

**Институт динамики систем и теории управления
имени В.М. Матросова СО РАН**

На правах рукописи

Дородных Никита Олегович

**МОДЕЛИ И МЕТОДЫ АВТОМАТИЗИРОВАННОЙ РАЗРАБОТКИ БАЗ
ЗНАНИЙ НА ОСНОВЕ ТРАНСФОРМАЦИИ
КОНЦЕПТУАЛЬНЫХ МОДЕЛЕЙ**

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

Научный руководитель: к.т.н., в.н.с. **Юрин Александр Юрьевич**

**диссертация на соискание ученой степени
кандидата технических наук**

Иркутск – 2017

Общая характеристика работы

Актуальность темы. Знание является стратегическим ресурсом и актуальность разработки новых методов и средств управления знаниями, в том числе, их применения при решении практических слабоформализованных задач в различных предметных областях, остается высокой. При этом «оцифровка» знаний и их представление в виде концептуальных моделей, декларативных программ и кодов баз знаний обеспечивает их эффективное использование. База знаний (БЗ) является основным компонентом систем искусственного интеллекта и сдерживающим фактором их широкого применения, т.к. разработка БЗ является одним из самых сложных и трудоемких этапов при создании интеллектуальных систем – «узким местом» проектирования систем подобного вида. На данном этапе решаются задачи моделирования предметной области, получения, концептуализации и формализации знаний с их описанием на определенном языке представления знаний (ЯПЗ).

Одним из способов повышения эффективности процесса разработки БЗ является применение методов получения знаний из различных источников, в том числе, концептуальных моделей, под которыми понимаются модели, представленные множеством понятий и связей между ними, определяющих смысловую структуру рассматриваемой предметной области, вместе со свойствами и характеристиками, классификацией этих понятий, по типам, ситуациям, признакам в данной области и законов протекания процессов в ней¹. При этом особый интерес представляет использование моделей, построенных с использованием программных средств концептуального, когнитивного, онтологического моделирования и CASE-средств путем их трансформации в программные коды. Однако, существующие методы и системы автоматизированного создания БЗ на основе концептуальных моделей обладают рядом недостатков, в частности: сложностью описания самих моделей для генерации кода; высокими квалификационными требованиями к пользователю; отсутствием возможности совместной распределенной и одновременной работы пользователей; отсутствием или ограниченностью генерации программного кода БЗ на различных ЯПЗ (частичное преобразование, скелетные коды). Наличие данных недостатков определяет актуальность задачи создания новых моделей, методов и средств, обеспечивающих разработку БЗ, в том числе, на основе трансформации концептуальных моделей. В свою очередь, существование множества форматов концептуальных моделей требует создания средства, которое будет обладать

¹ Аверкин А.Н., Гаазе-Рапопорт М.Г., Поспелов Д.А. Толковый словарь по искусственному интеллекту. М.: Радио и связь, 1992. – 256 с.

свойством расширяемости в части создания дополнительных модулей в форме программных компонентов трансформации моделей.

Значительный вклад в разработку и исследование моделей, методов и средств создания интеллектуальных систем (включая онтологии, БЗ и программные средства для их проектирования и синтеза программных кодов) внесли ученые: Аверкин А.Н., Баадер Ф., Вагин В.Н., Ван Хармелен Ф., Варшавский П.Р., Гаврилова Т.А., Голенков В.В., Грау Б., Грибова В.В., Грубер Т., Гуарино Н., Джарратано Дж., Джексон П., Еремеев А.П., Загорулько Ю.А., Клещев А.С., Колесников А.В., Кудрявцев Д.В., Ленат Д., Люгер Г., Массель Л.В., МакГиннесс Д., Мотик Б., Норвиг П., Осипов Г.С., Осуга С., Патель-Шнайдер П., Попов Э.В., Поспелов Д.А., Райли Г., Рассел С., Рыбина Г.В., Саэки Ю., Сова Дж., Стааб С., Финн В.К., Фоминых И.Б., Хорошевский В.Ф., Хоррокс Я., Шрайбер Г., Штудер Р., Частиков А.П. и др. В области автоматизации создания программных систем и их компонентов, разработки трансляторов, а также подходов трансформации моделей и программ можно отметить работы исследователей: Ахо А., Гасевик Д., Горбунов-Посадов М.М., Гринфилд Дж., Джоолт Ф., Ершов А.П., Клеппе А., Кук С., Менс Т., Опарин Г.А., Сабельфельд В.К., Ульман Дж., Фаулер М., Франкель Д., Чарнецкий К. и др.

Целью диссертационного исследования является разработка моделей и методов автоматизации проектирования и синтеза программных кодов БЗ в форме декларативных программ на основе трансформации концептуальных моделей и их программная реализация в форме инструментального средства для повышения эффективности обработки знаний.

Задачи исследования:

1. Выполнить анализ существующих подходов, методов и программных средств автоматизации создания интеллектуальных систем и систем разработки БЗ, в том числе в форме онтологий, обеспечивающих трансформацию концептуальных моделей в программные коды БЗ.

