

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Кафедра «Информационные технологии»

СИСТЕМЫ, ОСНОВАННЫЕ НА ЗНАНИЯХ

ПОСОБИЕ

**по курсу «Автоматизация принятия
управленческих решений» для студентов
экономических специальностей
дневной и заочной форм обучения**

Гомель 2009

УДК 004.895(075.8)
ББК 32.813я73
С40

*Рекомендовано научно-методическим советом
факультета автоматизированных и информационных систем
ГГТУ им. П. О. Сухого
(протокол № 7 от 09.03.2009 г.)*

Составитель: *Н. В. Водополова*

Рецензент: доц. каф. «Промышленная электроника» ГГТУ им. П. О. Сухого *В. А. Карнов*

С40 **Системы**, основанные на знаниях : пособие по курсу «Автоматизация принятия управленческих решений» для студентов экон. специальностей днев. и заоч. форм обучения / сост. Н. В. Водополова. – Гомель : ГГТУ им. П. О. Сухого, 2009. – 38 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <http://lib.gstu.local>. – Загл. с титул. экрана.

Пособие посвящено проблемам разработки систем искусственного интеллекта – одной из самых динамично развивающихся в последнее время отраслей научных исследований, имеющих большое практическое значение.

Для студентов экономических специальностей дневной и заочной форм обучения.

УДК 004.895(075.8)
ББК 32.813я73

© Водополова Н. В., составление, 2009
© Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», 2009

Искусственный интеллект

Искусственный интеллект – одно из самых молодых научных направлений. Прошло всего несколько десятилетий с момента начала работ в этой области. Но сегодняшний мир и научно-технический прогресс немыслимы без тех результатов, которые дали человечеству и обещают дать в будущем исследования в области искусственного интеллекта.

Причин столь исключительного положения работ в области искусственного интеллекта несколько. Наиболее важными из них являются следующие:

1. *Развитие вычислительной техники вступило в новую фазу: создание ЭВМ пятого и последующих поколений.* Отличительной чертой новых ЭВМ является их максимальная приближенность к пользователю, освобождение пользователя от программирования решения задач. Функции программиста передаются ЭВМ, сложность общения с ней не должна превосходить сложности общения с современными бытовыми системами. Для этого необходимо поднять «интеллектуальный» уровень ЭВМ, сделав ее способной к выполнению творческого профессионального труда программиста. В ЭВМ должна быть заложена большая сумма знаний о способах решения задач, специальные процедуры автоматического синтеза программ, а также средства общения с пользователем, максимально приближающие это общение к общению людей.

2. *XXI век – век новых информационных технологий.* В связи с массовым внедрением ЭВМ во все сферы человеческой жизни, развитием локальных, глобальных, национальных и межгосударственных сетей передачи и обработки данных, быстрым ростом хранилищ информации, распределенных в различных местах земного шара и доступных для любой ЭВМ, включенной в соответствующую сеть, станет возможным переход к

безбумажной технологии обработки информации. А это повлечет за собой изменение стиля человеческого общения в самых разнообразных деловых и бытовых взаимоотношениях.

3. *Технология производства в промышленности и сельском хозяйстве меняет свою основную парадигму.* Если раньше она ориентировалась на человека как основное звено в трудовом процессе, то роботизация производства заставляет по-новому оценить организацию трудового процесса. В роботизированных производствах нет необходимости в создании «человеческих условий» для производителей, в выполнении требований к агрегатам, климатическим условиям и т.п., которые определяются эргономикой и физиологией человека.

4. *Интеллектуальные системы в ближайшие годы должны сыграть революционизирующую роль в научных исследованиях и проектировании новых образцов новых изделий.* Интеллектуальные системы являются тем инструментом, без которого невозможно проектирование сверхсложных для человека изделий. Более того, именно интеллектуальные системы позволят ученым в приемлемые сроки решать те научные проблемы, на решение которых не хватает сейчас жизни нескольких поколений.

Очевидно, что это не все причины, побудившие бурное развитие нового научного направления, получившего название «искусственный интеллект», но и они дают представление об его глобальной значимости.

Единого определения, полностью описывающего эту научную область, нет. Наиболее традиционный взгляд на понятие «искусственный интеллект», отражает следующее определение.

Искусственный интеллект – это одно из направлений информатики (кибернетики), цель которого разработка аппаратно-программных средств, позволяющих пользователю-непрограммисту ставить и решать свои

задачи, традиционно считающиеся интеллектуальными, общаясь с ЭВМ на ограниченном подмножестве естественного языка.

Идеи создания искусственного подобия человеческого разума для решения сложных задач и моделирования мыслительной способности появились еще в глубокой древности. Впервые ее развил Р.Луллий, который еще в XIV в. пытался создать машину для решения различных задач на основе всеобщей классификации понятий. В XVIII в. Г.Лейбниц и Р.Декарт независимо друг от друга развили эту идею, предложив универсальные языки классификации всех наук. Эти идеи легли в основу теоретических разработок в области создания искусственного интеллекта.

Однако развитие искусственного интеллекта как научного направления стало возможным только после создания ЭВМ и появления новой науки – кибернетики.

Термин «*искусственный интеллект*» впервые был предложен в 1956 г. на научном семинаре в Станфордском университете, посвященном проблемам разработки логических задач.

В искусственном интеллекте, как отрасли науки, существует два направления:

- нейрокибернетика;
- кибернетика «черного ящика».

В настоящее время наметилась тенденция к их объединению.

Основная идея *нейрокибернетики* базируется на том факте, что единственный объект, способный мыслить, – это человеческий мозг. Отсюда нейрокибернетика ориентирована на *аппаратное моделирование* структур, подобных структуре мозга. Такие системы принято называть *нейросетями*.

Первые нейросети были созданы американскими учеными Г.Розенблаттом и П.Мак-Каллоком в конце 50-х гг. прошлого столетия.

Это были попытки создать системы, моделирующие человеческий глаз и его взаимодействие с мозгом. В 70-80 гг. количество работ по этому направлению искусственного интеллекта стало снижаться. Во многом неудачи исследователей были связаны с малой памятью и низким быстродействием существующих в то время компьютеров.

В середине 80-х гг. двадцатого столетия в Японии в рамках проекта разработки компьютера V поколения, основанного на знаниях, был создан компьютер VI поколения, или нейрокомпьютер. ЭВМ V поколения – *транспьютеры* – параллельные компьютеры с большим количеством процессоров, у которых ограничения по памяти и быстродействию практически сняты. *Нейрокомпьютеры* моделируют структуру мозга человека.

