



北京航空航天大学  
BEIHANG UNIVERSITY

经济管理学院



# 人工智能

## Artificial Intelligence

# 第五章 专家系统



# 课前索引

## 【熟手与新手】

对于一个从事教学工作多年的专业教师来说，当看到一道新的题目时

.....

- ✓ 经验丰富的老中医在诊断疑难杂症时.....
- ✓ 常年工作在一线的维修工作在检查设备故障时.....
- ✓ 当生物学家碰到新的生命物种时.....
- ✓ 如果是初出茅庐的新手碰到复杂问题时.....

为什么专家能够解决这些问题？

计算机能否模拟专家解决这些问题？

**-----专家系统是一类具有专门知识和经验的计算机智能程序系统。**

**-----专家系统（Expert System）是人工智能应用研究最活跃和最广泛的领域之一。**

# 本章主要内容

---

## 【发展】

1965年斯坦福大学的费根鲍姆和化学家勒德贝格合作研制DENDRAL系统，使得人工智能的研究从面向基本技术和基本方法的理论研究走向解决实际问题的具体研究，从探索广泛的普遍规律转向知识的工程应用，体现知识的巨大力量。

# 专家系统概念

什么是“专家”：经过不断的学习和工作，部分人在某一工作领域中积累了丰富的经验性知识以及解决问题时独有的思维方式，我们称这些有“经验”的人为该领域的专家。

- 你能说出它们是什么动物吗？它们有什么特征？



# 专家系统概念

- 但如果是仅仅知道这些动物的特征
- 你是否能够推断出它们是什么动物呢？
- 你是否能够用计算机推断出它们是什么动物呢？

## ——专家系统

专家系统就是一个具有智能特点（专门知识和经验）的计算机软件系统。它使用了某些领域的人类专家知识，并将这些知识通过编程嵌入到计算机内部，采用人工智能中的知识表示和知识推理技术来模拟通常由专家才能解决的复杂问题，达到具有与专家同等解决问题能力的水平。计算机嵌入了这些知识后，可以通过友好的用户界面同用户进行交流，并允许用户向计算机提出一些问题。

# 专家系统概念

ES包括以下三个方面的含义：

(1)ES是一种程序系统，但又具有智能，因此它不同于一般的程序系统，而是一种能运用专家知识和经验进行推理的启发式程序系统。

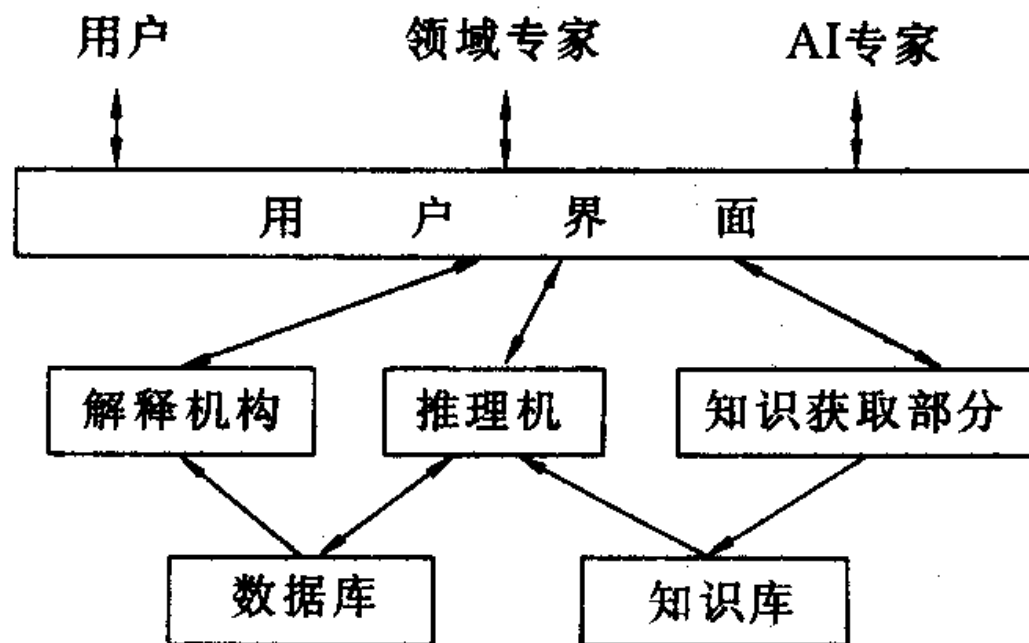
(2)ES的智能来源于领域专家的知识、经验及解决问题的诀窍。为此，ES内部必须包含有大量专家水平的领域知识与经验，并且能够在运行过程中不断地增长新知识和修改原有知识。

