

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ФОРМИРОВАНИЮ ГЕТЕРОГЕННЫХ СРЕД МОДЕЛИРОВАНИЯ

С.В. Смирнов

Институт проблем управления сложными системами РАН
443020, г. Самара, ул. Садовая, 61
E-mail: smirnov@iccs.ru

Исследуется проблема интеграции разнородных знаний при компьютерном моделировании сложных систем. Для ее решения предлагается использовать онтологический подход, который в технологических терминах конструирования гетерогенных сред компьютерного моделирования реализует общественный опыт в определении состава и взаимосвязи основных этапов процесса отражения и изучения реального мира. В работе предложены и проанализированы состав элементов среды моделирования, играющих роль тех или иных знаний, все возможные варианты многомодельных структур, возникающих при решении прикладных задач, принципы управления разнородными элементами среды моделирования, понятийные метамодели для описания и «взаимопонимания» элементов среды моделирования.

Ключевые слова: среда моделирования, метамодели, контекст моделирования, онтология, денотативная модель, приложение

Введение

Актуальной проблемой компьютерного моделирования является необходимость одновременного отражения различных сторон, ликов, граней у моделируемых объектов и согласованного манипулирования соответствующими неоднородными представлениями для решения задач. При моделировании сложных систем становится неизбежным построение гетерогенной, многомодельной среды: «так как сложная система принципиально является многоаспектной... она не может быть описана одной моделью и необходимо разрабатывать ряд моделей одной и той же системы, предназначенных для решения различных задач или только одной задачи» [1, с. 226]. Например, средства моделирования в системах поддержки принятия коллегиальных решений должны отражать различные позиции лиц, каждое из которых ответственно за определенную сферу деятельности, имеет свою точку зрения на объект и систему управления, располагает собственными моделями [2, 3].

Согласно риторике парадигмы работы со знаниями можно утверждать, что всякая модель есть составная часть знания о моделируемой системе или некоторой «целевой» предметной области (ПрО), а результативность использования многомодельных компьютерных сред обеспечивается механизмами компьютерной интеграции знаний. Возможности же и способы компьютерной интеграции знаний определяются организацией системы знаний о целевой ПрО.

Сергей Викторович Смирнов (д.т.н.), директор Института проблем управления сложными системами РАН.

В настоящее время такую организацию все чаще связывают с построением так называемых онтологий¹ [4]. При этом (что подчеркнуто и в [1]) определяющее значение приобретают метамодели, которые используются для описания различных эпистемических (т. е. играющих роль тех или иных знаний) элементов среды моделирования и обеспечивают «взаимопонимание» этих элементов.

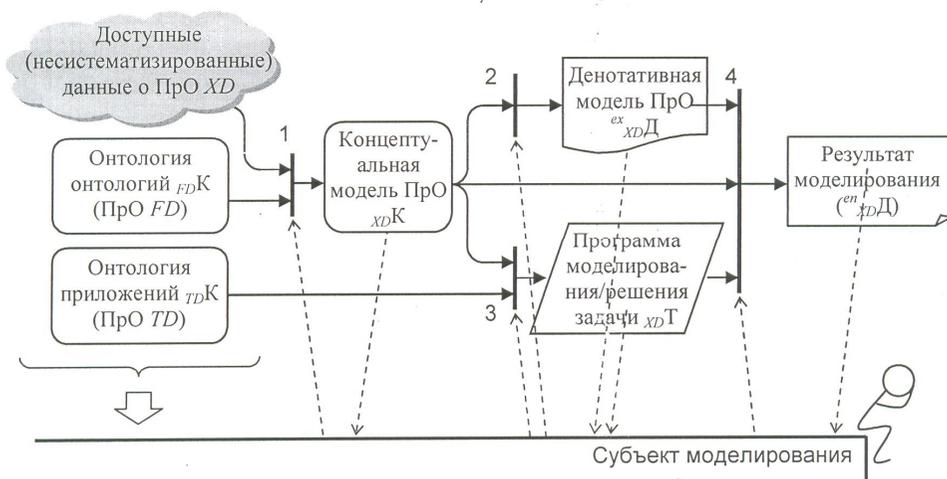
Результаты широкого исследования различных проблем построения и управления предметно-ориентируемыми средами компьютерного моделирования на основе онтологий [3, 4, 8-12] показали, что в задачах моделирования сложных систем необходимо и достаточно располагать фактически двумя понятийными метамоделями:

- описанием понятийной структуры, используемой для онтологической спецификации предметных областей моделирования;
- описанием совокупности понятий и отношений, составляющих онтологию произвольной задачи, или, проще говоря, приложения.

В статье фактически представлена интерпретация этих результатов, акцентирующая принципы реализации механизмов интеграции знаний при построении и управлении гетерогенными (или многомодельными) компьютерными средами.

1. Варианты многомодельных структур

На рис. 1 представлен укрупненный вариант схемы [10], отражающей, по нашему мнению, имеющийся значительный общественный опыт в определении состава и взаимосвязей основных этапов и эпистемических элементов технологии моделирования, которая реализуется при создании и функционировании моделирующих сред.



