
Интеллектуальные системы

УДК 004.822:004.89

О КОНЦЕПЦИИ ИНТЕГРИРОВАННОЙ МОДЕЛИ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Ю.А. Загорулько

Институт систем информатики им. А.П. Ершова СО РАН, г. Новосибирск
E-mail: zagor@iis.nsk.su

Рассматривается концепция интегрированной модели, которая объединяет взаимодополняющие методы и средства представления знаний. На базе этой модели строятся язык представления и обработки знаний и инструментальные средства, обеспечивающие в полном объеме известные на данный момент потребности разработки баз знаний прикладных интеллектуальных систем.

Ключевые слова:

Интегрированная модель представления знаний, сетевая модель, производственная модель, недоопределенная модель, императивная модель, онтология, объектно-ориентированная семантическая сеть, функциональная сеть.

Key words:

Integrated knowledge representation model, network model, production system model, subdefinite model, imperative model, ontology, object-oriented semantic network, functional network.

Введение

При создании интеллектуальной системы (ИС) самым трудоемким процессом остается разработка ее базы знаний. В связи с этим возникает насущная потребность в моделях представления знаний и построенных на их основе средствах создания баз знаний, обладающих не только высокой выразительностью и гибкостью, но и являющихся достаточно простыми и понятными в использовании как инженерам знаний – специалистам по представлению знаний, так и экспертам – специалистам в конкретных областях знаний.

В рамках работ по искусственному интеллекту накоплен широкий спектр средств и методов представления и обработки знаний. Так, семантические сети [1] используются как универсальная память для хранения любой информации, которую можно представить в терминах объектов и отношений между ними. Производственные системы [2] рассматриваются как удобное средство для выражения операционной семантики понятий предметной области и организации логического вывода. Фреймы [3], представляющие объектно-ориентированный подход в искусственном интеллекте, служат для повышения уровня представления знаний. В рамках парадигмы программирования в ограничениях (constraint programming) [4] знания можно

задавать в виде системы ограничений на значения параметров объектов предметной области. Аппарат недоопределенных типов данных и разработанный на его основе метод недоопределенных вычислительных моделей [5] дают возможность оперировать неточно заданными значениями и объектами. Большим потенциалом обладают языки логического программирования [6], позволяющие описывать сложные процессы обработки информации в декларативном стиле. В последнее время при разработке интеллектуальных систем успешно применяются дескриптивные логики [7] и построенные на их основе языки и системы представления знаний [8]. Широкую популярность у разработчиков баз знаний получили онтологии [9], с помощью которых можно достаточно просто представить непротиворечивую (согласованную) систему понятий моделируемой области знаний.

Как правило, существующие инструментальные системы и оболочки для построения ИС снабжены каким-то одним из упомянутых выше средств представления и обработки знаний. В то же время при создании прикладных ИС часто возникает необходимость их комплексного использования, поскольку ни одно из этих средств, взятое в отдельности, не может обеспечить в полном объеме потребности разработки реальной

прикладной системы. В связи с этим инструментальные средства разработки ИС должны включать различные взаимодополняющие друг друга методы и средства представления и обработки знаний. При этом возможны разные способы интеграции таких средств, начиная с интеграции на уровне подсистем и компонентов ИС и кончая интеграцией на уровне единого формализма.

Первый способ интеграции проще в реализации, но имеет ряд недостатков. Главным из них является отсутствие концептуального единства используемых методов и средств и как следствие этого – отсутствие концептуального единства создаваемых баз знаний. Это затрудняет разработку базы знаний, так как отдельные компоненты базы знаний приходится специфицировать не только с помощью разных средств представления знаний, но и использовать для этого разные программные системы, для интеграции которых в одну ИС приходится привлекать не только инженеров знаний, но и программистов. Также при данном способе интеграции приходится расходовать машинные ресурсы на организацию взаимодействия программных систем, реализующих отдельные методы и средства представления знаний.