(3)ES所要解决的问题一般是那些本来应该由领域专家才能解决的问题。

- (1) 应用于某专门领域。
- (2) 拥有专家级知识。
- (3) 能模拟专家的思维。
- (4) 能达到专家级水平。

# 专家系统概念

## 一、ES结构



专家系统的基本结构



# 专家系统概念

## 二、ES的分类

ES都是针对某一应用领域而建立的。不同应用领域的ES，其功能、设计方法及实现技术也各不相同。对ES的类型，可以有多种不同的划分方法。例如，可以按求解问题的性质分类，也可以按求解问题的要求分类，还可以按系统的体系结构分类等。

### 1. 按求解问题的性质分类

(1)解释型ES; (2)预测型ES; (3)设计型ES; (4)诊断型ES; (5)规划型ES; (6)监视型ES; (7)控制型ES; (8)调试型ES; (9)教学型ES; (10)修理型ES

制定用于新产品或新系统的调试,也可用于维修站对被修设备的调整量与试用于电子材料大数诊断等

机做厂做双耳、生厂上乙双耳寺的专家系统为

除了上述10种ES类型外，还有诸如决策型和管理型的ES。

决策型ES是对各种可能的决策方案进行综合评判和选优的一类ES。它同时具有解释、诊断、预测、规划等功能，并能对相应领域中的问题做出辅助决策和对决策做出解释。

管理型ES是在管理信息系统和办公自动化系统的基础上发展起来的一类ES。

# 专家系统概念

## 2. 按系统的体系结构分类

- (1) 集中式ES：对知识和推理进行集中管理的一类ES。目前，得到成功应用的ES多属于这一类型。
- (2) 分布式ES：指具有分布处理能力以及多ES、多知识元的协同处理能力的ES。  
系统中数据、知识及控制不但在逻辑上，而且在物理上都是分布的；  
系统中诸机构通过计算机网络实现互连，这些机构之间相互协作，可以求解单个机构难以解决，甚至不能解决的问题。  
目前，基于多Agent的研究是分布式ES研究的重点。
- (3) 神经网络ES：运用人工神经网络技术建造的ES。
- (4) 符号系统与神经网络相结合的ES：把基于连接主义的神经网络与基于符号处理的ES有机结合形成的一种混合型ES。这种ES可以充分发挥神经网络与符号系统的优点，克服它们的缺点，实现优势互补。

# 专家系统概念

## 三、ES的特点

在总体上，ES具有以下一些共同特点；

- (1) 可具有一个或多个专家的知识 and 经验，能以接近于人类专家的水平在特定领域工作。
- (2) 能高效、准确、迅速地工作，不会像人类专家那样产生疲倦和不稳定。
- (3) 使人类专家的领域知识突破了时间和空间的限制，ES程序可永久保存，并可复制任意多的副本或在网上供不同地区或不同部门的人们使用。
- (4) 能进行有效推理，包括各种精确性推理和非精确性推理等。
- (5) 具有透明性，能以可理解的方式解释推理过程。
- (6) 具有自学习能力，可总结规律，不断扩充和完善系统自身。
- (7) 能提高生产率，产生巨大的社会效益、经济效益等。
- (8) 对推动人工智能等其他学科的发展具有重大的作用。

# ES的基本结构

## 一、知识库

知识库是ES的知识存储器，用来存放求解问题的领域知识。

对领域问题的专家知识，首先需要用相应的知识表示方法将其表示出来，然后再进行形式化，并经编码放入知识库中。

知识库中的知识分为两大类型：一类是领域中的事实，称为事实性知识，这是一种广泛公用的知识，也即写在书本上的知识及常识；另一类是启发性知识，它是领域专家在长期工作实践中积累起来的经验总结。

领域专家所拥有的经验性、判定性知识，实际上是一种直觉性和诀窍性的知识。在问题求解过程中，这种知识是最难获得的，原因是不少专家很少意识到自己是如何使用这些知识解决问题的，甚至没有意识到自己在解决问题时究竟使用了多少这样的知识，而且让他们把这些直觉、诀窍、经验讲出来，本身就是一件比较困难的事情。但是，这些知识又恰恰是知识库的核心部分。

