

На правах рукописи



Гуськов Глеб Юрьевич

**МЕТОДЫ И СРЕДСТВА ФОРМИРОВАНИЯ
ПРЕДМЕТНЫХ ОНТОЛОГИЙ В
АВТОМАТИЗИРОВАННОМ ПРОЕКТИРОВАНИИ
ПРОГРАММНО-АППАРАТНЫХ КОМПЛЕКСОВ**

05.13.12 – Системы автоматизации проектирования (промышленность)

АВТОРЕФЕРАТ

диссертации на соискание ученой степени

кандидата технических наук

Ульяновск – 2018

Работа выполнена на кафедре «Информационные системы» Федерального государственного бюджетного образовательного учреждения высшего образования «Ульяновский государственный технический университет».

Научный руководитель:

Наместников Алексей Михайлович
кандидат технических наук, доцент,
доцент кафедры «Информационные
системы» ФГБОУ ВО «Ульяновский
государственный технический
университет»

Официальные оппоненты:

Ковалев Сергей Михайлович доктор
технических наук, профессор,
профессор кафедры «Автоматика и
телемеханика на железнодорожном
транспорте» ФГБОУ ВО «Ростовский
государственный университет путей
связи»,

Смагин Алексей Аркадьевич доктор
технических наук, профессор,
заведующий кафедрой
телекоммуникационных технологий и
сетей ФГБОУ ВО Ульяновский
государственный университет

Ведущая организация:

Федеральное государственное бюджетное учреждение науки «Институт автоматизации и процессов управления» Дальневосточного отделения Российской академии наук

Защита диссертации состоится «19» декабря 2018 г. в 15 часов 00 минут на заседании диссертационного совета Д 212.277.01 при Ульяновском государственном техническом университете по адресу: 432027, г. Ульяновск, ул. Северный Венец, 32 (ауд. 211, Главный корпус).

С диссертацией можно ознакомиться в библиотеке Ульяновского государственного технического университета, по адресу:

432027, г. Ульяновск, ул. Северный Венец, 32 и на сайте – <http://www.ulstu.ru/main?cmd=file&object=16152>

Автореферат разослан «__» _____ 2018 г.

Ученый секретарь

диссертационного совета 212.277.01,

доктор технических наук, профессор



Смирнов В.И.

Общая характеристика работы

Современное автоматизированное проектирование программно-аппаратных комплексов (ПАК) предполагает использование интеллектуального проектного репозитория, позволяющего выполнять поиск и повторное использование прототипов программного обеспечения на основе общего семантического представления предметных и проектных знаний. В настоящее время знания обычно представляются в форме онтологий.

Разработка онтологий весьма долгий и ресурсоёмкий процесс, требующий привлечения специалистов, обладающих широкими компетенциями, в том числе в онтологическом инжиниринге. На практике, как правило, разработчики не имеют достаточных знаний о предметной области (ПрО), которую им необходимо автоматизировать. Данная проблема появляется в связи с тем, что разработчики могут переходить с проекта на проект или вовсе менять место работы. Часто документы, регламентирующие предметную область, не фиксируют принятые при разработке программных продуктов семантические значения сущностей и отношений. Создание общей онтологии предметной области, учитывающей проектные решения программных продуктов позволяет повторно использовать найденные решения и сократить количество смысловых ошибок.

Основной проблемой использования онтологий в разработке программного обеспечения остаются высокие требования к разработчикам по знанию внутреннего устройства онтологий и их возможностей. Важность формализации концептов проблемной области для разработки ПАК привела к появлению специальных языков проектирования, которые включают в себя формализацию концептов (сущностей) проблемной области. Одним из распространенных средств проектирования является язык UML.

Средства UML позволяют разработать проект создаваемой системы, содержащий концептуальные элементы, такие как системные функции и процессы. Диаграммы на языке UML применимы к описанию конкретных особенностей системы, например, классов, составляющих архитектуру системы, таблиц и связей, описывающих схему базы данных, АРМ-операторов и серверов для создания схемы физического размещения.

Ряд отечественных и зарубежных ученых, в частности, Н.М. Боргест (автоматизированное проектирование в области самолетостроения, разработка робота-проектанта), В.А. Виттих (разработка методов и средств создания открытых мультиагентных систем для поддержки процессов принятия проектных решений), В.М. Курейчик (генетические алгоритмы в интеллектуальных САПР, эволюционное проектирование), С.В. Смирнов (разработка среды моделирования инженерных знаний, концептуальное моделирование), Загорулько, Т.А. Гаврилова, В.Ф. Хорошевский, А.С. Клещев, В.В. Грибова, Д.А. Новиков, В.В. Голенков, В.А. Виттих, , А.Ф. Тузовский (семантические модели представления предметных знаний в сфере автоматизированного проектирования), Т. Gruber, N. Guarino, M. Uschold, F.

Bobillo, U. Straccia, G. Rzevski (разработка интеллектуальных программных систем, основанных на онтологическом подходе) исследовали различные аспекты проблемы создания общей онтологии предметной области, учитывающей задачи автоматизации проектирования ПАК.

Актуальность темы исследования

В настоящее время постоянный рост требований к эффективности и качеству информационного обеспечения автоматизированного проектирования технических систем в условиях слабой формализации поставленных задач, особенно на ранних этапах проектирования, предполагает необходимость системного решения ряда научных задач:

- формирование семантического базиса представления проектов ПАК в интеллектуальном репозитории проектов;
- извлечение знаний из концептуальных моделей, использующих распространенные методологии разработки программно-аппаратных комплексов и представление их с помощью стандарта OWL;
- повышения эффективности автоматизированного проектирования ПАК за счет повторного использования прототипов, выбираемых из проектного репозитория на основе мер структурного подобия;
- формирование рекомендаций по развитию проекта на концептуальном уровне для предотвращения ошибок интерпретации предметной области.

Тема диссертации является актуальной так как работа посвящена разработке методов и средств представления проекта ПАК в форме OWL-онтологии в интеллектуальном проектном репозитории, включающем в себя инструменты поиска прототипа на основе структурного подобия и управления концептуальным развитием проекта ПАК.

Цель диссертационной работы

Целью диссертационной работы является снижение временных затрат на этапах проектирования и разработки программно-аппаратных комплексов за счет повторного использования проектных решений с помощью разработки и реализации моделей и алгоритмов поиска структурно-семантически схожих проектов по унифицированному онтологическому представлению знаний о предметной области, составленному на основе знаний, извлекаемых из проектных документов и формирования рекомендаций по управлению ходом развития проекта ПАК на основе интеграции классических и нечетких моделей прогнозирования временных рядов.

Задачи исследования

В соответствии с целью работы актуальными являются следующие задачи диссертационного исследования:

- провести сравнительный анализ современных методов, алгоритмов и систем поиска проектов ПАК схожих по архитектуре и предметной области;

- провести сравнительный анализ современных средств представления проектных диаграмм в интеллектуальном проектном репозитории;
- разработать формальную модель программного комплекса, определяемую прикладной онтологией, основанной на элементах объектно-ориентированных языков программирования с учетом жизненного цикла;
- разработать методику извлечения, формализации и использования знаний, извлеченных из систем контроля версий в рамках онтологического представления знаний;
- разработать методику и алгоритм трансляции проектных диаграмм и исходного кода в онтологическое представление проекта в рамках информационного обеспечения САПР;
- разработать способ управления автоматизированным проектированием проектных диаграмм проекта на основе прогнозирования временных рядов, выражающих развитие концептуальных диаграмм;
- разработать программные средства, позволяющие структурировать проекты в рамках экспертной выборки или репозитория крупного предприятия;
- исследовать возможности мер структурного подобия для эффективного отбора прототипа проекта;
- провести вычислительные эксперименты, позволяющие оценить эффективность предложенных моделей и алгоритмов в процессе проведения концептуального проектирования ПАК;
- внедрить результаты исследований в практику процесса проектирования технических систем предприятий.

