

OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.942

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ПОСТРОЕНИЮ СИСТЕМЫ ИМИТАЦИИ, НАСТРАИВАЕМОЙ НА КОНКРЕТНУЮ ПРЕДМЕТНУЮ ОБЛАСТЬ

Замятина Е.Б., Михеев Р.А.

*Пермский государственный национальный исследовательский университет
г. Пермь, Россия*

e_zamyatina@mail.ru

mikheev@prognoz.ru

В работе рассматривается модельно-ориентированный подход к построению системы имитации, настроенной на конкретную предметную область. Модельно-ориентированный подход базируется на онтологиях, определенных в системе имитации TriadNS. Система имитации TriadNS предназначена для проектирования и моделирования компьютерных сетей.

Ключевые слова: имитационное моделирование, онтология, адаптируемость, компьютерные сети.

Введение

Имитационное моделирование является широко применяемым, а иногда и единственным, методом анализа сложных динамических систем, каковыми являются, например, компьютерные сети. Однако для того, чтобы применение методов имитационного моделирования было эффективным, необходимо привлекать к исследованиям специалистов из разных областей знаний. Так, например, при разработке алгоритмов маршрутизации целесообразно применять теорию графов (нахождение кратчайшего расстояния, определение характеристик графа, вершинами которого являются вычислительные узлы компьютерной сети). Исследование трафиков обычно выполняется с применением теории массового обслуживания, для исследования параллельных и распределенных программных систем используют теорию сетей Петри. Поэтому важно, чтобы при разработке имитационной модели исследователь смог оперировать привычными для него терминами, понятиями. В этом случае целесообразно применять онтологический подход, поскольку онтологии представляют собой концепты (понятия предметной области) и связи между ними. В статье рассматривается и другой подход, который основан на использовании языкового инструментария. Кроме того, для тщательного исследования проблемы, ситуации, сложной динамической системы целесообразно разрабатывать различные их модели и преобразовывать (трансформировать) одну модель в

другую. Таким образом, возникает возможность строить многомодельные (полимодельные) комплексы [Соколов, 2005]. Полимодельные комплексы способствуют улучшению качества имитационного исследования (квалиметрия имитационной модели).

1. Применение онтологий в имитационном моделировании

Известно, что онтология – это описание типов сущностей предметной области, их свойств и отношений. Каждая предметная область (некая часть реального мира) может быть описана с помощью онтологий. Онтологии создаются и используются во множестве областей знаний, в том числе, известны примеры их успешного применения в имитационном моделировании. Однако создание онтологий для моделирования является достаточно сложной задачей, поскольку этот метод используют для исследования самых разнообразных систем, относящихся к различным предметным областям (химическим, физическим, транспортным и т.д.). Кроме того, методы имитационного моделирования основаны на математических, вероятностных и статистических расчетах, и, таким образом, онтологии для этих областей должны служить основой для всех остальных. Онтологии используют на различных этапах имитационного моделирования, начиная с этапа сбора информации о моделируемой системе и заканчивая этапом валидации модели [Sargent, 2004]. Примерами использования онтологий моделирования могут служить управляемые онтологиями среды

моделирования, а также подходы к объединению различных федератов, разрабатываемые для High Level Architecture (HLA) [Rathman, 2005]. Подход, разрабатываемый для HLA, использует онтологии для описания требований, которым должны удовлетворять интерфейсы федератов для успешного взаимодействия в федерации, а так же для разработки этих требований с учетом знаний о моделируемой предметной области. В работе [Liang, 2006] представлена онтология портов, рассматриваемая как средство автоматизации построения моделей из компонентов. Порты описывают интерфейс, определяющий границы компонентов или подсистем в конфигурации системы. Система представлена как конфигурация подсистем или компонентов, соединенных друг с другом через четко определенные интерфейсы. Онтологии успешно применяются и в других работах по имитационному моделированию [Benjamin, 2005], [Benjamin, 2006]. Итак, перед авторами стоит задача выполнить настройку системы имитации на конкретную предметную область с целью проведения качественных исследований. Рассмотрим различные подходы к решению этой задачи.

