

ИСПОЛЬЗОВАНИЕ ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ЛОГИЧЕСКОГО ПРОГРАММИРОВАНИЯ ДЛЯ СОЗДАНИЯ ЭКСПЕРТНЫХ СИСТЕМ В РАЗЛИЧНЫХ ПРЕДМЕТНЫХ ОБЛАСТЯХ

Жуков В.В.¹, Новикова Г.М.¹, Гитарская Л.М.¹, Кузнецова А.В.¹

¹ФГБОУ ВПО «Российский университет дружбы народов», Москва, Россия (117198, г. Москва, ул. Миклухо-Маклая, д. 6), e-mail: zhukpm@gmail.com

Рассматривается инструментальная среда разработки и технология проектирования экспертных систем. Обосновывается использование парадигмы логического программирования в качестве основы для создания экспертных систем, рассматриваются преимущества, а также примеры использования инструментальной среды логического программирования Visual Prolog для создания экспертных систем в таких предметных областях, как управление персоналом и юриспруденция. На примерах конкретных экспертных систем демонстрируются достоинства инструментов логического программирования, в частности наличие встроенного механизма логического вывода; возможность поддержки как логико-лингвистических, так и математических моделей; возможность работы со сложно структурированными терминами и лингвистическими переменными; интеграция парадигм логического и объектно ориентированного программирования; единый синтаксис языка, который используется как для описания механизмов работы с базой знаний, так и для создания графических интерфейсов экспертной системы, позволяющих оптимизировать процесс создания экспертных систем.

Ключевые слова: экспертные системы, логическое программирование, представление знаний, базы знаний, Visual Prolog, система управления персоналом, правовые системы.

THE USE OF LOGIC PROGRAMMING TOOLKIT FOR DIFFERENT KNOWLEDGE DOMAIN EXPERT SYSTEMS DEVELOPMENT

Zhukov V.V.¹, Novikova G.M.¹, Gitarskaya L.M.¹, Kuznetsova A.V.¹

¹Federal State Budget Educational Institution of Higher Professional Education "Peoples' Friendship University of Russia", Moscow, Russia (117198, Moscow, Miklukho-Maklaya street, 6), e-mail: zhukpm@gmail.com

This article is dedicated to expert systems development, specifically to designing technologies and creation toolkit. In the context of this article the experience of several authors in using logic programming toolkit for development expert systems for such domains as jurisprudence and personnel management is being summarized. The paradigm of logic programming as the basis of expert systems development is being justified. Several advantages of modern logic programming environment are being described: such as built-in inferential mechanism, both linguistic and mathematical models support, both complexly structured terms and linguistic variables processing, integration of logic and object-oriented programming paradigms, unified language syntax both to describe the mechanism of work with knowledge base and to create the graphic interface of the expert system. The advantages of development toolkit Visual Prolog are shown by the example of definite expert systems. Visual Prolog not only reduces expenses of production the expert system, but also simplify the development technology, shifting the system development process on to the knowledge engineer in this field.

Keywords: expert systems, logic programming, knowledge representation, knowledge bases, Visual Prolog, personnel management system, legal systems.

В настоящее время экспертные системы (ЭС) получили широкое распространение, и сегодня предметной областью их внедрения становится социально-экономическая и правовая сферы. Разрабатываются ЭС для автоматизации процесса создания договоров в сфере недвижимости, генерации совокупности требований в системе подбора персонала, проектируются экспертные системы, заменяющие банковского служащего или консультанта-продавца. Расширение области применения ЭС и спектра решаемых ими задач обусловлено совершенствованием экспертных знаний в области психологии, социологии,

юриспруденции, экономики, лингвистики. Однако создание ЭС связано с большими затратами не только на формирование знаний о предметной области, но и на разработку программных компонент, обрабатывающих эти знания. В то же время использование для создания ЭС инструментальных средств логического программирования позволяет сократить затраты и оптимизировать процесс разработки ЭС. Далее рассматриваются преимущества использования как принципа, так и инструментов логического программирования для создания ЭС в различных предметных областях.

Принцип логического программирования рассматривает процесс программирования как описание декларативных утверждений о предметной области, а вычисление как поиск определенных следствий из имеющегося описания. Описание задачи представляет собой совокупность утверждений на некотором формальном логическом языке, получение решения - логический вывод в некоторой формальной (аксиоматической) системе. В узком смысле в логическом программировании для описания предметной области используется логика предикатов первого порядка (ЛППП), а в качестве метода логического вывода (МЛВ) используется метод резолюций, основанный на доказательстве теорем через поиск противоречий.