Методы исследования

В диссертационной работе применяются методы онтологического анализа, дескрипционной логики, теории нечетких систем, прогнозирования временных рядов, а также объектно-ориентированного программирования при построении программного комплекса.

Научная новизна

Научная новизна результатов исследования заключается в следующем:

1. Предложены новые модели:
 - а. Онтологическая модель языка диаграмм UML, отличающаяся от известных тем, что она основана на мета-схеме языка UML, построенной в виде концептов и утверждений онтологии. Данная модель позволяет внести в онтологию информацию, извлеченную из любой диаграммы на языке UML и обеспечить гибкую поддержку новых спецификаций языка UML.
 - б. Онтологическая модель представления шаблона проектирования, отличающаяся от известных тем, что шаблон проектирования добавляется в онтологию в виде набора элементов, принадлежащих к классам онтологической модели языка UML, набора семантических ограничений и свойств шаблонов проектирования, описываемых в онтологии с помощью объектных свойств и свойств типов данных.

2. Разработаны новые меры архитектурного подобия программных проектов, отличающиеся новым способом вычисления степени выраженности шаблонов проектирования в рассматриваемых программных продуктах на основе их онтологических представлений.

3. Разработана методика переноса знаний из концептуальных моделей в онтологию OWL, отличающаяся тем, что отношения между элементами UML, такие как обобщения, ассоциации, реализации и зависимости, представлены набором элементов и объектных свойств с возможностью восстановления концептуальной модели по онтологическому представлению.

4. Разработан алгоритм трансформации UML диаграммы классов в онтологию формата OWL, отличающийся подходом к формированию иерархии классов UML-диаграммы и отношений между ними в OWL онтологию при помощи специфических наборов А-Вох утверждений.

5. Предложен новый алгоритм агрегации методов прогнозирования временных рядов показателей состояния проекта разработки программно-аппаратного комплекса, который отличается способом интеграции результатов применения четких и нечетких методов экспоненциального сглаживания.

Теоретическая значимость работы

Теоретическая значимость работы заключается в разработке и реализации новых эффективных моделей, алгоритмов и методики представления предметной области программно-аппаратного комплекса в информационном обеспечении САПР на основе формирования онтологий из UML-диаграмм и прогнозирования динамики концептуального развития проекта, обеспечивающих снижение количества смысловых ошибок проектировщика.

Практическая значимость работы

Разработанная система интеллектуального проектирования и анализа данных, реализующая предложенные модели и алгоритмы, была использована в ходе разработки:

1. Автоматизированной системы балансировки производственных мощностей системы управления АО «Авиастар-СП»;

2. «Системы интеллектуального поиска и анализа в Интернет-СМИ и социальных сетях» в рамках НИР НПЦ АО «НПО «Марс».

3. Подсистемы анализа данных при формировании прогнозов в ООО «Эверест Ресерч».

Основания для выполнения работы

Результаты диссертационной работы использовались в ряде НИОКР, выполненных в Ульяновском государственном техническом университете, направленных на решение научно-технических задач.

К наиболее важным результатам следует отнести:

1. Участие в выполнении гранта РФФИ №13-01-00324 «Исследование формальных методов грануляции слабоструктурированных информационных ресурсов на основе онтологии предметной области»;

2. Участие в выполнении гранта РФФИ №15-01-03000 А «Исследование формальных методов идентификации последовательности коротких сообщений на основе нечетких онтологических моделей»;

3. Участие в выполнении гранта РФФИ №15-41-02413 «Интеллектуальный анализ временных рядов на основе нечетких онтологий, извлекаемых из Интернет-ресурсов»;

4. Участие в выполнении гранта РФФИ №16-47-730715 р_а «Исследование и разработка метода нечеткого моделирования метрик для предикативной аналитики в задачах управления разработкой программного обеспечения»;

5. Участие в выполнении гранта РФФИ №16-47-730742 р_а «Интеграция онтологических моделей и проектных диаграмм при концептуальном проектировании сложных информационных систем»;

6. Участие в выполнении гранта РФФИ №16-47-732033 р_офи_м «Разработка моделей и средств онтологического анализа проектных диаграмм на основе методов машинного обучения»;

7. Участие в выполнении гранта РФФИ №16-47-732112 р_офи_м «Исследование и разработка методов прогнозирования временных рядов на основе многомодельного подхода»;

8. Участие в выполнении гранта РФФИ №17-07-00973 А «Исследование моделей и методов нечетких онтологий в задачах анализа программно-аппаратных комплексов»;

9. Участие в реализации государственного задания №2.1182.2017/4.6 «Разработка методов и средств автоматизации производственно-технологической подготовки агрегатно-сборочного самолетостроительного производства в условиях мультипродуктовой производственной программы»;

10. Участие в реализации государственного задания № 2.4760.2017/8.9 «Исследование и разработка моделей, методов и алгоритмов гибридизации нечетких предметных онтологий, логического вывода и интеллектуального анализа временных рядов»;

11. Участие в реализации хозяйственного договора с ООО «Эверест Ресерч» № «Д346» от 23.09.2014 г. «Разработка нечетких моделей и реализация нечетких методов прогнозирования временных рядов»;

12. Участие в реализации хозяйственного договора с ФНПЦ АО «НПО «Марс» от 01.02.17 № «72/17-УлГТУ» на выполнение НИР «Система интеллектуального поиска и анализа в Интернет-СМИ и социальных сетях».

Достоверность результатов диссертационной работы

Достоверность научных положений, выводов и рекомендаций подтверждена результатами вычислительных экспериментов, а также

результатами использования материалов диссертации в работе компаний ООО «Эверест-ресерч», ФНПЦ АО «НПО «Марс» и АО «Авиастар – СП».

Основные положения, выносимые на защиту

1. Модель онтологии языка UML, содержащая описание следующих элементов: класс, абстрактный класс, интерфейс, объект, пакет, метод и атрибут, позволяет сохранить знания о архитектуре ПАК в интеллектуальное хранилище проектов проектной организации.

2. Онтологическая модель шаблона проектирования, построенная на основе объектных свойств и свойств типов данных, адекватно представляет структурные особенности программного проекта.

3. Меры архитектурного подобия программных продуктов на основе вычисления степени выраженности шаблонов проектирования позволяют эффективно выделить прототипы среди существующих программных продуктов из репозитория.

4. Методика и алгоритм переноса знаний из концептуальных моделей и исходного кода проектов ПАК в онтологию OWL эффективно транслируют проектные знания в интеллектуальное хранилище проектов проектной организации.

5. Алгоритм агрегации методов, отличающийся от существующих подходом к объединению результатов работы четких и нечетких методов экспоненциального сглаживания прогнозирования временных рядов, позволяет управлять динамикой концептуального развития проекта программно-аппаратного комплекса.