2. Настройка на конкретную предметную область

Для настройки системы имитации можно воспользоваться либо программными средствами, которые предоставляют языковые инструментарии (или DSM-платформы), предназначенные для создания предметно-ориентированных языков (DSL, Domain Specific Language), либо программными средствами на основе онтологий. Рассмотрим первый способ.

2.1. Использование языкового инструментария для построения DSL

Существуют языковые инструментарии, которые позволяют строить DSL (метамодели), с использованием которых создаются модели соответствующих предметных областей (MetaEdit+, DSL Tools и др.). В Пермском университете был разработан языковой инструментарий MetaLanguage [Сухов, 2013], который лишён ряда ограничений, присущих другим DSM-платформам. В этом случае порядок построения системы имитации, настраиваемой на предметную область, сводится к следующим действиям:

- Создание новой метамодели с указанием её имени и описания (если это необходимо). Метамодель в данном случае – это предметно-ориентированный язык моделирования, который используется для создания моделей, ориентированных на решение конкретных задач.

Построение метамодели выполняется с помощью графического редактора моделей. При создании метамодели в первую очередь определяются базовые конструкции языка. Базовыми элементами, которые используются для

создания метамодели в MetaLanguage, являются сущность, отношение, ограничение. Таким образом, в процессе создания DSL определяются сущности метамодели, отношения между ними, задаются ограничения, налагаемые на метамодели.

- Используя полученный DSL, исследователь создает модели, содержащие объекты, описывающие конкретные сущности предметной области и связи между ними.

- По окончании построения модели необходимо проверить, удовлетворяет ли она ограничениям, которые были на неё наложены, – выполнить валидацию созданной модели. Если какие-либо ограничения не выполняются, пользователь будет проинформирован об этом и сможет внести изменения в модель.

Перечисленные выше программные средства предназначены не только для разработки имитационных моделей в конкретных системах имитации (например, GPSS-модели). Они имеют гораздо более широкое применение и используются, в частности, для построения моделей бизнес-процессов [Замятина и др., 2013].

2.2. Онтологический подход к построению предметно-ориентированной модели

Вторым способом настройки на предметную область является использование модельно-ориентированного подхода на основе онтологий. Этот подход используется для построения графического интерфейса различных программных продуктов [Грибова, 2005], [Грибова, 2006], [Грибова, 2010]. В качестве примера можно рассмотреть подход к построению пользовательских интерфейсов сетевой информационно-вычислительной системы, в рамках которого сначала строится высокоуровневое описание интерфейса, его модель, а затем по ней генерируется программный код с использованием технологии сборки [Веселов, 2009]. Авторы отмечают преимущества такого подхода, которые заключаются в следующем: 1) уменьшение временных затрат на разработку графического интерфейса и упрощение модификации, поскольку разработчику предоставляется только декларативное описание; 2) использование для сборки интерфейса как собственные разработки, так и заёмные компоненты. Реализация предлагаемого подхода позволяет использовать ресурсы географически удаленных кластеров членов сетевого общества и выполнять отдельные части проекта, не выходя за пределы своей корпоративной среды и понятий своей предметной области.

Для настройки системы имитации на предметную область в TriadNS используют онтологический подход, что обусловлено следующими причинами:

- В TriadNS разработаны базовая и предметно-ориентированная онтологии, которые используются для автоматического доопределения

модели по семантическим признакам.

- Компьютерные сети являются примером быстро меняющейся предметной области (новые типы сетей, новые протоколы, алгоритмы передачи сообщений, алгоритмы маршрутизации). Поэтому необходимо располагать языковыми и программными механизмами, позволяющими выполнять оперативную автоматическую или автоматизированную настройку на динамически изменяющуюся предметную область.

