

СПИСОК ЛИТЕРАТУРЫ

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений. — М.: Техносфера, 2005. — 1072 с.
2. Skarbek W., Koschan A. Colour image segmentation — a survey // Techn. Rep. 94–32. — Technical University Berlin. 1994. URL: <http://iristown.engr.utk.edu/~koschan/paper/coseg.pdf> (дата обращения: 09.04.2012).
3. Malik J., Belongie S., Leung T., Shi J. Contour and texture analysis for image segmentation // International Journal of Computer Vision. — 2001. — V. 43. — № 1. — P. 7–27.
4. Форсайт Д., Поппе Д. Компьютерное зрение. Современный подход. — М.: Вильямс, 2004. — 928 с.
5. Lucchese L., Mitra S. Color image segmentation: A state-of-the-art survey // Proc. of the Indian National Science Academy (INSA-A). — 2001. — V. 67. — № 2. — P. 207–221.
6. Стоянов А.К. Применение рекуррентной нейронной сети для решения задачи кластеризации // Известия Томского политехнического университета. — 2009. — Т. 315. — № 5. — С. 144–149.
7. Малинецкий Г.Г. Математические основы синергетики. — М.: Изд-во ЛКИ, 2007. — 312 с.

Поступила 11.04.2012 г.

УДК 004.652.8:004.825

ДИАГНОСТИКА МОДЕЛЕЙ БАЗ ДАННЫХ НА ОСНОВЕ ПРОДУКЦИОННОЙ СИСТЕМЫ

А.А. Мицель, В.Д. Сибилев, И.А. Красиков

Томский государственный университет систем управления и радиоэлектроники

E-mail: Krasikov86@sibmail.com

Представлена логическая структура и механизм работы проектируемой автоматизированной системы проверки моделей баз данных. Приведена формализация базы знаний продукционной системы искусственного интеллекта, используемая в синтаксическом анализе моделей баз данных. Рассмотрены примеры логического вывода на основе знаний, представленных в формализованной базе знаний продукционной системы искусственного интеллекта.

Ключевые слова:

База данных, синтаксические ошибки, семантические ошибки, логическая модель, метаданные, модель «сущность–связь», продукционная система, база знаний, правило-продукция.

Key words:

Database, syntax errors, semantic errors, logic model, metadata, model «entity–relationship», production system, knowledge base, rule–production.

Введение

Диагностика моделей баз данных на сегодняшний день является одной из важнейших задач, стоящих перед проектировщиком баз данных (БД), от выполнения которой зависит будущее всей разрабатываемой информационной системы. Процесс проектирования модели БД представляет собой логически сложную, трудоемкую задачу, требующую от проектировщика знания семантического моделирования БД и навыков в работе с инструментальными средствами построения моделей БД, называемых CASE-системами (Computer Aided Software/System Engineering).

В настоящее время специалистов в области проектирования БД готовят практически во всех вузах и иных учебных заведениях нашей страны, однако уровень знаний выпускаемых специалистов в данной области зачастую оставляет желать лучшего. Низкий уровень знаний проектировщиков БД проявляется в отсутствии навыка выделять объекты описываемой предметной области (ПО), формулировать законы взаимодействия объектов ПО друг с другом, строить на их основе модели БД, которые в конечном итоге сводятся к физическим БД. В таких моделях БД зачастую наблюдаются

различного рода аномалии обновления данных, их избыточность, потеря данных, либо в БД могут иметь место данные, не несущие никакой смысловой нагрузки. Перечисленные ошибки относятся к так называемым ошибкам нормализации БД, которые можно не допускать на этапе построения модели БД, владея методологией построения моделей БД и соблюдая применяемые к ним правила нормализации. В проектировании БД принят стандарт IDEF1X, который широко используется, начиная с 1993 г. и по сегодняшний день [1]. В стандарте IDEF1X четко и однозначно описаны все синтаксические правила построения моделей БД, на которых базируется большинство CASE-систем, однако знания методологии IDEF1X для успешного построения схем БД недостаточно, проектировщику необходимо также выявлять все функциональные зависимости (ФЗ) сущностей, вытекающие из делового регламента описываемой ПО. Правила нормализации БД, подробно описанные в книге [2], могли бы помочь студентам избежать большинства ошибок еще на этапе проектирования модели БД.