Апробация работы

Основные положения и результаты диссертационной работы докладывались, обсуждались и получили одобрение на следующих конференциях, семинарах и симпозиумах: VIII Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (г. Коломна, 2015 г.); Всероссийской научно-практической конференции «Нечеткие системы, мягкие вычисления и интеллектуальные технологии» (г. Санкт-Петербург, 2017 г.); XIV национальной конференции по искусственному интеллекту с международным участием «КИИ-2014» (г. Казань, 2014 г.); VI и VII Международных научно-технических конференциях «Открытые семантические технологии проектирования интеллектуальных систем» (г. Минск, 2016, 2017 гг.); I Международной научной конференции «Интеллектуальные информационные технологии в технике и на производстве» (г. Сочи, 2016 г.), XV национальной конференции по искусственному интеллекту с международным участием «КИИ-2016» (г. Смоленск, 2016 г.), 4-й Всероссийской научно-технической конференции аспирантов, студентов и молодых ученых ИВТ-2012 (Ульяновск), Всероссийской научно-технической конференции «Теоретические и практические аспекты развития отечественного авиастроения» (г. Ульяновск,

2012 г.), IX Всероссийской школе-семинаре аспирантов, студентов и молодых ученых «Информатика, моделирование, автоматизация проектирования» (г. Ульяновск, 2017 г.); Международной конференции «Interactive systems: Problems of Human-Computer Interaction collection of scientific papers.» (г. Ульяновск, 2017 г.); Международной конференции «Creativity in intelligent technologies & Data Science» (г. Волгоград, 2015 г.); 2-ом международном научно-практическом семинаре «Soft computing applications and knowledge discovery (SCAKD 2016)» в рамках 13-ой международной конференции «Concept lattices and their applications (CLA 2016)» (г. Москва, 2016 г.); II Международной научной конференции «Интеллектуальные информационные технологии в технике и на производстве» (г. Варна, Болгария, 2017 г.); 12-ой Международной конференции «Uncertainty Modelling in Knowledge Engineering and Decision Making» FLINS 2016 (г. Рубе, Франция, 2016 г.), Мировой конференции по мягким вычислениям (г. Баку, Азербайджан, 2018 г.), 17-ой Международной конференции по искусственному интеллекту и мягким вычислениям (г. Закопане, Польша, 2018 г.).

Научные публикации

По результатам работы было опубликовано 32 статьи, 10 из которых в журналах из перечня ВАК, а также 6 статей в изданиях, индексируемых в Scopus. Получено 1 свидетельство о государственной регистрации программ для ЭВМ.

Личный вклад

Все результаты, составляющие содержание диссертации, получены автором самостоятельно. Подготовка к публикации некоторых результатов проводилась совместно с соавторами, вклад соискателя был определяющим.

Структура и объем диссертации

Диссертация состоит из введения, четырех глав, заключения, списка использованной литературы и приложений. Общий объем диссертации - 220 страниц, из них 172 страницы текста, включая 38 рисунков и 14 таблиц. Библиография включает 103 наименования.

Краткое содержание работы

Во введении рассмотрена актуальность выполненного исследования, формулируются цель и задачи работы, определяются теоретическая значимость и практическая ценность результатов исследования, а также положения, выносимые на защиту.

В первой главе рассматриваются наиболее распространённые модели жизненного цикла (ЖЦ) и методологии разработки программного-аппаратных комплексов.

Модели ЖЦ и методологии разработки описывают процесс разработки с разных точек зрения, тем не менее, довольно часто они регламентируют одни и те же аспекты разработки относительно итераций, сотрудников, технических задач, форм отчётности, документации и т.д. Кроме того, на практике

специалисты высшего уровня, каждого отдельного предприятия стараются выбрать конкретную модель ЖЦ под проект и сформулировать принципы и подходы некоторой комбинации методологий разработки для отдельно взятой команды разработчиков. Вследствие нетривиального подхода к разработке в каждом конкретном случае имеет смысл рассмотреть наиболее распространённые модели ЖЦ и методологии разработки на предмет обобщения проблем управления знаниями о предметной области и её модели в рамках ПАК.

Таблица 1

**Сравнение моделей ЖЦ и методологий разработки программного обеспечения
ПАК**

	Гибкая методология разработки	Итерационная модель/ инкрементная модель	Каскадная модель/ итерационная модель
Масштаб применения методологии	Отдел, группы не более 7 человек	Предприятие, крупная проектная организация	Государственная структура, крупная проектная организация
Периодичность обновления продукта	От 2 до 4 недель	От 3 до 12 месяцев	От 12 месяцев
Качество проектирования	Низкое	Высокое	Исчерпывающее, избыточное
Качество документации	Низкое	Среднее	Очень высокое
Скорость модификации продукта	Высокая	Низкая	Очень низкая
Тип программного обеспечения	Программное обеспечение, обрабатывающее информацию в рамках сессии: Web-сервисы, Web-сайты, Системы управления информацией, Экспериментальное ПО	Программное обеспечение, хранящее персональную информацию, влияющую на качество жизни пользователей: программные инструменты, Web-сервисы, системы обработки персональной информации, исследовательские платформы.	Программное обеспечение, от которого напрямую зависит жизнь пользователей: встраиваемое ПО, авиационное ПО, навигационное ПО, медицинское ПО операционные системы, пакеты моделирования.

В случае использования гибкой методологии в качестве основы при разработке, управление знаниями осуществляется за счёт мероприятий, именуемых «Scrum Meeting». Документирование и формальное описание знаний о предметной области в рамках гибких методологий разработки практически игнорируется.

Инструмент управления знаниями, позволяющий автоматически извлекать знания из артефактов, порождаемых в ходе разработки, обеспечивающий непротиворечивость знаний, поддерживающий систему рекомендаций, основанных на данных о ходе разработки представленных в виде временных рядов, востребован как для разработки крупных ПАК, основанной на модели ЖЦ, так и для создания проектов со сложно прогнозируемым объёмом логики процессов, разрабатываемых в рамках гибких методологий.

Подобный инструмент позволит специалистам оптимизировать затраты времени на актуализацию формальных документов и сократит риски потери знаний при выбывании опытного специалиста из команды.

В рамках первой главы диссертационного исследования также рассматривается задача выбора формата представления знаний для реализации программной системы интеллектуального проектирования и разработки (СИПР), предоставляющей возможности управления знаниями о предметной области и её модели ПАК.

Третий раздел первой главы посвящён рассмотрению существующих систем управления разработкой программного обеспечения, которое является на текущий момент существенной частью каждой ПАК.

Во второй главе описана разработанная в ходе исследования онтология на основе метамодели языка UML. Разработанная онтология (п. 2.1.) построена из концептов, которые являются базовыми элементами диаграммы классов. Данный подход к построению онтологии позволяет вносить изменения и расширять множество элементов онтологии в случае необходимости. В ходе построения онтологии на основе метамодели языка UML была предложена следующая нотация (модель онтологии).

Построенная онтология описывается следующим выражением:

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

где $C^{prj} = \{c_1^{prj}, \dots, c_i^{prj}\}$ – множество концептов онтологии построенных на основе элементов языка UML : "Class", "Object", "Interface", "Relationship" и т.д.;

R^{prj} – множество связей между концептами онтологии, описывающих отношения между элементами языка UML;

F^{prj} – множество функций интерпретации, определённых на множестве R^{prj}

В п. 2.2. рассмотрено онтологическое представление шаблонов проектирования. Шаблоны проектирования заносятся в онтологию как множество экземпляров концептов разработанной онтологии. Семантика, ограничения и свойства шаблонов проектирования определяются с помощью ObjectProperties и DatatypeProperties онтологии.

Формально онтологическое представление шаблона проектирования можно представить следующим образом:

$$O_{tmp_i}^{prj} = \{inst(C_1^{prj}), \dots, inst(C_{rel1}^{prj}), \dots, r_{sameAs}\},$$

где $inst(C_1^{prj})$ – экземпляр концепта в онтологии, построенной на основе мета-схемы языка UML; $inst(C_{rel1}^{prj})$ – отношение между элементами шаблона проектирования, представленное в виде экземпляра концепта; r_{sameAs} – онтологическое отношение, обозначающее эквивалентность входящих в него экземпляров.