- Компьютерные сети могут включать большое количество вычислительных узлов, поэтому на имитационный эксперимент требуется большое количество времени. Выходом в данной ситуации является распараллеливание имитационного прогона, распределение объектов имитационной модели по узлам высокопроизводительной вычислительной среды. При этом необходимо сохранять каузальность событий. Для распараллеливания имитационного прогона существуют классические алгоритмы синхронизации (оптимистический и консервативный). Усилия многих авторов направлены на усовершенствование этих алгоритмов. В TriadNS также разрабатываются усовершенствованные алгоритмы синхронизации (на базе оптимистического алгоритма). Особенность этих алгоритмов заключается в том, что они используют знания о модели, а знания эти частично извлекаются из онтологий. Более того, для равномерного распределения вычислительной нагрузки на узлы также используют семантические знания о модели. Разработан мультиагентный подход, агенты действуют по правилам, извлекаемым из онтологий, как базовой, так и предметно-ориентированной [Миков, 2013].

- Верификация и валидация моделей также выполняются с применением онтологий [Замятина, 2013], из онтологий извлекаются правила, которым соответствует модель.

- Использование онтологий позволяет автоматически интегрировать компоненты имитационной модели, включать в уже разработанные модели компоненты, подготовленные в другой имитационной системе.

Так как онтологии применяются во всех перечисленных выше случаях, целесообразно использовать их и для настройки на конкретную предметную область. Исследования и проведенные эксперименты показали, что онтологии позволяют поддерживать адаптивность системы моделирования и ее открытость [Замятина, 2012].

Прежде, чем более подробно рассмотреть особенности разработки онтологий в TriadNS, рассмотрим принципы построения имитационной модели, лингвистических и программных средств сбора и обработки информации о модели и т.д.

3. Представление имитационной модели в симуляторе компьютерных сетей TriadNS

Симулятор компьютерных систем разработан на базе системы автоматизированного проектирования и моделирования вычислительных систем Triad [Миков, 1995].

3.1. Описание имитационной модели

Описание имитационной модели в Triad состоит из трех слоёв: слоя структур (*STR*), слоя рутин (*ROUT*) и слоя сообщений (*MES*). Таким образом, модель в системе Triad можно определить как $M = \{STR, ROUT, MES\}$.

Слой структур представляет собой совокупность объектов, взаимодействующих друг с другом посредством посылки сообщений. Каждый объект имеет полюсы (входные P_{in} и выходные P_{out}), которые служат соответственно для приёма и передачи сообщений. Слой структур можно представить графом. В качестве вершин графа следует рассматривать отдельные объекты. Дуги графа определяют связи между объектами. Объекты действуют по определённому алгоритму поведения, который описывают с помощью программы, которую в Triad принято называть рутинной (*route*). Рутинная представляет собой последовательность событий e_i , планирующих друг друга ($e_i \in E, i=1 \div n$, – множество событий, множество событий рутинной является частично упорядоченным в модельном времени). Выполнение события сопровождается изменением состояния объекта. Состояние объекта определяется значениями переменных рутинной. Таким образом, система имитации является событийно-ориентированной. Рутинная так же, как и объект, имеет входные (Pr_{in} и выходные Pr_{out}) полюса. Входные полюса служат соответственно для приёма сообщений, выходные полюса – для их передачи. В множестве событий рутинной выделено входное событие e_{in} . Все входные полюса рутинной обрабатываются входным событием. Обработка выходных полюсов осуществляется остальными событиями рутинной. Для передачи сообщения служит специальный оператор **out** (**out** <сообщение> **through** <имя полюса>). Совокупность рутин определяет слой рутин ROUT.

Слой сообщений (*MES*) предназначен для описания сообщений сложной структуры. Алгоритмом имитации называют совокупность объектов, функционирующих по определённым сценариям, и синхронизирующий их алгоритм. Последовательный алгоритм является событийно-ориентированным. Каждая рутинная имеет свой локальный календарь событий (*tloc_i*). При выполнении очередного события осуществляется поиск события с минимальным временем (поиск в календарях событий всех рутин) и управление передается рутинной, которая включает это событие.