Р и с. 1. Схема моделирования (включение субъекта в соответствии с направлением «открытых» стрелок подчиняется правилу «плавного поворота»; фигурная стрелка обозначает априорную и перманентную «включенность» субъекта для указанных элементов схемы; 1-3 – этапы моделирования)

¹ В искусственном интеллекте, концепции Semantic Web, онтологиям отводится место посредника между тезаурусом и моделью ПрО, но на практике чаще онтология и есть модель ПрО [5]. С уточнением, что речь идет о концептуальной модели, фиксирующей понятийную структуру ПрО, ясна прагматическая роль онтологий в моделирующих средах, которая хорошо согласуется с ранее опубликованными точками зрения на концептуальную составляющую моделей ПрО, см., например, [6, 7].

Продуктом онтологического анализа (или концептуализации; см. этап 1 на рис. 1) целевой предметной области XD является концептуальная (К-) модель $_{XD}K$ – онтология, которая представляет собой таксономическую иерархию понятий PrO , где каждое понятие с помощью механизма «атрибут – значение» описывает процедуры формирования и свойства каждого элемента своего объема, способности этих элементов вступать в связи с другими элементами (валентности – см. [12]), методы работы с этими элементами, различные ассоциированные с понятием ограничения. Другими словами, онтология PrO представляет знания о моделируемой действительности в объектно-ориентированном стиле [12].

С прагматических позиций важно то, что в таком варианте одновременно с декларативным описанием целевой PrO онтология непосредственно задает совокупность абстрактных типов данных, необходимых при разработке моделирующих алгоритмов и программ, и схему объектно-ориентированной базы данных для размещения денотативных моделей объектов (систем объектов) PrO .

Результатом этапа 2 – денотативного (Д-) моделирования объекта или множества объектов целевой PrO – является ее экзогенная модель $^{ex}_{XD}D$. Этот акт моделирования выполняется в рамках онтологии целевой PrO (на «языке» соответствующей К-модели).

На этапе 3 строится денотативная модель специальной «технологической» предметной области TD (речь о ней пойдет ниже, в подразделах 3.2-3.4). Функциональная часть этой модели имеет вид программы моделирования с некоторой актуальной прагматикой и реализует знания о допустимых «вычислениях», связанных с Д-моделями целевой PrO . Этот этап с весьма общей точки зрения можно рассматривать как спецификацию сценария решения задачи, описание воздействий на экзогенную денотативную модель целевой PrO , в результате которых она должна приобрести некоторые удовлетворяющие пользователя свойства. Можно сказать, что продуктом этапа 3 служит трансформационная (Т-) модель, а операционным базисом специфицированных ею трансформаций, аппаратом решения служит исчисление, основу которого составляют функциональные части К-модели целевой PrO . Для такой «компетентной» в целевой PrO XD Т-модели используем обозначение $_{XD}T$.

Наконец, на этапе вычислений (4 на рис. 1) знания, зафиксированные в $_{XD}T$, путем интерпретации на физической или виртуальной машине реализуются применительно к экзогенной денотативной модели целевой PrO $^{ex}_{XD}D$. Для субъекта моделирования ценен результат этого вычислительного эксперимента: либо эндогенная денотативная модель целевой PrO $^{en}_{XD}D$, либо зафиксированная апостериори последовательность воздействий на $^{ex}_{XD}D$, трансформирующая ее в $^{en}_{XD}D$, либо то и другое вместе.

Схема моделирования на рис. 1 по определению является многомодельной. Исключая $^{en}_{XD}D$ как производную аппликация $_{XD}T$ к $^{ex}_{XD}D$, а также априори фиксированные, стандартные понятийные метамодели $_{FD}K$ и $_{TD}K$, приходим к выводу, что для формального описания указанной схемы пригоден кортеж $(_{XD}K, ^{ex}_{XD}D, _{XD}T)$.

Для анализа с помощью этого формализма теоретически возможных схем интеграции моделей (и, следовательно, тех вариантов многомодельных структур, поддержание и управление которыми должна обеспечивать многомодельная среда) необходимо принять во внимание следующее:

– основным феноменом онтологического анализа является возможность формирования и сосуществование нескольких различных онтологий (К-моделей) целевой PrO ;

– «на языке» всякой онтологии можно построить в общем случае сколько угодно различных денотативных моделей ПрО;

– для всякой ПрО можно указать множество актуальных задач или множество вариантов решения некоторой одной задачи;

– для решения реальных задач наряду с моделированием целевой ПрО требуется, как правило, привлечение знаний из возможно нескольких обособленных проблемно- (или методо-) ориентированных, инструментальных (*PSD*-) ПрО;

– в качестве целевой ПрО в процессе моделирования может выступать *FD* (при построении новых онтологий).