В настоящее время используются три подхода к созданию нейросетей:

- *аппаратный* – создание специальных компьютеров, плат расширения, наборов микросхем, реализующих все необходимые алгоритмы;
- *программный* – создание программ и инструментариев, рассчитанных на высокопроизводительные компьютеры. Сети создаются в памяти компьютера, всю работу выполняют его собственные процессоры;
- *гибридный* – комбинация первых двух. Часть вычислений выполняют специальные платы расширения (сопроцессоры), часть – программные средства.

В основу кибернетики «черного ящика» лег принцип, противоположный нейрокибернетике. Не имеет значения, как устроено «мыслящее» устройство. Главное, чтобы на заданные входные воздействия оно реагировало так же, как человеческий мозг.

Это направление искусственного интеллекта было ориентировано на поиски алгоритмов решения интеллектуальных задач на существующих моделях компьютеров. Оказалось, что ни одна из существующих наук: философия, психология, лингвистика, – не может предложить универсального алгоритма человеческого мышления. Кибернетики были вынуждены создавать собственные модели.

Существенный прорыв в практических приложениях искусственного интеллекта произошел в середине 70-х гг. прошлого столетия, когда на смену поискам универсального алгоритма мышления пришла идея моделировать конкретные знания специалистов-экспертов. Первые такие системы появились в США и получили название *экспертных систем*. Основа их и новый подход к решению задач искусственного интеллекта – *представление знаний*. MYCIN и DENTRAL – ставшие уже классическими экспертные системы для медицины и химии.

Направления развития искусственного интеллекта

Все разработки в области искусственного интеллекта можно разделить на несколько направлений:

- ***представление знаний и разработка систем, основанных на знаниях***. Это направления является основным, связано с разработкой моделей представления знаний и созданием баз знаний, образующих ядро экспертных систем (ЭС);
- ***игры и творчество***. В основе игровых интеллектуальных задач: шахмат, шашек и т.д., – их лежит один из ранних подходов, основой которого являются лабиринтная модель и эвристики. В научном плане эти идеи считаются тупиковыми;

➤ **разработка естественных языковых интерфейсов и машинный перевод.** В 50-х гг. прошлого столетия это была одна из самых популярных тем исследования искусственного интеллекта. Первая программа в этой области – переводчик с английского на русский язык;

➤ **распознавание образов.** Это традиционное направление искусственного интеллекта, берущее начало у самых его истоков. Каждому объекту ставится в соответствие матрица признаков, по которой происходит его распознавание. Это направление близко к машинному обучению и тесно связано с нейрокибернетикой;

➤ **разработка новых архитектур компьютеров.** Это направление связано с поиском новых аппаратных решений и архитектур, направленных на обработку символьных и логических данных. Создаются ПРОЛОГ- и ЛИСП- машины, компьютеры V и VI поколений. Последние разработки посвящены компьютерам баз данных и параллельным компьютерам;

➤ **разработка интеллектуальных роботов.** Роботы – это электромеханические устройства, предназначенные для автоматизации человеческого труда. Идея создания роботов древняя, а само слово появилось в 20-х гг. двадцатого столетия. Его автор – чешский писатель Карел Чапек. К настоящему времени сменилось уже несколько поколений роботов. Конечная цель робототехники – самоорганизующиеся, или интеллектуальные, роботы. Основная проблема при создании интеллектуальных роботов – проблема машинного зрения;

➤ **разработка программного обеспечения.** В рамках этого направления

- разрабатываются специальные языки для решения задач не вычислительного плана. Это языки, ориентированные на

символьную обработку информации: *LISP*, *PROLOG*, *РЕФАЛ*, *SMALLTALK* и др.;

- создаются пакеты прикладных программ, ориентированные на программную разработку интеллектуальных систем. Например, *KEE*, *ARTS* и др.;
- создание «пустых» экспертных систем, или оболочек: *EXSYS*, *MI* и др., в которых можно наполнять базы знаний, создавая различные системы;

➤ **обучение и самообучение.** Это одна из активно развивающихся областей искусственного интеллекта. Включает модели, методы и алгоритмы, ориентированные на автоматическое накопление знаний на основе анализа и обобщения данных.

Как уже отмечалось ранее, в настоящее время не существует единого определения понятия «искусственный интеллект», полностью описывающего эту научную область. Это связано, прежде всего, с большим кругом задач, решаемых в ее рамках, что наглядно демонстрируют перечисленные направления развития искусственного интеллекта.

Однако среди многих точек зрения на эту область науки доминируют три.

1. Исследования в области искусственного интеллекта являются фундаментальными исследованиями, в рамках которых разрабатываются модели и методы решения задач, традиционно считавшихся интеллектуальными и не поддававшихся ранее формализации и автоматизации.

2. Новое направление связано с новыми идеями решения задач на ЭВМ, с разработкой принципиально иной технологии программирования,

с переходом к архитектуре ЭВМ, отвергающей классическую архитектуру, которая восходит еще к первым ЭВМ.

3. В результате работ в области искусственного интеллекта рождается множество прикладных систем, решающих задачи, для которых ранее создаваемые системы были непригодны.

Все три точки зрения взаимосвязаны. В области искусственного интеллекта развиваются фундаментальные исследования, новая технология программирования, новая архитектура технических средств, и все это используется для создания прикладных систем, предназначенных для работы в самых разнообразных областях.

Фундаментальные исследования, прежде всего, связаны с вопросами представления и манипулирования знаниями. Разработка новых технологий программирования и архитектур технических средств необходима для использования существующих и перспективных инструментальных средств проектирования систем, ориентированных на работу со знаниями. Создание прикладных интеллектуальных систем – наиболее прагматическая точка зрения.

Модели и методы знаний

Данные и знания. Основные определения

Исследования в области искусственного интеллекта, направленные на разработку способов представления и манипулирования знаниями, связаны с развитием

- структуры памяти ЭВМ;
- информационных структур для представления данных;
- моделей представления данных;
- концепции знаний;

- моделей представления знаний;
- методов работы со знаниями.

Развитие структуры памяти ЭВМ. Информация, с которой имеют дело ЭВМ, разделяется на *процедурную* и *декларативную*. Процедурная информация о веществе в *программах* в виде команд, которые выполняются в процессе решения задач, а декларативная – в *данных*, с которыми эти программы работают.

Стандартной формой представления информации в ЭВМ является *машинное слово*. Множество машинных слов образует *информационную базу*. Каждое машинное слово в памяти машины, т.е. в информационной базе, имеет свое уникальное имя – адрес ячейки памяти. По этому имени происходит извлечение информационных единиц из памяти ЭВМ и запись их в нее.