3.2. Сбор и обработка информации о функционировании имитационной модели

Для сбора, обработки и анализа имитационных моделей в системе TriadNS существуют специальные объекты – информационные процедуры и условия моделирования, реализующие алгоритм исследования. *Информационные процедуры* ведут наблюдение только за теми элементами модели (событиями, переменными, входными и выходными полюсами), которые указаны пользователем. Если в какой-нибудь момент времени имитационного эксперимента пользователь решит, что следует установить наблюдение за другими элементами или выполнять иную обработку собираемой информации, он может сделать соответствующие указания, подключив к модели другой набор информационных процедур. *Условия моделирования* анализируют результат работы информационных процедур и определяют, выполнены ли условия завершения моделирования.

3.3. Запуск модели на выполнение

Имитационная модель, описанная на языке Triad, может быть запущена на моделирование с помощью специального оператора *simulate*: *simulate* <список элементов моделей, за которыми установлено наблюдение> *on conditions of simulation* <наименование> (список фактических настроечных параметров)]<список входных и выходных фактических параметров интерфейса>]. Следует отметить, что исследователь может одновременно запустить на моделирование сразу несколько моделей, указав элементы, за которыми будут следить информационные процедуры, перечисленные в условиях моделирования. Кроме того, можно указать сразу несколько условий моделирования. Например, при моделировании компьютерных сетей целесообразно рассматривать эти компьютерные сети с разных точек зрения: исследователю может интересовать *стоимость* проектируемых компьютерных сетей, их *производительность*, их *надежность*, их *доступность*. Во всех этих случаях набор собираемых данных может быть различным, разными способами обрабатываются наблюдаемые данные, условия завершения моделирования также могут отличаться друг от друга. В этом случае целесообразно провести над моделью несколько имитационных экспериментов, используя различные критерии (условия моделирования).

4. Многомодельное имитационное исследование с использованием онтологий

Итак, как уже было сказано ранее, многомодельное (полимодельное) исследование позволяет привлечь к анализу результатов имитационного исследования специалистов различных предметных областей, дать им возможность работать в привычных для них

терминах, оперировать определенным кругом понятий. Для настройки на предметные области в TriadNS используют базовую и предметно-ориентированные онтологии. В настоящее время разработаны онтологии, учитывающие особенности реализации компьютерных сетей и онтологии, для представления модели в терминах системы массового обслуживания. Кратко рассмотрим особенности реализации.

4.1. Базовая онтология

Базовая онтология основывается на принципах построения имитационной модели в TriadNS. Базовую онтологию используют для представления семантических знаний, необходимых для доопределения моделей. Для представления онтологий был выбран язык OWL, поскольку существует большое количество инструментальных средств работы с онтологиями OWL, поддерживающих возможность публиковать созданные онтологии в сети Internet и объединять информацию из различных источников, как локальных, так и находящихся в глобальной сети. Для работы с онтологиями использовался инструментарий Jena OWL API. Автоматическое доопределение модели предполагает автоматическое извлечение рутины из базы рутин и ее наложение на терминальную вершину, поведение которой заранее не определено (вершина без указания алгоритма ее поведения и называется терминальной). Извлеченная рутинa должна соответствовать трем условиям автоматического доопределения:

- условию спецификации (проверка семантического имени, если семантическое имя – «маршрутизатор», то должна быть извлечена рутинa, реализующая алгоритм работы маршрутизатора),
- условию конфигурации (проверка количества входов и выходов вершины и рутины, количество их должно совпадать),
- условию декомпозиции (проверка графа-окружения, т.е. топология графа, образованного из вершин, смежных терминальной должно совпадать с «шаблоном» из базы знаний).

Семантические знания, которые необходимы для проверки условий автоматического доопределения модели, требуется извлечь из онтологического описания слоя структур. В качестве такого описания используется базовая онтология, импортируемая всеми создаваемыми онтологиями. В этой онтологии определены следующие классы: Model (класс, описывающий множество моделей языка Triad), SubMod (класс, описывающий множество всех экземпляров рутин и структур), Graph (класс, описывающий множество структур моделей, является подклассом SubMod), Routine (множество экземпляров рутин, является подклассом SubMod), Object (множество всех вершин структуры модели, является суперклассом для всех семантических типов) и т.д. Фрагмент базовой онтологии представлен на рисунке 1. Для работы с самими

онтологиями реализован класс `OntoManager`, поддерживающий загрузку нескольких онтологий, создание и сохранение онтологий. Кроме того, реализован класс `TypeManager` для работы с семантическими типами и для поиска соответствующего экземпляра рутин.