Тогда, варьируя элементами введенного кортежа и замещая их конечными множествами элементов того же рода, а также образуя множества допустимых кортежей, можно выявить следующие несводимые схемы интеграции моделей:

– $(_{XD}K, \{^{ex}_{XD}D_i\}, _{XD}T)$ – одновременная манипуляция несколькими Д-моделями целевой ПрО (базовый вариант);

– $\{(_{XD}K_j, \{^{ex}_{XD}D_{j,i}\}, _{XD}T)\}$ – совмещение различных взглядов на целевую ПрО, причем в некотором смысле (см. подраздел 3.2) имеем $_{XD}T_j \in _{XD}T$;

– $\{(_{XDk}K_k, \{^{ex}_{XD}D_{k,i}\}, _{XD}T)\}$ – совместное рассмотрение нескольких целевых ПрО (здесь также $_{XD}T_k \in _{XD}T$);

– $\{(_{XD}K, \{^{ex}_{XD}D_i\}, _{XD}T), \{(_{PSDk}K_k, \{^{ex}_{PSDk}D_{k,i}\}, _{XD}T)\}\}$ – моделирование целевой ПрО с использованием набора актуальных проблемно-ориентированных ПрО решения (под)задач, $_{PSDk}T_k \in _{XD}T$;

– $(_{FD}K, \{^{ex}_{FD}D_i\}, _{FD}T)$ – конструирование и редактирование онтологий ПрО: $_{FD}K$ – онтология онтологий, $^{ex}_{FD}D$ – пустая или подлежащая редактированию онтология некоторой ПрО, $_{FD}T$ – редактор онтологий;

– $(_{XD}K, \{^{ex}_{XD}D_i\}, \{_{XD}T_m\})$ – множественность Т-моделей (для организации различных или распределенных в пространстве-времени вычислений).

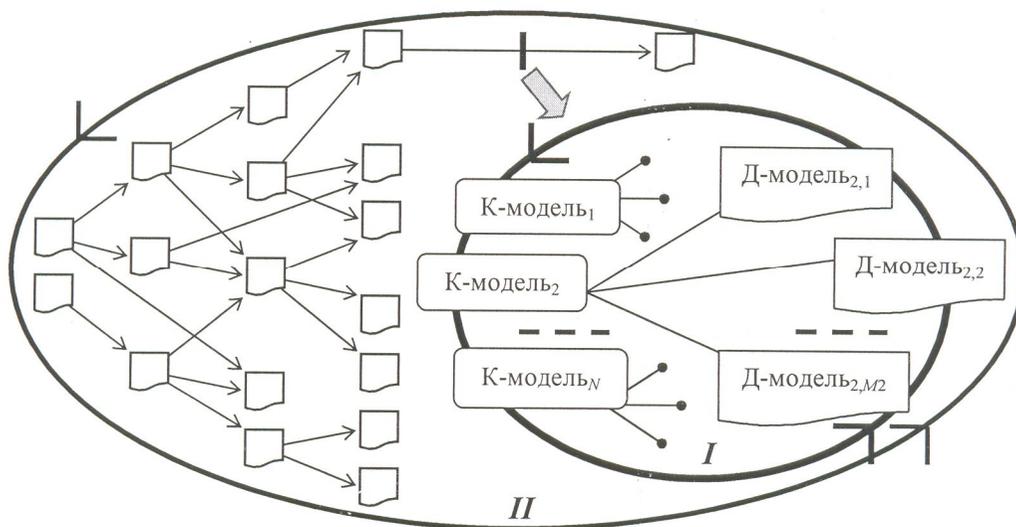
На этой основе возникают смешанно-сцепленные многомодельные структуры, когда промежуточные эндогенные Д-модели используются в качестве экзогенных в последовательной серии вычислительных экспериментов. При этом подструктуры, связанные с конструированием и редактированием онтологий, определяют эволюционный характер такого исследования, а варьирование Т-моделями при прочих равных условиях реализуют альтернативное развитие процесса моделирования.

2. Управление в гетерогенной среде моделирования

В многомодельной среде необходимо поддерживать такое управление гетерогенными моделями, которое обеспечит возможность реализации всех выделенных в разделе 1 схем интеграции моделей, т. е. многоаспектное представление целевой ПрО, решение актуальных для целевой ПрО междисциплинарных задач (соответствующие проблемно-ориентированные ПрО могут иметь собственное отражение в комплекте моделей различных типов) и в целом альтернативно-эволюционный характер решения этих задач. Для реализации этого требования необходимы два «контура» управления моделями на этапе вычислений (рис. 2).

В «задачном» контуре (*I* на рис. 2) путем переключения контекстов моделирования (КМ) – пар $\langle K, D \rangle$ – обеспечивается работа с несколькими Д-моделями из разных, вообще говоря, ПрО. При этом должен контролироваться состав допустимых ПрО: Т-модель должна быть компетентна в каждой из них, - и корректность контекстов моделирования: К- и Д-модели в КМ должны представлять одну и ту же ПрО, разумеется, первая должна быть онтологией, вторая – денотативной моделью, построенной в рамках этой онтологии.