Данная статья является продолжением работ [3, 4], в которых:

- приведены основные проблемы, с которыми сталкиваются как студенты, занимающиеся выполнением курсовых работ по дисциплине «Базы данных», так и преподаватель, выполняющий проверку студенческих курсовых работ;
- изложены причины, по которым использование существующих на сегодняшний день CASE-систем нежелательно в учебном процессе;
- предложена концепция автоматизированной системы управления курсовым проектированием (АСУКП) по дисциплине «Базы данных», основной задачей которой является обучение студентов навыкам семантического моделирования баз данных с использованием методологии IDEF1X (Integration DEFINitions 1 eXpanded);
- предложена концептуальная модель автоматизированной системы проверки курсовых работ (АСПК), являющаяся подсистемой АСУКП. Задачей АСПК является проверка синтаксической правильности IDEF1X диаграмм и установление их соответствия требованиям нормализации. Основной целью АСПК является снятие с преподавателя большей части аудиторной нагрузки, уходящей на проверку IDEF1X диаграмм логических моделей баз данных (БД), выполненных студентами;

Проектируемая АСПК ориентирована на контроль знаний студентов по дисциплине базы данных, обучающихся по дистанционной технологии. Несмотря на то, что в настоящее время существует множество компьютерных систем контроля знаний (СКЗ), основывающихся на тестировании студентов по различным дисциплинам, к числу которых относится и дисциплина базы данных: ЦКТ, АСТ, WITS, CALAT, TeachLab, PROUST, LISP-Tutor, а так же веб система обучения Moodle [5], функционально они отличаются от АСПК, поскольку критерием оценки знаний студентов в данных системах является анализ ответов на задаваемые системой тестовые вопросы, а не проверка правильности практических работ студентов, основанных на построении схем БД.

Существуют также системы визуального проектирования БД, именуемые CASE-системами, к которым относятся Erwin, Vantage Team Builder, S-Designor, DataBase Designer, Design/IDEF, Designer/2000, Silverrun и PRO-IV [6]. С помощью CASE-систем можно строить схемы БД в различных графических нотациях, однако данные системы ориентированы, в первую очередь, на профессионального проектировщика БД, а не на студента. В связи с этим их использование в учебном процессе нежелательно. Кроме того, с помощью существующих на сегодняшний день CASE-систем невозможно выполнить синтаксический анализ и проверку на нормализацию построенных схем БД [7]. Но стоит отметить инструмент моделирования данных ERwin Examiner, позволяющий проверять построенные в Erwin Data Modeler схемы

БД [8], однако в данной среде присутствует автоматическая поддержка синтаксиса графической нотации IDEF1X построения схем БД, что позволяет студенту выполнять синтаксически правильные схемы БД, не зная методологии IDEF1X.

Компонентами логической модели БД являются: словарь (гlossарий) предметной области, содержащий определения сущностей, атрибутов и доменов, трехуровневая IDEF1X диаграмма пользовательских данных, состоящая из ER (Entity-Relation), KB (Key Based), FA (Full Attributes) уровней детализации предметной области [9].

Целью данной работы является выявление ключевых моментов в диагностике моделей баз данных и рассмотрение возможности диагностирования моделей баз данных с помощью АСПК. Для достижения поставленной цели необходимо решить следующие задачи, касающиеся АСПК:

- рассмотреть ключевые моменты в проверке логической модели БД преподавателем и найти их схожесть с процессом автоматизированной проверки модели;
- описать ее логическую структуру и механизм работы;
- разработать ее функциональную модель и базу знаний;
- проверить работоспособность базы знаний в оболочке экспертной системы.

Ниже рассматривается механизм работы АСПК, основывающийся на производственной системе искусственного интеллекта, разработана логическая структура метаданных, используемая для представления проектируемой студентом и хранения правильных, с точки зрения преподавателя, моделей БД, приведена ее функциональная модель.

1. Логическая структура и механизм работы автоматизированной системы проверки курсовых работ

В соответствии с вышесказанным, задачами АСПК являются:

- проверки правильности имен, определений сущностей и атрибутов, имеющих в глоссарии, заключающиеся в установлении их соответствия объектам описываемой предметной области, исключение тавтологии (рекурсии) в их определениях;
- синтаксический анализ IDEF1X диаграмм баз данных ER, KB, FA уровней;
- установление соответствия IDEF1X диаграмм FA уровня требованиям нормализации (семантический анализ);

При проверке курсового проекта непосредственно преподавателем анализ IDEF1X модели БД, описанной в проекте, основывается на знании преподавателем методологии IDEF1X и ее применении в анализе предметной области конкретной темы курсового проекта. В процессе проверки курсовой работы у преподавателя формируется собственное представление о моделируемых пользова-

тельских данных, отраженное в виде IDEF1X модели БД, которой должна соответствовать IDEF1X модель БД студента, в случае правильно выполненной курсовой работы. Установление соответствия между IDEF1X моделями БД студента и преподавателя также является критерием правильности курсовой работы при ее проверке с помощью АСПК [4].

Для каждой темы курсового проекта в АСПК существует как минимум один вариант правильной IDEF1X диаграммы, созданный преподавателем, представленный метаданными (МД), имеющими логическую структуру, удовлетворяющую синтаксису методологии IDEF1X [3].