2. Создать предметно-ориентированный декларативный язык для описания моделей трансформаций.

3. Разработать модели и методы автоматизации процесса проектирования и создания программных компонентов интеллектуальных систем, обеспечивающих синтез кода БЗ на основе трансформации концептуальных моделей.

4. Разработать инструментальное средство, реализующее предлагаемые модели и методы.

5. Разработать методику создания БЗ на основе анализ концептуальных моделей с использованием предлагаемого инструментального средства.

6. Апробировать предлагаемые модели, методы и инструментальное средство на примере разработки БЗ для решения задач в области технической диагностики и прогнозирования в нефтехимии и учебном процессе.

7. Оценить эффективность автоматизированного создания БЗ на основе трансформации концептуальных моделей по сравнению с классическим методом разработки.

Объектом исследования являются алгоритмическое и программное обеспечение создания программных средств для обработки знаний в вычислительных машинах, комплексах и компьютерных сетях.

Предметом исследования являются модели, методы и алгоритмы для проектирования и синтеза программных кодов БЗ на основе трансформации концептуальных моделей.

Методы исследования:

В работе использовались методы объектно-ориентированного программирования, трансформации моделей, построения трансляторов и предметно-ориентированных языков, а также методы и средства искусственного интеллекта и онтологического моделирования.

Научная новизна:

1. Разработан новый предметно-ориентированный декларативный язык описания моделей трансформаций (TMRL), включающий конструкции для описания не только преобразуемых структур и связей между ними, но и механизма взаимодействия с внешними программными компонентами трансформаций. Это позволяет абстрагироваться от конкретики специализированных языков трансформации моделей общего назначения и использовать созданные ранее компоненты.

2. Предложен метод автоматизации процесса создания программных компонентов интеллектуальных систем для проектирования БЗ и синтеза их кода на основе трансформации концептуальных моделей, отличием которого от известных является использование нового предметно-ориентированного декларативного языка описания моделей трансформаций в совокупности с интерактивным визуальным построением правил трансформации и оригинальной моделью типового программного компонента.

3. Впервые разработано инструментальное средство, реализующее предложенный метод и позволяющее интерактивно создавать компоненты трансформации концептуальных моделей, а также проектировать с их помощью БЗ.

4. Создана оригинальная методика автоматизированной разработки БЗ, отличием которой от известных является использование: концептуальных моделей в качестве исходных данных и специализированных программных

компонентов и языков (TMRL и RVML) в качестве инструментальных средств.

В целом в диссертации предложены новые модели, методы и средства разработки БЗ на основе трансформации концептуальных моделей, позволяющие значительно сократить сроки и стоимость разработки, а также снизить требования к квалификации разработчика в части знания языков программирования и модельных трансформаций за счет применения интерактивного инструментального средства для создания компонентов трансформаций.

Практическая значимость результатов:

Основные научные результаты по теме диссертации получены в рамках: проектов РФФИ 15-37-20655, 15-07-03088, 15-07-05641, 16-37-00122, соглашения № 8770 ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009-2013 гг.

Предложенные в рамках диссертационной работы модели, методы и веб-ориентированная программная система позволяют снизить трудозатраты и сократить сроки разработки программных компонентов создания БЗ интеллектуальных систем. Практическая значимость результатов определяется их использованием в учебном процессе ИрННТУ в рамках курсов «CASE-средства» и «Инструментальные средства информационных систем», а также при выполнении работ по хозяйственному договору с ОАО «ИркутскНИИхиммаш» по разработке проблемно-ориентированного редактора БЗ.

Достоверность результатов проведенных исследований подтверждается обоснованным использованием методов и технологий трансформации моделей, опубликованных в открытой печати, публикацией полученных результатов, работоспособностью разработанного инструментария, решением тестовых и прикладных задач. Теоретические аспекты развивают известные методы и алгоритмы.

Соответствие диссертации паспорту научной специальности:

В соответствии с паспортом специальности 05.13.11, диссертация охватывает решение задач создания программных средств различного назначения (в частности, систем проектирования и синтеза БЗ), включает исследование языков и систем программирования (построение предметно-ориентированного языка трансформации концептуальных моделей в код БЗ и поддерживающего его программной системы), а также моделей, методов, алгоритмов и программных инструментов для организации взаимодействия программ и программных систем. Отраженные в диссертационной работе положения соответствуют пунктам 1, 2 и 3 области исследования специальности 05.13.11:

1. Модели, методы и алгоритмы проектирования и анализа программ и программных систем, их эквивалентных преобразований, верификации и тестирования.

2. Языки программирования и системы программирования, семантика программ.

3. Модели, методы, алгоритмы, языки и программные инструменты для организации взаимодействия программ и программных систем.