«Проектный» контур (*II* на рис. 2) связан с макроуправлением вычислениями. Соответствующий механизм многомодельной среды для организации взаимосвязанных серий экспериментов должен структурировать сведения об исполненных актах вычислений (В-актах) в форме растущего ациклического графа: В-акт x связан с предшествующим ему В-актом x^* в том смысле, что при выполнении x используется хотя бы одна эндогенная Д-модель В-акта x^* ; в общем случае в В-актах используются различные Т-модели. Каждая такая структура определяет отдельный проект исследования на моделях. Надлежащим образом контролируя рост этой структуры, механизм управления проектами способен обеспечить в многомодельной среде транзакционные свойства В-актов.



Р и с. 2. Управление моделями на этапе вычислений:
I – задачный и *II* – проектный «контуры» управления

3. О метамоделях

3.1. Общие замечания. Из двух метамodelей на рис. 1 в действительности только FDK представляет собой метаспецификацию – онтологию онтологий, или метаонтологию, а TDK – «обычная» онтология, которая отнесена к метамоделям условно, благодаря лишь особой роли в организации работы многомодельной среды.

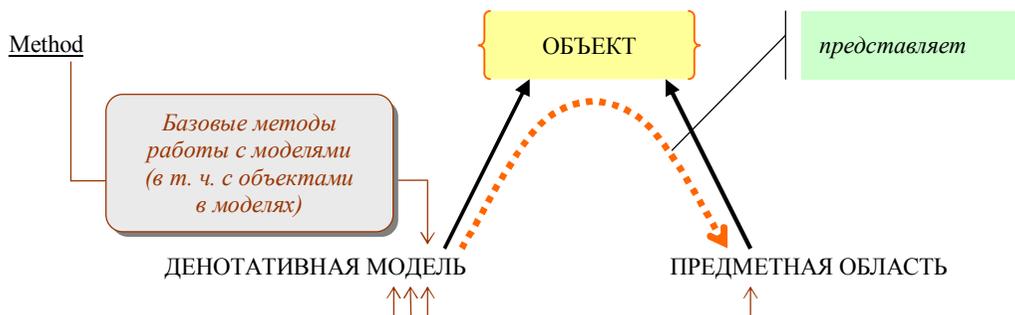
Вопрос о метамодели для понятийного описания ПрО имеет давнюю историю и непосредственно связан с проблематикой аналитической философии, оснований математики и развитием теории понятийно-ориентированных языков [13-15]. Варианты такого рода метамодели, дополнительно отражающие прагматику создания инструментария для построения предметно-ориентированных сред компьютерного моделирования, предложены в [8, 12].

Менее определено обстоит дело со второй метамоделью – понятийной структурой приложений, ориентируемых на решение задач моделирования. Обычно подобные метамодели явно не фиксируются, но так или иначе строятся соответствующие денотаты как спецификации априори очерчиваемого для конкретного приложения (а если посмотреть шире, то машины вывода, пакета прикладных программ и т. п.) класса решаемых задач. Например, в вычислительном моделировании такими денотатами служат специальные решетки задач [16]. Примером же явной фиксации метамодели класса задач может служить определение грамматики формального языка

ка описания задач [17]. Исследуя подобные примеры, можно констатировать, что метамодель задач (зафиксирована ли она формально и явно или «понятна» благодаря восприятию некоторого контекста) реферирует концептуальную понятийную модель некоторой целевой или проблемно-ориентированной области решения задач. Причем в последнем случае об обнаружении задачи допустимого класса говорят тогда, когда объектам целевой ПрО удастся корректно сопоставить объекты проблемно-ориентированной области (где имеется механизм решения задач).

Обобщение подобных представлений возможно путем выбора в качестве основы для спецификации задач моделирования нового признака их классификации: задачи можно различать, указывая, во-первых, предметную область, где они возникают, и, во-вторых, предметные области, которые приходится дополнительно рассматривать для решения задач (в частности, указывая проблемно-ориентированные области) [9].

3.2. Принципы онтологического описания задач. Онтологическую основу «простой» задачи моделирования определяет следующий принцип: несмотря на бесконечное разнообразие мыслимых задач, их общность состоит в том, что всякая из них возникает «в пределах» некоторой ПрО применительно либо к какому-то объекту данной ПрО, либо к системе таких объектов. Поэтому онтология приложений $_{TD}K$ включает два фундаментальных понятия: «Предметная область» (представленная ее онтологией) и «Денотативная модель» (модель некоторой системы объектов ПрО), которые связаны ограниченным по мощности отношением принадлежности: всякая Д-модель представляет, конкретизирует некоторую ПрО и в каждой ПрО может одновременно рассматриваться любое количество Д-моделей (рис. 3).