Рабочими компонентами АСПК являются:

- метаданные модели студента (МДС) – набор таблиц базы данных соответствующей МД структуры, хранящих сведения о проектируемой студентом логической модели;
- метаданные модели преподавателя (МДП) – набор таблиц базы данных аналогичной МД и МДС структуры, хранящих реализации правильных схем БД, удовлетворяющих требованиям нормализации;
- продукционная система искусственного интеллекта, осуществляющая проверку синтаксической правильности схемы БД, выполненной студентом;

Механизм работы АСПК заключается в последовательном выполнении следующих шагов:

1. Проверка на принадлежность определений сущностей и атрибутов данных в глоссарии определений сущностей и атрибутов, имеющимся в МДП. При установлении данного соответствия имена и определения сущностей и атрибутов заносятся в МДС, а сами сущности и атрибуты связываются на основе данных определений с соответствующими сущностями и атрибутами в МДП. После чего студент выполняет построение схемы БД в графическом редакторе схем АСУКП.

2. Проверка синтаксиса построенной в АСУКП схемы БД с помощью продукционной системы искусственного интеллекта, формализация базы знаний которой приведена в разделе 2 данной статьи. Если в проверяемой продукционной системой схеме БД отсутствуют синтаксические ошибки, то данная схема считается синтаксически правильной, и сведения о ней заносятся в МДС, в противном случае студенту выдается сообщение об ошибке, и схема не фиксируется в МДС до тех пор, пока студент не исправит ошибку. Если в существующую схему БД добавляется новая сущность, не определенная ранее в глоссарии, то АСПК возвращается на шаг 1, где производится анализ на соответствие данной сущности структуре БД в МДП, таким образом, устанавливается принадлежность нового объекта модели БД к описываемой предметной области.

3. Проверка схемы БД на соответствие требованиям нормализации. Данная проверка заключается в установлении семантического соответствия схе-

мы БД в МДС схеме БД в МДП, получаемого на основе результатов SQL запросов к таблицам в МДС и МДП, связанных между собой по определениям сущностей в глоссарии на шаге 1. Если соответствие между МДС и МДП установлено, то логическая схема БД считается нормализованной, т. е. семантически правильной.

Схема БД в МДП, заданная преподавателем заранее, должна удовлетворять требованиям третьей нормальной формы (3 НФ), касательно сущностей, которые:

- не должны допускать повторяющиеся кортежи, кортежи отношений должны быть не упорядоченными, атрибуты сущностей должны быть не упорядочены и различаться по наименованию, все значения атрибутов должны быть атомарными – требования 1НФ.
- должны удовлетворять требованиям 1НФ, и в них не должно быть неполных функциональных зависимостей неключевых атрибутов от составного первичного ключа – требования 2НФ.
- должны удовлетворять требованиям 2НФ, и между всеми ее неключевыми атрибутами не должно быть транзитивных зависимостей – требования 3НФ [10].

Только удовлетворяющая перечисленным выше требованиям нормализации схема БД считается семантически правильной, так же следует отметить, что существуют требования нормализации схем БД и более высокого уровня, чем 3НФ, но в рамках данной статьи они рассматриваться не будут, поскольку на практике, как правило, достаточным оказывается соответствие проектируемой схемы БД требованиям 3НФ [11].

Формулировка задачи семантического анализа в теоретико-множественном представлении выглядит следующим образом:

Если обозначить множество реализованных в МДС схем БД как ММДС, а множество семантически правильных, с точки зрения преподавателя, схем реализованных в МДП как ММДП, то формально задачу семантического анализа схемы БД студента можно представить в виде:

$$\exists M((M \in \text{ММДС}) \wedge (M \in \text{ММДП})) \rightarrow S,$$

где M – проверяемая схема БД, а S – логический вывод, обозначающий отсутствие семантических ошибок в проверяемой схеме БД.

2. Формализация базы знаний продукционной системы искусственного интеллекта, используемой в автоматизированной системе проверки курсовых работ для синтаксического анализа схем базы данных

В статье приведена формализация экспериментальной базы знаний продукционной системы искусственного интеллекта, знания в продукционной системе представляются правилами-продукциями или продукциями, являющимися импликациями вида «если–то», имеющими обозначение:

$$\alpha \rightarrow \beta, \quad (1)$$

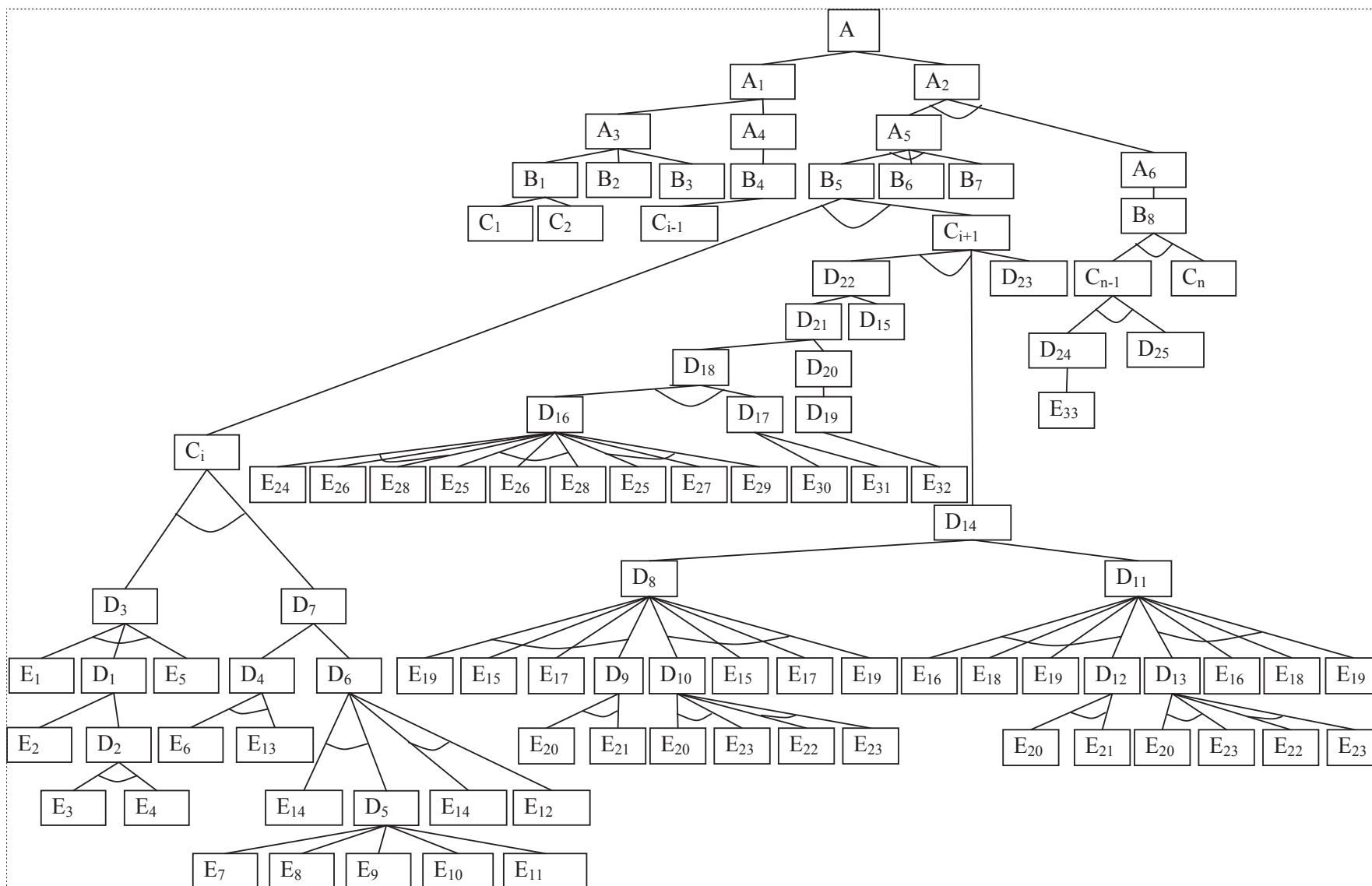


Рис. 1. Функциональная модель системы АСПК

где α – посылка; β – заключение. Для более подробного раскрытия условия применимости можно использовать выражение следующего вида:

$$P_1 \wedge P_2 \wedge \dots \wedge P_n \rightarrow B, \quad (2)$$

где P_i ($i=1,2,\dots,n$) – условия применимости, образующие конъюнктивную форму; B – заключение, которое может трактоваться как действие [12].

База знаний разделена на множества, элементы которых являются посылки и заключения, соответствующие синтаксическим правилам графотекстового языка IDEF1X описания логических моделей БД:

- E – входные данные продукционной системы.
- D – промежуточные данные, полученные из E .
- C – данные, не являющиеся прямыми следствиями из описания синтаксиса графической модели, но связанные с описанием ее семантики.
- B – выводы о наличии синтаксических и отсутствии или наличии семантических ошибок в анализируемой модели.
- A – принятые решения о принадлежности или не принадлежности анализируемой модели к множеству синтаксически и семантически правильных моделей БД.

На рис. 1 приведена функциональная модель системы АСПК.

