац. техн. ун-та*. 2011. Т.15, № 5(45). - С.3-11.
- [14] **Гаврилова, Т.А.** Инженерия знаний. Модели и методы / Т.А. Гаврилова, Д.В. Кудрявцев, Д.И. Муромцев. СПб.: Издательство «Лань», 2016. 324 с.
- [15] **Болотова, Л.С.** Системы искусственного интеллекта: модели и технологии, основанные на знаниях / Л.С. Болотова. - М.: Финансы и статистика, 2012. 664 с.
- [16] **Массель, Л.В.** Онтологический инжиниринг для поддержки принятия стратегических решений в энергетике / Л.В. Массель, Т.Н. Ворожцова, Н.И. Пяткова // *Онтология проектирования*. 2017. Т.7, №1 (23). - С.66-76. - DOI: 10.18287/2223-9537-2017-7-1-66-76.
- [17] **Бармина, О.В.** Интеллектуальная система управления взаимодействием бизнес-процессов в проектно-ориентированных организациях / О.В. Бармина, Н.О. Никулина // *Онтология проектирования*. 2017. Т.7, №1 (23). - С.48-65. - DOI: 10.18287/2223-9537-2017-7-1-48-65.
- [18] **Bobillo, F.** Fuzzy description logics with general t-norms and datatypes / F. Bobillo, U. Straccia // *Fuzzy Sets and Systems*. 2009. 160 (23). P.3382–3402.
- [19] Official documentation on Protégé. - <http://protrgrwiki.stanford.edu/index.php/ProtegeUserDocs>.
- [20] **Ярушкина, Н.Г.** Применение способа интеграции нечётких временных рядов и нечётких онтологий в задачах диагностики технических систем / Н.Г. Ярушкина, В.С. Мошкин, Г.Р. Ишмуратова, И.А. Андреев, И.А. Мошкина // *Онтология проектирования*. 2018. Т.8, №4 (30). - С.594-604. - DOI: 10.18287/2223-9537-2018-8-4-594-604.
- [21] **Althof, Klaus-Dieter.** A Review of Industrial Case-Based Reasoning Tools / Klaus-Dieter Althof, Eric Auriol, Ralph Barlette, Michel Manago // *AI Intelligence*, 1995. – p. 159.
- [22] **Люгер, Д.Ф.** Искусственный интеллект: стратегии и методы решения сложных проблем / Д.Ф. Люгер // Пер. с англ. – 4-е изд. – М.: Издательский дом «Вильямс», 2003. 864 с.
- [23] **Левенштейн, В.И.** Двоичные коды с исправлением выпадений, вставок и замещений символов. Доклады Академии Наук СССР, 1965. Т. 163, №4. С.845-848.
- [24] **Zadeh I.** Fuzzy Logic, Neural Networks and Soft Computing / I. Zadeh // *Communication on the ACM*, 1994. – Vol. 37, No 3. – p.77–84.
- [25] **Крюков, К.В.** Меры семантической близости в онтологии / К.В. Крюков, Л.А. Панкова, В.А. Пронина, В.С. Суховеров, Л.Б. Шипилина // *Проблемы управления*. 2010, № 5. С.2–14

- [26] **Черняховская, Л.Р.** Программа подбора прецедентов проблемных ситуаций при управлении проектами / Л.Р. Черняховская, О.В. Бармина, Н.О. Никулина // Свидетельство о государственной регистрации программы для ЭВМ №2018615125 от 26.04.2018.
- [27] Official documentation on JColibri. - <http://gaia.fdi.ucm.es/research/colibri/jcolibri/>.
- [28] **Van Rijsbergen, C.J.** Information Retrieval (2nd ed.). Butterworth-Heinemann. 1979. <http://www.dcs.gla.ac.uk/Keith/Preface.html>.
-

Сведения об авторах



Антонов Вячеслав Викторович, 1956 г. рождения. Окончил Башкирский государственный университет (1979), кандидат технических наук (2007), доктор технических наук (2015). Профессор кафедры автоматизированных систем управления Уфимского государственного авиационного технического университета. В списке научных трудов около 30 работ в области построения интеллектуальных систем поддержки принятия решений. AuthorID (РИНЦ): 530537. Author ID (Scopus): 57200254522; ORCID 0000-0002-5402-9525; Researcher ID (WoS): AАН-5121-2019. Antonov.V@bashkortostan.ru.

Бармина Олеся Владимировна, 1988 г. рождения. Окончила Уфимский государственный авиационный технический университет (2011), магистратуру (2014), а также аспирантуру по направлению «Информатика и вычислительная техника» (2018). Ведущий аналитик ООО "Нордиджи", г. Санкт-Петербург. Научные интересы: реинжиниринг бизнес-процессов, управление программными проектами, интеллектуальные системы поддержки принятия решений. В списке трудов более 20 работ в указанных областях. ORCID 0000-0003-4404-1376; obarmina@outlook.com.



