

## МЕТОД РАЗРАБОТКИ ИМИТАЦИОННОЙ МОДЕЛИ НА ОСНОВЕ ОНТОЛОГИЙ

*Е.А. Побелустик, С.А. Альховик*

Аннотация. Проведено исследование возможности применения методов и средств онтологического анализа для создания предметно-ориентированных языков моделирования. Предложен метод разработки имитационной модели, отличающийся комплексным применением методов и средств онтологического анализа на этапах построения содержательного описания объекта моделирования, концептуальной и формальной моделей, и описания имитационной модели.

Ключевые слова: онтология, методология построения онтологий, онтологический анализ, имитационная модель.

### 1. ВВЕДЕНИЕ

Системы имитационного моделирования являются одним из важнейших универсальных средств исследования больших (сложных) систем, функционирующих в реальном времени. Имитационная модель сложной системы также является сложной и требует больших затрат времени исследователя для ее построения. Она также требует больших затрат машинного времени в сеансе имитации, что обуславливает использование многопроцессорных машин или компьютерных сетей. Предлагается использовать онтологии предметных областей для повышения эффективности работы с имитационными моделями.

Онтология – это описание типов сущностей, существующих в предметной области, их свойств и отношений. Каждая предметная область может быть описана с помощью онтологий. Однако любая предметная область имеет свою собственную лексику и требует специализированных средств грамматического разбора.

Как результат, несмотря на изобилие программных продуктов общего назначения, универсальный подход к извлечению знаний о предметной области себя не оправдывает. Даже если цель состоит в том, чтобы реализовать извлечение знаний в рамках крупной коммерческой или технической предметной области, организации придется адаптировать, по крайней мере, часть выбранного решения извлечения знаний, чтобы гарантировать, что оно отвечает ее особым целям и соответствует ее специфическим требованиям.

Онтологии предлагают массу возможностей усовершенствовать извлечение информации в рамках предметной области, но часто несовместимы с конкретной технологией построения имитационной модели. Урегулировать эту несовместимость трудно, поскольку взаимодействие между специалистами, которые разбираются в построении имитационной модели, и теми, кто должен будет использовать механизмы построения имитационной модели, крайне затруднено. Специалисты разрабатывают онтологию, не понимая потребности своих пользователей. В то же время, конечные пользователи пытаются применять существующие онтологии, но далеко не так эффективно.

Более разумная альтернатива — найти способ максимально эффективного использования уже имеющихся инструментов извлечения информации и уже применяемых

онтологий. Зачастую эти инструменты обладают полезными возможностями, которые компания не в состоянии в полной мере реализовать, если не разработает онтологию, адаптированную к своим собственным нуждам.

## 2. МЕТОД РАЗРАБОТКИ ИМИТАЦИОННОЙ МОДЕЛИ НА ОСНОВЕ ОНТОЛОГИЙ

Для задачи в общем случае разработан алгоритм получения решения поставленной проблемы, который основан на технологических этапах создания имитационных моделей и независимо от способа проектирования системы и назначения моделирования состоит из следующих шагов:

1. Составить описание системы – определение структуры системы, выделение протекающих в ней процессов, определение списка задач, которые будут решаться с помощью ИМ, формулировка цели моделирования (от цели зависит то, какие процессы в реальной системе следует выделить и отразить в модели, а от каких процессов абстрагироваться). На вход этого процесса поступает документация, устное описание реальной системы. Результатом является подробная документация с описанием системы и всех ее компонентов, цели и требования моделирования.

2. Построение онтологии некоторой предметной области по составленному описанию системы, т.е. составляется концептуальная модель системы – переход от реальной системы к логической схеме ее функционирования, определение основных элементов и процессов, которые будут моделироваться. Исходными данными для процесса являются цели и требования моделирования, документация с описанием системы. Результат – концептуальная модель в виде онтологии.

3. Разработанная онтология экспортируется в формат на основе *XML (RDF, OWL и XML Schema)*, допускающий ее дальнейшее использование и преобразование, т.е. выбор языка формализации – на основе концептуальной модели происходит выбор средств и метода имитационного моделирования, с помощью которых будет разработана ИМ системы с учетом поставленных целей и требований.

4. Для конкретных извлеченных данных разрабатываются шаблоны на языке *XSLT*.

Составляется формальное описание системы – с учетом выбранных средств и метода имитационного моделирования происходит реализация описания системы в терминах математических понятий, составляются алгоритмы элементов и процессов системы. Исходными данными для этапа является концептуальная модель. Результат – формальное описание системы.

5. После применения разработанных шаблонов к данным получается новый *XML*-документ, содержащий требуемые данные в структурированном виде согласно правилам структуризации.

Для простых ИМ этот этап зачастую объединяется с этапом составления формального описания, а для сложных – на нем решаются вопросы: синхронизации компонент модели, задания начальных условий, обработки результатов и сбора статистики. Результатом выполнения этапа является документация с описанием ИМ.