# ES的基本结构

## 二、数据库

数据库也称为全局数据库或综合数据库，用来存储有关领域问题的**事实、数据、初始状态(证据)**和推理过程中得到的各种中间状态及目标等。它相当于ES的工作存储器，用来存放用户回答的事实、已知的事实和由推理得到的事实。

随着问题的不同，数据库的内容也可以是动态变化的。数据库存放的是该系统当前要处理对象的一些事实。

例如，在医疗ES中，数据库存放的仅是当前患者的情况，如姓名、年龄、症状等及推理过程中得到的一些中间结果、病情等；

在气象ES中，数据库存放的是当前气象要素，如云量、温度、气压以及推理得到的**中间结果**等。

# ES的基本结构

## 三、推理机

推理机是一组用来控制、协调整个ES的程序。它根据数据库当前输入的数据，利用知识库中的知识，按一定的推理策略，去求解当前的问题、解释外部输入的事实和数据，推导出结论并向用户提示等。

推理机所采用的推理方法可以是正向推理、逆向推理、或正逆向结合的双向推理，并且，在这三种推理方式中，都包含有精确推理和非精确推理。

对于一些大中型ES，由于其知识库中的知识数量很多，因此其推理机制由知识库管理系统和推理机两个主要部分组成。其中，知识库管理系统实现对知识库中知识的合理组织和有效管理，并能根据推理过程的需求去搜索、运用知识和对知识库中的知识做出正确的解释；而推理机则主要用于生成并控制推理的进程和使用知识库中的知识。

推理机和知识库相分离，是ES的一大特点。这不仅便于对知识库的管理，而且还可以实现具有可塑性、通用性的系统。

# ES的基本结构

## 四、解释机构

ES应该能够以用户便于接受的方式解释自己的推理过程。

例如，回答用户提出的“为什么？”，给用户说明“结论是如何得出的？”等。通过这种解释，既可以使ES更取信于用户，又可以帮助系统建造者发现知识库及推理机中的错误。

解释机构实际上也是一组程序，它包括系统提示、人机对话、能书写规则的语言以及解释部分程序，其主要功能是解释系统本身的推理结果，回答用户的提问，使用户能够了解推理的过程及所运用的知识和数据。目前，大多数ES的解释机构都采用人机对话的交互式解释方法。

# ES的基本结构

## 五、知识获取机构

知识获取是ES的一种辅助功能，它可为修改知识库中的原有知识和扩充新知识提供相应手段。

其基本任务是把知识加入到知识库中，并负责维持知识的一致性及完整性，建立起性能良好的知识库。

在不同ES中，知识获取的功能和实现方法差别较大，有的系统首先由知识工程师向领域专家获取知识，然后再通过相应的知识编辑软件把知识送到知识库中；

有的系统自身就具有部分学习功能，由系统直接与领域专家对话获取知识；

有的系统具有较强的学习功能，可在系统运行过程中通过归纳、总结，得出新的知识。

无论采取哪种方式，知识获取都是目前ES研制中的一个重要问题。



# ES的基本结构

## 六、用户界面

用户界面是ES的另一个关键组成部分，它作为ES与外界的接口，主要用于系统和外界之间的通讯与信息交换。

ES的使用者包括最终用户、领域专家、知识工程师。

最终用户和领域专家一般都不是计算机专业人员，因此用户界面必须适应非计算机人员的需求，不仅应把系统的输出信息转换为便于用户理解的形式，而且还应使用户能方便地操纵系统运行。