Создание ЭС в среде логического программирования состоит в описании предметной области с использованием синтаксиса языка, поддерживаемого инструментальной средой, а также в определении алгоритмов работы интерфейса системы. Интерфейс должен обеспечивать открытость ЭС – возможность интерактивного расширения описания предметной области, а также активное участие человека (эксперта/пользователя) в процессе решения поставленной задачи. В частности, ЭС должна не только выдавать экспертную оценку, но и показывать путь рассуждения, ведущего к результирующему умозаключению.

Основой ЭС является база знаний (БЗ), которая состоит из двух частей – экстенциональной и интенциональной [1]. В рамках парадигмы логического программирования экстенциональная часть базы знаний хранит множество фактов о предметной области, которые описываются в терминах языка исчисления предикатов первого порядка набором атомарных аксиом с константным значением термов и составляют базу фактов логической программы. Интенциональная часть базы знаний хранит описание объектов предметной области, их связей и свойств, выраженных (в рамках парадигмы логического программирования) в виде правильно построенных выражений ЛППП специального вида, называемых хорновскими дизъюнктами, которые составляют базу правил логической программы.

Общая архитектура ЭС, созданная инструментальными средствами логического программирования, приведена на рис. 1.

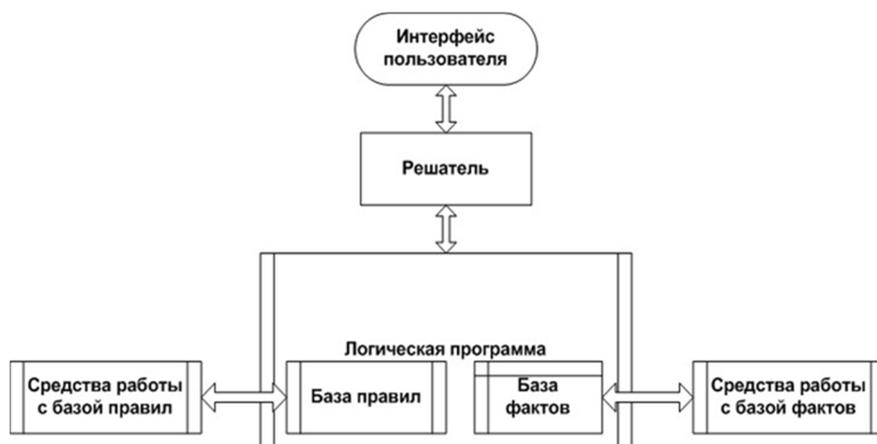


Рис. 1. Общая архитектура системы

Реализация ЭС в инструментальной среде логического программирования имеет ряд преимуществ. Рассмотрим их более подробно на примере использования интегрированной среды логического программирования Visual Prolog 7.4 [7].

Главным преимуществом создания ЭС в среде логического программирования является наличие встроенного механизма дедуктивного вывода, что определяет эффективность и результативность работы ЭС только полнотой и непротиворечивостью экспертных знаний о предметной области, составляющих БЗ логической программы. Рассмотрим упрощенный пример. Пусть в БЗ ЭС определен шаблон факта, сигнализирующего о наличии у некоторого человека того или иного симптома. Данный шаблон возможно представить в соответствии с листингом 1.

Листинг 1. Шаблон факта наличия симптома

имеет_симптом (человек Человек, симптом Симптом).

Кроме того, в БЗ определено правило относительно заболевания «грипп». Считаем, что человек болеет гриппом, если у него наблюдается некоторое множество симптомов. На языке Visual Prolog это можно записать в соответствии с листингом 2.

Листинг 2. Правило относительно заболевания «грипп»

болеет (Человек, грипп) :-

имеет_симптом (Человек, боль_в_горле),

имеет_симптом (Человек, озноб),

имеет_симптом (Человек, потливость).

Тогда дальнейшая работа с построенной системой сводится исключительно к определению фактов, имеющих место в реальной жизни. Например, при наличии фактов, указанных в листинге 3, возможно автоматически при помощи МЛВ получить диагноз, приведенный в листинге 4.

Листинг 3. Множество определенных фактов

имеет_симптом (Иван, боль_в_горле),

имеет_симптом (Иван, озноб),

имеет_симптом (Иван, потливость).

Листинг 4. Запрос и результат его выполнения

болеет (Иван, X),

X = грипп.