Таблица 2

Сравнение форм представления знаний

	Логическая модель	Семантическая сеть	Продукционная модель	Фрейм	Онтология
Описание иерархии концептов ПрО	Логические переменные	Узлы сети (название)	Логические переменные	Фрейм	T-Box
Описание внутреннего строения концепта ПрО	Нет	Нет	Нет	Слоты – свойства, преобразования.	Свойства данных
Описание связей между концептами ПрО	В виде высказываний	В виде связей между узлами (название)	В виде продукционных правил	Слоты- связи	Объектные свойства
Поддержка базовых логических свойств связей между концептами ПрО	В виде высказываний	Нет	В виде продукционных правил	Нет	Да, на уровне объектных свойств
Соответствие концептов ПрО объектно-ориентированной модели	Нет	Нет	Нет	Да	Да
Наличие графического отображения знаний ПрО	Нет	Граф	Граф - семантическая сеть	Таблица, нет представления множества фреймов	Граф (Концепты + экземпляры)
Контроль непротиворечивости знаний	Да	Нет	В виде набора правил продукции	Нет	Да (за счёт МЛВ)
Возможность применения существующих машин логического вывода (МЛВ)	Нет, является базовой МЛВ	Нет, продукционные модели	Нет, является МЛВ	Нет, ЛВ осуществляется через слоты-преобразования	Да
Интерпретируемость результатов и доказательства логического вывода (ЛВ)	Нет	Нет	Нет	Нет	Да
Возможность сравнения/ объединения документов	Нет	Да	Нет	Да	Да
Наличие общепризнанной нотации, стандарта, мета-схемы	Модель	Модель	Модель	Модель	OWL 2:W3C
Редакторы документов в данном формате	Единичные разработки	Единичные разработки	Единичные разработки	Единичные разработки	Protege и др.
Совместимые машины логического вывода	Нет	Нет	Нет	Нет	Fact++, Pellet и др.
Наличие программных библиотек	Единичные разработки	Community	Единичные разработки	Единичные разработки	OWL API Java(Official) OWL API C++/.NET (Community)

Онтологическое представление шаблона проектирования состоит из набора экземпляров концептов и экземпляров отношений онтологии, определённых на базе метамодели языка UML.

Так как основной целью диссертационной работы является сокращение времени на разработку программного обеспечения при разработке ПАК путём повторного использования архитектурных программных решений, были построены новые метрики для вычисления подобия между проектируемыми и уже реализованными программными проектами.

Для вычисления меры архитектурного подобия программных проектов в п. 2.3. предложена новая формула вычисления степени выраженности шаблона проектирования в программном проекте:

$$\mu_{prj,tmp} = \frac{|C_{prj} \cap C_{tmp}| + |R_{prj} \cap R_{tmp}|}{|C_{tmp}| + |R_{tmp}|}$$

где $\mu_{prj,tmp}$ – мера выраженности шаблона проектирования в программном продукте,

C_{prj} и C_{tmp} – экземпляры концепта онтологии программного проекта или шаблона проектирования,

R_{prj} и R_{tmp} – экземпляры отношения онтологии программного проекта или шаблона проектирования.

Для вычисления архитектурного подобия между программными продуктами необходимо вычислить меру выраженности для каждого шаблона проектирования из составленного множества шаблонов.

После вычисления меры выраженности каждого отобранного шаблона проектирования в каждом из рассматриваемых программных проектов, становится возможным вычисление меры подобия между программными проектами по одной из трёх предложенных метрик (п. 2.4.).

Первая метрика позволяет рассчитать меру подобия с помощью выделения наиболее выраженного шаблона проектирования в каждом из проектов:

$$\mu_{dc_\gamma, dc_\delta} = \bigvee_{tmp \in (dc_\gamma \cap dc_\delta)} \mu_{dc_\gamma \cap dc_\delta}(tmp),$$

где dc_γ и dc_δ – UML диаграмма классов проекта, представленная в виде выражений A-box разработанной онтологии; $\mu_{dc_\gamma \cap dc_\delta}(tmp)$ – мера подобия проектов γ и δ .

Подобная метрика показывает хорошие результаты для сравнительно небольшого количества сложных комбинированных шаблонов проектирования. Подобные шаблоны проектирования основываются на предметной области и в меньшей мере соответствуют шаблонам проектирования в привычном понимании промышленного программирования.

Вторая метрика учитывает меру выраженности шаблонов проектирования, превышающую определённое пороговое значение. Экспериментальным путём

было подобрано пороговое значение, равное 0,3. В случае, если мера выраженности шаблона проектирования ниже 0,3 можно сделать вывод об отсутствии в программном проекте данного шаблона проектирования, и как следствие, такой шаблон проектирования необходимо исключить из рассмотрения:

$$\mu_{dc_\gamma, dc_\delta} = \left(\bigvee_{tmp \in (dc_\gamma \cap dc_\delta) \geq 0,3} \mu_{dc_\gamma \cap dc_\delta} \right) / N,$$

где N – число шаблонов проектирования с мерой выраженности выше 0,3 для каждого из проектов.

Третья, предлагаемая в работе метрика, схожа по семантике со второй метрикой, но накладывает дополнительное условие на учёт вклада меры выраженности шаблона проектирования в меру архитектурного подобия проектов:

$$\mu_{dc_\gamma, dc_\delta} = \left(\bigvee_{tmp \in (dc_\gamma \cap dc_\delta) \geq 0,3} \tilde{\mu}_{dc_\gamma \cap dc_\delta} \right) / N,$$

где $\tilde{\mu}_{dc_\gamma \cap dc_\delta}$ – взвешенная мера выраженности шаблона проектирования в программном проекте.

П. 2.5 второй главы рассматривает проектирование ПАК, как цикличное проектирование. Разработан алгоритм трансляции, позволяющий синхронизировать изменения между онтологией, проектными диаграммами и исходным кодом проекта.

Алгоритм системы интеграции принимает на вход UML – диаграмму классов в форме XMI документа. XMI документ представляет из себя XML файл определённой внутренней структуры. Алгоритм представлен на рисунке 1.

Руководителю проекта необходимо выполнить прогнозирование возможных вариантов развития проекта и своевременно принять проектные и организационные решения, повышающие эффективность разработки средства автоматизации. Одним из наиболее выразительных способов представления знаний о ходе разработки являются временные ряды основных показателей проекта. В процессе автоматизированного проектирования руководитель проекта может эффективно отследить состояние и динамику реализации проекта на основе модели описывающей проект:

$$P = \langle TS_{releases}, TS_{commits}, TS_{tasks}, TS_{concepts}, TS_{states}, F_{states} \rangle, \text{ где}$$

$TS_{releases}$ – временной ряд сборок версии программного обеспечения, поставляемого клиентам (release);

$TS_{commits}$ – временной ряд изменений исходного кода (commits);

TS_{tasks} – временной ряд задач (tasks);

$TS_{concepts}$ – временной ряд изменённых, добавленных и удалённых концептов онтологии предметной области, извлечённых из диаграммы классов проекта за последний этап проектирования (для классической разработки с моделью ЖЦ)

или извлеченных из исходного кода программы за последний этап или спринт разработки;