Более предпочтительным выглядит другой способ интеграции, при котором все необходимые методы и средства объединяются в одном формализме (модели и/или языке представления знаний). Такая интеграция не только создает предпосылки для построения мощных баз знаний (за счет совместного использования различных взаимодополняющих друг друга методов и средств), но и обеспечивает концептуальное единство создаваемых баз знаний ИС, так как все компоненты базы знаний становятся связанными общими понятиями. На основе такого формализма могут быть созданы инструментальные средства, с помощью которых эксперты смогут сами строить базу знаний ИС в полном объеме, не прибегая к помощи посредников – инженеров знаний и программистов. Благодаря этому значительно упрощается разработка базы знаний ИС и повышается ее качество. Кроме того, интеграция всех требуемых средств в рамках одной модели (языка) создает предпосылки для создания на ее основе эффективных прикладных ИС, так как устраняются неизбежные расходы ресурсов на организацию взаимодействия программных систем, реализующих отдельные методы и средства представления знаний.

В статье рассматривается концепция интегрированной модели представления знаний, которая объединяет различные взаимодополняющие друг друга методы и средства представления знаний. На базе этой модели может быть построен язык представления и обработки знаний и инструментальные средства, обеспечивающие в полном объеме известные на данный момент потребности разработки баз знаний прикладных интеллектуальных систем.

Требования к интегрированной модели представления знаний

Для того чтобы модель представления знаний была пригодна для широкого класса интеллектуальных систем, она должна обеспечивать представление всех видов знаний, необходимых для разработки и функционирования ИС.

Прежде всего, в базе знаний ИС необходимо представлять знания о предметной области (ПО), в рамках которой будет функционировать система (рис. 1). Сюда входят знания об основных сущностях (понятиях и объектах) предметной области, а также знания о том, каким образом эти сущности связаны между собой. К последним относятся знания об отношениях, непосредственно связывающих понятия, а также более сложные виды знаний, представляющие различного рода зависимости между понятиями ПО, как логические, так и функциональные. В частности, при разработке большого класса ИС возникает необходимость в представлении знаний о ПО в виде правил «ЕСЛИ–ТО». Также довольно часто требуется представлять знания в виде ограничений на значения параметров, описывающих некоторый объект или систему объектов.

Для лучшего структурирования ПО и обеспечения более лаконичного ее описания важно иметь возможность выстраивания понятий предметной области в иерархию «общее–частное» и поддержки наследования свойств по этой иерархии.

Базы знаний большинства ИС также содержат конкретные знания из предметной области (предметные знания), которые представляются в виде экземпляров понятий (конкретных объектов) и связей между ними – экземпляров отношений или ограничений, заданных на значениях атрибутов экземпляров понятий.

Другим важным типом знаний, которые нужно представлять в ИС, являются знания о решаемых в рамках моделируемой ПО задачах и способах их решения (методах и алгоритмах). Эти знания характеризуют проблемную область (ПрО) интеллектуальной системы. Такого рода знания могут быть как декларативными, так и процедурными. К первому типу относятся знания, описывающие пространство решаемых ИС задач, включая разбиение задач на подзадачи и связь подзадач с методами их решения, ко второму – знания, представляющие как методы решения задач, так и конкретные алгоритмы.

Кроме знаний, представленных в БЗ, практически в любых ИС необходимо представлять знания, описывающие фрагмент действительности (ситуацию), который задает контекст, и входные данные для задач, решаемых ИС. Такие знания, как и предметные знания, обычно представляются в виде экземпляров понятий и отношений и/или ограничений, заданных на значениях атрибутов экземпляров понятий.