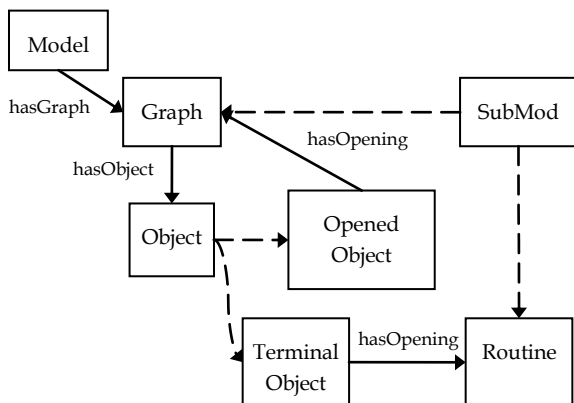


Рисунок 1 – Фрагмент базовой онтологии

4.2. Онтология для представления компьютерных сетей

Предметно-ориентированная онтология системы `TriadNS` дополняет базовую онтологию. Введены специализированные для области компьютерных сетей подклассы основных классов базовой онтологии: `ComputerNetworkModel` – модель компьютерной сети; `ComputerNetworkStructure` – структура модели компьютерной сети; `ComputerNetworkNode` – элемент компьютерной сети, изначально содержит подклассы `WorkStation`, `Server`, `Router`; `ComputerNetworkRoutine` – рутин элемента компьютерной сети и т.д.

В онтологии компьютерных сетей определены два специальных свойства для полюса вершины, представляющей элемент компьютерной сети:

- `isRequired(ComputerNetworkRoutinePole, Boolean)` определяет, обязательно ли полюс должен быть соединен с другим полюсом;
- `canConnectedWith(ComputerNetworkRoutinePole, ComputerNetworkRoutine)` определяет семантический тип элемента, с которым полюс можно соединить.

Для построения модели может быть использован язык `Triad` или графический редактор. Графический редактор по сути является визуальным языком и используется для построения как имитационной модели, так и онтологий соответствующей предметной области.

4.3. Онтология для представления модели компьютерной сети в виде системы массового обслуживания

Онтология семантических типов элементов систем массового обслуживания (СМО) дополняет базовую онтологию в следующем. Во-первых, вводятся специализированные для области систем массового обслуживания подклассы основных

классов базовой онтологии: `MpsModel` – модель системы массового обслуживания, `MpsStructure` – структура модели системы массового обслуживания и т.д. Во-вторых, вводятся четыре подкласса класса вершины, соответствующие основным элементам систем массового обслуживания: `Generator`, `Queue`, `Channel`, `Terminator`. В-третьих, вводятся классы для представления специфической для теории очередей информации: типа распределения времени, типа процедуры выборки из очереди и т.д. Иерархия семантических типов в онтологии представляется в виде иерархии классов рутин, верхним элементом которой является подкласс класса `Routine` базовой онтологии – класс `MpsRoutine`, то есть рутин системы массового обслуживания. При формировании иерархии семантических типов большую роль играет множественное наследование. На рисунке 2 представлен фрагмент системы массового обслуживания.