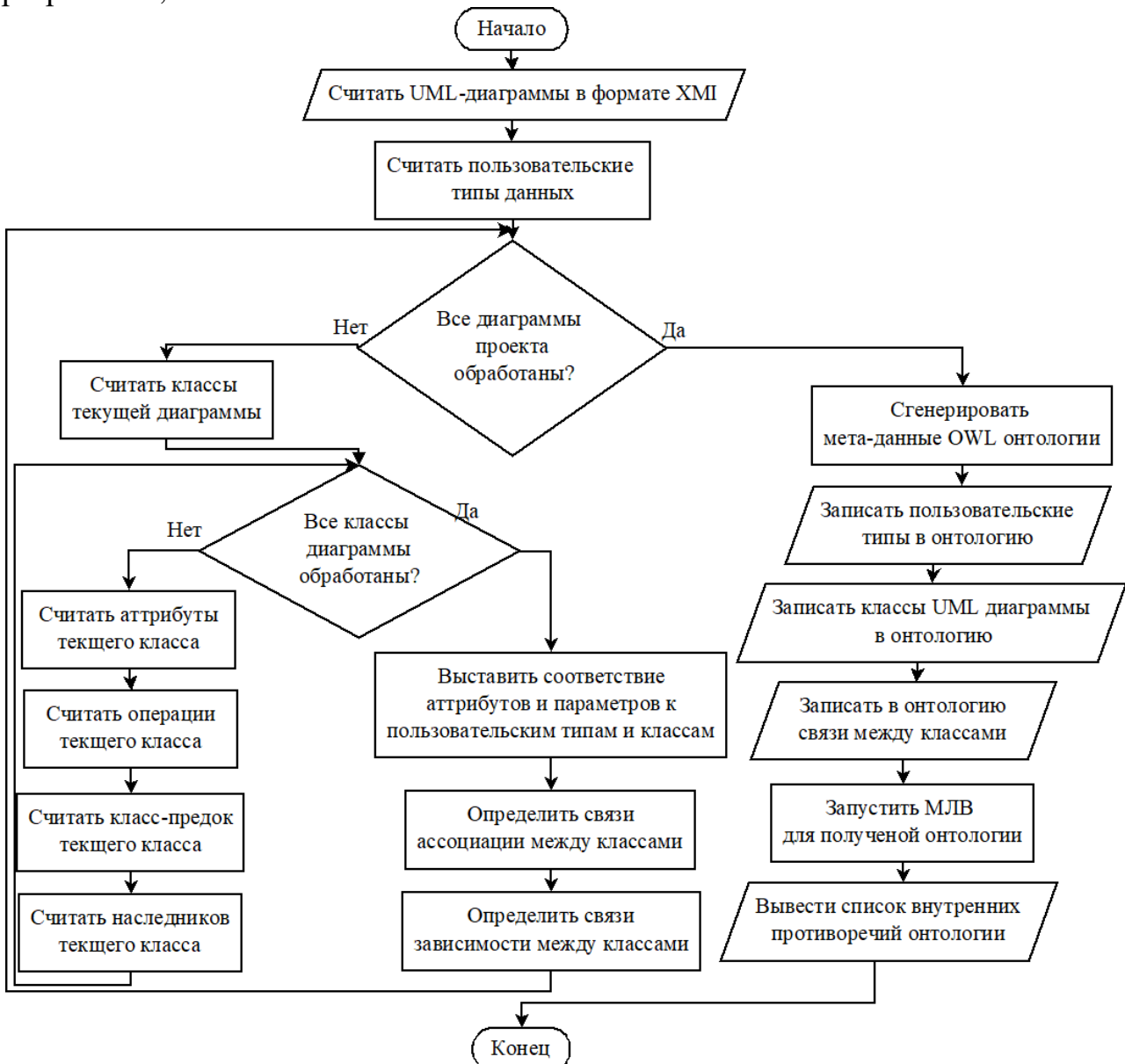


Рис. 1. Блок-схема алгоритма переноса знаний из набора UML-диаграмм в онтологию формата OWL.

TS_{states} – временной ряд состояний программного проекта, основанный на интеграции показателей остальных временных рядов, характеризующий проект;
 F_{states} – прогноз состояний программного продукта, полученный с помощью выбранного коллектива или комбинации методов.

$$TS_{releases} = \langle P_t^{rel} \rangle, \text{ где}$$

$t \in [0; N], N$ – количество точек временного ряда $TS_{releases}$,

P_1^{rel} – точка временного ряда $TS_{releases}$;

$$P_i^{rel} = \langle Label, Date_{plan}, Date_{fact} \rangle, \text{ где}$$

$Label$ – метка или название сборки, позволяющее её идентифицировать,

$Date_{plan}$ – дата сборки по плану,

$Date_{fact}$ - дата поставки сборки по факту.

$$TS_{commits} = \langle P_t^{com} \rangle, \text{ где}$$

$t \in [0;N], N$ – количество точек временного ряда $TS_{commits}$,

P_1^{com} – точка временного ряда $TS_{commits}$;

$$P_i^{com} = \langle Label, Date, Operation, Type, Author \rangle, \text{ где}$$

$Label$ – уникальный номер и комментарий к изменению исходного кода, позволяющее её идентифицировать,

$Operation$ – изменений: {Addition, Edit, Remove, All}

$Type$ – тип вносимых изменений: {Gui, Business Logic, Subject area}

$Date$ – дата внесения изменения в исходный код.

$Autor$ – автор изменений.

$$TS_{tasks} = \langle P_t^{task} \rangle, \text{ где}$$

$t \in [0;N], N$ – количество точек временного ряда TS_{tasks} ,

P_1^{task} – точка временного ряда TS_{tasks} ;

$$P_i^{task} = \langle Label, Date_{plan}, Date_{fact} \rangle, \text{ где}$$

$Label$ – уникальный номер и название задачи в программе, позволяющее её идентифицировать,

$Date_{plan}$ - дата реализации задачи по плану,

$Date_{fact}$ - дата реализации задачи ошибки по факту.

$$TS_{concepts} = \langle P_t^{concept} \rangle, \text{ где}$$

$t \in [0;N], N$ – количество точек временного ряда $TS_{concepts}$,

$P_1^{concept}$ – точка временного ряда $TS_{concepts}$;

$$P_i^{concept} = \langle Label, Date, \rangle, \text{ где}$$

$Label$ – уникальный номер версии проектного решения,

$Date$ – дата создания новой версии проектного решения,

$$TS_{states} = \langle P_t^{state} \rangle, \text{ где:}$$

$t \in [0;N], N$ – количество точек временного ряда TS_{states} ,

P_1^{state} – точка временного ряда TS_{states} ;

$$P_i^{state} = \langle Date, State \rangle, \text{ где}$$

$Date$ – дата присвоения проектному решению данного состояния,

$State$ – одно из множества состояний проекта {Start, Unexpected fast, fast, Normal, Stable, Chaotic, Danger, Fail}

$$F_{states} = \langle P_t^{state} \rangle, \text{ где}$$

$t \in [0;N], N$ – количество точек временного ряда F_{states} ,

P_1^{state} – точка временного ряда F_{states} ;

$$P_i^{state} = \langle Date, State \rangle, \text{ где}$$

$Date$ – дата присвоения проектному решению данного состояния,

$State$ – одно из множества состояний проекта {Start (0), Unexpected fast (4), fast (3), Normal(2), Stable(1), Chaotic (-1), Danger (-2), Fail(-3)}

Временные ряды могут быть спрогнозированы с помощью классических и нечётких моделей, но эффективность прогноза существенно возрастает при использовании коллективов и комбинаций моделей.

В п 2.6. описаны разработанные методы: комбинация моделей с использованием весов, полученных с помощью информационных критериев, комбинация с использованием нечётких весов и коллектив, полученный агрегирующим алгоритмом. В ходе исследования был реализован массив методов для прогнозирования временных рядов, состоящий из классических методов экспоненциального сглаживания и нечетких методов прогнозирования временных рядов. Часть нечётких методов были предложены в ходе работы по хозяйственному договору с ООО «Эверест ресерч» и основаны на интеграции нечеткой модели и традиционного экспоненциального сглаживания. Отличительной особенностью коллективов и комбинаций моделей прогнозирования, реализованных в рамках данного диссертационного исследования, является интеграция результатов работы четких и нечетких моделей.