用户界面应尽可能地拟人化，尽可能地使用接近于自然语言的计算机语言，并能理解和处理声音、图像等多媒体信息。

# 如何构建专家系统

建立专家系统的一般步骤如下：

**(1) 设计初始知识库。**知识库的设计是建立专家系统最重要和最艰巨的任务。初始知识库的设计包括：

**①问题知识化**，即辨别所研究问题的实质，如要解决的任务是什么，它是如何定义的，可否把它分解为子问题或子任务，它包含那些典型数据等。

**②知识概念化**，即概括知识表示所需要的关键概念及其关系，如数据类型、已知条件（状态）和目标（状态）、提出的假设以及控制策略等。

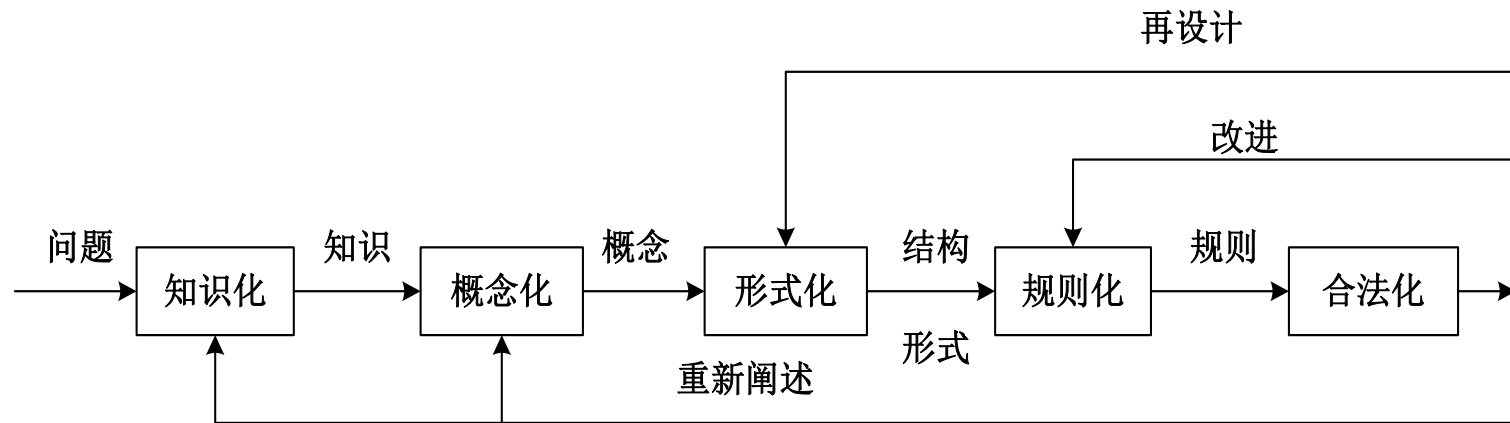
**③概念形式化**，即确定用来组织知识的数据结构形式，应用人工智能中各种知识表示方法把与概念化过程有关的关键概念、子问题及信息流特性等变换为比较正式的表达，它包括假设空间、过程模型和数据特性等。