Научное и народнохозяйственное значение диссертации заключается в повышении эффективности процессов обработки и передачи знаний (автоматизированная разработка БЗ) в вычислительных машинах, комплексах и компьютерных сетях.

Личный вклад автора:

Все выносимые на защиту научные положения получены соискателем лично. В основных научных работах по теме диссертации, опубликованных в соавторстве, лично соискателем разработаны: в [1, 2, 7, 9] – модели и методы автоматизации создания программных компонентов для автоматического создания БЗ на основе трансформации концептуальных моделей; [3, 4, 6, 8, 12-14] – алгоритмическое и программное обеспечение автоматизированного создания продукционных БЗ на основе концептуальных моделей; [5, 10, 11] – апробация разработанных моделей, методов и программных средств автоматизации создания БЗ путем трансформации концептуальных моделей.

Апробация работы:

Основные результаты диссертационной работы, её отдельные положения, а также результаты конкретных прикладных исследований и разработок обсуждались на научных семинарах ИДСТУ СО РАН, на международных, региональных научных и научно-практических конференциях: XII Международном форуме управления знаниями «International Forum on Knowledge Asset Dynamics. IFKAD-2017» (г. Санкт-Петербург, 2017 г.); II, III Российско-монгольских конференциях молодых ученых по математическому моделированию, вычислительно-информационным технологиям и управлению (Россия, г. Иркутск – Монголия, п. Ханх, 2013, 2015 гг.); Международной научно-практической конференции «Фундаментальная информатика, информационные технологии и системы управления: реалии и перспективы. ФИТМ-2014» (г. Красноярск, 2014 г.); XLIV Международной конференции и XIV Международной конференции молодых ученых «Информационные технологии в науке, образовании и управлении. IT + S&E`15» (г. Гурзуф, 2015 г.); Пятнадцатой национальной конференции по искусственному интеллекту с международным участием. КИИ-2016 (г. Смоленск, 2016 г.); VI, VII Международных конференциях «Системный анализ и информационные

технологии. САИТ» (г. Светлогорск, 2015, 2017 гг.); VI, VII Международной научно-технических конференциях «Открытые семантические технологии проектирования интеллектуальных систем. OSTIS» (Беларусь, г. Минск, 2016, 2017 гг.); IV Всероссийской конференции «Математическое моделирование и вычислительно-информационные технологии в междисциплинарных научных исследованиях» (г. Иркутск, 2014 г.); XV, XVI Всероссийских конференциях молодых ученых по математическому моделированию и информационным технологиям (г. Тюмень, г. Красноярск, 2014, 2015 гг.); XXI, XXII Байкальских Всероссийских конференциях с международным участием и Школах-семинарах научной молодежи «Информационные и математические технологии в науке и управлении» (г. Иркутск, 2016, 2017 гг.); Конференциях «Ляпуновские чтения» (г. Иркутск, 2014, 2015, 2016 гг.).

Публикации:

Результаты диссертационного исследования опубликованы в 41 печатной работе, в том числе 4 статьи в рецензируемых журналах, включенных в перечень ВАК, 1 статья в рецензируемом журнале, индексируемого в Web of Science и Scopus, 1 коллективная монография, а также получены 2 свидетельства об официальной регистрации программ для ЭВМ Федеральной службы по интеллектуальной собственности, патентам и товарным знакам.

Структура и объем диссертации. Диссертационная работа состоит из введения, четырех глав, заключения и списка литературы, включающего 269 наименований, и 8 приложений. Объем составляет 161 страницу основного текста, включая 86 рисунков, 9 таблиц, список сокращений и условных обозначений.

Краткое содержание работы

Во введении обосновывается актуальность диссертационной работы, формулируется цель и задачи исследования, определяется научная новизна и практическая значимость результатов.

В первой главе дан анализ современных направлений исследований и существующих программных средств повышения эффективности создания интеллектуальных систем и их БЗ. В частности, рассмотрено применение: систем концептуального, когнитивного и онтологического моделирования, а также CASE-средств; различных оболочек и специализированных сред разработки экспертных систем (ЭС); редакторов БЗ, а также методологий и средств проектирования БЗ с использованием специализированных графических нотаций; подходов и программных средств автоматизации разработки гибридных ЭС.

Также в главе, подробно рассмотрена проблематика трансформаций моделей, включая подходы и языки трансформации; приведен обзор работ,

направленных на решение задач автоматизированного создания БЗ на основе преобразования концептуальных моделей. Представлено детальное сравнение существующих решений, определены их основные недостатки (ограничения).

На основе аналитического обзора делаются выводы о перспективности использования подходов, основанных на порождающем программировании, в частности, модельно-управляемом подходе (Model Driven Development, MDD) и концептуальных моделей в качестве основных артефактов при разработке БЗ, также отмечается отсутствие единых принципов построения систем разработки БЗ на основе трансформации концептуальных моделей, наличие узкой специализации существующих решений и их направленность на высококвалифицированных программистов, что, в целом, обуславливает актуальность работы.