Таблица 1. Правила-продукции продукционной системы

0) $E_7 \rightarrow D_5;$	29) $E_{25} \wedge E_{27} \wedge E_{29} \rightarrow D_{16};$
1) $E_8 \rightarrow D_5;$	30) $E_{30} \rightarrow D_{17};$
2) $E_9 \rightarrow D_5;$	31) $E_{31} \rightarrow D_{17};$
3) $E_{10} \rightarrow D_5;$	32) $E_{32} \rightarrow D_{19};$
4) $E_{11} \rightarrow D_5;$	33) $D_{16} \wedge D_{17} \rightarrow D_{18};$
5) $E_3 \wedge E_4 \rightarrow D_2;$	34) $D_{19} \rightarrow D_{20};$
6) $E_{14} \wedge E_5 \rightarrow D_6;$	35) $D_{18} \rightarrow D_{21};$
7) $E_{14} \wedge E_{12} \rightarrow D_6;$	36) $D_{20} \rightarrow D_{21};$
8) $E_2 \rightarrow D_1;$	37) $D_{21} \rightarrow D_{22};$
9) $D_2 \rightarrow D_1;$	38) $D_{15} \rightarrow D_{22};$
10) $E_6 \wedge E_{13} \rightarrow D_4;$	39) $D_{14} \wedge D_{22} \rightarrow C_{H+1};$
11) $E_{20} \wedge E_{21} \rightarrow D_9;$	40) $D_{23} \rightarrow C_{H+1};$
12) $E_{20} \wedge E_{23} \rightarrow D_{10};$	41) $C_1 \rightarrow B_i;$
13) $E_{22} \wedge E_{23} \rightarrow D_{10};$	42) $C_2 \rightarrow B_i;$
14) $E_{20} \wedge E_{21} \rightarrow D_{12};$	43) $C_{H+1} \rightarrow B_4;$
15) $E_{20} \wedge E_{23} \rightarrow D_{13};$	44) $C_i \wedge C_{H+1} \rightarrow B_5;$
16) $E_{22} \wedge E_{23} \rightarrow D_{13};$	45) $C_{N-1} \wedge C_N \rightarrow B_8;$
17) $E_1 \wedge D_1 \wedge E_5 \rightarrow D_3;$	46) $B_1 \rightarrow A_3;$
18) $D_4 \rightarrow D_1;$	47) $B_2 \rightarrow A_3;$
19) $D_6 \rightarrow D_1;$	48) $B_3 \rightarrow A_3;$
20) $E_{19} \wedge E_{15} \wedge E_{17} \wedge D_9 \rightarrow D_8;$	49) $B_4 \rightarrow A_4;$
21) $D_{10} \wedge E_{15} \wedge E_{17} \wedge E_{19} \rightarrow D_8;$	50) $B_5 \wedge B_6 \wedge B_7 \rightarrow A_5;$
22) $E_{16} \wedge E_{18} \wedge E_{19} \wedge D_{12} \rightarrow D_{11};$	51) $B_8 \rightarrow A_6;$ 52) $A_3 \rightarrow A_{1i};$
23) $D_{13} \wedge E_{16} \wedge E_{18} \wedge E_{19} \rightarrow D_{11};$	53) $A_4 \rightarrow A_{1i};$
24) $D_3 \wedge D_7 \rightarrow C_i;$	54) $A_5 \wedge A_6 \rightarrow A_2;$
25) $D_8 \rightarrow D_{14};$	55) $A_1 \rightarrow A_i;$
26) $D_{11} \rightarrow D_{14};$	56) $A_2 \rightarrow A_i;$
27) $E_{24} \wedge E_{26} \wedge E_{28} \rightarrow D_{16};$	57) $D_{24} \wedge D_{25} \rightarrow C_{N-1};$
28) $E_{25} \wedge E_{26} \wedge E_{28} \rightarrow D_{16};$	58) $E_{33} \rightarrow D_{24}$

Формализация базы знаний системы АСПК в виде правил-продукций, полученных подстановкой характеристик анализируемой модели БД в формулы (1), (2), приведена в табл. 1.

Логический вывод в данной продукционной системе является прямым по образцу (вывод управляемый данными) со стратегией разрешения конфликтного множества правил-продукций в глубину.

3. Демонстрация работоспособности базы знаний

Продемонстрируем работоспособность базы знаний продукционной системы на основе прямого вывода по образцу от исходной последовательности предпосылок E к целевому состоянию системы A . В процессе вывода будет использоваться стратегия разрешения конфликтного множества правил, соответствующая алгоритму поиска решения в глубину. Поиск решения прекращается при достижении продукционной системой одного из терминальных состояний:

1. «Решение найдено» – состояние продукционной системы, соответствующее достижению в ходе поиска по образцу заданной цели – доказывает истинность утверждения о принадлежности анализируемой модели БД к множеству синтаксически правильных моделей с точки зрения методологии IDEF1X;
2. «Решение не найдено» – терминальное состояние продукционной системы, при котором за заранее указанное количество шагов либо по завершении просмотра всей базы знаний системой не была достигнута заданная цель поиска – указывает на невозможность доказательства заданного утверждения, что говорит о возможной синтаксической ошибке в анализируемой модели БД.