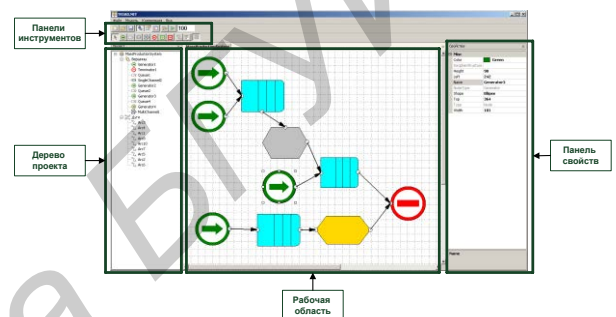


Рисунок 2 – Фрагмент графического представления модели СМО

Заключение

В работе представлен онтологический подход к настройке системы имитации на конкретную предметную область в симуляторе компьютерных сетей `TriadNS`. Настройка на конкретную предметную область позволяют улучшить качество имитационных исследований благодаря тому, что к ним могут быть привлечены специалисты из разных областей знаний, но при этом они могут работать в привычной для них предметной области и использовать определенный круг понятий, концептов, терминов. В настоящее время построены базовая и предметно-ориентированная системы имитации для компьютерных сетей и систем массового обслуживания. Коллектив авторов предполагает разработать программные и языковые средства, позволяющие выполнить преобразования (трансформации) одной модели в другую (многомодельное имитационное исследование).

Работа выполнена при поддержке РФФИ (проекты № 12-07-00302-а «Теоретические и технологические основы интеллектуального управления жизненным циклом сложных изделий» и № 13-07-96506_р-ю-а «Разработка деонтических основ мультиагентных технологий формирования баз знаний для виртуальных лабораторий»), и гранта Министерства образования и науки (проект № 8.5782.2011) «Автоматизация и оптимизация этапов имитационного моделирования».

Библиографический список

- [Веселов, 2009] Веселов А.В., Остапкевич М.Б., Пискунов С.В. Автоматизированная генерация пользовательских интерфейсов для сетевой информационно-вычислительной системы // Тр. 7-й Междунар. конф. памяти акад. А.П. Ершова "Перспективы систем информатики". Новосибирск: Сиб. науч. изд-во, 2009. С. 84–90.
- [Грибова, 2005] Грибова В.В., Клещев А.С. Использование методов искусственного интеллекта для проектирования пользовательского интерфейса // Информационные технологии. - 2005. - №8. - С.58-62.
- [Грибова, 2006] Грибова В.В., Клещев А.С. Управление проектированием и реализацией пользовательского интерфейса на основе онтологий // Проблемы управления. 2006. №2. С.58-62.
- [Грибова, 2008] Грибова В.В. Модель генерации кода пользовательского интерфейса для различных типов диалога // Научно-технические ведомости СПбГПУ. – Санкт-Петербург: Изд-во Политехнического университета. – 2008. – №3. – С.145-151.
- [Грибова, 2010] Грибова В.В., Черкезишвили Н.Н. Развитие онтологического подхода для автоматизации разработки пользовательских интерфейсов с динамическими данными // Информационные технологии. - 2010. - №10. - С. 54-58.
- [Замятина и др., 2013] Замятина Е.Б., Лядова Л.Н., Сухов А.О. О подходе к интеграции систем моделирования и информационных систем на основе DSM-платформы MetaLanguage // В кн.: Технологии разработки информационных систем ТРИС-2013. Материалы IV Международной научно-технической конференции / Т. 1. Таганрог: Издательство Технологического института ЮФУ, 2013. С. 61-70.
- [Замятина и др., 2011] Замятина Е.Б., Миков А.И., Михеев Р.А. Применение онтологий в системе проектировании и моделирования компьютерных сетей TRIADNS. Труды Конгресса по интеллектуальным системам и информационным технологиям. «IS&IT'11», Научное издание в 4-х томах, М. Физматлит, 2011, Т.1, стр. 253-260
- [Замятина и др., 2012] Замятина Е.Б., Миков А.И. Программные средства системы имитации Triad.Net для обеспечения ее адаптируемости и открытости. Информатизация и связь. 2012. № 5. С. 130–133.
- [Миков и др. а, 2010] Миков А.И., Замятина Е.Б. Проблемы повышения эффективности и гибкости систем имитационного моделирования. Проблемы информатики, №4(8), Новосибирск, Институт вычислительной математики и математической геофизики СО РАН, 2010. С.49-64.
- [Миков и др. б, 2010] Миков А.И., Замятина Е.Б., Козлов А.А. Мультиагентный подход к решению проблемы равномерного распределения вычислительной нагрузки. Natural and Artificial Intelligence, ITHEA, Sofia, Bulgaria, 2010, P.173-180.
- [Соколов, 2005] Соколов Б.В., Юсупов Р.М. Концептуальные основы квалиметрии моделей и полимодельных комплексов // Имитационное моделирование. Теория и практика: сб. докл. 2-й всерос. науч.-практ. конф. ИММОД–2005. Т. 1. СПб: ЦНИИТС. 2005. С. 65–70.
- [Сухов, 2013] Сухов А.О. Инструментальные средства создания визуальных предметно-ориентированных языков моделирования / Фундаментальные исследования. – 2013. – № 4 (ч. 4). – С. 848-852.
- [Benjamin, 2005] Benjamin P., Akella K.V., Malek K., Fernandes R. An Ontology-Driven Framework for Process-Oriented Applications // Proceedings of the 2005 Winter Simulation Conference / M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds., – pp. 2355-2363.
- [Benjamin, 2006] Benjamin P., Patki M., Mayer R. J. Using Ontologies For Simulation Modeling // Proceedings of the 2006 Winter Simulation Conference/ L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds. –pp.1161-1167
- [Liang, 2003] Liang V.-C, Pardis C.J.J. A Port Ontology for Automated Model Composition // Proceedings of the 2003 Winter Simulation Conference / S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds., - pp. 613-622.
- [Mikov, 1995] Mikov A.I. Formal Method for Design of Dynamic Objects and Its Implementation in CAD Systems // Gero J.S. and F.Sudweeks F.(eds), Advances in Formal Design Methods for CAD, Preprints of the IFIP WG 5.2 Workshop on Formal Design Methods for Computer-Aided Design, Mexico, Mexico, 1995. pp.105-127.
- [Mikov, 2007] Mikov A., Zamyatina E., Kubrak E. Implementation of simulation process under incomplete knowledge