6. Запрограммировать ИМ – этот этап включает программирование, отладку и тестирование ИМ. Результатом является готовый модуль ИМ, который может быть использован для исследования, а также техническая документация.

В ПТКИ *BelSim 2#* генерирование структуры ИМ происходит автоматически на основе *XML*-документа.

7. Спланировать и провести натурные эксперименты – подготавливаются начальные данные для ИМ, проводится их статистический анализ. Этот этап может проводиться параллельно с составлением описания и программированием ИМ.

8.Верифицировать ИМ – доказательство утверждений соответствия алгоритма функционирования ИМ замыслу моделирования путем формальных и неформальных исследований [1, с. 126].

9.Проверить адекватность ИМ – на основе различных математических критериев сравниваются результаты, полученные при натуральных экспериментах, с результатами, полученными на основе ИМ для одинаковых начальных условий. На основе проверки адекватности ИМ делается вывод о точности получаемых результатов и о дальнейшем использовании ИМ для исследований.

10.Откалибровать ИМ – уменьшение неточности результатов ИМ, если модель оказывается неадекватной моделируемой системе. На этом этапе может происходить корректировка параметров, добавление дополнительных элементов либо их замена на более точные. Если предпринятые меры являются неуспешными, то происходит откат на более ранние этапы с целью доработки описания системы либо ее концептуальной модели.

11.Исследовать свойства ИМ – определяются основные свойства и возможности ИМ: длительность переходного процесса; погрешность имитации, обусловленная датчиками псевдослучайных чисел; оценка устойчивости результатов моделирования, исследование чувствительности ИМ [1, с. 134].

12.Спланировать и провести имитационные эксперименты – определяются изменяемые и регистрируемые параметры (факторы) ИМ, определяются начальные данные и проводится компьютерный эксперимент. Компьютерный эксперимент – это выполнение модели при различных значениях ее параметров (факторов) и наблюдение ее поведения с регистрацией характеристик поведения. Результат – массив значений входных и выходных факторов ИМ.

13.Интерпретировать результаты – анализируются результаты имитационных экспериментов, проводится их статистический анализ, строятся графики и таблицы.

14.Разработать анимированное представление ИМ – для более удобного указания исходных данных, а также для наглядного отображения процесса моделирования и результатов может быть построено специальное анимированное представление ИМ.

15.Разработать пользовательский интерфейс ИМ – для более простого использования ИМ, для нее может быть разработан специальный интерфейс позволяющий упростить ввод данных и анализа результатов. Этот интерфейс может включать в себя анимированное представление ИМ, и использован как конечный инструмент для поддержки принятия решений.

16.Поддержка принятия управленческого решения – на основе анализа результатов имитационного моделирования лицом принимающим решение происходит принятие управленческого решения.

Рассмотрим более подробно предложенный метод разработки имитационной модели на основе онтологий.

Практическая разработка онтологии включает:

- 1) определение классов в онтологии;
- 2) расположение классов в таксономическую иерархию;
- 3) определение слотов и описание их допустимых значений;
- 4) заполнение значений слотов экземпляров.

После этого можно создать базу знаний, определив отдельные экземпляры этих классов, введя в определенный слот значение и дополнительные ограничения для слота.

Выделим некоторые фундаментальные правила разработки онтологии. Они выглядят довольно категоричными, но во многих случаях помогут принять верные проектные решения.

– Не существует единственно правильного способа моделирования предметной области - всегда существуют жизнеспособные альтернативы.

– Лучшее решение почти всегда зависит от предполагаемого приложения и ожидаемых расширений.

– Разработка онтологии - это обязательно итеративный процесс.

Понятия в онтологии должны быть близки к объектам (физическим или логическим) и отношениям в интересующей предметной области. Скорее всего, это существительные (объекты) или глаголы (отношения) в предложениях, которые описывают предметную область.

После того как определена начальная версия онтологии, можно оценить и отладить ее, используя ее в каких-то приложениях и/или обсудив ее с экспертами предметной области. В результате начальную онтологию скорее всего нужно будет пересмотреть. И этот процесс итеративного проектирования будет продолжаться в течение всего жизненного цикла онтологии.

Для формального описания онтологий как правило используют *OWL (Web Ontology Language)*, в аббревиатуре буквы намеренно переставлены местами, чтобы получилось английское слово "сова") - это логический язык. Применительно к *OWL* под онтологиями понимаются иерархические структуры понятий (классов), связанных отношениями (свойствами). *OWL* предоставляет средства для логического описания семантики (т.е. смысла) понятий, благодаря чему последние могут согласованно использоваться как людьми, так и приложениями в различных информационных системах (*Web*-сайтах, базах данных, экспертных системах, системах поддержки принятия решений и т.д.) [2].

После построения онтологии некоторой предметной области необходимо разработанную онтологию преобразовать в структуру *XML*-описания для программы разработки имитационных моделей.

Рассмотрим подробнее данный этап.