Во второй главе приводится описание разработанных моделей и методов автоматизации процесса создания программных компонентов (модулей) интеллектуальных систем, обеспечивающих генерацию кодов БЗ путем трансформации концептуальных моделей. Разработку программных компонентов предлагается осуществлять на основе компонентно-каркасного подхода, путем «клонирования» (копирования) типового шаблонного программного компонента для и его дальнейшей специализации. При этом типовой компонент может быть рассмотрен как «каркас» с «разъемами/слотами/гнездами»² (далее в тексте будет использовано понятие «слот»), специализация которого, в свою очередь, осуществляется путем интерпретации декларативной программы в форме модели трансформации, содержащей описание трансформаций и вызываемых блоков (модулей) (далее в тексте будет использовано понятие «блок») анализа и генерации.

Предлагается следующая модель программного компонента: $M_{TPC} = \langle M_T, A_{IN}, CG_{OUT}, I \rangle$, где M_T – модель трансформации; A_{IN} – анализатор (слот для модуля анализатора) входных моделей; CG_{OUT} – генератор (слот для модуля генератора) выходных моделей; I – интерфейс взаимодействия с внешними системами, обеспечивающий доступ к анализатору и генератору.

При этом $M_T = \langle MM_{IN}, MM_{OUT}, T \rangle$, где MM_{IN} – метамодель исходной (входной) концептуальной модели, $MM_{IN} \in \{MM_{CM}, MM_{PR}, MM_{ONT}\}$, где MM_{CM} – метамодель концептуальной модели, MM_{PR} – метамодель обобщенной модели продукций, MM_{ONT} – метамодель обобщенной модели онтологии; MM_{OUT} – метамодель целевой модели представления знаний (БЗ),

² Горбунов-Посадов М.М. Расширяемые программы. М.: Полиптих, 1999. – 321 с.

$MM_{OUT} \in \{MM_{PR}MM_{ONT}, MM_{CLIPS}, MM_{OWL}\}$, где MM_{CLIPS} – метамодель языка CLIPS; MM_{OWL} – метамодель языка OWL; T – оператор преобразования моделей, $T: CM \rightarrow KB$, где CM – исходная концептуальная модель; KB – целевая БЗ. Определено три типа оператора преобразования: $T = \langle T_{CM-KB}, T_{CM-UM}, T_{UM-KB} \rangle$, где T_{CM-KB} – оператор преобразования исходной концептуальной модели в код БЗ на целевом ЯПЗ; T_{CM-UM} – оператор преобразования исходной концептуальной модели в модель онтологии или унифицированный формат представления знаний в форме продукций; T_{UM-KB} – оператор преобразования модели онтологии или продукций в код БЗ на целевом ЯПЗ (CLIPS или OWL, соответственно).

В основе модели трансформации M_T и рассмотренного далее метода лежит концепция модельных трансформаций³.

В процессе проектирования и синтеза конкретного экземпляра программного компонента, необходимо сформировать модель трансформации M_T , которая определяет правила преобразования (программу трансформации) исходных концептуальных моделей в целевые БЗ, а также определить (вставить в гнезда) соответствующие блоки анализатора и генератора, из которых произойдет сборка конкретного программного компонента.

Для представления и хранения модели трансформации M_T разработан предметно-ориентированный язык (Domain Specific Language, DSL) – Transformation Model Representation Language (TMRL). Грамматика TMRL принадлежат к классу контекстно-свободных грамматик (КС-грамматик – LL(1)). Конструкции TMRL позволяют в декларативном виде описывать элементы модели трансформации, в частности, правила соответствия элементов метамodelей, а также механизм взаимодействия с внешними программными компонентами анализа и генерации.

TMRL является специализированным языком, ориентированным только на поддержку представления и хранения модели трансформации M_T . Вследствие чего TMRL не имеет прямых аналогов, с которыми можно было бы осуществить полное и корректное сравнение. Наиболее близкими к TMRL являются универсальные языки модельных трансформаций: ATL (ATLAS Transformation Language), языки стандарта Query/View/Transformation (QVT-R, QVT-C, QVT-O), Epsilon и др. Основным отличием TMRL от данных языков является его простота, достигаемая за счет ограниченного набора конструкций

³ Czarniecki K., Helsen S. Feature-based survey of model transformation approaches // IBM Systems Journal. 2006, vol. 45, no. 3, pp. 621-645.

(14 конструкций-терминалов), а также возможность описания механизма взаимодействия с внешними программными компонентами (блоками анализа и генерации).