Рассмотренные факты и правило представляют собой часть БЗ ЭС, и в то же время, в рамках инструментальной среды Visual Prolog, это фрагмент специализированной базы данных. Здесь проявляется еще одно существенное преимущество Visual Prolog – система способна решать задачи, связанные с управлением базами данных. В ней развита работа с указателями и шаблонами данных, присутствуют механизмы поддержки как внутренней, так и внешней базы данных (в том числе загрузка и выгрузка текущего состояния системы, описываемого множеством фактов предметной области). Последнее свойство позволяет при необходимости производить интеграцию системы, написанной на Visual Prolog, с другими языками и системами программирования. Кроме того, единый синтаксис языка, который используется как при формировании, обработке и выборке данных, так и при описании правил предметной области, дает возможность быстрой разработки исходных кодов, поскольку механизмы управления БЗ системы Visual Prolog также описываются в предикатной форме.

Современные интегрированные среды разработки программ на основе логического программирования (в частности, Visual Prolog 7.4) имеют встроенные средства создания графических интерфейсов для взаимодействия ЭС с экспертом и пользователем. Ввиду единообразия синтаксиса данные средства также описываются в предикатной форме, что позволяет осуществлять глубокую интеграцию интерфейса и БЗ ЭС. Существенным преимуществом является, в частности, возможность автоматического создания и построения интерфейсов ЭС на основе данных из БЗ.

Еще одно преимущество использования систем логического программирования заключается в их постепенной интеграции с другими парадигмами программирования. Объектно ориентированная составляющая позволяет разработчикам ЭС использовать комплексные модели для представления знаний [3]. Конструкции структурного программирования «если-то-иначе», «для-каждого» дают возможность программистам осуществлять более быструю разработку исходных кодов. Пример выполнения некоторой операции над всеми симптомами человека приведен в листинге 5.

Листинг 5. Использование структурной составляющей Visual Prolog

foreach имеет_симптом (Иван, X)

% Некоторое действие

end foreach

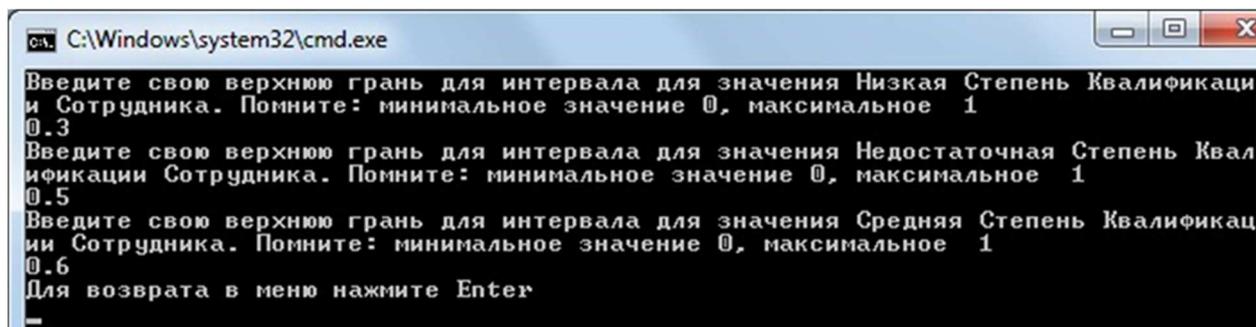
Следующие преимущества создания ЭС в среде логического программирования рассмотрим на примере разработки ЭС оценки персонала компании (ЭСОП) средствами Visual Prolog [4]. ЭСОП предназначена для оценки таких характеристик, как квалификация, надёжность и лояльность персонала компании, причем объектом анализа может быть как отдельный сотрудник, так и множество сотрудников, составляющих проектную группу, отдел, департамент, всю компанию в целом. Систему можно отнести к классу ЭС диагностического типа, однако ЭСОП может не только выводить результирующее умозаключение по оценке, но и выдавать рекомендации для принятия эффективных решений в области управления персоналом. Очевидно, что предлагаемые решения носят рекомендательный характер, оставляя право принятия решений за человеком. В то же время в ЭСОП реализован режим работы в качестве информационно-справочной системы.

Работа ЭС базируется на специальных знаниях о предметной области [5], для представления которых необходимо использовать как математическую, так и логико-лингвистическую модель представления знаний. Математическая модель необходима для расчета числовых значений показателей, в то время как логико-лингвистическая модель позволяет формировать экспертные заключения на естественном языке. Важными компонентами ЭС являются блоки обучения и пополнения знаний, а также компонента, позволяющая настраивать коэффициенты математических и логико-лингвистических моделей.