Для представления данных и команд в последнее время используются одинаковые по числу разрядов машинные слова. Это позволило рассматривать их в ЭВМ в качестве одинаковых информационных единиц и выполнять операции и над командами, и над данными.

Развитие информационных структур для хранения данных. Появились способы описания данных в виде векторов и матриц, возникли списочные структуры, иерархические структуры. В настоящее время в языках программирования высокого уровня используются *абстрактные типы данных*, структура которых задается программистом. Появление *баз данных* (БД) ознаменовало еще один шаг на пути организации работы с декларативной информацией. В базах данных могут одновременно храниться большие объемы информации, а специальные средства, образующие *систему управления базами данных* (СУБД), позволяют эффективно манипулировать с данными.

Модели представления данных. В 70-х годах прошлого столетия различали три основные модели представления данных: *реляционные*, *сетевые* и *иерархические*. В настоящее время появилось второе поколение таких моделей, в рамках которых происходит постепенное слияние данных и знаний.

В развитых моделях представления данных сейчас выделяют два компонента: *интенциональные* и *экстенциональные представления*. Оба компонента хранятся в базе данных. При этом в ее экстенциональную часть входят конкретные факты (например, записи таблицы базы данных), касающиеся конкретной предметной области, а в интенциональную часть – схемы связей между атрибутами (например, между именами полей таблиц базы данных). Таким образом, экстенциональные представления описывают конкретные объекты из предметной области, конкретные события, происходящие в ней, или конкретные явления и процессы, а интенциональные представления фиксируют те закономерности и связи, которым все эти конкретные объекты, события, явления или процессы обязаны в данной проблемной области удовлетворять.

Экстенциональные представления относятся к данным. Относительно интенциональных представлений единого мнения нет. Разработчики баз данных говорят в этом случае о *схемах баз данных*, а представители искусственного интеллекта – о *знаниях в проблемной области*.

Концепция знаний. возникла по мере развития исследований в области интеллектуальных систем.

Знания – это выявленные закономерности предметной области (принципы, связи, законы), позволяющие решать задачи в этой области.

Знания в информационных системах хранятся в *базах знаний*. База знаний является ядром любой интеллектуальной системы.

Знания могут быть классифицированы как

- *поверхностные* – знания о видимых взаимосвязях между отдельными событиями и фактами в предметной области;
- *глубинные* – абстракции, аналогии, схемы, отображающие структуру и процессы в предметной области.

Следует особо подчеркнуть, что основой современной концепции знаний является утверждение, что знания объединяют в себе многие черты процедурной и декларативной информации.

Исходя из особой значимости знаний для любой интеллектуальной системы, отличительные особенности знаний, модели представления знаний, а также методы работы со знаниями рассмотрим более подробно.

Особенности знаний

Этой форме представления информации в ЭВМ присущи следующие особенности.

1. *Внутренняя интерпретируемость.* Каждая информационная единица должна иметь уникальное имя, по которому интеллектуальная система находит ее, а также отвечает на запросы, в которых это имя упомянуто. Когда данные, хранящиеся в памяти, были лишены имен, отсутствовала возможность их идентификации системой. Данные могла идентифицировать только программа, извлекающая их из памяти по указанию программиста, написавшего программу.

При переходе к знаниям в память ЭВМ вводится информация о некоторой *протоструктуре информационных единиц*. Эта структура представляет собой специальное машинное слово, в котором указано, в каких разрядах памяти хранятся сведения, например, о фамилии, годе рождения, специальности и стаже работы сотрудника предприятия. Все эти

атрибуты могут играть роль имен для тех машинных слов, которые соответствуют строкам таблицы. По ним можно осуществлять поиск нужной информации. Каждая строка таблицы представляет собой экземпляр протоструктуры. В настоящее время СУБД обеспечивают реализацию внутренней интерпретируемости всех информационных единиц, хранимых в базе данных.

2. *Структурированность*. Информационные единицы должны обладать гибкой структурой. Каждая информационная единица может быть включена в состав любой другой, и из каждой информационной единицы можно выделить некоторые ее составляющие. Это позволяет устанавливать между отдельными информационными единицами отношения типа «часть-целое», «род-вид» или «элемент-класс».

3. *Связность*. В информационной базе между информационными единицами должна быть предусмотрена возможность установления связей различного типа. Эти связи должны характеризовать отношения между информационными единицами.

Семантика отношений может носить декларативный или процедурный характер. Например, две или более информационные единицы могут быть связаны отношением «одновременно», две информационные единицы – отношением «причина-следствие» и т.п. Такие отношения характеризуют декларативные знания. Если между двумя информационными единицами установлено отношение «аргумент-функция», то оно характеризует процедурное знание, связанное с вычислением определенных функций.

4. *Семантическая метрика*. На множестве информационных единиц в некоторых случаях полезно задавать отношение, характеризующее ситуационную близость информационных единиц, т.е. силу ассоциативной связи между информационными единицами. Такое

отношение дает возможность выделять в информационной базе некоторые типовые ситуации (например, «покупка», «отчисление студента» и т.п.).

5. *Активность.* С момента появления ЭВМ и разделения используемых в ней информационных единиц на данные и команды создалась ситуация, при которой данные пассивны, а команды активны. Все процессы, протекающие в ЭВМ, инициируются командами, а данные используются этими командами лишь в случае необходимости. Для интеллектуальной системы эта ситуация неприемлема. Как и у человека, в интеллектуальной системе актуализации тех или иных действий способствуют знания, имеющиеся в системе. Таким образом, выполнение программ в интеллектуальной системе должно инициироваться текущим состоянием информационной базы. Появление в базе фактов или описаний событий, установление связей может стать источником активности системы.

Перечисленные особенности информационных единиц определяют ту грань, за которой данные превращаются в знания, а базы данных перерастают в *базы знаний*. Совокупность средств, обеспечивающих работу со знаниями, образует *систему управления базой знаний*.

Модели представления знаний

Постановка и решение любой задачи всегда связаны с ее «погружением» в предметную область. Решая, например, задачу составления расписания обработки деталей на металлорежущих станках, мы манипулируем конкретными объектами (конкретные станки, детали, интервалы времени и т.п.) и общими понятиями («станок», «деталь», «тип станка» и т.п.). Все они образуют единое пространство понятия «предметная область».

Предметная область – это все предметы и события, которые составляют основу общего понимания необходимой для решения задачи информации.

Гипотетически предметная область представляется состоящей из реальных или абстрактных объектов, которые называются *сущностями*.