Структура программы на TMRL состоит из 3 основных блоков, рассмотрим ее на примере программы, описывающей преобразование диаграммы классов UML в модель онтологии (элементы языка выделены жирным шрифтом):

Блок 1: Описание элементов и отношений исходной метамодели:

```
Source Meta-Model UML-diagram-class {
  Elements [
    Model,
    Class attributes (xmi.id, name),
    ... ]
  Relationships [
    Model is associated with Namespace.ownedElement,
    Namespace.ownedElement is associated with Class,
    DataType(xmi.id) is Attribute(type),
    ... ] }
```

Данный блок содержит описание диаграммы классов UML: «UML-diagram-class», включая элементы модели (раздел «Elements»). В данном примере – это элементы «Model» и «Class», при этом элемент «Class» обладает свойствами «xmi.id» и «name». Помимо описания элементов, исходная метамодель содержит описание связей между элементами метамодели (раздел «Relationships»), в том числе по идентификаторам, например связь атрибута с типом данных («DataType(xmi.id) **is** Attribute(type)»).

Блок 2: Описание элементов и отношений целевой метамодели:

```
Target Meta-Model Ontology {
  Elements [
    ExtendedOntology attributes (id, name),
    Class attributes (id, name),
    ... ]
  Relationships [
    Ontology is associated with Class,
    ... ] }
```

Блок содержит описание модели онтологии: «Ontology». Структура блока аналогична структуре блока исходной метамодели.

Блок 3: Описание правил преобразования моделей:

```
Transformation UML-diagram-class to Ontology {
  Rule Model to Ontology priority 1 [
    Ontology(name) is Model or ModelElement.name
    Ontology(id) is Model(xmi.id)
  ]
  Rule (Class, ModelElement.name) to Class priority 2 [
    Class(name) is or Class(name) or ModelElement.name
```

```

Class(id) is Class(xmi.id)
] ...
Rule (AssociationEnd, MultiplicityRange) to Lhs priority 7 [
Lhs is AssociationEnd
Lhs(operator) is "AND" [
    if (MultiplicityRange(lower) is "1") and
    (MultiplicityRange(upper) is "-1")
    ... ]
... ]...}

```

Данный блок содержит описание правил преобразования элементов исходной («UML-diagram-class») метамодели в целевую («Ontology»).

Метод создания программных компонентов, использующий предложенную модель типового программного компонента и предметно-ориентированный декларативный язык состоит из 4 основных этапов:



Рисунок 1. Методика разработки БЗ на основе трансформации концептуальных моделей

типového программного компонента на основе сформированной модели трансформации M_T и выбранных блоков анализатора и генератора.

Особенностями метода, определяющими его новизну, является использование нового предметно-ориентированного декларативного языка описания моделей трансформаций (TMRL) в совокупности с интерактивным визуальным построением правил трансформации и оригинальной моделью

Этап 1: создание метамодели исходной концептуальной модели: на основе анализа XML-схемы или на основе анализа исходной концептуальной модели.

Этап 2: выбор целевой метамодели БЗ: CLIPS, OWL или метамодель внутреннего представления знаний системы в форме онтологии или продукций, определяющий выбор блоков генераторов при создании (сборке) программного компонента на этапе 4.

Этап 3: создание модели трансформации M_T .

Этап 4: автоматическое создание программного компонента путем специализации

типового программного компонента.

Для унифицированного промежуточного представления и хранения, а также изменения знаний, полученных из концептуальных моделей, предлагается использовать модель онтологии и продукций. Данные модели позволяют абстрагироваться от особенностей описания знаний в различных ЯПЗ, используемых при реализации БЗ (например, CLIPS, JESS, Drools, RuleML, SWRL, OWL, RDF и др.), и хранить знания в собственном независимом формате.

Методика автоматизированной разработки БЗ на основе трансформации концептуальных моделей с использованием специализированных программных компонентов и языков (TMRL и RVML) в качестве инструментальных средств, может быть представлена в виде последовательности действий представленных на рисунке 1.

В третьей главе описывается веб-ориентированная система программирования БЗ – Knowledge Base Development System (KBDS), реализующая предлагаемые модели и методы. Основной особенностью KBDS (Рис. 2) является расширяемость в части создания новых программных компонентов, обеспечивающих импорт и анализ различных форматов концептуальных моделей и генерацию кода БЗ на разных ЯПЗ.