using domain ontology //Proceedings of the 6-th EUROSIM Congress on Modeling and Simulation. University of Ljubljana, Slovenia, 2007, Vol.2. Book of full papers. p.1-7.

[Mikov, 2013] Mikov A., Zamyatina E., Kozlov A., Ermakov S. Some Problems of the Simulation Model Efficiency and Flexibility. Proceedings of «2013 8th EUROSIM Congress on Modelling and Simulation EUROSIM 2013», Cardiff, Wales, United Kingdom. P. 532-538.

[Rathnam, 2004] Rathnam T., Paredis C.J.J. Developung Federation Object Models Using Ontologies // Proceedings of the 2004 Winter Simulation Conference / R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, eds.,– pp. 1054-1062.

[Sargent, 2004] Sargent R.G. Some Recent Advances In The Process World View. // Proceedings of the 2005 Winter Simulation Conference / M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.,– pp. 293-298.

ONTONTOLOGICAL APPROACH TO A SIMULATION SYSTEM CONSTRUCTING FOR THE PARTICULAR SUBJECT AREA

Zamyatina E.B., Mikheev R.A.

*Perm State National Researching University,
Perm, Russian Federation*

e_zamyatina@mail.ru

mikheev@prognoz.ru

This paper discusses the problem of simulation system constructing for the specific domain. Authors suggest the ontological approach using. The simulation system TriadNS is considered. This simulation system is devoted for computer networks design and analysis. Authors represent base ontology and some another ontologies describing the concepts of specific domain.

Introduction

Introduction considers the actuality of problem. Indeed the adjusting of simulation model for the specific domain permits differ specialists take part in common investigations. So computer networks may be considered from the point of view of graph theory, or as queue networks, or as Petri nets and so on. The investigations become to be more quality.

Main Part

Main part includes the overview of ontology using in simulation. The overview shows that ontological approach may be useful during design, verification and validation of simulation model. Furthermore the simulation system TriadNS is described, and base ontology and ontologies for specific domains are presented.

Conclusion

The results of investigations and new directions in investigations are presented. The authors intend to design rules of transformation of one simulation model to another.