Сущности предметной области находятся в определенных *отношениях* друг к другу (ассоциациях), которые также можно рассматривать как сущности и включать в предметную область. Между сущностями наблюдаются различные отношения подобия. Совокупность подобных сущностей составляет *класс сущностей*, являющийся новой сущностью предметной области.

Отношения между сущностями выражаются с помощью суждений.

Суждение это мысленно возможная ситуация, которая может иметь место для предъявляемых сущностей или не иметь места. В языке (формальном или естественном) суждениям отвечают *предложения*. Суждения и предложения также можно рассматривать как сущности и включать в предметную область.

Языки, предназначенные для описания предметных областей, называются *языками представления знаний*. Универсальным языком представления знаний является естественный язык. Однако использование естественного языка в системах машинного представления знаний наталкивается на большие трудности в виду присущих ему двусмысленностей, нерегулярностей и т.п. Но главное препятствие заключается в отсутствии формальной семантики естественного языка, которая имела бы достаточно эффективную операционную поддержку.

В настоящее время существуют десятки моделей (или языков) представления (описания) знаний для различных предметных областей. Большинство из них может быть сведено к следующим классам:

- логические модели;
- сетевые модели;
- продукционные модели;
- фреймовые модели.

Логические модели

Описания предметных областей, выполненные в логических языках, называются (формальными) *логическими моделями*. А их основе лежит *формальная система*, задаваемая четверкой вида $M = \langle T, P, A, B \rangle$.

Множество T есть *множество базовых элементов* различной природы, например, слов из некоторого ограниченного словаря, деталей детского конструктора, входящих в состав некоторого набора и т.п. Важно, что для множества T существует некоторый способ определения принадлежности или непринадлежности произвольного элемента к этому множеству. Процедура такой проверки может быть любой, но за конечное число шагов она должна давать положительный или отрицательный ответ на вопрос, является ли x элементом множества T . Обозначим такую процедуру $\Pi(T)$.

Множество P есть *множество синтаксических правил*. С их помощью из элементов T образуют *синтаксически правильные совокупности*. Например, из слов ограниченного словаря строятся синтаксически правильные фразы, из деталей детского конструктора с помощью гаек и болтов собираются новые конструкции. Декларируется существование процедуры $\Pi(P)$, с помощью которой за конечное число шагов можно получить ответ на вопрос, является ли совокупность X синтаксически правильной.

В множестве синтаксически правильных совокупностей выделяется некоторое подмножество A . Элементы A называются *аксиомами*. Как и для

других составляющих формальной системы, должна существовать процедура $\Pi(A)$, с помощью которой для любой синтаксически правильной совокупности можно получить ответ на вопрос о принадлежности ее к множеству A .

Множество B есть множество *правил вывода*. Применяя их к элементам A , можно получать новые синтаксически правильные совокупности, к которым снова можно применять правила из множества B . Так формируется *множество выводимых* в данной формальной системе *совокупностей*. Если имеется процедура $\Pi(B)$, с помощью которой можно определить для любой синтаксически правильной совокупности, является ли она выводимой, то соответствующая формальная система называется *разрешимой*. Это показывает, что именно правила вывода являются наиболее сложной составляющей формальной системы.

Для знаний, входящих в базу знаний, можно считать, что множество A образуют все информационные единицы, которые введены в базу знаний извне, а с помощью правил вывода из них выводятся новые *производные знания*. Другими словами, формальная система представляет собой генератор порождения новых знаний, образующих множество *выводимых* в данной системе *знаний*. Это свойство логических моделей делает их притягательными для использования в базах знаний. Оно позволяет хранить в базе лишь те знания, которые образуют множество A , а все остальные знания получать из них по правилам вывода.

Сетевые модели

Сетевые модели формально можно задать в виде

$$H = \langle I, C_1, C_2, \dots, C_n, \Gamma \rangle$$

Множество I есть множество информационных единиц.

C_1, C_2, \dots, C_n – множество типов связей между информационными единицами.

Отображение Γ задает между информационными единицами, входящими в множество I , связи из заданного набора типов связей.

В зависимости от типов связей, используемых в модели, различают

- *классифицирующие сети*;
- *функциональные сети*;
- *сценарии*.

В *классифицирующих* сетях используются отношения структуризации. Такие сети позволяют в базах знаний вводить разные иерархические отношения между информационными единицами. *Функциональные* сети характеризуются наличием функциональных отношений. Их часто называют *вычислительными моделями*, т.к. они позволяют описывать процедуры «вычислений» одних информационных единиц через другие. В *сценариях* используются отношения типа «средство-результат», «орудие-действие» и т.п.

Если в сетевой модели допускаются связи различного типа, то ее обычно называют *семантической сетью*.

Семантика – это наука, устанавливающая отношения между символами и объектами, которые они обозначают, т.е. наука, определяющая смысл знаков.

Семантическая сеть – это ориентированный граф, вершины которого – *понятия*, а дуги – *отношения* между ними.

Понятиями обычно выступают абстрактные или конкретные объекты (множество I). *Отношения* – это связи типа «имеет частью», «это», «принадлежит» и т.п. (отображение Γ).

Проблема поиска решения в базе знаний типа семантической сети сводится к задаче поиска фрагмента сети, составляющего некоторой подсети, соответствующей поставленному вопросу.

Продукционные модели

Продукционная модель, или модель, основанная на правилах, позволяет представить знания в виде предложений типа

ЕСЛИ (*условие*), ТО (*действие*), где

условие – некоторое предложение-образец, по которому осуществляется поиск в базе знаний;

действие – действия, выполняемые при успешном исходе поиска (они могут быть промежуточными, выступающими далее как *условия*, и целевыми, завершающими работу системы).

При использовании продукционной модели база знаний состоит из набора правил. Программа, управляющая перебором правил, называется *машиной вывода*. Чаще всего вывод бывает *прямой* (от данных к поиску цели) или *обратный* (от цели для ее подтверждения – к данным). *Данные* – это исходные факты, на основании которых запускается машина вывода – программа, перебирающая правила из базы.

Фреймовые модели

Фрейм – в переводе с английского языка – каркас или рамка.

Под *фреймом* понимается абстрактный образ или ситуация, имеющие жесткую структуру информационных единиц. В общем виде она выглядит следующим образом:

(Имя фрейма:

Имя слота1 (значение слота1),

Имя слота2 (значение слота2),

...