Множество фактов (экстенциональная составляющая БЗ) представляют собой значения показателей, оценивающих работу как отдельного сотрудника, так и группы сотрудников. Множество правил (интенциональная составляющая БЗ) включают правила, описывающие математические, а также логические связи между показателями и умозаключениями.

Функциональность ЭСОП позволяет:

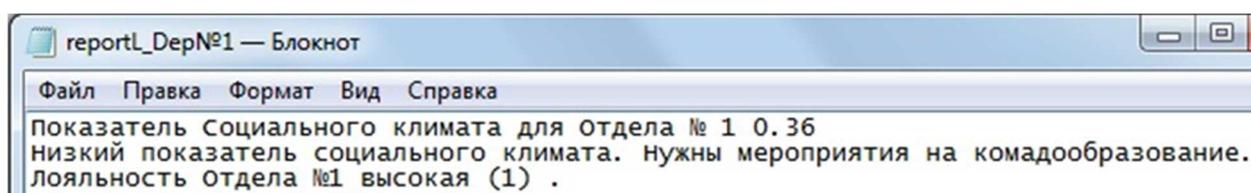
- 1) производить расчеты значений обобщенных показателей, таких как квалификация, лояльность, надежность;
- 2) настраивать значения лингвистических переменных – определять интервалы числовых значений для таких показателей, как «низкая квалификация сотрудников отдела», «высокая лояльность персонала компании» и т.д. (рис. 2);



```
C:\Windows\system32\cmd.exe
Введите свою верхнюю грань для интервала для значения Низкая Степень Квалификации
Сотрудника. Помните: минимальное значение 0, максимальное 1
0.3
Введите свою верхнюю грань для интервала для значения Недостаточная Степень Квал
ификации Сотрудника. Помните: минимальное значение 0, максимальное 1
0.5
Введите свою верхнюю грань для интервала для значения Средняя Степень Квалификац
ии Сотрудника. Помните: минимальное значение 0, максимальное 1
0.6
Для возврата в меню нажмите Enter
=
```

Рис. 2. Пример настройки лингвистических переменных

- 3) диагностировать состояние объектов организационно-функциональной структуры, таких как сотрудник, отдел, проектная группа, компания в целом по различным признакам идентификации в терминах лингвистических переменных;
- 4) вырабатывать рекомендации на основе результатов диагностики (рис. 3).



```
reportL_Dep№1 — Блокнот
Файл  Правка  Формат  Вид  Справка
Показатель Социального климата для Отдела № 1 0.36
низкий показатель социального климата. Нужны мероприятия на коадообразование.
Лояльность Отдела №1 высокая (1) .
```

Рис. 3. Фрагмент работы системы ЭСОП

Перечислим преимущества использования для создания ЭСОП инструментальной среды логического программирования Visual Prolog.

1. Язык включает набор встроенных предикатов, например $=$, SUM, $*$, которые позволяют реализовать в рамках ЭСОП математическую модель для расчета значений показателей в предметной области.
2. Visual Prolog имеет встроенные механизмы работы с логико-лингвистической моделью, позволяющие расширять описание предметной области в процессе логического вывода.
3. Среда логического программирования дает возможность интегрировать в рамках логической программы математическую и логико-лингвистическую модель, что позволяет в рамках единой инструментальной среды производить как математические расчеты, так и осуществлять дедуктивный вывод.

Сегодня востребованной областью применения ЭС становится юриспруденция. ЭС приходят на смену справочно-правовым системам, получившим широкое распространение с начала 80-х годов [2]. Среди основных целей построения и использования ЭС в области юриспруденции можно выделить определение полноты и непротиворечивости законодательной базы (ЗБ). Будем понимать под полнотой ЗБ такое ее системное построение, когда все аспекты некоторой ситуации в полной мере регулируются законом, а схожие ситуации и понятия одинаково регулируются некоторым набором нормативно-правовых актов (НПА). Под непротиворечивостью ЗБ понимается отсутствие таких ситуаций, когда

необходимость применения различных НПА влечет необходимость применения несовместных действий.

