

УДК 004.89

В.И. Межуев

Бердянский государственный педагогический университет, Украина
mejuev@ukr.net

Особенности компьютерного моделирования предметных областей и систем

В статье анализируются проблемы компьютерного моделирования предметных областей (ПрО) и систем. Рассмотрены свойства и этапы познания предметных областей и систем, формулируется способ перехода предметной области в класс систем. Метод основан на определении и использовании метамоделей, которые структурируют свойства ПрО и позволяют решать возникающие над ПрО задачи.

Введение

Понятие предметной области (ПрО) в компьютерных науках было введено в начале 80-х годов прошлого века, когда была осознана необходимость использования семантических моделей для представления информации в компьютерных системах. Несмотря на широкое использование понятия предметной области в рамках теории и практики построения информационных систем (ИС), на сегодняшний день отсутствует не только формальное, но и конструктивное содержательное определение данного понятия. Популярное в литературе определение ПрО как «части действительности» не отражает сущности ПрО и не дает возможности четкого выделения моделей ПрО, проведения их классификации, анализа используемых для моделирования ПрО средств и методов, а следовательно, и понимания классов задач, которые можно решать над ПрО в ИС.

Системный подход и системный анализ являются важнейшими методологиями современного естествознания. Поэтому представляется необходимым анализ понятия предметной области в контексте его сходств и различий с понятием системы. Заметим также широкую разработанность методов моделирования, являющихся частью общей теории систем (ведущей начало от работ Людвиг фон Бергаланфи [1]). Предметом рассмотрения в данной статье является понятие предметной области в контексте системного подхода, а **целью работы** - разработка способа перевода ПрО в класс систем, что позволит использовать методы теории систем как для моделирования структуры ПрО, так и решения возникающих над ПрО задач.

Понятия предметной области и системы

Как и любая часть действительности, ПрО может быть рассмотрена как объект, к которому применим системный подход. Однако понятия предметной области и системы не являются тождественными. Осознание действительности как ПрО мы определяем как *необходимый этап разработки системы*. Любую часть заинтересовавшей нас действительности можно классифицировать как ПрО, однако *системой ПрО становится только после познания ее структуры* (то есть существенных связей и отношений между элементами ПрО).

Важным аспектом является то, что ПрО может и не являться системой, однако ИС над ней может быть построена. Таким образом, познание законов существования

ПрО не является необходимым условием для построения ИС. Под законами существования мы понимаем глубинные, важнейшие связи и отношения ПрО, а не дескриптивную (содержательно-описательную) модель ПрО, которая выделяется при проектировании ИС (прежде всего, баз данных). Класс задач, решаемый в этом случае над ПрО посредством ИС, является весьма ограниченным и сводится в основном к хранению и поиску необходимой информации. Решение же возникающих над ПрО задач анализа, объяснения, прогнозирования и др. требует понимания и отражения в ИС законов существования ПрО.

Заметим, что мы будем использовать понятие «информационная система» в широком смысле, то есть как компьютерную систему, предназначенную для сбора, хранения, обработки, передачи информации. Программные системы (программы), базы данных, базы знаний, информационные хранилища и т.д. являются частными случаями информационных систем.

Также неверно говорить, что любое представление ПрО в ИС является системой, так как оно осуществлено в *системе* функционирующих в ИС понятий и отношений. Модель ПрО может перейти в ранг системы только путем отражения существенных свойств и отношений *самой* ПрО.

Отметим также иную проблему, которая часто ускользает от внимания при разработке ИС. Существующей практикой в разработчике ИС является рассмотрение ПрО не как части действительности (онтологический аспект), а как знания про эту действительность (гносеологический аспект). Иными словами, теоретическую основу построения ИС над ПрО составляют логические модели, то есть неявно принимается за истину утверждение, что закономерности функционирования ПрО имеют форму логических утверждений.

Однако в общем случае законы логики не отражают существенные связи и отношения ПрО как таковой, а являются законами функционирования понятий, в которых формулируются свойства ПрО. Только применение математических методов к описанию ПрО позволяет решить задачу построения моделей, адекватных отражаемой действительности.