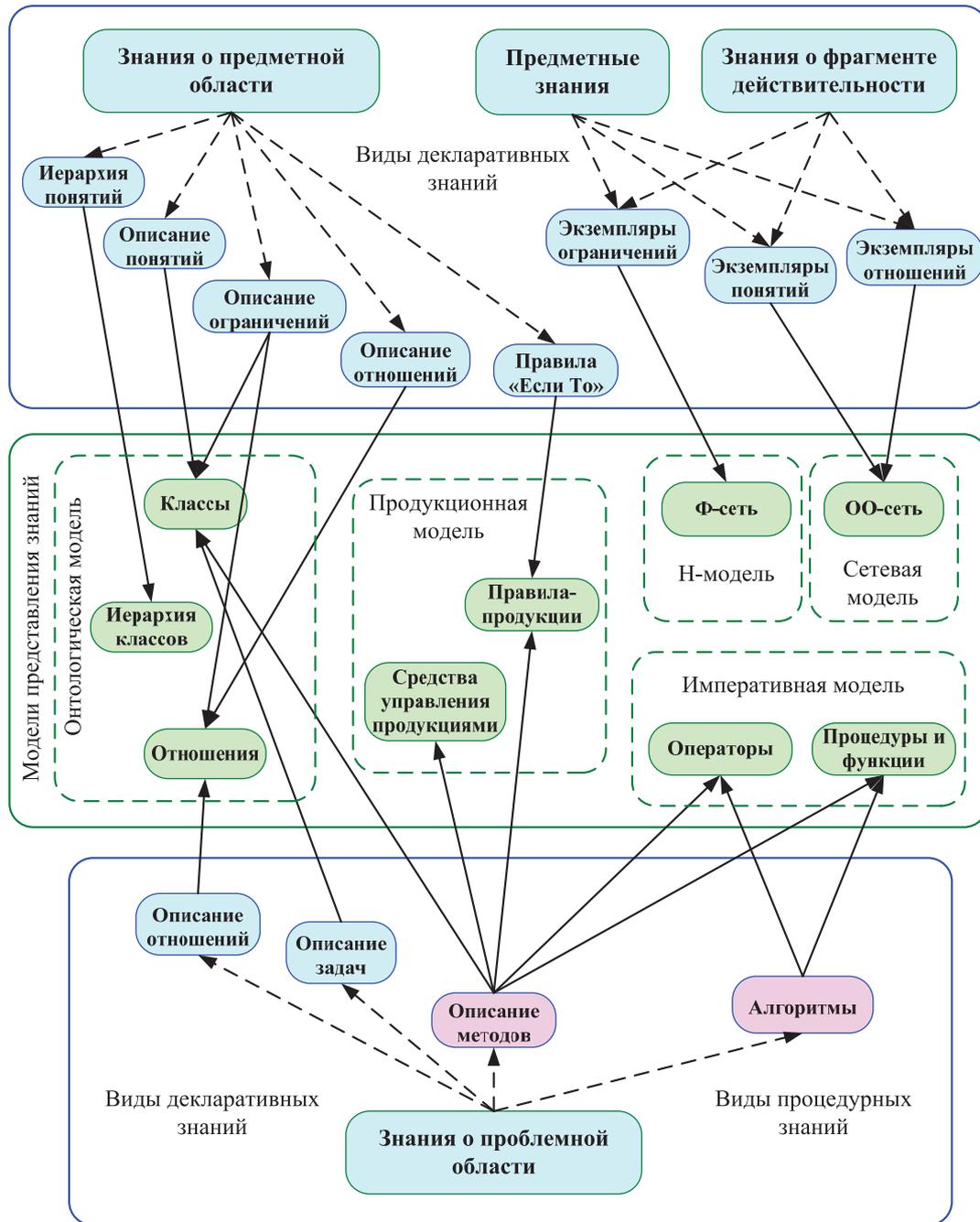


Рис. 1. Виды знаний и возможные способы их представления

При описании предметных знаний, контекста и входных данных задачи довольно часто возникает необходимость представлять знания об объектах, имеющих неточные значения атрибутов, путем задания оценок таких значений, например, в виде множеств возможных значений и/или ограничений на значения. Для представления таких знаний необходимо обеспечить возможность использования в экземплярах понятий атрибутов, имеющих недоопределенные значения, и задания экземпляров ограничений, связывающих такие атрибуты.