Никулина Наталья Олеговна, 1971 г. рождения. Окончила Уфимский государственный авиационный технический университет (1994), кандидат технических наук (1998). Доцент кафедры автоматизированных систем управления. Научные интересы: исследования в области управления проектами, системного моделирования и проектирования информационных систем в организационном управлении. В списке трудов более 40 работ в указанных областях. AuthorID (РИНЦ): 396597; ORCID 0000-0002-8594-2093. nikulina@outlook.com.

Поступила в редакцию 08.12.2019, после рецензирования 25.02.2020. Принята к публикации 05.03.2020.

Decision-making support in software project management based on fuzzy ontology

V.V. Antonov¹, O.V. Barmina², N.O. Nikulina¹

¹Ufa State Technical Aviation University, Ufa, Russia

²Nordigy LCC, Saint-Petersburg, Russia

Abstract

The article deals with knowledge management in software projects and decision-making support of its participants. The novelty of research is justified by the need to reduce the degree of uncertainty in decision-making and is confirmed by statistics indicating a sufficiently large number of unsuccessful projects. It is proposed to organize decision-making support based on the cases analysis of previously occurring problem situations. Also, it is proposed to use knowledge engineering methods, in particular, ontological analysis of the subject area. A formation of a fuzzy ontology method is developed for decision-making support in the software development. The novelty of the mathematical model lies in the combination of various inference mechanisms when making decisions on the basis of the entire complex of knowledge about the subject area. Increasing amount of information does not allow using simple searches for quick retrieval of the

necessary information. Therefore, an algorithm is proposed for generating a search query that takes into account fuzzy properties, fuzzy taxonomic and associative relations between classes of objects in ontology. The effectiveness of the proposed solutions is confirmed by an experiment on the prototype of an intelligent decision-making support system.

Keywords: *project management, fuzzy ontology, knowledge management, search, retrieval, decision support, incidents, case management.*

Citation: *Antonov VV, Barmina OV, Nikulina NO. Decision-making support in software project management based on fuzzy ontology [In Russian]. Ontology of designing. 2020; 10(1): 121-140. DOI: 10.18287/2223-9537-2020-10-1-121-140.*

Acknowledgments: The work is supported by RFBR grant № 19-08-00937 «Methods and models of the intellectual decision-making support in software project management realized on manufacturing enterprises»

List of figures and tables

- Figure 1 – The sequence of building a fuzzy ontology
 Figure 2 – Q_i search query data model
 Figure 3 – The precedent search algorithm scheme in the fuzzy ontology of decision-making support in software project management
 Figure 4 – Defining a fuzzy data type in the Protégé 5.2.0 ontology editor
 Figure 5 – Defining a data type for the data property in the Protege 5.2.0 ontology editor
 Figure 6 – Problem situation case search architecture
 Figure 7 – An example of the description of the linguistic variable «Project Duration»
 Figure 8 – An example class with the specified annotation fuzzyLabel
 Figure 9 – Software implementation of the Search-plugin for Protege 5.2.0 for finding precedents for a problem situation
 Figure 10 – Plugin Search-plugin launch
 Figure 11 – Plugin Search-plugin to Protege 5.2.0
 Figure 12 – Listing a SPARQL query for use cases
 Figure 13 – Evaluation results of the F-measure for searching for similar precedents for a problem situation (experiment fragment)
 Table 1 – Key elements of FuzzyOWL
 Table 2 – An example of determining fuzzy attributive proximity
 Table 3 – An example of determining fuzzy taxonomic proximity
 Table 4 – Experiment plan
 Table 5 – Results comparing tools precedents search
 Table 6 – Experiment results

References

- [1] Project Management Trends [In Russian]. <https://gantbpm.ru/trendy-proektnogo-upravleniya/>.
- [2] **Chernyakhovskaya LR, Malakhova AI.** Development of intellectual decision support models and methods based on ontology of software projects organization management [In Russian]. *Ontology of designing*. 2013; 4: 42–52.
- [3] A guide to the project management body of knowledge (PMBok Guide), Sixth ed. / Project Management Institute, Inc., 14 Campus Boulevard, Newton Square, Pennsylvania 19073-3299 USA, 2017, 756 p.
- [4] **Aamodt A, Plaza E.** Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communications*. IOS Press. 1994; 7(1): 39-59.
- [5] The Access Group: Inbox Insight Survey. http://www.theaccessgroup.com/media/250643/5_project_lessons_to_take_into_2014_whitepaper_v2.pdf.
- [6] Complete collection project management statistics. <https://www.wrike.com/blog/complete-collection-project-management-statistics-2015/>.
- [7] Top project management trends. http://www.esi-intl.co.uk/resource_centre/white_papers/top-project-management-trends-2016.pdf.
- [8] **Dyubua D, Prad A.** Theory of opportunity. Applications to the representation of knowledge in computer science [In Russian]. Moscow: Radio and communication Publs.; 1990. 288 p.
- [9] **Larichev OI.** Theory and Decision Making Methods [In Russian]. Moscow: Logos Publs.; 2006. 392 p.
- [10] **Madera AG.** Business processes and process management in the face of uncertainty: quantitative modeling and optimization [In Russian]. Moscow: LENAND Publs.; 2019. 160 p.