Заметим, что понятие адекватности модели ПрО существенно отличается в логическом и математическом подходе:

- адекватность математического метода определяется практикой (экспериментом), как степень соответствия формальной модели характеру изучаемой с ее помощью ПрО;
- адекватность логической модели является внутренним свойством самой логической системы (например, соответствие описания, выраженного в форме требований и спецификаций, свойствам модели ПрО).

В методологии разработки компьютерных систем прослеживается их четкое разделение на 2 основных направления [2]:

- выполнение вычислений;
- накопление и обработка информации.

Выполнение вычислений имеет своей сущностью математическое моделирование ПрО с целью решения возникающих естественнонаучных, а также инженерных задач. Второе направление основано на моделировании ПрО путем представления ее свойств в информационной системе, под которой *традиционно* понимается база данных (БД), база знаний (БЗ), информационное хранилище (ИХ) [3].

Различия в данных подходах обусловлены языком описания ПрО (естественным либо формальным), математическим аппаратом, используемым для моделирования, а

также тем, что в естественнонаучном моделировании речь идет о сравнительно небольших массивах числовой информации (в моделировании ПрО в ИС речь идет о представлении больших массивов данных, выраженных в нечисловой форме).

Однако, несмотря на различия в данных подходах, их единой сущностью является моделирование ПрО с целью решения возникающих в них задач. Таким образом, актуальной является задача анализа различных подходов к отражению ПрО в ИС с целью интеграции их в единой теории моделирования предметных областей.

Заметим, что первым этапом в рассмотрении любой ПрО является ее описание, чаще всего осуществляемое на естественном языке. Поэтому является необходимым разработка метода перехода от содержательно-описательной модели ПрО, к модели, выраженной формально-математическими средствами. Возможность такого перехода предоставляют *онтологии*, являющиеся по определению Тома Грубера точными, то есть выраженными формальными средствами, спецификациями концептуализации [4]. В наших предыдущих работах [5], [6] была проанализирована возможность использования онтологий как моделей предметных областей, в частности, для моделирования свойств операционной системы реального времени OpenComRTOS.

Однако, будучи важным классом моделей, логические системы, к которым относят и онтологии [7], отражают структуры наших знаний о ПрО. В данной статье мы рассматриваем возможность построения моделей ПрО при помощи математических средств, отражающих структуру ПрО как части действительности. Данный подход основан на структурировании свойств ПрО при помощи определенной метамодели, построенной на основании того или иного математического аппарата (алгебры, геометрии, теории графов и др.).

Этапы познания и свойства предметных областей и систем

Системность есть общий принцип бытия материи [8], то есть является внутренним свойством сущностей любой природы, будь то материальные или же идеальные объекты. Таким образом, свойство *системности* присуще и ПрО. Любые проблемы в рассмотрении ПрО (как и являющиеся предметом данной статьи определение и способы моделирования ПрО) связаны с недостатком системности в понимании ее свойств.

Заметим, что системность присуща также любому виду человеческой практики, включая и процесс описания, моделирования и представления ПрО в ИС. Таким образом возникает задача моделирования процесса моделирования ПрО с целью обнаружения эффективной последовательности действий, приводящих к построению адекватной действительности модели ПрО.

В контексте системного подхода основной способ моделирования ПрО мы определяем как постепенное (итерационное) увеличение системности модели ПрО при переходе от одного уровня ее описания (и понимания) к другому.

Этот процесс обусловлен противоречиями между моделью и реальной ПрО, то есть простотой и сложностью, точностью и неточностью, определенностью и неопределенностью и т.д. и направлен на максимальное приближение модели ПрО к познаваемой действительности.

С точки зрения системного подхода процесс познания ПрО состоит в следующем: вначале мы воспринимаем внешние свойства ПрО – то, что ПрО является единой и изолированной. Все, что мы можем сказать о внутренних свойствах ПрО – это присущая им разнородность: мы можем видеть, что ПрО составлена из элементов, но не можем ничего сказать о ее структуре. Лишь после понимания сущности элементов ПрО мы можем познать отношения между элементами, т.е. структуру ПрО. Именно

на этом этапе мы можем классифицировать ПрО как систему. Исходя из анализа отношений между элементами ПрО, мы можем рассмотреть ПрО как статическую или динамическую, структурную (отражающую конкретное состояние) или же функциональную (отражающую процессы перехода) систему и т.д. Данная классификация предполагает ответ на вопросы: меняются ли свойства и отношения ПрО со временем, какие события происходят в системе и какие состояния она принимает и др.