В качестве входных данных продукционной системы используются параметры, соответствующие синтаксически правильному описанию зависимой именованной сущности, имеющей именованную входящую идентифицирующую связь, изображенную в соответствии с требованиями уровня «сущность–связь» представления модели данных.

Данный пример вывода по образцу был проанализирован на достоверность принятого решения с помощью программной оболочки экспертной системы «Expert system», позволяющей работать со знаниями, представленными в продукционной системе искусственного интеллекта [13].

Результаты нескольких контрольных прогонов базы знаний изображены на рис. 2–4.

Прогон 1. Исходная последовательность фактов: $E_1, E_2, E_5, E_8, E_{14}, E_{25}, E_{26}, E_{28}, E_{31}, E_{20}, E_{21}, E_{19}, E_{15}, E_{17}$ – независимая сущность, удовлетворяющая требованиям именованной сущностей, имеющая 1 или более обязательную не идентифицирующую связь, именованную в соответствии с требованиями именования связей.

Таблица 2. Прямой вывод, перебор в глубину

№ шага	Рабочая память	Конфликтное множество	Активируемое правило
0	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn	–	–
1	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn	8, 10, 13, 16, 28, 31, 45	45
2	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8	8, 10, 13, 16, 28, 31, 51	51
3	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6,	8, 10, 13, 16, 28, 31	31
4	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17	8, 10, 13, 16, 28	28
5	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16	8, 10, 13, 16, 33	33
6	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18	8, 10, 13, 16, 35	35
7	E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21	8, 10, 13, 16, 37	37
8	E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22	8, 10, 13, 16	16
9	E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13	8, 10, 13	13
10	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10,	8, 10, 21	21
11	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8	8, 10, 25	25
12	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14	8, 10, 39	39
13	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci+1	8, 10	10
14	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci+1, D4	8, 18,	18
15	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci+1, D4, D7	8	8
16	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci+1, D4, D7, D1	17	17
17	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci+1, D4, D7, D1, D3	24	24
18	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci+1, D4, D7, D1, D3, Ci	44	44
19	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci1, D4, D7, D1, D3, Ci, B5	50	50
20	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci1, D4, D7, D1, D3, Ci, B5, A5	54	54
21	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci1, D4, D7, D1, D3, Ci, B5, A5, A2	56	56
22	B7, B6, E23, E22, E15, E17, E19, E31, E25, E26, E28, E2, E1, E5, E6, E13, Cn, B8, A6, D17, D16, D18, D21, D22, D13, D10, D8, D14, Ci1, D4, D7, D1, D3, Ci, B5, A5, A2, A	–	Решение найдено

Цель поиска: B5 – доказать принадлежность модели БД, включающей описываемый графический объект, к множеству синтаксически правильных моделей, с точки зрения методологии IDEF1X.

Прогон 2. Исходная последовательность фактов: соответствует выборке данных прогона 1, но без включения факта E5 – имя сущности не занесено в глоссарий модели.

Цель поиска: B5 – доказать принадлежность модели БД, включающей описываемый графический объект, к множеству синтаксически правильных моделей, с точки зрения методологии IDEF1X.

Прогон 3. Исходная последовательность фактов: E19, E5, E17, E4, E23, E7, E10, E13, E11, E9, E8, E6, E12, E1 – случайная выборка фактов.

Цель поиска: B5 – доказать принадлежность модели БД, включающей описываемый графический объект, к множеству синтаксически правильных моделей с точки зрения методологии IDEF1X.

Следует отметить, что в ходе выполнения курсовой работы студент может построить собственный

вариант синтаксически и семантически правильной модели БД, не предусмотренный преподавателем, в таком случае преподаватель должен зафиксировать новый вариант правильной модели БД в МДП.

Заключение

Разработана логическая структура АСПК, структура базы данных, хранящей проектируемые студентом и правильные, с точки зрения преподавателя, модели БД. Предложена функциональная модель АСПК, и формализована ее база знаний в виде правил продукций, которые были протестированы в оболочке экспертной системы.

Предлагаемое в данной статье решение не имеет аналогов среди существующих на сегодняшний день CASE-систем проектирования БД, ориентированных на задачи построения синтаксически правильных схем баз данных, а не обучения студентов. Авторы полагают, что предложенное в данной статье решение задачи проверки схем баз данных перспективно и требует его дальнейшей реализации.