Временные ряды выделенных показателей невозможно интерпретировать как состояние программного продукта, и как следствие, невозможно дать рекомендации исходя из их прогноза. Поэтому было решено выделять семантически значимые комбинации и последовательности показателей, которые получают семантическую метку описывающую изменение состояния программного обеспечения.

Таблица 3

Соответствия последовательностей и комбинаций показателей, изменениям, вносимым в проект

Последовательность или комбинация показателей	Изменение состояния
ЕСЛИ P_{i+1} не существует	TO $P_i^{state} : \{State - Start\}$
ЕСЛИ $P_i^{rel} : \{Date_{plan} > Date_{fact}\} \ \&\& \ P_i^{state} : \{State - Start, Unexpected fast, Fast\}$	TO $P_i^{state} : \{State - Normal\}$
ЕСЛИ $P_i^{rel} : \{Date_{plan} > Date_{fact}\}$	TO $P_i^{state} : \{State - Danger\}$
ЕСЛИ $>70\% P_i^{task} : \{Date_{fact} > Date_{plan}\}$	TO $P_i^{state} : \{State - Unexpected fast\}$
ЕСЛИ $>30\% P_i^{task} : \{Date_{fact} > Date_{plan}\}$	TO $P_i^{state} : \{State - Fast\}$
ЕСЛИ $>30\% P_i^{commit} : \{Operation - All\}$	TO $P_i^{state} : \{State - Chaotic\}$
ЕСЛИ $>50\% P_i^{commit} : \{Operation - All\} \ \&\& \ P_i^{state} : \{State \neq Start, Unexpected fast, Fast\} \ \&\& \ P_i^{rel} : \{Date_{plan} > Date_{fact}\}$	TO $P_i^{state} : \{State - Fail\}$
ЕСЛИ Last $(P_i^{commit} : \{Type - Subject area\}).Date > Last (P_t^{concept}).Date$	TO : Необходимо обновить проектную часть

В третьей главе приведена архитектура и функциональные возможности разработанной системы интеллектуального проектирования и разработки программно-аппаратных комплексов.

Система интеллектуального проектирования и разработки (СИПР) – модульное программное решение, предназначенное для автоматизации работы с

проектными документами, исходным кодом, данными из систем контроля версий и отслеживания ошибок. В качестве унифицированного формата хранения знаний (УФХЗ) в СИПР используется онтология в формате OWL, Т-Box которой построен на основе мета-схемы нотации языка UML

Основные функции СИПР:

1. Индексирование проектных документов – формирование индекса в виде документа УФХЗ.
2. Индексирование исходного кода ПАК – формирование индекса в виде документа УФХЗ.
3. Хранение и обновление индексов ранее спроектированных и разработанных проектов в виде документа УФХЗ.
4. Хранение и обновление знаний предметной области в виде шаблонов проектирования, представленных документами УФХЗ.
5. Вычисление меры структурно-семантического подобия проектов на основе предметной области.
6. Проверка онтологии проекта в формате УФХЗ на согласованность с помощью машины логического вывода

СИПР состоит из двух подсистем:

- Подсистема поддержки интеллектуального проектирования (ПИП) программных продуктов на основе знаний, хранящихся в онтологии УФХЗ.
- Подсистема интеллектуальной разработки (ПИР) ПАК. Данная подсистема позволяет на основании знаний, полученных из прикладных систем разработки программного обеспечения, генерировать рекомендации по дальнейшему развитию проектируемого ПАК.

В основе ПИР лежит подход к представлению данных в виде временных рядов с последующей обработкой. Разработанная система была внедрена в состав инструментов, используемых при проектировании в ФНПЦ АО «НПО «Марс» в рамках выполнения НИР «Системы интеллектуального поиска и анализа в Интернет-СМИ и социальных сетях» и АО «Авиастар-СП» в рамках создания ПАК при оптимизации ресурсов на основе производственно-технологического моделирования в условиях мультипродуктовой производственной программы.

В качестве хранилища проиндексированных документов использована онтология в формате OWL. Для обеспечения работы системы и хранения внутренних данных была использована база данных PostgreSQL.

Модули системы СИПР реализованы на разных языках программирования с использованием различных IDE. Объединение результатов работы осуществляется за счёт использования УФХЗ. Система СИПР включает следующие модули:

- Модуль извлечения знаний из проектных диаграмм, реализованный на языке программирования C# в среде разработки Visual Studio.

- Модуль извлечения знаний из исходного кода ПАК, реализованный на языке программирования Java в среде разработки IntelliJ IDEA.

- Модуль извлечения знаний из систем контроля версии и систем отслеживания ошибок, реализованный на языке программирования Java в среде разработки IntelliJ IDEA.

- Модуль определения структурно-семантического подобия, реализованный на языке программирования Java в среде разработки IntelliJ IDEA.

- Модуль прогнозирования проектных временных рядов, реализованный на языке программирования C++ в среде разработки Visual Studio.

- Модуль формирования рекомендаций по управлению развитием программного обеспечения, входящего в состав или ПАК, реализованный на языке программирования Java в среде разработки IntelliJ IDEA.

В четвертой главе проводится анализ адекватности разработанных моделей и методов на основе вычислительных экспериментов.

Для осуществления анализа временных затрат на процесс проектирования и разработки ПАК проведен ряд экспериментов. Учитывалось общее время разработки проекта, а также временные затраты на анализ существующих решений, проектирование, разработку и тестирование/отладку.

В результате проведения первого эксперимента необходимо было проверить гипотезу о том, что разработанная система СИПР, позволит существенно сократить время, затрачиваемое на анализ уже реализованных проектов.

Эксперимент заключался в сравнении времени, затрачиваемом на анализ проектов с помощью только ключевых слов, и с помощью применения системы СИПР к результатам поиска по ключевым словам.

Рассматриваемые проекты были загружены из открытого репозитория github.com.

Соответствие проектов, отобранных для анализа с помощью традиционного и предлагаемого подходов предметной области устанавливалось с помощью экспертов. Задача отбора и анализа проектов на стадии проектирования новой ПАК является нетривиальной и творческой, как правило оценки качества решения подобных задач даются экспертами. Экспертами выступали специалисты: разработчики, проектировщики и менеджеры проектов, входящие в состав научной группы кафедры «Информационные системы» УлГТУ и специалисты предприятий: ФНПЦ АО «НПО «Марс» и АО «Авиастар-СП». Результаты первого эксперимента приведены на рисунке 2.

Исходя из результатов, продемонстрированных в ходе проведения первого эксперимента, можно сделать вывод о существенном сокращении объема работы и как следствие временных затрат на этапах анализа существующих решений и проектирования нового ПАК.

Для проекта разрабатываемого совместного с ФНПЦ АО «НПО «Марс» количество прототипов проектов сократилось 5,96 раза, а для проекта разрабатываемого совместного с АО «Авиастар-СП» - в 2,9 раза.

Качество работы системы определяется количеством релевантных проектов, входящих в выборку.



Рис. 2. Результаты эксперимента по поиску структурно-семантически схожих проектов.

Для проекта, разрабатываемого совместно с ФНПЦ АО «НПО «Марс», качество выборки по ключевым словам составило 12,4%, а с помощью системы СИПР 59,2%, что привело к повышению точности в 4,77 раза.