Рассмотренные выше виды знаний, необходимые для полноценного функционирования ИС,

не могут быть представлены средствами одной модели. Для этого требуются средства нескольких моделей. Так, для представления знаний о предметной области требуется задействовать средства двух моделей – онтологической и продукционной (рис. 1). Часть предметных знаний и знаний о фрагменте действительности может быть представлена средствами сетевой модели (в объектно-ориентированной семантической сети), другая часть – средствами недоопределенной модели (в функциональной сети или Ф-сети). Для представления знаний о проблемной области требуются средства трех моделей – онтологической, продукционной и императивной.

Таким образом, возникает необходимость в модели представления знаний, которая обладает набором методов и средств, достаточным для представления в ИС всех необходимых знаний. Прежде всего, такая модель должна обеспечивать представление как декларативных, так и процедурных знаний, что является необходимым требованием, так как без представления этих двух типов знаний невозможна разработка полноценной ИС. Кроме того, она должна обеспечивать представление и обработку как точных, так и неточных данных, что делает ее пригодной для решения задач в условиях неопределенности знаний и данных.

Состав и изобразительные возможности интегрированной модели представления знаний

Исходя из перечисленных выше требований, в данной работе предлагается концепция интегрированной модели представления знаний, объединяющей в рамках одного формализма различные взаимодополняющие друг друга средства и методы представления и обработки знаний следующих моделей – онтологической, сетевой, продукционной, императивной и недоопределенной. (Последняя включает аппарат недоопределенных типов данных и разработанный на его основе метод недоопределенных вычислительных моделей [5], который, как показано в [10], фактически является универсальным методом программирования в ограничениях.) Выше было показано, что комбинация методов и средств представления знаний, предоставляемых этими моделями, обеспечивает в полном объеме известные на данный момент потребности прикладных ИС в представлении и обработке знаний.

Интегрированная модель представления знаний (ИМПЗ-модель) может быть представлена шестеркой:

$$M = \langle ONT, SN, FN, PS, IP, NF \rangle,$$

ONT – прикладная онтология, описывающая основные понятия и отношения (классы объектов и отношений) предметной и проблемной областей приложения; *SN* – объектно-ориентированная семантическая сеть (ОО-сеть), классы объектов и отношений которой, как и их свойства, определяются онтологией *ONT*; *FN* – функциональная сеть, включающая локальные вычислительные модели, заданные на значениях атрибутов объектов семантической сети *SN*; *PS* – система правил-продукций, описанных в терминах классов и отношений онтологии *ONT*; *IP* – средства императивного программирования; *INF* – общий механизм вывода/обработки знаний и данных, представленных в ИМПЗ-модели, объединяющий механизмы вывода/обработки каждой из интегрируемых моделей.

Представление знаний о предметной области ИС и решаемых ею задачах обеспечивается прикладной онтологией.

При этом онтология включает:

- 1) конечное непустое множество классов *C*, представляющих понятия (сущности) ПО;

- 2) конечное множество бинарных отношений *R*, определенных на классах *C*;
- 3) конечное множество атрибутов *A*, описывающих декларативные свойства классов и отношений;
- 4) множество типов данных *T*, «поставляющих» значения для атрибутов;
- 5) множество ограничений *F*, заданных на атрибутах классов понятий;
- 6) множество математических свойств бинарных отношений *P*.

Все классы онтологии *C* выстроены в иерархию «общее–частное», по которой организуется наследование свойств (включая атрибуты, отношения и ограничения) вышестоящих классов нижестоящими.

Благодаря включению в ИМПЗ-модель аппарата недоопределенных типов данных все простые типы данных, входящие в *T*, имеют недоопределенные расширения, обеспечивающие возможность представления недоопределенных значений в виде интервала или множества допустимых значений и оперирования ими. Это позволяет, в частности, задавать атрибутам объектов ОО-сети недоопределенные значения.