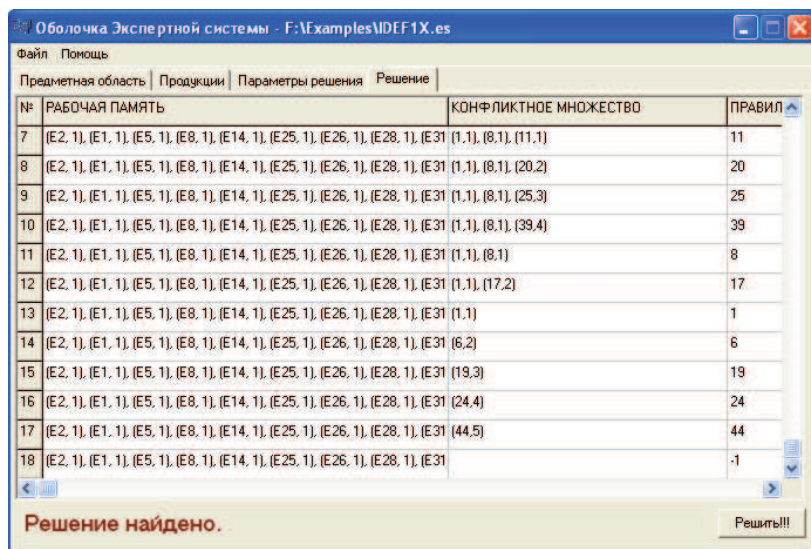


Рис. 2. Прогон выборки данных соответствующей правильному описанию объекта графической модели БД

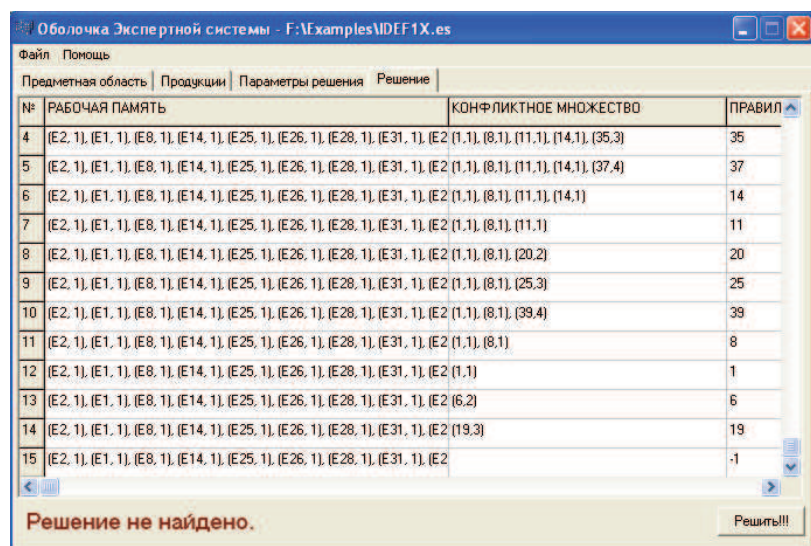


Рис. 3. Прогон выборки данных соответствующей описанию объекта графической модели БД, содержащей синтаксическую ошибку

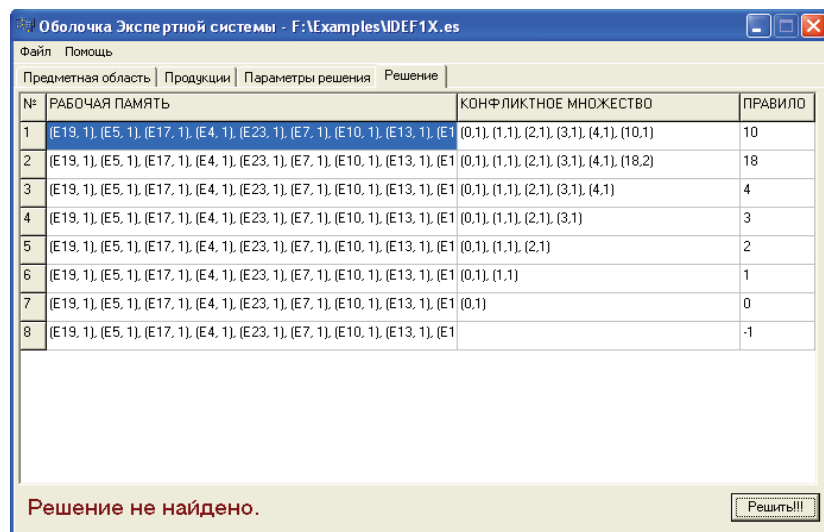


Рис. 4. Прогон выборки данных соответствующей случайному (неправильному) описанию объекта графической модели БД