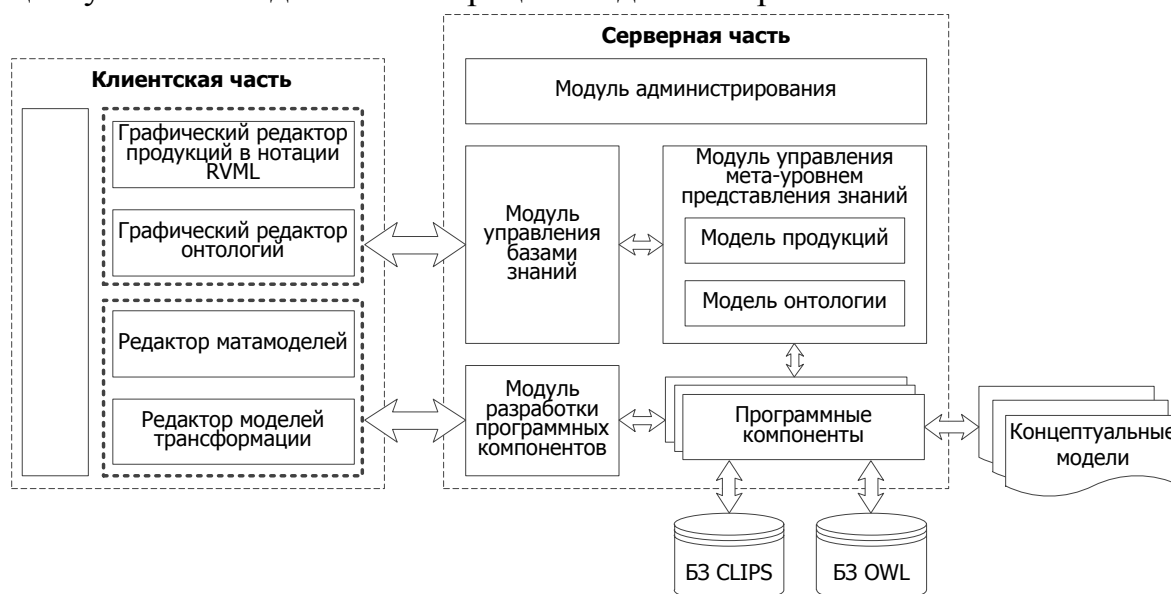


Рисунок 2. Архитектура KBDS

Пример экранной формы (веб-страница) редактора моделей трансформаций представлен на рисунке 3. На форме изображены понятия исходной и целевой метамодели и их соответствия.

KBDS реализует программный интерфейс для взаимодействия со сторонними программными средствами. Для апробации взаимодействия использовалась система программирования БЗ – Personal Knowledge Base Designer.

Редактор трансформации

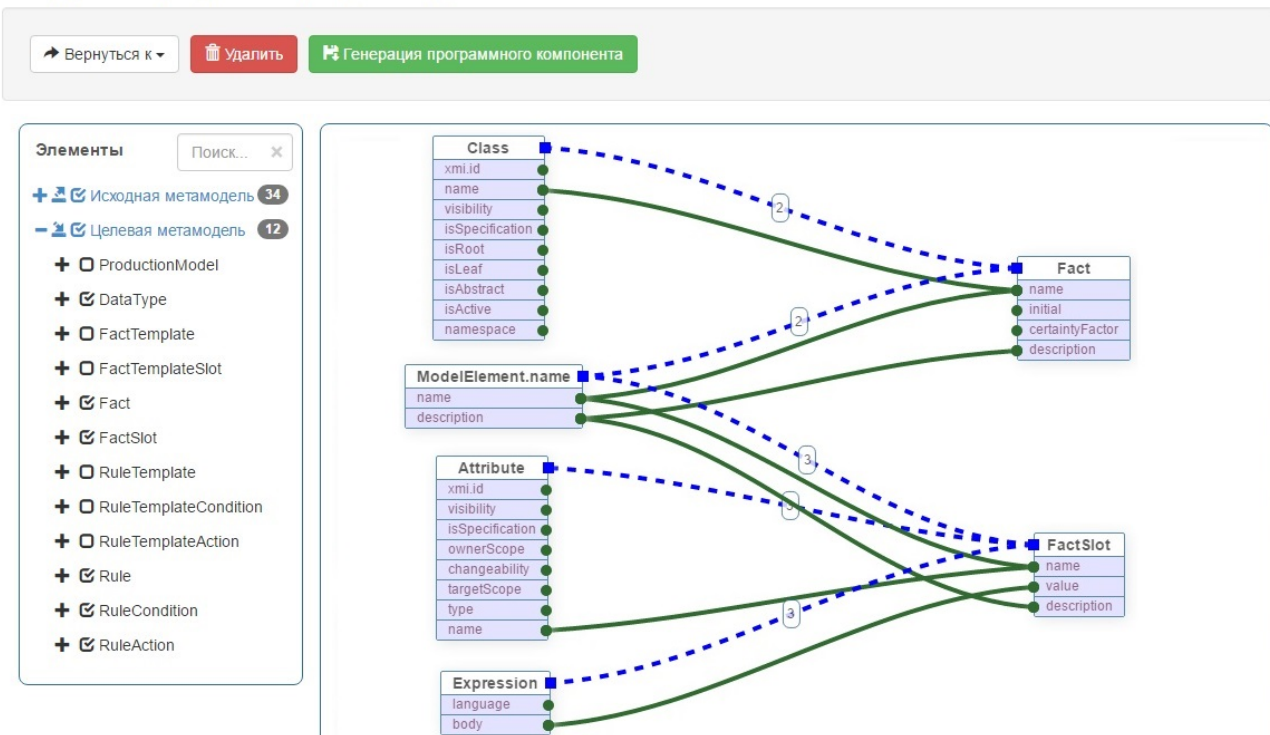


Рисунок 3. Пример экранной формы KBDS: редактор моделей трансформаций (пример фрагмента модели трансформации диаграмм классов UML в модель продукций)