- [11] **Antonov VV, Kulikov GG.** Formal domain model based on fuzzy relations [In Russian]. *Software and systems*. 2011; 2(94): 48-51.
- [12] **Antonov VV, Kulikov GG.** Semantic and mathematical language of intellectual system structure description based on fuzzy logic [In Russian]. *Software and systems*. 2011; 3(95): 33-35.
- [13] **Antonov VV, Kulikov GG, Antonov DV.** Development a formal domain model using fuzzy clustering [In Russian]. *Vestnik UGATU*. 2011; 15(5): 3-11.
- [14] **Gavrilova TA, Kudryavtsev DV, Muromtsev DI.** Knowledge engineering. Models and methods [In Russian]. SPb: Lan' Publs; 2016. 324 p.
- [15] **Bolotova LS.** Artificial intelligence system: models and technologies based on knowledge [In Russian]. Moscow: Finance and statistics Publs. 2012. 664 p.
- [16] **Massel LV, Vorozhtsova TN, Pjatkova NI.** Ontology engineering to support strategic decision-making in the energy sector [In Russian]. *Ontology of designing*. 2017; 7(1): 66-76. DOI: 10.18287/2223-9537-2017-7-1-66-76.
- [17] **Barmina OV, Nikulina NO.** Intellectual system for managing the interaction of business processes in project-oriented organizations [In Russian]. *Ontology of designing*. 2017; 7(1): 48-65. DOI: 10.18287/2223-9537-2017-7-1-48-65.
- [18] **Bobillo F, Straccia U.** Fuzzy description logics with general t-norms and datatypes. *Fuzzy Sets and Systems*. 2009; 160 (23): 3382-3402.
- [19] Official documentation on Protégé. <http://protrgrwiki.stanford.edu/index.php/ProtegeUserDocs>.
- [20] **Yarushkina NG, Moshkin VS, Ishmuratova GR, Andreev IA, Moshkina IA.** Application of fuzzy time series and fuzzy ontology integration in diagnostics of technical systems [In Russian]. *Ontology of designing*. 2018; 8(4): 594-604. DOI: 10.18287/2223-9537-2018-8-4-594-604.
- [21] **Althof K-D, Auriol E, Barlette R, Manago M.** A Review of Industrial Case-Based Reasoning Tools. *AI Intelligence*. 1995, 159 p.
- [22] **Luger DF.** Artificial Intelligence: Strategies and Methods for Solving Complex Problems [In Russian]. Moscow: Williams Publishing House; 2003. 864 p.
- [23] **Levenshtein VI.** Binary codes capable of correcting deletions, insertions, and reversals, DAN SSSR, 1965, 163, No. 4, 845–848 [in Russian] (English translation: *Soviet Physics Doklady*, 1965; 10(8): 707–710).
- [24] **Zadeh I.** Fuzzy Logic, Neural Networks and Soft Computing // *Communication on the ACM*, 1994; 37(3): 77–84.
- [26] **Kryukov KV, Pankova LA, Pronina VA, Suhoverov VS, SHipilina LB.** Measures of semantic proximity in ontology [In Russian]. *Management issues.*, 2010; 5: 2–14.
- [27] **Chernyakhovskaya LR, Nikulina NO, Barmina OV.** The program of selection of precedents of problem situations in project management. Certificate of state registration of computer programs №2018615125; 26.04.2018.
- [28] Official documentation on JColibri. <http://gaia.fdi.ucm.es/research/colibri/jcolibri/>.
- [29] **Van Rijsbergen CJ.** Information Retrieval (2nd ed.). Butterworth-Heinemann. 1979. <http://www.dcs.gla.ac.uk/Keith/Preface.html>.
-

About the authors

Vyacheslav Viktorovich Aatonov (b. 1956) graduated from the Bashkir State University in 1979. PhD in Technical Science (2007), Doctor of Technical Science (2015). Professor of Automated management systems department of the Ufa State Aviation Technical University. He is the co-author about 30 scientific publications in the field of intellectual decision-making support systems development. AuthorID (RCI): 530537. Author ID (Scopus): 57200254522; ORCID 0000-0002-5402-9525; Researcher ID (WoS): AAH-5121-2019. Antonov.V@bashkortostan.ru.