Для обработки *XML* - документов можно применить различные языки программирования, например, *Java* с использованием низкоуровневых *API*, таких как *SAX* или *DOM*, но наиболее удобным представляется язык *XSLT (eXtensible Stylesheet Language – Transformation)*, который получил статус рекомендации W3C 16 ноября 1999г.) – расширяемый язык таблиц стилей для трансформации. Одним из преимуществ этого языка является его схожесть с продукционными правилами и относительная схожесть с инструкциями на естественном языке, а главное этот язык позволяет трансформировать структуру *XML* документов в другое представление и/или формат. Общая схема трансформации документа представлена на *рисунке 1*. Для того чтобы трансформировать документ необходимо правила трансформации *XSLT* применить к документу. Трансформация выполняется с помощью *XSLT*-процессора. В результате трансформации *XSLT*-процессор генерирует необходимое представление исходного документа (сериализация документа) [3,с.547].

После применения разработанных шаблонов к данным получается новый *xml*-документ, содержащий требуемые данные в структурированном виде согласно разработанным правилам структуризации.

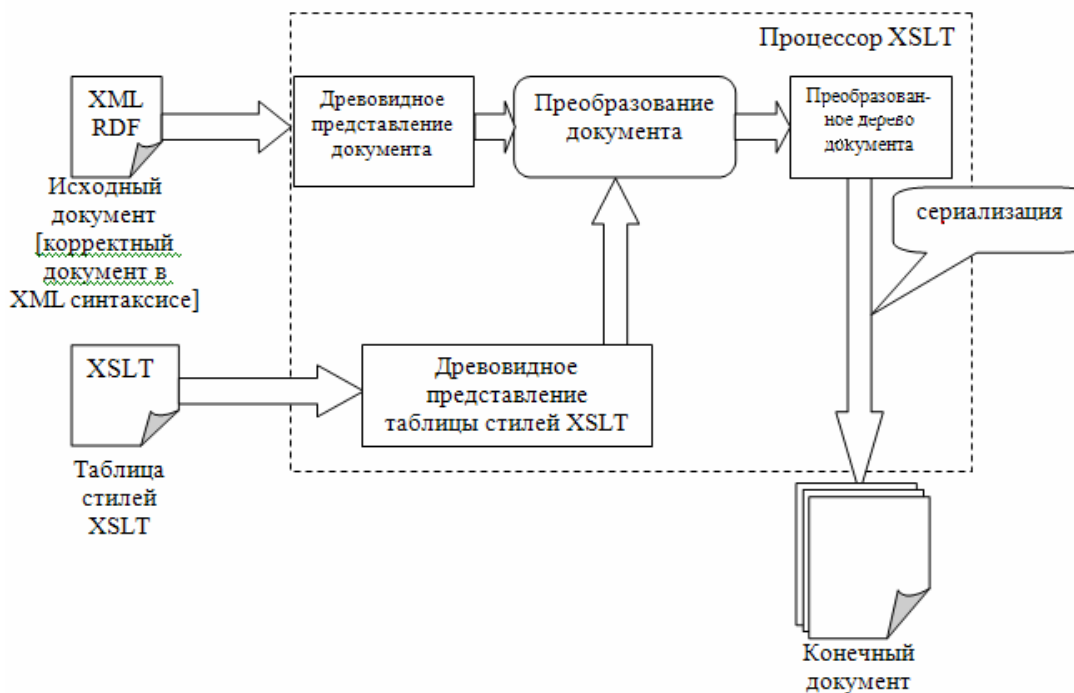


Рис.1 – Общая схема преобразования документа с использованием XSLT

### 3. ЗАКЛЮЧЕНИЕ

С учетом сложности процесса исследования и построения имитационных моделей сложных систем, использование онтологического подхода позволит сократить время и трудоемкость разработки формальной модели предметной области и языка моделирования, а также последующих моделей на его основе.

#### Литература

1. **Максимей, И.В.** Имитационное моделирование на ЭВМ / И.В. Максимей. – М.: Радио и связь, 1988. – 232 с.: ил. – ISBN 5-256-00001-2.
2. **Noy, F.** Ontology Development 101: A Guide to Creating Your First Ontology / Natalya F. Noy, Deborah L. McGuinness//Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.
3. **Кэй, М.** XSLT. Справочник программиста / М. Кэй – Пер. с англ. – СПб.: Символ-Плюс, 2002. – 1016 с.

#### **Побелустик Елена Александровна**

Выпускник магистратуры кафедры Автоматизированные системы управления

Белорусско-Российский университет, г. Могилев

Тел.: +375 (222) 245608

E-mail: [orkhidey@yandex.ru](mailto:orkhidey@yandex.ru)

#### **Альховик Сергей Александрович**

Доцент кафедры Автоматизированные системы управления

Белорусско-Российский университет, г. Могилев

Тел.: +375(222) 25-24-47

E-mail: [asa@tut.by](mailto:asa@tut.by)