СПИСОК ЛИТЕРАТУРЫ

1. Integration Definition For Information Modeling (IDEFIX) // Federal Information Processing Standards Publication 184. – 1993. – 147 p.
2. Дейт К. Введение в системы баз данных. – Киев: Диалектика, 1998. – 781 с.
3. Сибилев В.Д., Тимаков А.О. Концепции автоматизированного управления курсовым проектом по дисциплине «Базы данных» // Доклады Томского гос. ун-та систем управления и радиоэлектроники. Т. 2. Автоматизированные системы обработки информации, управления и проектирования: Сб. науч. трудов. – Томск: Изд-во Том. гос. ун-та систем управления и радиоэлектроники, 2001. – С. 29–38.
4. Красиков И.А. Автоматизированная проверка курсовых работ по дисциплине базы данных // Проблемы и перспективы образования в России: Матер. X Междунар. научно-практ. конф. – Новосибирск, 2011. – С. 97–102.
5. Романенко А.В. Контроль знаний в электронных обучающих комплексах // Доклады Томского гос. университета систем управления и радиоэлектроники. Ч. 1. Технические науки: Сб. науч. трудов. – Томск: Изд-во Том. гос. ун-та систем управления и радиоэлектроники, 2007. – С. 34–39.
6. Вендров А.М. CASE-технологии. Современные методы и средства проектирования информационных систем // ЦИТ форум. 2012. URL: <http://citforum.ru/database/case/> (дата обращения: 07.12.2012).
7. Красиков И.А. Коммуникативные и методологические проблемы обучения студентов проектированию баз данных // Тестирование в сфере образования: проблемы и перспективы развития: Матер. III Всероссийской научно-практ. конф. с междунар. участием. – Красноярск, 2010. – С. 186–190.
8. Маклаков С. Проверка качества модели данных с помощью ERwin Examiner // Журнал «КомпьютерПресс». 2012. URL: <http://compress.ru/Article.aspx?id=9437> (дата обращения: 07.12.2012).
9. Кузнецов С. Развитие логических моделей данных // ЦИТ форум. 2012. URL: http://citforum.ru/database/articles/ref_vs_nav_models/. (дата обращения: 07.12.2012).
10. Пушников А.Ю. Введение в системы управления базами данных. Ч. 2. Нормальные формы отношений и транзакции. – Уфа: Изд-во Башкирского ун-та, 1999. – 138 с.
11. Кузнецов С. Базы данных: достижения и перспективы на пороге XIX столетия // ЦИТ форум. 2012. URL: http://citforum.ru/database/classics/nfs_report/ (дата обращения: 07.12.2012).
12. Геловани В.А., Башлыков А.А., Бритков В.Б., Вязилов Е.Д. Интеллектуальные системы поддержки принятия решений в нештатных ситуациях с использованием информации о состоянии природной среды. – М.: Эдиториал УРСС, 2001. – 304 с.
13. Павлов С.Н. Системы искусственного интеллекта. В 2-х частях. Ч. 1. – Томск: Эль Контент, 2011. – 176 с.

Поступила 18.07.2012 г.

УДК 004.421

АДАПТИВНЫЙ АЛГОРИТМ ОБРАБОТКИ ПОТОКА НАВИГАЦИОННЫХ ДАННЫХ НА ОСНОВЕ МЕТОДА ДИАГНОСТИЧЕСКОЙ ФИЛЬТРАЦИИ

С.А. Хруль, Д.М. Сонькин

Томский политехнический университет
E-mail: siberianodis@gmail.com

Рассмотрена задача обработки потока навигационных данных для повышения достоверности и уменьшения объема исходных данных. С учетом анализа существующих подходов разработан адаптивный алгоритм отсева избыточных и ошибочных данных, работающий в режиме реального времени, основанный на диагностической фильтрации, позволяющий уменьшить поток навигационных данных, а также существенно снизить нагрузку на канал передачи данных. Приведен пример практического использования в рамках действующей системы мониторинга транспорта.

Ключевые слова:

Фильтрация навигационных данных, методы фильтрации данных, мобильные объекты, спутниковая навигационная система.

Key words:

Filtering navigation information, data filtering methods, mobile objects, navigation system.

Современным системам спутникового мониторинга подвижных объектов, использующим терминальное оборудование на мобильных объектах, необходима высокая достоверность принимаемых навигационных данных, достигнуть которой без использования различных алгоритмов обработки принимаемой информации не представляется возможным. Одним из решений, позволяющим повысить достоверность и уменьшить объем навигационной информации, передаваемой пользователю, является фильтрация ложных и избыточных данных, полученных от ГЛОНАСС/GPS модуля,

входящего в состав аппаратуры мобильного терминала [1].

GPS/ГЛОНАСС модуль принимает сигналы со спутников и после их обработки формирует сообщение, включающее ряд параметров о состоянии мобильного объекта, а также навигационную информацию. Производители GPS/ ГЛОНАСС-приемников используют различные алгоритмы окончательной обработки спутниковых сигналов, а также различные протоколы и форматы передачи навигационных сообщений терминалу. Наиболее распространенным форматом является NMEA-