Имя слотаN (значение слотаN))

Значением *слота* может быть практически что угодно (числа или математические соотношения, тексты на естественном языке или программы, правила вывода или ссылки на другие слоты данного фрейма или других фреймов). В качестве значения слота может выступать набор слотов более низкого уровня, что позволяет во фреймовых представлениях реализовать «принцип матрешки».

Связи между фреймами задаются значениями специального слота.

Методы работы со знаниями

В зависимости от целевой функции методы работы со знаниями можно сгруппировать следующим образом:

- приобретение знаний;
- пополнение знаний;
- обобщение и классификация знаний.

Из перечисленных групп методов работы со знаниями безусловно важны все, однако первая группа методов является основополагающей.

Приобретение знаний

Приобретением знаний называется выявление знаний из источников и преобразование их в нужную форму, а также перенос в базу знаний интеллектуальной системы.

Из определения следует, что процесс приобретения знаний сводится к извлечению знаний, их формализации и представлению в интеллектуальной системе.

Извлечение знаний. Источниками знаний могут быть книги, архивные документы, содержимое других баз знаний и т.п., т.е. некоторые

объективизированные знания, переведенные в форму, которая делает их доступными для потребителя.

Другим типом знаний являются *экспертные знания*, которые имеются у специалистов, но никак не зафиксированы во внешних по отношению к нему хранилищах. Экспертные знания являются *субъективными*. Еще одним видом субъективных знаний являются *эмпирические знания интеллектуальной системы*. Такие знания добываются системой путем наблюдения за окружающей средой при наличии у нее средств наблюдения.

Ввод в базу знаний объективизированных знаний не представляет собой проблемы, а вот выявление и ввод субъективных и особенно экспертных знаний достаточно трудны. Преодолением этой проблемы при разработке интеллектуальной системы занимается инженер по знаниям.

Чтобы разработать методологию извлечения субъективных знаний, получаемых от эксперта, прежде всего, надо четко различать две формы репрезентации знаний. Одна форма связана с тем, как и в каких моделях хранятся эти знания у человека-эксперта. При этом эксперт не всегда осознает полностью, как репрезентированы у него знания. Другая форма связана с тем, как инженер по знаниям, проектирующий интеллектуальную систему, собирается их описывать и представлять. От степени согласованности этих двух форм репрезентации между собой зависит эффективность работы инженера по знаниям.

Формы репрезентации знаний человека изучаются в когнитивной психологии. Примеры:

- *представление класса понятий через его элементы*. Понятие «птица» репрезентируется рядом «чайка, воробей, дятел, ...»;
- *представление понятий класса с помощью базового прототипа, отражающего наиболее типичные свойства*

объектов класса. Понятие «птица» репрезентируется прототипом «нечто с крыльями, клювом, летает, ...»;

- *представление с помощью признаков.* Для понятия «птица» – наличие крыльев, клюва, двух лап, перьев, ...).

Кроме понятий репрезентируются и отношения между ними. Как правило, отношения между понятиями определяются процедурным способом, а отношения между составляющими понятием (определяющими структуру понятия) – декларативным способом. Наличие двух видов описаний заставляет при выборе модели представления знаний использовать различные их виды, т.е. появляются комбинированные модели представления знаний. Например, использование семантической сети и продукционной системы.

Для извлечения знаний у эксперта инженер по знаниям может воспользоваться следующими режимами работы:

- протокольный анализ;
- интервью;
- игровая имитация профессиональной деятельности.

Протокольный анализ заключается в фиксации, например путем записи на какой-то носитель «мыслей вслух» эксперта во время решения проблемы и в последующем анализе полученной информации. В режиме *интервью* инженер по знаниям ведет с экспертом активный диалог, направляя его в нужную сторону. При *игровой имитации* эксперт помещается в ситуации, похожие на те, с которых протекает его профессиональная деятельность. Наблюдая за его действиями в различных ситуациях, инженер по знаниям, формирует свои соображения об экспертных знаниях, которые впоследствии могут быть уточнены с экспертом в режиме интервью. Принципы игровой имитации нашли применение в разнообразных деловых играх и специальных тренажерах.

Каждый из способов извлечения знаний имеет свои преимущества и недостатки. Так, при анализе протоколов инженеру по знаниям нелегко отделить понятия, важные для включения в словарь предметной области, от тех, которые при «мыслях вслух» появляются случайно. Кроме того, в протоколах обнаруживаются пробелы, когда рассуждение эксперта как бы прерывается и продолжается уже на основе пропущенных шагов вывода. Заполнение подобных ниш возможно лишь в режиме интервью. Таким образом, во всех трех подходах к извлечению знаний из экспертов необходим этап интервью, что делает его одним из важнейших методов приобретения знаний.

Существует не менее двух десятков стратегий интервьюирования. Чаще всего используются три: разбиение на ступени, репертуарная решетка и подтверждение сходства.

При *разбиении на ступени* эксперту предлагается назвать наиболее важные, по его мнению, понятия предметной области и указать между ними отношения структуризации, т.е. отношения типа «род-вид», «элемент-класс», «целое-часть» и т.п. Эти понятия используются на следующем шаге опроса как базовые. Стратегия нацелена на создание иерархии понятий предметной области, выделение в понятиях тесно связанных между собой групп – *кластеров*.

Стратегия *репертуарной решетки* направлена на выявление характеристических свойств понятий, позволяющих отделять одни понятия от других. Методика состоит в предъявлении. Эксперту троек понятий с предложением назвать признаки для каждой двух понятий, которые отделяли бы их от третьего. Так как каждое понятие входит в несколько троек, то на основании такой процедуры происходит уточнение объемов понятий и формируются «симптокомплексы» понятий, с помощью которых эти понятия могут идентифицироваться в базе знаний.

Стратегия подтверждения сходства состоит в том, что эксперту предлагается установить принадлежность каждой пары понятий из предметной области к некоторому отношению сходства (толерантности). Для этого эксперту задается последовательность достаточно простых вопросов, цель которых заключается в уточнении того понимания сходства, которое вкладывает эксперт в утверждение о сходстве двух понятий предметной области.

Формализация знаний и представление их в базе знаний. – это процесс описания полученных от эксперта знаний с помощью одного из языков представления знаний (моделей знаний) и разработка базы знаний. Если формализовать знания или представить их в базе знаний не удастся, инженер по знаниям возвращается к работе с экспертом.

Пополнение, обобщение и классификация знаний

Проблема **пополнения знаний** возникла при решении задач понимания естественного языка, обучения, поиска ответов на вопросы к базе знаний, анализа ситуации, сцен и др. Знания могут быть представлены в виде фактов, хранящихся в базе знаний, или в виде описаний ситуаций, поступающих на вход интеллектуальной системы.