Р и с. 3. Фрагмент онтологии приложений. Базовые понятия:

- ↖, ↗ – связи понятий (объектов-классов) между собой и с объектами-атрибутами (реализуют отношения «является_видом» и «является_частью» соответственно);
- объекты-атрибуты указаны курсивом, подчеркнут терминальный класс;
- ОБЪЕКТ – понятие, априори включаемое во всякую понятийную структуру [12]

Развитие подхода к унификации описания задач состоит в следующем:

– в общем случае «пространство действий» приложения, направленного на решение некоторой одной задачи, задается совокупностью нескольких КМ (онтология ПрО, денотативная модель данной ПрО) и в этом смысле имеем (см. раздел 1)

$$_{XD}T_j \in _{XD}T;$$

– в каждом текущем КМ (КМТ) «объектом действий» является Д-модель КМТ;

– «инструментарием действий» (операционным базисом целевой трансформации Д-модели) служат функциональные составляющие собственно онтологии задач $_{TD}K$ и онтологии КМТ, т. е. методы и демоны этих онтологий;

– в качестве основного регламентирующего правила «организации действий», направленных на решение задачи (т. е. порядка выполнения моделирующей программы, являющейся функциональной частью Т-модели $_{XD}T$), выступает требование фиксации КМ в начале действий либо, в общем случае, необходимость последовательной смены в ходе решения одного КМ другим.

Отсюда вытекают наиболее общие требования к комплектности функциональных составляющих онтологии приложений:

– во-первых, она должна включать заготовку, каркас приложения – пользовательской программы оригинальных содержательных действий, реализующих решение задачи. В традиционном понимании это каркас главной (или основной, выполняемой первой и управляющей вызовом подпрограмм) программы, онтологически позиционируемый как полиморфный метод модели $_{TD}K$ в целом, который подлежит конкретизации в каждой Т-модели;

– во-вторых, необходимо располагать методами, реализующими управление контекстами моделирования, к которым в частности относятся методы работы с различными моделями «в целом»: открытие, закрытие и т. п. (все подобные методы характеризуют понятие «денотативная модель»).

Наконец, в [12] показано, что именно понятие «денотативная модель» онтологии задач целесообразно «нагрузить» ординарными базовыми методами работы с моделируемыми объектами, такими как порождение, уничтожение, изменение значения атрибута и т. п. (см. рис. 3).

3.3. Формирование компетенции трансформационной модели и текущего набора обрабатываемых моделей. Формально онтология $_{TD}K$ никак не ограничивает формирование «пространства действий» конкретного приложения – денотативной модели $_{XD}T$.

Действительно, при выполнении Т-модели допустимо породить и активизировать в качестве КМТ пару $\langle_{TD}K, _{XD}T\rangle$. В этом КМ можно произвольно модифицировать пространство действий $_{XD}T$ и, в частности, изменить состав предметных областей, где решается задача. Другими словами, допустимо произвольное, «на лету», декларативное изменение компетенции программного приложения. Хотя подобные решения и имеют смысл, на практике чаще востребованы приложения с фиксированным набором областей применения. И для таких случаев целесообразно располагать механизмами, преодолевающими описанный рост сложности управления в многомодельной среде.

Соответствующая организационно-технологическая идея состоит в априорном формировании компетенции приложения $_{XD}T$ на этапе его разработки:

– во-первых, с момента создания любая Т-модель $_{XD}T$ должна быть компетентна для работы в КМ $\langle_{TD}K, _{XD}T\rangle$;

– во-вторых, при действии в контексте моделирования $\langle_{TD}K, _{XD}T\rangle$ в новой Т-модели порождается набор объектов класса «Предметная область», который фиксирует компетенцию создаваемого приложения.

Последующая содержательная разработка данного приложения (создание в рамках этой уже частично определенной Т-модели главной программы приложения, необходимых подпрограмм, диалоговых форм и т. п.) проводится при ограничении использования базовых ординарных методов работы с объектами. Соответствующее соглашение состоит в том, чтобы при выполнении разработанного приложения исключить изменение априори сформированного в Т-модели набора объектов класса «Предметная область», а манипуляцию объектами класса «Денотативная модель»

вести опосредованно, с помощью «макрометодов» TDK для управления контекстами моделирования.

Таким образом, в конструируемой многомодельной среде у всякой Т-модели при решении задачи будут существенно и целенаправленно ограничены возможности динамического формирования «пространства действий»: допустимые КМ определяются априори зафиксированными компетенциями Т-модели, но в этих пределах выбор денотативных моделей для целевой трансформации остается свободным. При этом услуги по образованию новых и уничтожению имеющихся КМ, по переключению между различными КМ и т. п. (которые связаны не только с манипулированием системой объектов в $PrO\ TD$, но и включают, в частности, значительный объем работы с файловой системой ЭВМ по поиску, открытию и закрытию Д-моделей) выполняются высокоуровневыми функциями управления контекстами моделирования.

3.4. Понятие «актор» и связанное с ним расширение онтологии приложений. Общее представление о процессе решения всякой задачи как о последовательности действий в сменяемых друг друга КМ предполагает, что информационное взаимодействие между Д-моделями, определяющими различные контексты, будет осуществляться с помощью средств представления данных базовой платформы программирования многомодельной среды.

Поскольку семантика каждого из используемых КМ, вообще говоря, уникальна (она определяется онтологией контекста) и, следовательно, уникален характер информации, которой «обмениваются» Д-модели в процессе решения задачи, то затруднительно предложить эффективный унифицированный механизм реализации подобных информационных обменов.