В четвертой главе описывается апробация разработанных моделей, методов и средств, включающая:

- разработку программных компонентов для анализа концептуальных моделей в форме:
 - диаграмм классов UML CASE-средства IBM Rational Rose Enterprise;
 - концепт-карт IHMC SmartTools стандарта XTM;
 - проверку адекватности предлагаемых моделей и метода на примере разработки продукционной БЗ в формате CLIPS для прогнозирования развития деградационных процессов аппаратов в нефтехимии, на основе созданных программных компонентов. БЗ включает 14 шаблонов фактов, 12 шаблонов правил, 4 начальных факта и 20 конкретных правил. Результаты автоматизированного создания БЗ сравнивались с созданной БЗ в ручном режиме;
 - оценку эффективности предлагаемого метода, по временному критерию.
- Оценка эффективности проводилась на базе ИрНИТУ с участием 60 студентов Института кибернетики им. Е.И. Попова. Было проанализировано 20 тестовых вариантов (примеров) разработки БЗ статических ЭС. Результаты оценки временных затрат представлены на рисунке 4, где:

- A1: разработка БЗ с использованием предлагаемого метода и средства;
- A2: разработка БЗ с использованием CASE-средства (IBM Rational Rose) и средства создания БЗ CLIPS – ClipsWin;
- A3: разработка БЗ с использованием средства разработки БЗ CLIPS – ClipsWin, но без использования визуального UML-моделирования.

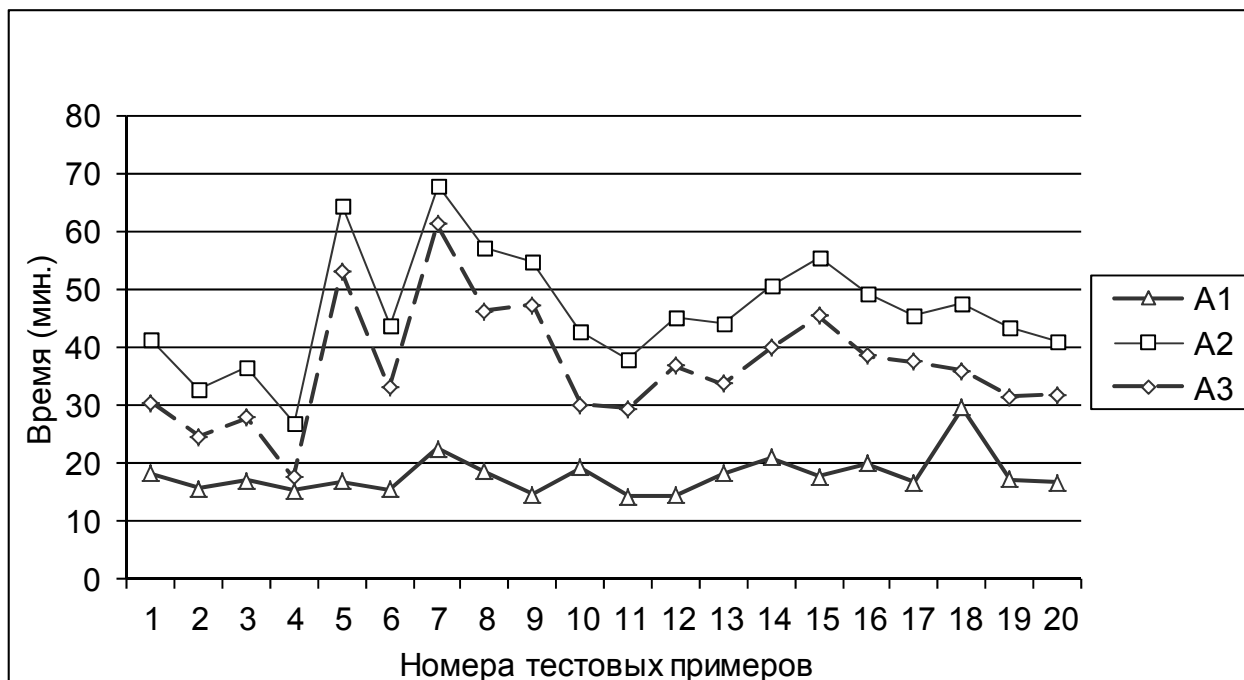


Рисунок 4. Результаты оценки временных затрат

Анализ результатов показал, что эффективность разработки БЗ может быть повышена, в среднем на 60.3% по сравнению с A2 и на 48.2% по сравнению с A3 за счет автоматической кодогенерации на основе визуальных моделей, что в свою очередь позволяет:

- использовать результаты этапов концептуализации и формализации в форме диаграмм классов UML, рассматривая последние не как статические графические артефакты, а как основу для формирования программного кода в соответствии с идеологией модельно-управляемого подхода;
- снизить риск ошибок проектирования, за счет возможности быстрого прототипирования БЗ и получения их программного кода;
- исключить ошибки программирования, за счет автоматического отображения элементов концептуальной модели в CLIPS.