Функциональные зависимости между атрибутами сущностей $A_i \in A$ могут задаваться в описаниях классов объектов и отношений в виде ограничений $F_j \in F$, связывающих значения атрибутов объектов. (В описаниях бинарных отношений ограничения задаются на атрибутах их аргументов.) Такие ограничения представляют собой множество логических выражений (формул), связывающих арифметическими, теоретико-множественными и логическими операциями значения атрибутов одного или нескольких объектов. Следует заметить, что благодаря включению в ИМПЗ-модель метода недоопределенных моделей с помощью ограничений задаются не только условия корректности значений связанных ими атрибутов, но и функциональные интерпретации каждого из этих значений относительно других, что обеспечивает потенциальную возможность вычисления (уточнения) неизвестных (неточных) значений атрибутов через известные (или более точные).

В зависимости от потребностей представления знаний о ПО любому бинарному отношению $R_k \in R$ может быть «приписано» одно или несколько непротиворечащих друг другу математических свойств $P_m \in P$ из следующего набора – рефлексивность, симметричность, транзитивность, антирефлексивность, антисимметричность, – а также задано обратное отношение.

Конкретные знания о ПО, как и знания о моделируемом в ИС фрагменте действительности или контексте задачи, представляются в объектно-ориентированной семантической сети *SN*, включающей множество экземпляров классов (объектов) и отношений. Для представления этого вида знаний также служат экземпляры ограничений, накладываемых на атрибуты объектов (экземпляров классов) ОО-сети и хранимые в функциональной сети, связанной с ОО-сетью через атрибуты объектов.

Так, если в описании какого-либо класса или отношения заданы ограничения, то при создании каждого его экземпляра на основе этих ограничений строится локальная вычислительная модель (ЛВМ), вершинами которой являются входящие в ограничения атрибуты объекта (в случае отношения – атрибуты его объектов-аргументов), а дугами – существующие между ними функциональные отношения.

Построенные таким образом ЛВМ всех объектов и экземпляров отношений ОО-сети объединяются в глобальную вычислительную модель, которую будем называть функциональной сетью *FN*. Для уточнения значений слотов объектов, ЛВМ которых представлены в функциональной сети, используется метод недоопределенных вычислительных моделей [5].

Описание предметных знаний в виде правил «ЕСЛИ–ТО» средствами системы продукционных правил *PS* является естественным и удобным для многих областей. В интегрированной модели система продукционных правил работает над семантической сетью, состав понятий и отношений которой определяется онтологией. В связи с этим посылка и заключение любого правила могут описываться в терминах классов и отношений онтологии. Это позволяет не только удобно представлять экспертные знания в виде правил «ЕСЛИ–ТО», но и строить довольно мощные по своим изобразительным и операционным возможностям правила и с их помощью описывать на достаточно высоком уровне довольно сложные процессы вывода и обработки информации.

Рассмотрим, какие средства предоставляет ИМПЗ-модель для представления знаний о решаемых в рамках моделируемой ПрО задачах и методах их решения.

Знания о пространстве задач могут быть представлены в прикладной онтологии, где описываются типы задач и методов, предназначенных для их решения, и задается разбиение задач на подзадачи и связь задач с методами их решения (знания второго вида также могут быть представлены средствами ОО-сети). При этом сами методы решения могут быть представлены продукционными правилами, объединенными в группы и/или продукционные модули. Для этих целей в модель включаются средства структурирования множества правил и управления ими, такие как операторы активации и деактивации групп правил, которые могут сочетаться с императивными средствами управления.

Для поддержки эффективной реализации методов и алгоритмов, а также обеспечения создания полноценных прикладных ИС в ИМПЗ-модель включены традиционные для императивной модели (парадигмы программирования) средства. Они применяются для разработки алгоритмов в случаях, когда их реализация другими методами (например, с помощью продукционных правил) была бы слишком неэффективной по времени и/или затратной по памяти, а также для реализации численных вычислений и обеспечения интерфейсных возможностей ИС.