Вместе с тем опыт показывает, что типичной причиной, вызывающей необходимость совместного рассмотрения в процессе решения одновременно нескольких PrO , является многоаспектность, «многоликость» объектов, которые фигурируют в проблемных ситуациях, стимулирующих постановку задач компьютерного моделирования. Естественно именовать подобных фигурантов «актерами» конкретных моделирующих приложений.

Например, в дискретно-событийных имитационных моделях фундаментальную роль играют активные объекты, онтологическая спецификация которых обычно формулируется в терминах, разработанных для абстрактного понятия «транзакт» [18]. В то же время по крайней мере части подобных объектов резонно сопоставить разнообразные ресурсные компоненты описываемой системы, т. е. понимать эти объекты как определенные «ресурсы» [19]. Таким образом, у многих объектов в задачах имитационного моделирования легко обнаружить, по меньшей мере, два лица: одно представляет объект в мире динамики (лицо – транзакт), а другое (или несколько других) – в мире статики (лицо – набор ресурсов неизменного состава).

Возвращаясь к проблеме взаимосвязи контекстов моделирования, можно констатировать, что если различные стороны, аспекты объектов – акторов приложения оказываются описанными в разных онтологиях, то смена КМ в процессе моделирования вызывается в том числе и необходимостью отразить эволюцию, целевую трансформацию каждого актора применительно ко всем, вообще говоря, его ипостасям.

В таблице отражена допустимая логика представления многоликих акторов $\{x_1, x_2, \dots, x_n\}$ в Т-модели, которая «компетентна» в предметных областях $PrO\ 1$, $PrO\ 2$, ..., $PrO\ m$ с непустыми наборами обрабатываемых Д-моделей каждой из этих PrO .

Таблица фиксирует следующее:

– актер x_1 имеет «лица» (см. затененные ячейки), воплощаемые в каждой из открытых Д-моделей ПрО 2, причем «значением» лица в $2,1Д$ -модели является ее объект $2,1x_1$, в $2,2Д$ – $2,2x_1$, ..., в $2,k2Д$ – $2,k2x_1$;

– актер x_2 в данный момент не имеет реализации ни в одной из сфер компетентности Т-модели;

– актер x_3 имеет лицо в каждой из предметных областей Т-модели, но лишь в одной из множества открытых Д-моделей каждой ПрО;

– актер x_n имеет смешанный (в сравнении с x_1 , x_2 и x_3) вариант представления в денотативных моделях в сферах компетентности Т-модели.

Пример представления акторов приложения (Т-модели)

Актор приложения (<i>Old</i> в Т-модели)	Значение лица актора в Д-модели предметной области (<i>Old</i> «объекта-значения» лица в Д-модели)												
	ПрО 1			ПрО 2				...	ПрО m				
	$1,1Д$	$1,2Д$...	$1,k1Д$	$2,1Д$	$2,2Д$...	$2,k2Д$...	$m,1Д$	$m,2Д$...	$m,kmД$
x_1					$2,1x_1$	$2,2x_1$		$2,k2x_1$					
x_2													
x_3	$1,1x_3$		$2,k2x_3$...	$m,1x_3$...	
...													
x_n	$1,1x_n$	$1,2x_n$		$1,k1x_n$	$2,1x_n$								

Анализ возможностей представления акторов приложения приводит к следующим заключениям.

1. Лицо актора воплощается в некоторой конкретной Д-модели определенной ПрО. Локально лицо реализует определенный объект Д-модели, который (точнее, его *Old*) играет роль значения лица актора.

2. Содержательное воплощение лиц данного актора в разных ПрО безусловно различно, хотя их значения могут находиться в любом отношении друг к другу. Например, может оказаться, что $1,1x_3 = 2,k2x_3$, а $2,k2x_3 < m,1x_3$.

3. Значения лиц, расположенные в одном и том же столбце таблицы, также могут находиться в любом соотношении. Содержательно это, в частности, означает, что с некоторой точки зрения, в рамках некоторой концептуализации допустимо тождество отдельных лиц у разных акторов, которые в целом далеко не тождественны.

4. Нет оснований утверждать, что актер в Д-модели не может располагать более чем одним лицом (в таблице такая иллюстрация не представлена).

5. Из предыдущего положения вытекает, что на содержательном уровне лица актора в разных Д-моделях определенной ПрО могут, вообще говоря, качественно различаться (быть объектами разных классов).