Olesya Vladimirovna Barmina (b. 1988) graduated from the Ufa State Aviation Technical University in 2011, master's degree (2014), post-graduate course (2018) in computer science (Department of technical cybernetics). Now she is working as Senior Data Analyst at Nordigy LLC. Research fields are project management, AI, ontology engineering. She is the co-author more than 20 scientific articles and abstracts. ORCID 0000-0003-4404-1376. obarmina@outlook.com.

Natalya Olegovna Nikulina (b. 1971) graduated from the Ufa State Aviation Technical University in 1994, PhD in Technical Science (1998). She is an associate professor of Automated management systems department of the Ufa State Aviation Technical University. Research fields are project management, system modeling and designing information-systems in organizational management. She is the co-author more than 40 scientific articles and abstracts. AuthorID (RCI): 396597; ORCID 0000-0002-8594-2093. [nikulinano@outlook.com](mailto:nikulino@outlook.com).

Received December 8, 2019. Revised February 25, 2020. Accepted March 05, 2020.

Онтологии в борьбе с глобальной проблемой COVID-19

The emerging COVID-19 outbreak has reminded everyone the early versions of the SARS coronavirus outbreak and the Middle East coronavirus outbreak. These coronavirus outbreaks have made us realize the critical importance of systematically studying these coronavirus infectious diseases using the cutting edge technologies. Ontology provides a unique angle to study the coronavirus infectious diseases.

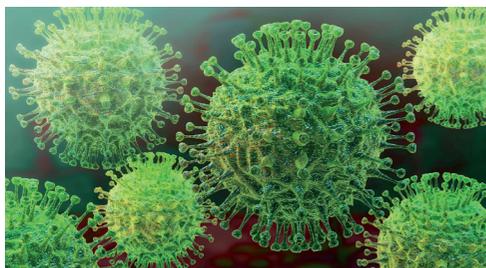


Фото с сайта Азбука здоровья: «Коронавирус: всё, что нужно знать»
<https://azzdorovya.ru/bolezni/infekcionnye/koronavirus-vsjo-chto-nuzhno-znat/amp/>

The **Ontology of Coronavirus Infectious Disease (CIDO)** is a community-driven open-source biomedical ontology in the area of coronavirus infectious disease. The CIDO is developed to provide standardized human- and computer-interpretable annotation and representation of various coronavirus infectious diseases, including their etiology, transmission, epidemiology, pathogenesis, diagnosis, prevention, and treatment. Its development follows the OBO Foundry Principles. CIDO has now been approved to be an OBO Foundry library ontology: <http://obofoundry.org/ontology/cido.html>.

CIDO was initiated and co-led by Dr. Hong Yu, director of Department of Respiratory and critical Care Medicine, Guizhou Province People's Hospital, Guizhou University Medical College, Guiyang, Guizhou, China, and Dr. Yongqun Oliver He, Associate Professor, University of Michigan Medical School, Ann Arbor, MI, USA.

Many graduate students, clinical doctors, and researchers in the He laboratory and Guizhou Province People's Hospital have also participated in the CIDO development.

Many researchers in the University of Michigan, including Dr. Brian Athey and Dr. Gil Omenn, have participated in the CIDO development.

Dr. Junguk Hur from University of North Dakota Medical School is also a CIDO developer. Dr. Hur is also interested in applying CIDO for more efficient literature mining.

Dr. Barry Smith, a well-known ontologist from the University at Buffalo, is also a member of our CIDO development team. Dr. Smith developed the **Basic Formal Ontology (BFO)**, a top level ontology used by over 300 ontologies around the world. BFO has now been approved to become ISO/IEC standard 21838. CIDO is aligned with BFO, and by doing so, CIDO is aligned with the other hundreds of ontologies.

Dr. Xiaolin Yang, an experienced ontologist and bioinformatician from the Institute of Basic Medical Sciences, School of Basic Medicine, Peking Union Medical College and Chinese Academy of Medical Sciences, Beijing, China, has also participated in our collaborative CIDO development and applications.

Everyone who would like to contribute to the CIDO development is welcome to join our development team. Thanks.

Collaborators:

CORD-19-on-FHIR - Semantics for COVID-19 Discovery

CIDO Source Link (free for downloading): OWL format: <https://raw.githubusercontent.com/CIDO-ontology/cido/master/src/ontology/cido.owl>

Tracker link: <https://github.com/CIDO-ontology/cido/issues>

CIDO in OBO Foundry: <http://obofoundry.org/ontology/cido.html>.

Web Deposition:

Ontobee: <http://www.ontobee.org/ontology/CIDO>

BioPortal: <http://biportal.bioontology.org/ontologies/CIDO>

Number of Terms (including imported terms)

Class (1725), ObjectProperty (33), AnnotationProperty (51), Instance (17)

Ontologists and designers of all countries and subject areas, join us!