При описании ситуации человек обычно не стремится сообщить все подробности, относящиеся к этой ситуации, считая их несущественными, само собой разумеющимися и т.д. Однако большинство опущенных деталей необходимо для того, чтобы полностью понять ситуацию. В связи с этим требуются механизмы восстановления пропущенной информации. Например, если человек воспринимает сообщение «Воронам же, гнездящимся на ветках, все нипочем», то он скорее всего не будет думать, что ветки с воронами висят в воздухе. Он представит дерево, на ветках которого расположены гнезда, а в них

вороны. Однако даже такой тривиальный и естественный вывод требует достаточно большого числа рассуждений: ветка является частью дерева; птицы выют гнезда на деревьях; деревья растут на земле и т.д. Даже такой простой пример предполагает использование сложной процедуры вывода на основе знаний, хранящихся в памяти.

Существует несколько подходов к пополнению знаний: модели «здравого смысла»; сценарии; подход, опирающийся на идею о том, что физические закономерности внешнего мира могут быть описаны в рамках специальных логик. Однако все эти подходы в той или иной мере используют идею продукционных правил.

Система продукций (правил) отражает закономерности, присущие заданной проблемной области. Например, для ситуаций, описывающих статические пространственные отношения между объектами (свойство вложенности, которое выполняется в «матрешках»), могут быть использованы такие продукции:

$$X \text{ <содержится в> } Y, Y \text{ <содержится в> } Z \Rightarrow X \text{ <содержится в> } Z$$

или

$$X \text{ <находится на> } Y, Y \text{ <находится на> } Z \Rightarrow X \text{ <находится над> } Z$$

Пополняя постепенно базу знаний продукциями такого типа, можно в интеллектуальной системе создать достаточно мощные средства для порождения новых фактов из тех, которые в ней имеются.

Процедуры пополнения знаний отличаются друг от друга структурой продукционных правил и стратегиями управления в системе продукций.

Способность человеческого мышления к обобщению лежит в основе любого научного исследования. Именно в результате обобщения

получаются новые знания, т.е., не следующие непосредственно из ранее известных знаний. В системах, моделирующих мышление, *обобщение* понимают как процесс получения знаний, объясняющих имеющиеся факты и способных объяснить, классифицировать или предсказывать новые знания.

В общем виде задача обобщения формулируется следующим образом: по совокупности наблюдений (фактов) F , совокупности требований и допущений к виду результирующей гипотезы H и совокупности базовых знаний и предположений, включающих знания об особенностях предметной области, выбранном способе представления знаний, наборе допустимых операторов, эвристик и др., сформировать гипотезу H : $H \Rightarrow F$ (H «объясняет» F).

Существует несколько методов обобщения, цель использования каждого из них – создание *модели обобщения*. Все они находят применение в экспертных системах, системах распознавания образов, в задачах автоматизированного исследования закономерностей явлений физики, химии, биологии, медицины. Наличие подсистем, решающих задачи обобщения, обязательно для систем ситуационного управления, т.к. именно это позволяет получать по текущей ситуации ее иерархическую модель в виде «слоеного пирога», а следовательно, и принимать правильные решения.

Инструментальные средства создания интеллектуальных систем

Инструментальные средства создания интеллектуальных систем – это комплекс программно-аппаратных средств, позволяющих разработчику создавать и совершенствовать специализированные системы.

В общей проблеме окружения, ориентированного на создание интеллектуальных систем, надо различать аппаратную и программную составляющую.

Доступная разработчику аппаратура является тем базисом, на котором создаются инструментальные средства. Это с одной стороны. С другой – и сами системы, ради которых эти инструментальные средства разрабатываются, будут эксплуатироваться на этой аппаратуре. Поэтому аппаратная часть в значительной мере определяет архитектуру и функциональные возможности будущей системы, а также инструментальные средства для их работы.

По архитектуре систем различают резидентные и кросс-системы.

Резидентные системы поддержки обеспечивают создание программных комплексов для тех же вычислительных средств, на которых они сами реализованы. Преимущества резидентных систем определяются прежде всего удобством использования единой программно-аппаратной среды для всех этапов разработки и сопровождения системы. Часто в резидентных системах средства поддержки разработки реализуются на базе типовых (штатных) средств программного обеспечения и/или путем прямого включения в соответствующую инструментальную среду.

Кросс-системы позволяют проектировать программные комплексы для ЭВМ, не совместимых с инструментальными средствами. В результате на целевой (рабочей) ЭВМ специальные технологические программы могут вообще отсутствовать либо включать минимальный набор средств, поддерживающих собственно развертывание системы. По мере становления индустрии баз знаний можно ожидать, что кросс-системы станут основным средством для тиражирования знаний в конкретные прикладные системы.

Накопленный во многих странах опыт решения различных интеллектуальных задач показывает, что современные универсальные ЭВМ, несмотря на постоянное совершенствование, недостаточно эффективны. Поэтому проблема построения специализированных вычислительных систем и аппаратных средств для решения таких задач привлекает внимание исследователей многих стран.

На первом этапе развития искусственного интеллекта (в конце 50-х – начале 60-х годов) не существовало языков и систем, ориентированных специально на области знаний. Появившиеся к тому времени универсальные языки программирования казались адекватным инструментом для создания любых, в том числе и интеллектуальных, систем. Такая уверенность основывалась на том факте, что в этих языках программирования можно выделить декларативную и процедурную компоненты. Соответственно это – описания и последовательности операторов, реализующие заданные алгоритмы над заданными описаниями. Казалось, что на этой базе могут быть интерпретированы любые модели и системы представления знаний. Однако сложность и трудоемкость таких интерпретаций оказалась настолько велика, что прикладные системы для реализации были недоступны.

Это послужило толчком к развитию языков программирования, которые могли бы использоваться в качестве программных инструментальных средств разработки интеллектуальных систем. Таковыми являются ЛИСП, Рефал, Снобол, ПРОЛОГ (основной конкурент ЛИСПа), современные продукционные языки. Однако ни один из них не стал универсальным языком программирования искусственного интеллекта. Однако важно то, что языки этой группы способствовали переосмыслению самого понятия программы. В области искусственного

интеллекта это послужило толчком к развитию объектно-ориентированного программирования и разработке языков представления знаний.

Опыт, накопленный в ходе реализации интеллектуальных систем, позволяет говорить о том, что в области искусственного интеллекта сформировались определенные парадигмы программирования. Основными из них являются функциональная, логическая, объектно-ориентированная парадигмы и продукционное программирование.