Таким образом, интерес всегда представляют собой задачи рассмотрения ПрО, как совокупности объектов (сущностей), имеющих определенные устойчивые отношения – структуру. Поэтому в целях решения возникающих над ПрО задач ее свойства должны быть определенным образом упорядочены. Для этого мы предлагаем использовать метамоделю, позволяющую структурировать свойства ПрО (и таким образом получить структурную модель ПрО).

Заметим, что в естественнонаучном моделировании любая ПрО описывается весьма небольшим количеством свойств. Признавая тезис о потенциальной бесконечности свойств предметных областей, отметим, что при математическом моделировании мы абстрагируемся от несущественных свойств ПрО на основании некоторой заданной априори информации.

Например, проблема (потенциальной) бесконечности свойств ПрО и связанное с ней отражение в ИС огромных массивов информации не является актуальной при моделировании физических систем, которое осуществляется в рамках определенной физической теории, манипулирующей ограниченным числом свойств ПрО.

Вообще говоря, моделирование структуры и поведения объекта в рамках заданной теории является типичной задачей естественнонаучного моделирования (например, расчет траектории движения тела в рамках механики Ньютона). Предлагаемый нами подход также основывается на моделировании свойств ПрО в рамках определенной математической теории, которая является основой метамоделю для порождения моделей ПрО. Особенность нашего подхода состоит в его опоре на методы современной системной инженерии, рассмотрению которых посвящен следующий раздел статьи.

Способы компьютерного моделирования предметных областей

В настоящее время в западной системной инженерии выделяются два основных направления компьютерного моделирования предметных областей:

- Domain-Specific Modeling (DSM) – предметно-ориентированное моделирование (или моделирование, ориентированное на предметные области).
- General-Purpose Modeling (GPM) – моделирование общего назначения. К GPM относятся такие языки моделирования, как UML, SysML, XML и др., позволяющие выражать знания и представлять различные аспекты систем, вне зависимости от их природы.

Также выделяют подход Model-Driven Engineering (MDE), так называемая инженерия, порождаемая из моделей, к инструментам которой, прежде всего, относят CASE-системы. Сущность MDE-подхода состоит в разработке DSM-моделей в понятиях, более близких к ПрО, а не данных и алгоритмах. Наиболее известным расширением MDE является MDA (Model-Driven Architecture) – подход, предложенный группой OMG [9].

Таким образом, GPM-подход применим к различным предметным областям, тогда как понятия DSM являются конкретными для ПрО. Сущность DSM-подхода

состоит в создании языков моделирования, специально ориентированных на решение задач определенной предметной области. Такие языки называют предметно-ориентированными языками (с англ. Domain Specific Language, DSL), в отличие от языков программирования или моделирования общего назначения.

Примерами DSL являются:

- язык бухгалтерии;
- язык банковского дела;
- язык для создания компьютерных игр;
- язык построения параллельных приложений;
- язык медицинской томографии и др.

Отличие DSL от языка программирования состоит в том, что последний может строиться в понятиях: ветвление, цикл, линейная последовательность операторов, а для поддержки ПрО – использоваться библиотека программных функций. В DSL применяют специфичные для предметной области понятия. Например, DSL-язык для конструирования одежды будет включать понятия: выкройка, линия разреза, шов, борт, ворот и др.

В общем случае DSL-языки могут быть визуальными или текстовыми. Для построения предметно-ориентированных языков используют так называемые метамодели, по существу являющиеся формализмами (или, лучше сказать, *полуформализмами*) как ER-нотация, языки онтологий, MOF и другие.

Рассмотрим, например, ER-нотацию, предложенную Петером Ченом в 1976 году [10]. ER-нотация также является отражением естественнонаучного системного подхода и акцентирует внимание на элементах и связях между элементами ПрО. Однако заметим, что связи между сущностями в ER-походе являются дескриптивными (т.е. представляют собой наименования отношений) и не отражают структуру реальной системы. Вообще говоря, применимость DSL-языков чаще всего ограничивается описанием ПрО, что не позволяет решать возникающие над ПрО задачи.