В заключении сформулированы основные научные результаты диссертационной работы.

Результаты, выносимые на защиту

1. Предложены новые модели и методы автоматизации процесса проектирования и создания программных компонентов интеллектуальных

систем, обеспечивающих синтез кода БЗ на основе трансформации концептуальных моделей.

2. Разработан новый предметно-ориентированный декларативный язык описания моделей трансформаций – Transformation Model Representation Language (TMRL).

3. Разработано инструментальное средство – система программирования БЗ, реализующая предложенные модели, методы и язык.

4. Создана оригинальная методика автоматизированной разработки БЗ на основе анализа концептуальных моделей, основанная на применении предлагаемых методов и средств.

5. Произведена апробация моделей методов и средств на примере создания БЗ ЭС для прогнозирования технического состояния нефтехимического оборудования и в учебном процессе.

6. Получены результаты оценки эффективности метода создания БЗ на основе концептуальных моделей.

Список основных публикаций по теме диссертации

Статьи в журналах из перечня ВАК

1. Бычков И.В., Дородных Н.О., Юрин А.Ю. Подход к разработке программных компонентов для формирования баз знаний на основе концептуальных моделей // Вычислительные технологии. – 2016. – Т.21. – №4. – С. 16-36.

2. Грищенко М.А., Дородных Н.О., Николайчук О.А., Юрин А.Ю. Применение модельно-управляемого подхода для создания продукционных экспертных систем и баз знаний // Искусственный интеллект и принятие решений. – 2016. – №2. – С. 16-29.

3. Дородных Н.О., Юрин А.Ю. Использование диаграмм классов UML для формирования продукционных баз знаний // Программная инженерия. – 2015. – №4. – С. 3-9.

4. Дородных Н.О., Юрин А.Ю. Web-сервис для автоматизированного формирования продукционных баз знаний на основе концептуальных моделей // Программные продукты и системы. – 2014. – №4. – С. 103-107.

5. Berman A.F., Grishchenko M.A., Dorodnykh N.O., Nikolaychuk O.A., Yurin A.Y. A model-driven approach and a tool to support creation of rule-based expert systems for industrial safety expertise // In Processing of the International Forum on Knowledge Asset Dynamics. IFKAD-2017. 2017, pp. 2034-2050. (Web of Science & Scopus).

Монографии

6. Грищенко М.А., Дородных Н.О., Юрин А.Ю. Модельно-управляемый

подход. Алгоритмическое и программное обеспечение для создания продукционных баз знаний и экспертных систем. LAP, 2015. – 129 с.

Другие основные работы по теме диссертации

7. Дородных Н.О., Коршунов С.А., Юрин А.Ю. Концепция подхода к созданию программных компонентов генерации баз знаний на основе трансформации концептуальных моделей // Информационные и математические технологии в науке и управлении. – 2016. – №2. – С. 111-120.

8. Дородных Н.О., Юрин А.Ю. Формирование баз знаний продукционного типа на основе UML-моделей // Информатика и кибернетика. – Д.: ДонНТУ, – 2016. – № 3(5). – С.44-50.

9. Дородных Н.О., Юрин А.Ю. Разработка программных компонентов для формирования баз знаний на основе трансформации концептуальных моделей // Пятнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2016. Труды конференции. В 3-х томах. Т. 1. Смоленск: Универсум, 2016. – С. 33-40.

10. Дородных Н.О. Web-based software for automating development of knowledge bases on the basis of transformation of conceptual models // Материалы VII Международной научно-технической конференции OSTIS-2017. Минск: БГУИР. 2017. С. 145-150.

11. Дородных Н.О. Программная система автоматизации разработки web-сервисов для генерации баз знаний // Труды конференции «Системный анализ и информационные технологии. САИТ-2017». – М: ФИЦ ИУ РАН. – С. 222-229.

12. Дородных Н.О., Николайчук О.А., Юрин А.Ю. Автоматизация создания продукционных баз знаний на основе концептуальных моделей // Труды конференции. В 2-х т. «Системный анализ и информационные технологии. САИТ-2015». – М: ИСА РАН. – Т1. – С. 281-288.

Свидетельства о государственной регистрации программ для ЭВМ

13. Дородных Н.О. Web-ориентированный редактор моделей трансформаций (Web Transformation Model Editor): Свидетельство о государственной регистрации программ для ЭВМ № 2017618430 от 01.08.2017. М.: Федеральная служба по интеллектуальной собственности, патентам и товарным знакам, 2017.

14. Дородных Н.О. RVML editor (Web Knowledge Base Designer): Свидетельство о государственной регистрации программ для ЭВМ № 2017618446 от 01.08.2017. М.: Федеральная служба по интеллектуальной собственности, патентам и товарным знакам, 2017.