Для проекта, разрабатываемого совместно с АО «Авиастар-СП», качество выборки по ключевым словам составило 12,5%, а с помощью системы СИПР 27%, что привело к повышению точности в 2,18 раза.

Результаты первого эксперимента демонстрируют преимущество во временных затратах при использовании разработанной системы. Потери релевантных проектов составили 20% в первом случае и 25% во втором случае, при сокращении объёма выборки в среднем 4,5 раза.

Данный результат характеризует использование системы положительно. Подсистему поддержки интеллектуального проектирования (ПИП) системы СИПР можно рекомендовать применять в случае большого размера выборки с относительно малым количеством релевантных проектов.

Результатом применения подсистемы ПИР стало управление проектом в рамках разработки «Автоматизированной балансировки мощностей системы управления АО «АВИАСТАР-СП». На рисунке 3 изображены временные ряды состояний, структурно-семантически схожих программных проектов, проекта НИР и прогноз, полученный от подсистемы ПИР.

Из результатов эксперимента видно, что несмотря на неполноту временных рядов структурно-семантически схожих проектов, удалось скорректировать развитие НИР относительно прогнозных значений.

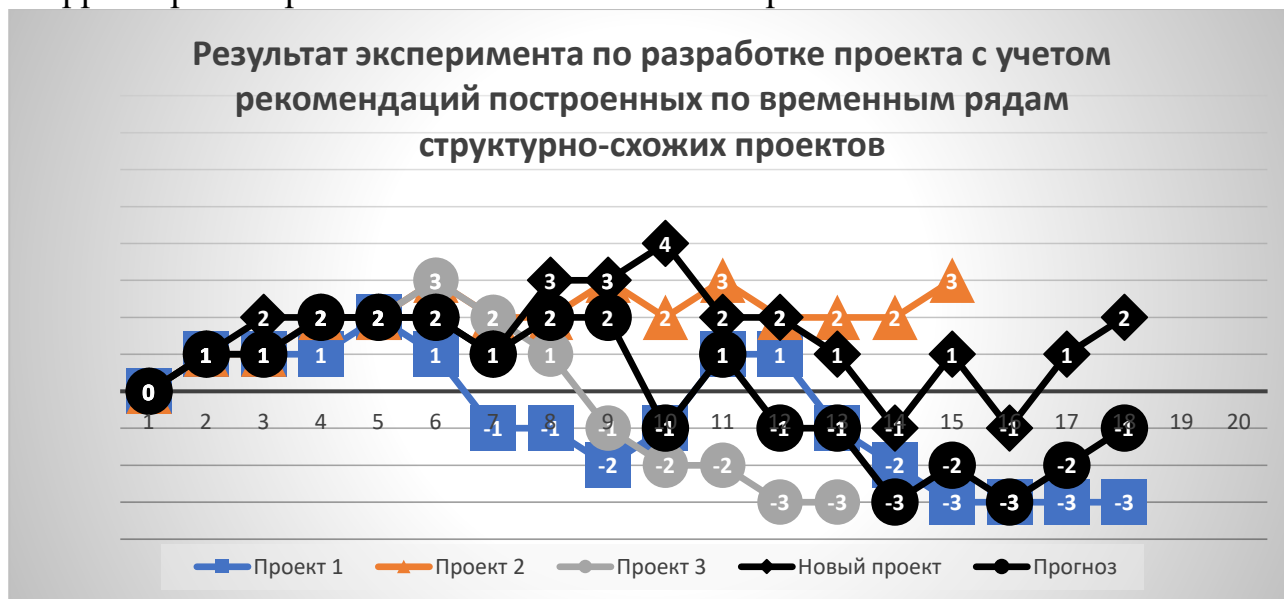


Рис. 3. Разработка проекта с учетом рекомендаций построенных по временным рядам структурно-схожих проектов.

Для использования четких моделей прогнозирования использовалось приравнивание состояний проекта к целым числам от -3 до 4, где Fail – это -3, Start – 0, а Unexpected fast – 4.

В качестве алгоритма агрегирования для построения коллектива моделей использовался агрегирующий алгоритм Вовка.

В заключении сформулированы основные выводы и результаты диссертационной работы.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

В ходе диссертационного исследования получены следующие научные результаты:

1. Онтологически-ориентированная модель языка диаграмм UML и онтологическая модель шаблона проектирования.

2. Меры архитектурного подобия программных проектов; меры выраженности шаблона проектирования в рассматриваемых программных продуктах.

3. Методика переноса знаний из концептуальных моделей в онтологию OWL.

4. Алгоритм трансформации UML – диаграммы классов в онтологию формата OWL.

5. Алгоритм агрегации методов прогнозирования временных рядов показателей состояния проекта разработки программно-аппаратного комплекса.

6. Для решения задач исследования и проверки научных результатов была разработана программная система интеллектуального проектирования и разработки, отличающаяся от известных:

- поддержкой проектов на каждом этапе жизненного цикла;
- унифицированным форматом хранения знаний о проекте;

- блоком анализа, позволяющего оценить состояние проекта автоматически.

7. Проведены вычислительные эксперименты, подтверждающие эффективность предложенных моделей и алгоритмов. Для проекта, разрабатываемого совместно с ФНПЦ АО «НПО «Марс», количество прототипов проектов сократилось 5,96 раза, а для проекта, разрабатываемого совместно с АО «Авиастар-СП» - в 2,9 раза.

8. Проведены вычислительные эксперименты, подтверждающие эффективность прогнозирования развития проекта ПАК на основе временных рядов. Для проекта, разрабатываемого совместно с ФНПЦ АО «НПО «Марс», точность выбора прототипа проектирования возросла в 4,77 раза за счет использования системы СИПР, а для проекта, разрабатываемого совместно с АО «Авиастар-СП», точность возросла в 2,18 раза.

9. Внедрение СИПР в проектной организации ФНПЦ АО «НПО МАРС» и на промышленном предприятии АО «Авиастар-СП» позволили успешно и в срок реализовать в первом случае проект «Система интеллектуального поиска и анализа», во втором случае – проект «Автоматизированной системы балансировки производственных мощностей».

СПИСОК ПУБЛИКАЦИЙ

Статьи, опубликованные в журналах, рекомендованных Перечнем

ВАК России:

1. Гуськов, Г.Ю. Прогнозирование временных рядов коллективами методов / Артюхов М.В., Г.Ю. Гуськов, А.А. Романов, И.А. Тимина // Радиотехника. – М.: Радиотехника. – № 6. – 2015. – С. 48-54.

2. Гуськов, Г.Ю. Интеграция нечетко-гранулярных и онтологических методов в задаче анализа временных рядов / Т.В. Афанасьева, Г.Ю. Гуськов, А.М. Наместников, Н.Г. Ярушкина // Автоматизация процессов управления. – 2015. – №2(40). – С. 72-79.

3. Гуськов, Г.Ю. Разработка многоагентной системы извлечения знаний из гетерогенных источников / Г.Ю. Гуськов, В.С. Мошкин, А.М. Наместников, А.А. Филиппов, Н.Г. Ярушкина // Радиотехника. – М.: Радиотехника. – 2016. – №9. – С. 57-63.

4. Гуськов, Г.Ю. Интеллектуальная система управления проектами разработки программного обеспечения / Г.Ю. Гуськов, А.М. Наместников, И.А. Тимина, Н.Г. Ярушкина // Вестник Ростовского государственного университета путей сообщения, –2016.–№ 3.(63) – С. 31-36.

5. Гуськов, Г.Ю. Система управления программными проектами на основе онтологического подхода / Г.Ю. Гуськов, А.М. Наместников // Автоматизация процессов управления. – 2016. – №3(45). – С. 88-94.