Сущность нашего подхода состоит в построении DSL на основании математических абстракций, отражающих сущность ПрО как части действительности. В данном случае метамодель позволяет отразить не только структуру ПрО, но и применить соответствующие математические методы для вычисления свойств ПрО.

В основу метамодели могут быть положены различные математические формализмы: алгебра, геометрия, теория множеств, теория графов и др. Данные формализмы определяют как модельные объекты, так и методы моделирования ПрО. Порожденный из математической метамодели DSL позволяет оперировать методами метамодели, в то же самое время находясь в рамках содержательных понятий ПрО.

В нашем подходе мы определяем такие необходимые элементы структуры модели ПрО, как свойства и методы оперирования этими свойствами.

Например, в физическом моделировании, мы говорим про величины и математические методы, в информационных системах это БД и СУБД, в базах знаний это предикаты и правила вывода, в программных системах про данные и алгоритмы и т.д.

В зависимости от видов свойств можно выделить модели, основанные на величинах и процедурах измерения, и основанные на предикатах логические модели. Отношение в нашем подходе мы рассматриваем как частный случай свойства, акцентируя внимание на методах, применимых к данным свойствам. В этом состоит иное отличие от ER подхода, акцентирующего внимание на отношениях между сущностями. Смещение акцента с взаимосвязи между элементами ПрО на дающий результат метод позволяет построить технологию моделирования ПрО в ИС.

Структура метамодели и модели ПрО

Мы выделяем вертикальный и горизонтальный уровни рассмотрения ПрО. Горизонтальный уровень отражает этапы разработки ПрО (а именно – описание, моделирование и представление ПрО в ИС), вертикальный – переход от абстрактного к конкретному в рассмотрении ПрО (рис. 1).

Как было отмечено нами выше, рассмотрение действительности как ПрО мы определяем как этап в развитии системы. Чтобы перейти в разряд систем, должны быть познаны существенные связи и отношения ПрО, т.е. структура ПрО. Для этого и предлагается использовать структурирующие свойства ПрО метамодель (вертикальный уровень на рис. 1).

Первым этапом в рассмотрении ПрО является ее описание. Заметим, что переход от описания к модели ПрО также можно осуществить при помощи определенной метамодели. Традиционно для структурирования описания ПрО используют деревья и графы (например, онтологический подход), отражающие подчиненность понятий ПрО.



Рисунок 1 – Вертикальный и горизонтальный уровни рассмотрения ПрО

Метамодель включает в себя модельные объекты и методы определенной математической теории, а также правила порождения моделей ПрО.

$$\Omega = \{ \overset{\Delta}{T}, \{C\}, \{R\} \}.$$

$\{T\}$ – множество типов для порождения объектов модели ПрО.

$\{C\}$ – множество математических методов, применимых к модельным объектам $\{T\}$.

$\{R\}$ – совокупность правил построения моделей ПрО и генерации кода.

Например, в теории графов в качестве типов мы будем использовать вершины и ребра, а методов – правила обхода, нахождения минимального пути и т.д.

Метамодель Ω используется для порождения моделей предметных областей $M_1, M_2 \dots M_N$ путем применения множества правил $\{R\}$. Именно наличие $\{R\}$ отличает метамодель Ω от математической теории.

$$\Omega \Rightarrow M_1, M_2 \dots M_N$$

Модель ПрО рассматривается нами как:

$$M = \{\{O\}, \{P\}, \{L\}\}^{\Delta}$$

$\{O\}$ – множество объектов – экземпляров $\{T\}$.

$\{P\}$ – множество заданных на $\{O\}$ свойств ПрО.

$\{L\}$ – множество законов существования или использования ПрО.

Особенность предложенной модели ПрО состоит в отделении и смещении акцента с объектов на свойства ПрО. Объекты модели являются экземплярами типов $\{T\}$, задающими структуру свойств ПрО и позволяющими применить адекватный ПрО метод.

Также смещается акцент с результата моделирования, которым становится не база данных, база знаний или информационное хранилище, а абстрактный модельный объект, предназначенный для решения определенного класса задач над ПрО. БД, БЗ, ИХ остаются важными частными случаями модельных объектов, полученных в результате моделирования предметных областей. Заметим, что в качестве результата моделирования ПрО также может быть рассмотрен и программный код, формальная (математическая) модель и др.