Функциональная парадигма – программа состоит из независимых функций, каждая из которых определяет правило преобразования своих аргументов для получения некоторого результата. Эти функции определяются путем композиции системных или определяемых программистом функций с использованием структур типа альтернативы или рекурсии. Наиболее известным представителем языков, опирающихся на функциональную парадигму, является ЛИСП.

Логическая парадигма основана на использовании механизма доказательства теорем, который позволяет выяснить, противоречиво ли некоторое множество логических формул. При этом программа может рассматриваться как набор логических формул совместно с теоремой (запросом), которая должна быть доказана. При этом программист избавляется от необходимости определения точной последовательности шагов для выполнения вычислений. Типичным представителем этой парадигмы является язык ПРОЛОГ.

Объектно-ориентированная парадигма – программа состоит из объектов и сообщений. Такой способ программирования вносит в программу модульность посредством абстракции данных и наследования. Поэтому особенно хорошо подходит для ситуаций, когда имеется ясная иерархическая классификация объектов. Это позволяет избежать

дублирования и локализовать описания механизмов работы с информацией. Одним из первых языков этого типа является *Smalltalk*.

В современной практике интеллектуального программирования отдельные парадигмы редко используются в чистом виде. Более того, в настоящее время наметился отдельный подход к программированию интеллектуальных систем на базе *продукционной парадигмы*, в рамках которой взаимодействие правил-продукций может опираться на функциональную парадигму, а унификация условий применимости правил – на логическую. При этом сам продукционный процессор может быть построен на базе механизмов асинхронного управления, близких по своей идее к объектно-ориентированной парадигме.

Таким образом, современный этап программирования интеллектуальных систем характеризуется тенденцией смешанного использования разных парадигм.

Экспертные системы

Структура экспертной системы

В середине 70-х годов прошлого столетия в исследованиях по искусственному интеллекту сформировалось самостоятельное направление, получившее название «экспертные системы». Цель исследований, проводимых в этом направлении, состоит в разработке программ или устройств, которые при решении задач, трудных для эксперта-человека, получают результаты, не уступающие по качеству и эффективности решениям эксперта.

Термин «экспертная система» отражает как ее основную характеристику, так и способ создания системы. В процессе разработки экспертная система наполняется информацией – знаниями о предметной

области, которые берутся у соответствующих специалистов, обладающих высокой квалификацией. Если полученные знания в достаточной степени отражают все особенности предметной области, а также удалось построить эффективную систему работы с ними, то от экспертной системы можно ожидать нетривиальные решения задач. Таким образом, система строится экспертами и сама становится экспертом, помогающим принимать решения.

В настоящее время существует уже огромное количество экспертных систем. Такой интерес к экспертным системам со стороны пользователей вызван по крайней мере тремя причинами.

1. Экспертные системы ориентированы на решение широкого круга задач в неформализованных областях, т.е. на приложения, которые до недавнего времени считались малодоступными для вычислительной техники.

2. С помощью экспертных систем специалисты, не знающие программирования, могут самостоятельно разрабатывать интересующие их приложения, что позволяет резко расширить сферу использования вычислительной техники.

3. Экспертные системы при решении практических задач достигают результатов, не уступающих, а иногда и превосходящих возможности экспертов, не имеющих «под рукой» экспертную систему.

В настоящее время экспертные системы применяются в различных областях деятельности человека. Наибольшее распространение экспертные системы получили в проектировании интегральных микросхем, в поиске неисправностей, в военных приложениях и автоматизации программирования.

Экспертные системы – это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных

областях и тиражирующие этот эмпирический опыт для консультаций менее квалифицированных пользователей.

Обобщенная структура экспертной системы представлена на рис.1

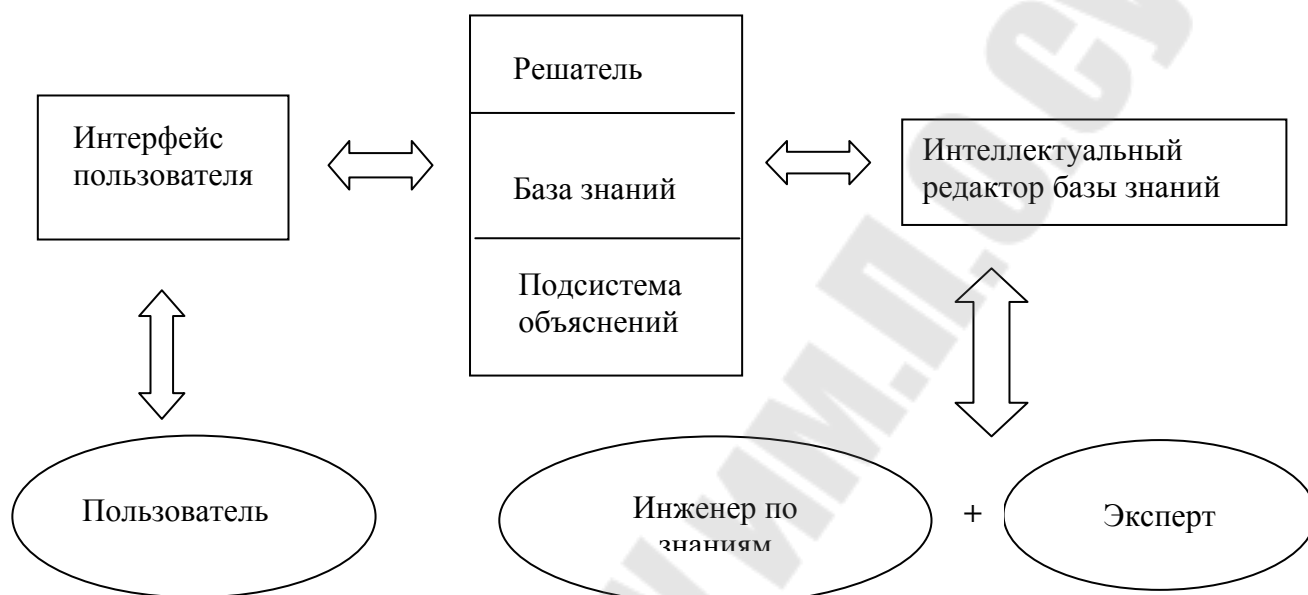


Рис.1

Особо отметим, что на рис.1 представлена обобщенная структура экспертных систем. Реальные систем могут иметь более сложную структуру.

Рассмотрим структурные компоненты экспертных систем.