На рис. 4 представлен фрагмент онтологии приложений, где показаны классы «Актор» и «Лицо», связанные отношением принадлежности «один ко многим»: любой актер может иметь произвольное количество лиц, но каждое лицо принадлежит лишь одному актору. Лицо актора означает в некотором определенном КМ, для идентификации которого используются атрибуты, указывающие имена К- и Д-моделей; атрибут «Объект-значение» хранит *Old* объекта-значения лица в указанной

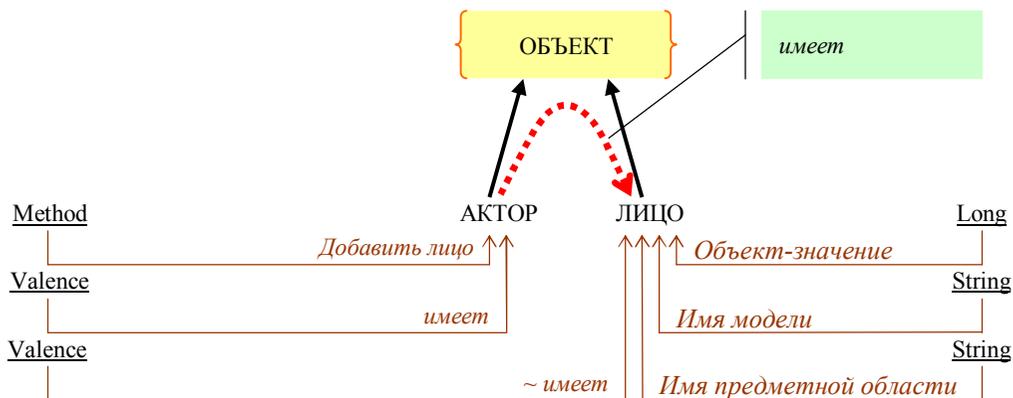
Д-модели. Атрибуты «имеет» (в том числе с тегом «~») являются валентностями [12].

В ходе решения задачи создание (а при необходимости и уничтожение) многоликих акторов осуществляется в КМ $\langle_{TD}K, x_D T\rangle$. Для оснащения актора лицом в одной из обрабатываемых Д-моделей класс «Актор» целесообразно снабдить высокоуровневым методом, который:

- порождает для формируемого лица в указываемой Д-модели объект-значение нужного класса;
- порождает в Т-модели $x_D T$ новый объект класса «Лицо», означивая надлежащим образом его характеристические атрибуты и связывая новое лицо с актуальным актором Т-модели.

Разумеется, уничтожение актора должно повлечь уничтожение всех его лиц, но для сохранения целостности Д-моделей никак не должно отражаться на объектах, реализующих лица (в Д-моделях, содержащих эти объекты).

Для актора чрезвычайно важно предельно упростить переход от работы с представлением одного его аспекта к работе с представлением другого его аспекта. Иначе говоря, речь идет о «прозрачности» для разработчика приложений перехода от манипулирования объектом-значением одного лица актора к действиям с объектом-значением другого лица этого актора. Такой переход должен, безусловно, сопровождаться надлежащим переключением КМ. При этом нетрудно заметить, что введение метода, реализующего такой «гиперпереход», фактически приведет к неявному расширению набора базовых методов работы с объектами (см. рис. 3).



Р и с. 4. Фрагмент расширенной онтологии приложений: описание «многоликих» акторов многомодельной среды

Действительно, с одной стороны, выполнение всякого метода осуществляется в ходе решения какой-либо задачи, т. е. в приложении, когда доступны сведения о существовании и состоянии акторов. С другой стороны, суть рассматриваемого метода перехода – «оттолкнувшись» от некоторого исходного актуального объекта (который, вообще говоря, лишь предположительно является объектом-значением лица одного из акторов выполняемого приложения), получить другой объект во вполне определенной, явно указываемой Д-модели (этот другой объект согласно предположению об исходном объекте будет объектом-значением другого лица актора). Сопоставляя два последних предложения, заключаем, что рассматриваемый метод может быть применен к любому объекту во всякой Д-модели, т. е. де-факто является базовым ординарным методом работы с объектами. Применение этого метода к про-

извольному объекту в любой ПрО не приводит к ошибке и интерпретируется в предположении, что обрабатываемый объект является значением лица некоторого актора. Когда это верно, то метод корректно выполняет смену КМТ и переход к работе с нужным объектом-значением лица актора, который, по существу, указан неявно.

Заключение

Проблема создания и использования многомодельных сред компьютерного моделирования естественным образом формулируется в рамках проблематики разработки механизмов компьютерной интеграции разнородных знаний. Онтологический подход к этой проблеме позволяет взять за основу возможных решений общность и четкую определенность процессов отражения и изучения реального мира.

В проведенных исследованиях и разработках на передний план выдвигалась задача формирования теоретических положений о многомодельных средах компьютерного моделирования, характерной особенностью которых является построение и организации целенаправленной трансформации семантически гетерогенных объектных пространств, описывающих моделируемую реальность.

Предложенный способ представления многоаспектных сущностей – «многоликих» акторов задач моделирования – призван не только отражать и сохранять содержательный интегральный образ моделируемой реальности, но и радикально упростить коммуникации между моделями, представляющими в гетерогенной среде моделирования различные аспекты, «миры» решаемой задачи.