Применение метамодели для построения моделей программных систем

В случае применения предложенного метода для построения программных систем (ПС), процесс описания и моделирования ПрО соответствует процессу проектирования ПС: выдвижение требований и спецификаций к ПС (в общем случае, этап описания ПрО), моделирование ПС, валидация и верификация модели ПС, реализация ПС, Тестирование ПС, сопровождение и поддержка ПС.

Следуя классическому определению Никлауса Вирта [11], мы рассматриваем программу (программную систему) как структуры данных плюс алгоритмы. Поэтому в конечном счете свойства ПрО должны быть преобразованы в данные, а законы существования ПрО – вычисляться в алгоритмах ПС. Метамодель, в этом случае, задает структуры данных и методы, применимые к вычислению свойств ПрО.

В качестве примеров модельных объектов метамодели можно привести структуры данных – массивы разной размерности, списки, деревья, графы и др. Существует множество применимых к данным структурам методов, позволяющих построить модель ПрО.

Табл. 1 отражает переход от абстрактного к конкретному в процессе моделирования ПрО. Заметим также, что важным аспектом нашего подхода является преобладающая роль семантики над синтаксисом: метамодель задает синтаксис модели ПрО, модель ПрО определяет семантику ПрО.

Таблица 1 – Переход от абстрактного к конкретному в моделировании ПрО

Метамодель (синтаксис модели ПрО)	Модель (семантика ПрО)	Программная система (представление ПрО)	Исполнение программы
Множество типов $\{T\}$ – структуры данных	Свойства ПрО	Типы данных	Данные
Множество методов $\{C\}$ вычислений над $\{T\}$	Законы существо- вания ПрО	Алгоритмы	Процесс вычисления

Практические примеры применения метамоделей для построения моделей ПрО

Предложенный подход был использован нами для моделирования программных систем, где метамоделю была основана на теории графов (т.е. модельными объектами являлись узлы и ребра графа) [12]. Методы теории графов (обход, поиск минимального пути и др.) были использованы нами для генерации программного кода (путем обхода графа модели программной системы), построения таблицы маршрутизации, распределения сетевого трафика и решения других задач. Заметим, что данная метамоделю применялась нами как для построения моделей программных приложений, так и моделей топологий вычислительной сети. В случае построения программного приложения узлами графа являлись задачи (процессы) и сущности синхронизации, а ребра графа определяли взаимодействия (вызовы функций). В случае моделирования топологий в качестве узлов графа рассматривались вычислительные устройства (процессоры), а ребрами служили HW-связи между ними.

В статье [13] в качестве метамоделю для моделирования физических явлений и процессов нами было предложено использовать геометрические структуры, задающие распределения физических свойств в пространстве. Вместе с ускорением процесса моделирования путем применения технологии визуальной разработки данный подход позволил упростить использование различных математических методов вычисления физических свойств.

Геометрическая метамоделю также применялась нами для построения моделей в задачах компьютерной томографии, где модельные объекты задавали пространственные распределения интенсивности излучения (томограммы или же рентгеновские снимки). Задачей, решаемой нами в рамках данного подхода, являлось восстановление внутренней структуры трехмерного объекта по следам функций на плоских срезах данного объекта. Вообще говоря, к геометрической метамоделю может быть применено множество математических методов. Например, теорию интерфлетации и интерлинации функций [14], в которой плоское и линейное распределения свойств являются базовыми модельными понятиями.

Нами были построены также иные компьютерные инструменты, использующие метамоделю для построения моделей предметных областей. Отметим, например, визуальную среду для представления суждения о свойствах ПрО в виде векторов в логическом пространстве [15]. Здесь основой метамоделю для порождения модели ПрО является векторная алгебра. Интерес в данном подходе представляет смешение двух видов формализмов (логики силлогизмов и векторной алгебры), что позволило расширить класс решаемых над ПрО задач.

Выводы

1. Сформулирован принцип первичности ПрО по отношению к системе. Перевод ПрО в ранг системы связан с познанием существенных связей и отношений ПрО.

2. Структурирование свойств ПрО возможно на основе метамоделю, что является необходимым условием для применения математических методов решения возникающих над ПрО задач.