Эффективной с точки зрения поставленных задач и адекватной по выразительной мощности моделью представления знаний для таких ЭС является ЛППП. Это обуславливается следующими допущениями: каждый НПА можно рассматривать как совокупность норм права (НП); структура НП включает в себя гипотезу, диспозицию и санкцию и говорит о следующем: если выполнена гипотеза, то должна быть выполнена и диспозиция, иначе должна быть применена санкция. Указанную структуру можно представить следующим образом:

гипотеза → *диспозиция*

гипотеза ∧ ¬*диспозиция* → *санкция*

Выполнимость гипотезы определяется набором юридических фактов. Если в текущей ситуации каждый из них истинен, то истинна и гипотеза, а значит необходимо применить соответствующую НП. Данное заключение описывается следующей формулой:

$\text{Факт}_1 \wedge \text{Факт}_2 \wedge \dots \wedge \text{Факт}_n \rightarrow \text{гипотеза}$

Использование МЛВ позволяет определить непротиворечивость множества построенных формул, и, следовательно, непротиворечивость всей ЗБ. Нахождение НП, противоречащих друг другу, осуществляется на основе анализа работы МЛВ: работу МЛВ можно представить в виде дерева, узлы которого содержат некоторые факты и правила, которые к ним применяются и порождают новые факты; если разделить факты на изначальные и выведенные, то при получении противоречия, поднимаясь по дереву вверх до уровня изначальных фактов, можно определить и правила, которые фундаментально привели к данному противоречию.

Проблему полноты ЗБ в рамках модели ЛППП можно разрешить с использованием сложно структурированных термов. Используя функции (функциональные символы в рамках ЛППП), доменными значениями которых являются схожие понятия и ситуации, можно сделать предположение, что если существует правило, регулирующее некоторое доменное значение, то должно существовать и аналогичное правило, регулирующее либо каждое другое доменное значение отдельно, либо весь домен значений в целом. Однако проведение такого анализа без участия эксперта может привести к накоплению в БЗ фактов, не соответствующих действительности.

Использование современных инструментальных средств логического программирования для создания ЭС в области юриспруденции также имеет ряд ключевых преимуществ.

1. Интеграция с другими парадигмами программирования позволяет строить и использовать комплексные модели представления знаний, в частности гибридную модель на основе ЛППП и системы фреймов.
2. Декларативный характер написания программ (в том числе пополнения БЗ) позволяет экспертам предметной области заниматься модернизацией системы без участия программистов; при этом не требуется специализированных инструментальных средств для создания интерфейса ЭС-эксперт.
3. Возможность эффективной реализации методов работы с текстами, в частности юридического характера, описанными в [6].

Таким образом, современный уровень развития инструментальных средств логического программирования позволяет сегодня создавать ЭС в различных предметных областях. Использование в основе организации экспертных систем принципа логического программирования, а в качестве инструментальной среды разработки таких языков, как Visual Prolog, не только снижает издержки на создание ЭС, но и упрощает технологию ее разработки, переключая основной процесс создания системы на инженера по знаниям в данной предметной области.

Список литературы

1. Вагин В.Н. Новикова Г.М. Инструментальная система ДИССПО для создания интегрированных экспертных систем // Материалы семинара «ЭВМ новых поколений и перспективы их использования в народном хозяйстве». – М. : МДНТП, 1989. – С. 59-64.
2. Васильев Д. Компьютер вместо юрисконсульта // Наука и жизнь. – 2001. – № 3.
3. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. – СПб. : Питер, 2000. – С. 37-38.
4. Гитарская Л.М., Новикова Г.М. К вопросу о применении экспертных систем в системе управления персоналом // Труды XLV Всероссийской конференции по проблемам математики, информатики, физики и химии. – РУДН, 2012. – С. 137-139.
5. Новикова Г.М. Оценка компетентности персонала в корпоративной системе управления // Компетенции и компетентность современного специалиста. – М. : Граница, 2011. – С. 78-95.
6. Савинич Л.В., Стефанюк В.Л. Выражение обусловленности в естественном языке // Информационные технологии и вычислительные системы. – 2008. – № 1. – С. 30-37.
7. Страница среды логического программирования Visual Prolog 7.4 [Электронный ресурс]. – URL: <http://www.visual-prolog.com/> (дата обращения: 13.05.2014).

Рецензенты:

Стефанюк В.Л., д.т.н., профессор, главный научный сотрудник лаборатории искусственного интеллекта Института проблем передачи информации, вице-президент Российской ассоциации искусственного интеллекта, г. Москва.

Хачумов В.М., д.т.н., профессор, заведующий лабораторией «Методы интеллектуального управления» ФГБУН «Институт системного анализа» Российской академии наук, г. Москва.