В целом полученные результаты позволяют расширить арсенал механизмов интеграции разнородных знаний при моделировании сложных систем, для которых многоаспектность представления является одним из коренных следствий слабоформализуемости и определяющей характеристикой сложности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Емельянов В.В., Курейчик В.В., Курейчик В.М.* Теория и практика эволюционного моделирования. – М.: ФИЗМАТЛИТ, 2003.
2. *Виттих В.А.* Процессы управления в социотехнических системах // Проблемы управления и моделирования в сложных системах: Труды VII Международной конф. (27 июня – 1 июля 2005 г., Самара, Россия). – Самара: СамНЦ РАН, 2005. – С. 32-42.
3. *Vittikh V.A., Smirnov S.V.* Modeling Environment To Support Decision-Making Processes During Controlling Of Complex Systems // Proc. of the 16th IMACS World Congress 2000 on Scientific Computation, Applied Mathematics and Simulation (August 2000, Lausanne, Switzerland). CD ROM Index 715-3.
4. *Смирнов С.В.* Онтологии в задачах моделирования сложных систем // Проблемы управления и моделирования в сложных системах: Труды II международной конф. (20-23 июня 2000 г., Самара, Россия). – Самара: СамНЦ РАН, 2000. – С. 66-72.
5. *Нариньяни А.С.* Кентавр по имени ТЕОН: Тезаурус + Онтология // Труды международного семинара по компьютерной лингвистике и ее приложениям «Диалог'2001». Т. 1 (2001 г., Аксаково, Россия). – С. 184-188.
6. *Тамм Б.Г., Пуусетт М.Э., Таваст Р.Р.* Анализ и моделирование производственных систем. – М.: Финансы и статистика, 1987. – 191 с.
7. *Христьяновский Д.Г., Эрлих А.И.* Проблемы моделирования в прикладных интеллектуальных исследованиях // Труды III конф. по искусственному интеллекту (октябрь 1992 г., Тверь, Россия). Т. 2. – Тверь: Российская ассоциация ИИ, 1992. – С. 78-81.
8. *Смирнов С.В.* Онтологическая относительность и технология компьютерного моделирования сложных систем // Известия Самарского научного центра РАН. 2000. Т. 2. № 1. – С. 66-71.
9. *Смирнов С.В.* Онтологический анализ предметных областей моделирования // Известия Самарского научного центра РАН. 2001. Т. 3. № 1. – С. 62-70.
10. *Смирнов С.В.* Онтологии в прикладных интеллектуальных системах: прагматический подход // Девятая Национальная конф. по искусственному интеллекту с международным участием КИИ-2004 (28 сентября – 2 октября 2004 г., Тверь, Россия): Труды конф., Т. 3. – М.: ФИЗМАТЛИТ, 2004. – С. 1059-1067.

11. *Смирнов С.В., Гинзбург А.Н.* Формирование и использование сосуществующих контекстов моделирования сложной системы // Проблемы управления и моделирования в сложных системах: Труды VIII международной конф. (24-28 июня 2006 г., Самара, Россия). – Самара: СамНЦ РАН, 2006. – С. 550-557.
12. *Смирнов С.В.* Прагматика онтологий: объектно-ориентированная модель знаний о предметной области // Одиннадцатая Национальная конф. по искусственному интеллекту с международным участием КИИ-2008 (28 сентября – 3 октября 2008 г., Дубна, Россия): Труды конф. Т.3. – М.: ЛЕ-НАНД, 2008. – С. 208-216.
13. *Фреге Г.* Смысл и денотат // Семиотика и информатика. 1977. Вып. 8.
14. *Куайн В.* Онтологическая относительность // В кн.: Современная философия науки: знание, рациональность, ценности в трудах мыслителей Запада. 2-е изд., перераб. и доп. – М.: Логос, 1996. – С. 40-61.
15. *Плесневич Г.С.* Понятийно-ориентированные языки в инженерии знаний // Новости искусственного интеллекта. 2003. № 6. – С. 3-9.
16. *Тьюгу Э.Х.* Концептуальное программирование. – М.: Наука, 1984. – 320 с.
17. *Кораблин М.А.* Конструирование специфицирующих оболочек для пакетов прикладных программ // УСиМ. 1990. № 2. – С. 43-49.
18. *Киндлер Е.* Языки моделирования. – М.: Энергоатомиздат, 1985. – 288 с.
19. *Емельянов В.В., Ясиновский С.И.* Введение в интеллектуальное имитационное моделирование сложных дискретных систем и процессов. Язык РДО. – М.: АНВИК, 1998. – 427 с.

Статья поступила в редакцию 4 октября 2011 г.

ONTOLOGY-BASED APPROACH TO THE BUILDING OF HETEROGENEOUS MODELING ENVIRONMENT

S.V. Smirnov

Institute for the Control of Complex Systems of RAS
61, Sadovaya st., Samara, 443020

The paper investigates the problem of integration of heterogeneous knowledge in computer-based modeling of complex systems. For solving this problem, it is proposed to use ontology-based approach, which implements in technological terms the common experience of determining the composition and interrelations of main stages of the process of real world modeling and analysis. The paper proposes and analyses the elements of the modeling environment, which act as various pieces of knowledge, all the possible variants of multimodel structures, arising from solving practical tasks, principles of controlling heterogeneous elements of modeling environment, conceptual metamodels for description and “interaction” of modeling environment elements.

Keywords: modeling environment, metamodel, modeling context, formal ontology, denotative model, program application.