3. Рассмотрена применимость подхода к построению моделей и генерации кода программных систем. Рассмотрение ПС через призму метамоделю позволяет преобразовать свойства ПрО в структуры данных, а законы существования ПрО – в методы вычисления свойств.

4. Показана применимость подхода к построению моделей иных ПрО, в частности, решения задач компьютерной томографии, а также для построения системы векторной логики.

Литература

1. Берталанфи Л. Общая теория систем – критический обзор / Л. Берталанфи // Исследование по общей теории систем : Сборник. – М. : Прогресс, 1969.
2. Петров В.Н. Информационные системы / Петров В.Н. – СПб. : Питер, 2002. – 688 с.
3. Цаленко М.Ш. Моделирование семантики в базах данных / Цаленко М.Ш. – М. : Наука, 1989. – 288 с.
4. Gruber T.R. A translation approach to portable ontologies / T.R. Gruber // Knowledge Acquisition. – 1993. – 5(2). – P. 199-220.
5. Межуев В.И. Использование онтологий как моделей предметных областей / В.И. Межуев // Искусственный интеллект. – 2009. – № 4. – С. 4-11.
6. Межуев В.И. Моделирование свойств операционной системы реального времени OpenComRTOS при помощи OWL-DL онтологий / В.И. Межуев // Збірник наукових праць ДонНТУ серії «Інформатика, кібернетика та обчислювальна техніка». – 2009. – Вип. 10(153). – С. 39-46.
7. Guarino Nicola. Understanding, building and using ontologies / Nicola Guarino // International Journal of Human-Computer Studies. – Vol. 46. – Is. 2-3. – P. 293-310.
8. Перегудов Ф.И. Введение в системный анализ / Ф.И. Перегудов, Ф.П. Тарасенко. – М. : Высшая школа, 1989. – 320 с.
9. [Электронный ресурс]. – Режим доступа : <http://www.omg.org/>
10. Pin-Shan Chen Peter. The Entity-Relationship Model – Toward a Unified View of Data / Peter Pin-Shan Chen // ACM Transactions on Database Systems. – Vol. 1. – Is. 1. – P. 9-36.
11. Вирт Н. Алгоритмы + структуры данных = программы / Вирт Н. ; [пер. с англ.]. – М. : Мир, 1985. – 406 с.
12. Mezhujev Vitaliy. OpenComRTOS Visual Modelling Environment: the Tool for distributed Parallel Applications Development / Vitaliy Mezhujev, Eric Verhulst // Науковий вісник Чернівецького університету : збірник наук. праць. – Чернівці : ЧНУ, 2008. – Вип. 423 : Фізика. Електроніка. : Тематичний випуск «Комп'ютерні системи та компоненти». Частина I. – С. 88-94.
13. Межуев В.И. Метамоделль для візуального моделювання багатовимірних предметних галузей та її практичні застосування / В.И. Межуев, О.М. Литвин // Управляючі системи та машини. – 2010. – № 5.
14. Литвин О.М. Інтерлінація функцій та деякі її застосування / Литвин О.М. – Харків : Основа, 2002. – 544 с.
15. Mezhujev Vitaliy. Vector logic: theoretical principles and practical implementations / Vitaliy Mezhujev // Вісник ЗНУ : Збірник наукових статей. Фізико-математичні науки. – Запоріжжя : ЗНУ, 2006. – С. 91-97.

В.И. Межуев

Особенности компьютерного моделирования предметных областей и систем

У статті аналізуються проблеми комп'ютерного моделювання предметних областей (ПрО) і систем. Розглянуті властивості та етапи пізнання ПрО і систем, формулюється спосіб переходу предметної області в клас систем. Метод оснований на визначенні та використанні метамоделей, що структурують властивості предметних областей та дозволяють розв'язувати задачі, що виникають над ПрО.

V.I. Mezhujev

Features of Computer Modelling of Application Domains and Systems

The problems of computer modelling of application domains (AD) and systems are analyzed in the paper. Properties and stages of cognition of AD and systems are considered, the way of transition of AD into a class of systems is formulated. The method is based on use of the metamodels, which structurize properties of AD and allow to solve the tasks of AD.

Статья поступила в редакцию 26.05.2010.