В дополнение к традиционным императивным средствам, таким как процедуры и функции, операторы присваивания, ветвления, цикла и т. п., ИМПЗ-модель включает механизм альтернатив, который является мощным средством поддержки процессов вывода и обработки информации в условиях, когда неточно заданы не только объекты, но и отношения между ними. Данный механизм реализуется оператором альтернативы, который позволяет задавать и перебирать альтернативы (варианты вычислений), обеспечивая при возникновении противоречия во время обработки очередной из них переход к следующей альтернативе.

Интеграция указанных методов и средств выполняется на основе объектно-ориентированного подхода. При этом:

- понятия и отношения описываемых ПО и ПрО представляются классами объектов и отношений онтологии *ONT*, инкапсулирующими в себе семантические свойства и ограничения на свои атрибуты и аргументы;
- объекты (экземпляры классов) и экземпляры отношений представляются в семантической сети *SN*, структура и свойства которой определены в онтологии;
- ограничения *F* на значения атрибутов объектов представляются в функциональной сети *FN*, связанной через эти атрибуты с семантической сетью *SN*;
- все необходимые процессы вывода и обработки информации (реализующие процесс решения задачи) осуществляются системой продукционных правил *PS*, работающей над семантической сетью, механизмом удовлетворения ограничений, реализуемым в функциональной сети, а также средствами императивного программирования *IM*.

Взаимосвязи интегрируемых моделей показаны на рис. 2. Здесь видно, что сетевая и продукционная модели используют сущности, определенные в онтологической модели. В частности, структура и свойства элементов семантической сети определяются в онтологии, а посылки и заключения продукционных правил описываются в терминах классов и отношений онтологии. В свою очередь, средства оперирования, предоставляемые онтологической моделью, используют средства оперирования сетевой модели (для того, чтобы привести ОО-сеть в соответствие с онтологией). Средства оперирования продукционной модели используют и функционал сетевой модели (для сопоставления посылок правил с семантической сетью и выполнения действий над нею, определенных в правилах), и средства оперирования, заданные в императивной модели. Кроме того, сетевая модель использует средства оперирования недоопределенной модели, а последняя – сущности, определенные в сетевой модели.

Выводы

В связи с отсутствием инструментальных систем, обеспечивающих в полном объеме потребности разработчиков баз знаний интеллектуальных

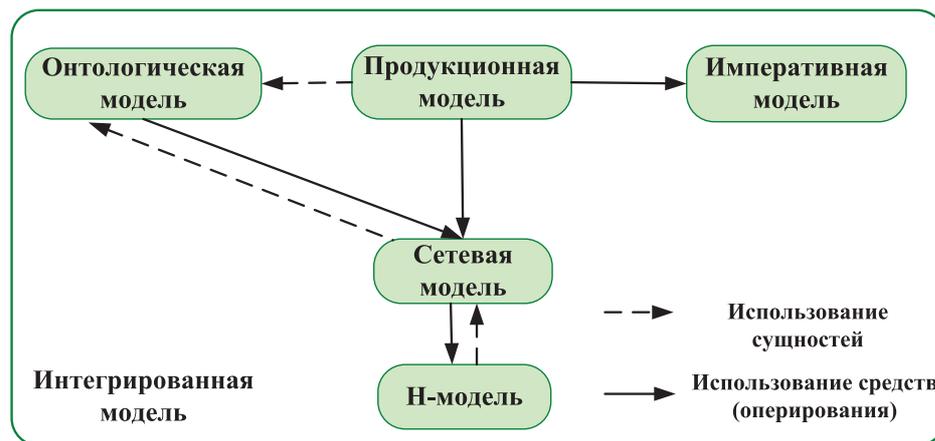


Рис. 2. Взаимодействие интегрируемых моделей

систем, в качестве базиса для представления знаний в ИС предложена модель, интегрирующая взаимодополняющие друг друга методы и средства представления и обработки знаний следующих моделей: онтологической, сетевой, продукционной, недоопределенной и императивной.