6. Гуськов, Г.Ю. Подход к поиску похожих по структуре проектов, основанный на онтологии языка uml / Г.Ю.Гуськов, А.М. Наместников, Н.Г. Ярушкина // Радиотехника. – М.:Радиотехника. – 2017. – № 6 – С. 122-127.

7. Гуськов, Г.Ю. Подход к поиску похожих по структуре проектов, основанный на онтологии языка uml / Т.В. Афанасьева, Г.Ю.Гуськов, А.М.

Наместников, В.Н. Негода, А.А. Романов, М.К. Самохвалов, Н.Г. Ярушкина // Радиотехника. – М.:Радиотехника. – 2017. – № 6 – С. 122-127.

8. Гуськов Г.Ю., Интеграция проектных диаграмм и онтологий в задаче балансировки мощностей авиастроительного предприятия / Т.В. Афанасьева, Г.Ю.Гуськов, А.М. Наместников, В.Н. Негода, А.А. Романов, М.К. Самохвалов, Н.Г. Ярушкина // Автоматизация процессов управления. –2017. –№ 4 (50). С. 85-93.

9. Гуськов, Г.Ю. Мягкие вычисления в интеллектуализации процессов проектирования в САПР / Т.В. Афанасьева, А.М. Наместников, В.С. Мошкин, Г.Ю. Гуськов, Н.Г. Ярушкина, // Нечеткие системы и мягкие вычисления. – Тверь: Изд-во ТвГУ –2017. – № 1(12). – С. 19-44.

10. Гуськов, Г.Ю. Применение нечеткой базы знаний проблемной области в задаче поиска архитектурно подобных программных проектов / А.А Филиппов, В.С. Мошкин, Г.Ю. Гуськов, Н.Г. Ярушкина // Нечеткие системы и мягкие вычисления. – Тверь: Изд-во ТвГУ –. 12. № 2. С. 107-120.

проиндексированные в SCOPUS

11. Guskov G., Yarushkina N., Afanasieva T., Zavarzin D. Fuzzy trends data mining in knowledge discovery process // Proceedings of the First Conference Creativity in Intelligent Technologies and Data Science, CIT&DS 2015. 2015. pp. 115-123.

12. Guskov G., Afanasieva T., Yarushkina N., Zavarzin D., Romanov A. Time series forecasting using combination of exponential models and fuzzy techniques // Proceedings of the First International Scientific Conference Intelligent Information Technologies for Industry (ИТИ'16). 2016. Vol. 1. pp. 41 - 50.

13. Guskov, G., Afanasieva, T.,Yarushkina, N., ACL-scale as a tool for preprocessing of many-valued contexts // CEUR Workshop Proceedings. 2016

14. Guskov, G., Afanasieva, T., Yarushkina, N. The study of basic fuzzy series forecasting models /Uncertainty Modelling in Knowledge Engineering and Decision Making - Proceedings of the 12th International FLINS Conference, FLINS 2016

15. Guskov G., Namestnikov A., Yarushkina N. Approach to the search for similar software projects based on the UML ontology // Intelligent Information Technologies for Industry. Sochi. 2017. Vol. 2. pp. 3–10.

16. Guskov G., Yarushkina N., Moshkin V., Filippov A., Developing a Fuzzy Knowledge Base and Filling It with Knowledge Extracted from Various Documents // Artificial Intelligence and Soft Computing. Zakopane. 2018. Vol. 1. pp. 799-811.

в иных изданиях

17. Гуськов, Г.Ю. Интеллектуальный веб-сервис экспресс анализа экономического состояния предприятия / Г.Ю. Гуськов, И.Г. Перфильева, А.А. Романов, И.А. Тимина, Н.Г. Ярушкина // Четырнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2014 труды конференции. – Казань. – 2014. – Т.2. – С. 213-221

18. Гуськов, Г.Ю. Построение массива методов для прогнозирования временных рядов / М.В. Артюхов, Г.Ю. Гуськов, А.А. Романов, И.А. Тимина // Материалы VIII международной конференции Интегрированные модели и мягкие вычисления в искусственном интеллекте: труды конференции. – Т.3. – М.: Физматлит. – 2015. – С. 332 – 341.

19. Гуськов, Г.Ю. Программная система преобразования UML-диаграмм в онтологии на языке OWL / Г.Ю. Гуськов, А.М. Наместников // Пятнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2016: труды конференции. – 2016. – Т.3. – С. 270-278.

20. Гуськов Г.Ю., Правила трансформации проектных диаграмм UML в онтологическое представление / Г.Ю. Гуськов, А.М. Наместников, Н.Г. Ярушкина // Сборник: Нечеткие системы, мягкие вычисления и интеллектуальные технологии (НСМВИТ-2017) труды VII всероссийской научной-практической конференции. Санкт-Петербург, 2017. С. 58-64.

21. Гуськов, Г.Ю. Алгоритм и программа преобразования проектных диаграмм UML в предметную онтологию OWL / Г.Ю. Гуськов // Вузовская наука в современных условиях: Труды конференции. – Ульяновск: Изд-во УлГТУ. – 2017. –Т. 2. – С. 129-132.

22. Гуськов, Г.Ю. Онтология, разработанная на основе мета-схемы языка UML / Г.Ю. Гуськов, А.М. Наместников // Сборник: Нечеткие системы, мягкие вычисления и интеллектуальные технологии (НСМВИТ-2017), труды VII всероссийской научной-практической конференции. Санкт-Петербург, 2017. С. 19-25

23. Гуськов, Г.Ю. Разработка нечеткой базы знаний на основе анализа контекстов проблемной области / Г.Ю. Гуськов, А.А. Филиппов, В.С. Мошкин, Н.Г. Ярушкина // В сборнике: Нечеткие системы и мягкие вычисления. Промышленные применения, Ульяновск, 2017. С. 295-304

24. Guskov G., Namestnikov A. Ontological mapping for conceptual models of software system // Open Semantic Technologies for Intelligent Systems. Minsk. 2017. Vol. 1. pp. 111-117.

25. Гуськов, Г.Ю., Разработка и реализация архитектуры программного обеспечения для прогнозирования метрик ИТ проектов / Т.В. Афанасьева, Г.Ю. Гуськов Романов А.А. // Сборник: Информатика, моделирование, автоматизация проектирования IX Всероссийская школа-семинар аспирантов, студентов и молодых ученых : сборник научных трудов. 2017. С. 112-116.

26. Guskov G., Namestnikov A., Yarushkina N. System of integration a conceptual models with fuzzy and crisp OWL ontologies // In proceedings: INTERACTIVE SYSTEMS: Problems of Human-Computer Interaction collection of scientific papers. 2017. С. 190-199

27. Guskov G., Namestnikov A. Approach to determining the structural similarity of software projects // Open Semantic Technologies for Intelligent Systems. Minsk. 2018. Vol. 1. pp. 269-272.

Свидетельство о регистрации программы для ЭВМ

28. Гуськов Г.Ю., Ярушкина Н.Г., Романов А.А., Мошкина И.А. Интегрированная система нечетких методов прогнозирования временных рядов. Свидетельство о регистрации программы для ЭВМ RUS 2015617046 12.05.2015

Гуськов Глеб Юрьевич

Методы и средства формирования предметных тезаурусов в автоматизированном проектировании программно-аппаратных комплексов

Автореферат

Подписано в печать. _____ . Формат 60x80/16

Усл. печ. л. 1,17 Тираж 120 экз. Заказ

ИПК «Венец» УлГТУ, 432027, г. Ульяновск, Северный Венец, 32.