Пользователь – специалист предметной области, для которого предназначена система. Обычно его квалификация недостаточно высока, и поэтому он нуждается в помощи и поддержке своей деятельности со стороны экспертной системы.

Инженер по знаниям – специалист по искусственному интеллекту, выступающий в роли промежуточного буфера между экспертом и базой знаний.

Эксперт – специалист в предметной области.

Интерфейс пользователя – комплекс программ, реализующих диалог пользователя с экспертной системой как на стадии ввода информации, так и получения результатов.

База знаний – ядро экспертной системы, совокупность знаний предметной области, записанная на машинный носитель в форме, понятной эксперту и пользователю (обычно на некотором языке, приближенном к естественному). Параллельно такому «человеческому» представлению существует база знаний во внутреннем «машинном» представлении.

Решатель – программа, моделирующая ход рассуждений эксперта на основании знаний, имеющихся в базе знаний.

Подсистема объяснений – программа, позволяющая пользователю получить ответы на вопросы «Как была получена та или иная рекомендация?» и «Почему система приняла такое решение?».

Интеллектуальный редактор базы знаний – программа, представляющая инженеру по знаниям возможность создавать базу знания в диалогом режиме.

Классификация экспертных систем

Как отмечалось ранее, существует огромное количество экспертных систем. Все они могут быть классифицированы по различным критериям. По видам решаемых задач различают:

1. Экспертные системы *интерпретации данных*. Под интерпретацией понимается определение смысла данных, результаты которого должны быть согласованными и корректными. Например,

обнаружение и идентификация различных типов океанских судов (система *SIAP*).

2. *Диагностика*. Под диагностикой понимается обнаружение неисправности в некоторой системе. Неисправность – это отклонение от нормы. Такая трактовка позволяет с единых теоретических позиций рассматривать и неисправность оборудования, и заболевания живых организмов, и всевозможные природные аномалии. Например, диагностика и терапия сужения коронарных сосудов (*ANGY*), диагностика ошибок в аппаратуре и математическое обеспечение ЭВМ (*CRIB*).

3. *Мониторинг* – непрерывная интерпретация данных в реальном масштабе времени и сигнализация о выходе тех или иных параметров за допустимые пределы. Например, контроль за работой электростанций (*СПРИНТ*), помощь диспетчерам атомного реактора (*REACTOR*) и др.

4. *Проектирование* – подготовка спецификаций (чертежей, пояснительных записок и т.д.) на создание «объектов» с заранее определенными свойствами. Например, проектирование конфигураций ЭВМ VAX (система *XCON*), синтез электрических цепей (*SYN*) и др.

5. *Прогнозирующие системы* логически выводят вероятные следствия из заданных ситуаций. Например, предсказание погоды (*WILLARD*), прогнозы в экономике (*ECON*), оценка будущего урожая (*PLANT*) и др.

Этапы разработки экспертных систем

Разработка экспертных систем находится на уровне скорее искусства, нежели науки. Процесс разработки промышленной экспертной системы, опираясь на традиционные технологии, можно разделить на

шесть более или менее независимых этапов, практически не зависящих от предметной области:

- выбор проблемы;
- разработка прототипа;
- доработка до промышленной системы;
- оценка;
- стыковка;
- поддержка.

Каждый из этих этапов важен и необходим, однако, первые два являются основополагающими. Поэтому подробно рассмотрим именно их.

Выбор проблемы является подготовительным этапом. На этом этапе необходимо

- определение проблемной области и задачи;
- нахождение эксперта и назначение коллектива разработчиков;
- определение предварительного подхода к решению проблемы;
- анализ расходов и прибыли от разработки;
- подготовка подробного плана разработки.

Разработка прототипа. Прототипная система является усеченной версией экспертной системы, спроектированной для проверки правильности кодирования фактов, связей и стратегий рассуждений эксперта. Существует шесть стадий разработки прототипа.

1. *Идентификация проблемы* – уточняется задача, планируется ход разработки прототипа, определяются:

- необходимые ресурсы (время, люди, ЭВМ и т.д.);
- источники знаний (книги, методики, дополнительные эксперты и др.);
- имеющиеся аналоговые экспертные системы;

- цели (распространение опыта, автоматизация рутинных действий и др.);
- классы решаемых задач и т.д.

2. *Извлечение знаний.*

3. *Структурирование и концептуализация знаний.*

Концептуализация знаний – разработка неформального описания знаний о предметной области в виде графа, таблицы, диаграммы или текста, которое отражает основные концепции и взаимосвязи между понятиями предметной области.

Такое описание называется *полем знаний*.

На этом этапе выявляется структура полученных знаний о предметной области, т.е. определяются:

- терминология;
- список основных понятий и их атрибутов;
- отношения между понятиями;
- структура входной и выходной информации;
- стратегии принятия решений и т.д.

4. *Формализация знаний*, т.е. разработка базы знаний на языке, который, с одной стороны, соответствует структуре поля знаний, а с другой – позволит реализовать прототипы системы на следующей стадии программной реализации.

5. *Реализация* – разработка программного комплекса, демонстрирующего жизнеспособность подхода в целом. Чаще всего первый прототип системы отбраковывается на этапе реализации действующей экспертной системы.

6. *Тестирование* – выявление ошибок в подходе и реализации прототипа и выработка рекомендаций по доводке системы до промышленного варианта.

СОДЕРЖАНИЕ

Искусственный интеллект	3
Направления развития искусственного интеллекта	7
Модели и методы знаний	10
<i>Данные и знания. Основные определения</i>	10
<i>Особенности знаний</i>	13
<i>Модели представления знаний</i>	15
<i>Логические модели</i>	17
<i>Сетевые модели</i>	18
<i>Продукционные модели</i>	20
<i>Фреймовые модели</i>	20
<i>Методы работы со знаниями</i>	21
<i>Приобретение знаний</i>	21
<i>Пополнение, обобщение и классификация знаний</i>	25
Инструментальные средства создания интеллектуальных систем	27
Экспертные системы	31
<i>Структура экспертной системы</i>	31
<i>Классификация экспертных систем</i>	34

СИСТЕМЫ, ОСНОВАННЫЕ НА ЗНАНИЯХ

Пособие по курсу «Автоматизация принятия управленческих решений» для студентов экономических специальностей дневной и заочной форм обучения

Составитель: **Водополова** Наталья Витальевна

Подписано к размещению в электронную библиотеку
ГГТУ им. П. О. Сухого в качестве электронного
учебно-методического документа 11.06.09.

Рег. № 47Е.

E-mail: ic@gstu.gomel.by

<http://www.gstu.gomel.by>