Благодаря тому, что интеграция указанных методов и средств выполнена на основе объектно-ориентированного подхода, ИМПЗ-модель позволяет представлять одновременно декларативные и процедурные знания, а также знания о вычислениях (т. е. знания, позволяющие уточнять значения параметров описываемого объекта или ситуации). Кроме того, такой способ интеграции позволяет добиться эффективной реализации интеллектуальной системы и сделать доступными все интегрируемые средства в рамках одного языка представления и обработки знаний.

На основе ИМПЗ-модели построен объектно-ориентированный язык представления и обработки знаний, который включает средства для совместно-

го описания как декларативных, так и вычислительных свойств понятий и отношений некоторой ПО, а также средства оперирования их экземплярами в продукционном стиле. Благодаря этому на таком языке можно естественным образом описывать и решать широкий класс задач, в том числе задачи, требующие сочетания логического вывода и вычислений над неточно заданными значениями.

Набор языковых и программных средств, реализующих интегрированную модель представления знаний, образует интегрированную систему представления и обработки знаний, которая послужила базисом инструментальных средств разработки интеллектуальных систем. На основе этих средств были построены оболочки, предназначенные для разработки различных типов систем, основанных на знаниях, — от оболочек информационных систем, обеспечивающих пользователю содержательный доступ к знаниям и данным определенной области знаний [11], до оболочек экспертных систем и систем поддержки принятия решений [12].

СПИСОК ЛИТЕРАТУРЫ

1. Deliyanni A., Kowalski R.A. Logic and Semantic Networks // Comm. ACM. — 1979. — V. 22. — № 3. — P. 184–192.
2. Davis R., King J. An overview of production systems // Computer Science Department Stanford University, Report STAN-CS-75-524. — 1975. — P. 38.
3. Минский М. Структура для представления знания // Психология машинного зрения. — М.: Мир, 1978. — С. 249–338.
4. Mayoh B. Constraint Programming and Artificial Intelligence // Constraint Programming. NATO ASI Series F: Computer and Systems Sciences. — 1993. — V. 131. — P. 17–50.
5. Нариньяни А.С. Недоопределенность в системах представления и обработки знаний // Изв. АН СССР. Техн. кибернетика. — 1986. — № 5. — С. 3–28.
6. Братко И. Алгоритмы искусственного интеллекта на языке Prolog. — М.: Издательский дом «Вильямс», 2004. — 637 с.
7. Baader F., Calvanese D., McGuinness D.L., Nardi D., Patel-Schneider P.F. The Description Logic Handbook: Theory, Implementation, Applications. — Cambridge: Cambridge University Press, 2003. — 574 p.
8. OWL Web Ontology Language Guide. W3C Recommendation 10 February 2004. URL: <http://www.w3.org/TR/owl-guide/> (дата обращения: 10.09.2012).
9. Guarino N. Formal Ontology in Information Systems // Proceedings of FOIS'98. — Trento, Italy, 6–8 June 1998. — Amsterdam: IOS Press, 1998. — P. 3–15.
10. Telerman V., Ushakov D. Data Types in Subdefinite Models // Art. Intell. and Symbolic Mathematical Computation, Lecture Notes in Computer Science / ed. J. Calmet et al. — Berlin: Springer, 1996. — V. 1138. — P. 305–319.
11. Загорюлько Ю.А. Программная оболочка для построения интеллектуальных научных Интернет-ресурсов // Информационные и математические технологии в науке и управлении: Труды XVII Байкальской Всеросс. конф. — Иркутск: ИСЭМ СО РАН, 2012. — Т. 3. — С. 50–56.
12. Загорюлько Г.Б., Загорюлько Ю.А. Проект открытого расширяемого инструментария для построения интеллектуальных СПИР // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS-2011): Матер. Междунар. научно-техн. конф. — Минск: БГУИР, 2011. — С. 233–236.

Поступила 14.09.2012 г.