**④形式规则化**，即编制规则把形式化了的知识变换为由编程语言表示的可供计算机执行的语句和程序。

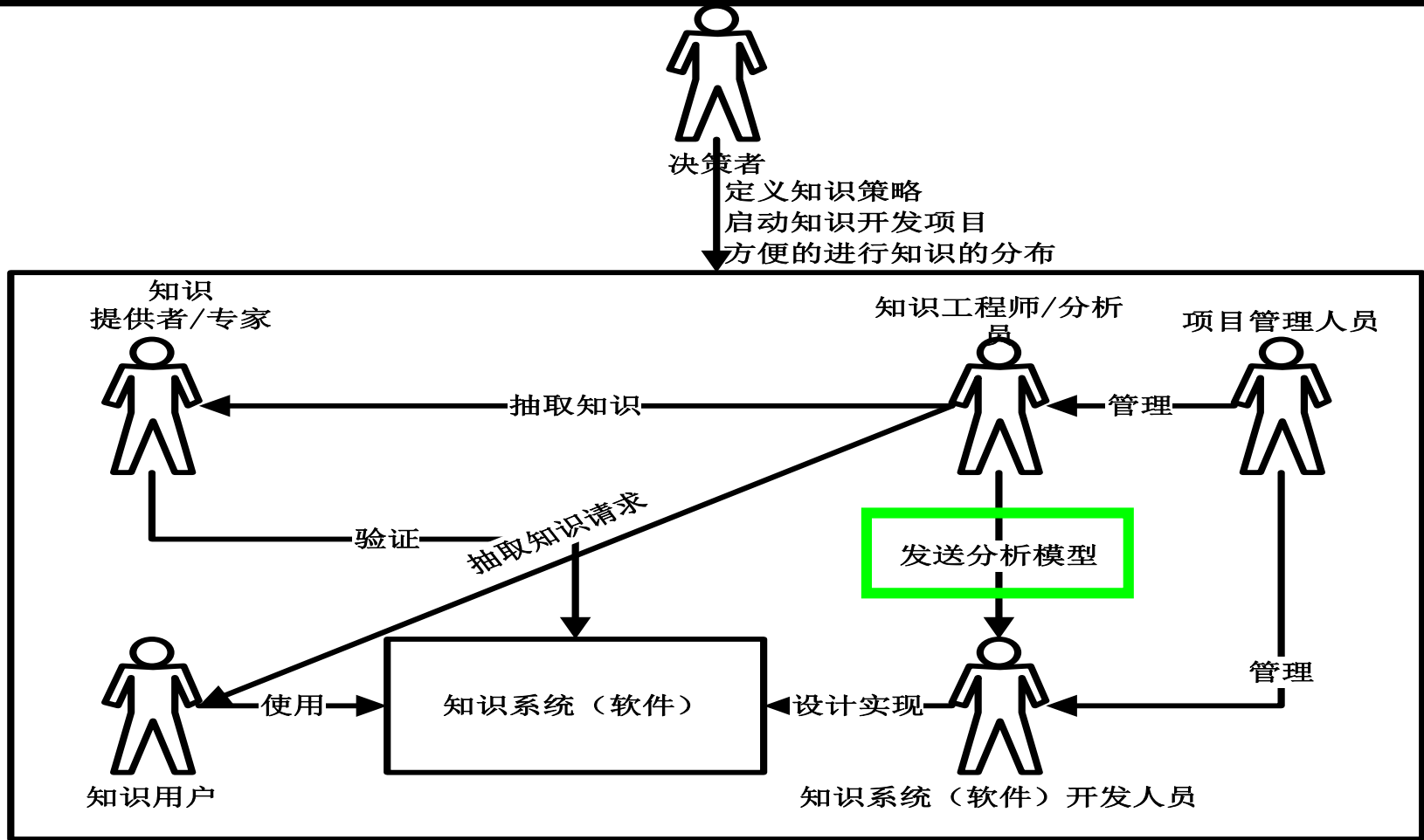
**⑤规则合法化**，即确认规则化了的知识的合理性，检验规则的有效性。

**(2) 原型机的开发与试验**。在选定知识表达方法之后，即可着手建立整个系统所需要的试验子集，它包括整个模型的典型知识，而且只涉及与试验有关的足够简单的任务和推理过程。

**(3) 知识库的改进与归纳。**反复对知识库及推理规则改进试验，归纳出更完善的结果。经过相当长时间（数月至2、3年）的努力，使系统在一定范围内达到人类专家水平。



# 专家系统的建设过程



# ES的开发语言与环境

## 专家系统开发工具

- 人工智能语言：LISP, Prolog, C/C++, Java
- 专家系统外壳:又称为骨架系统，从成功应用的专家系统演变而来的。  
如：

MYCIN----->E-MYCIN

PROSPECTOR的知识获取系统----->KAS

- 通用型专家系统工具：
  - OPS系列, CLIPS (JESS)：规则型
  - ART：规则+框架
- 专家系统开发环境：ES开发的工具包，提供多种方便的构件

# ES的开发语言与环境

## 一、程序设计语言

程序设计语言包括通用程序设计语言和人工智能语言。它们是开发专家系统的最基础的语言工具。

作为通用程序设计语言的主要代表有C、PASCAL、ADA等；

作为人工智能语言的主要代表有以 LISP为代表的函数型语言，以PROLOG为代表的逻辑型语言，以及C++、JAVA等为代表的面向对象语言。

LISP(List Processing Language)是1960年研究出来的，在专家系统发展的早期，有许多著名的专家系统都是用这种语言开发出来的。例如 MYCIN和PROSPECTOR（勘探者）等。

# ES的开发语言与环境

---

PROLOG (Programming in Logic) 语言是1972年研制成功的一种逻辑程序设计语言。由于它具有简洁的文法以及一阶逻辑的推理能力，因而被广泛地应用于人工智能的许多研究领域。

C++ 语言既是一种通用程序设计语言，又是一种很好的人工智能语言，它以其强大的功能和面向对象特征，在人工智能中得到了广泛的应用。

已有不少人直接用它来开发专家系统或各种专家系统工具。尤其是 Visual C++ 的发展，为专家系统对多媒体信息的处理、可视化界面的设计、基于网络的分布式运行等提供了一种很好的语言环境。

对于基于网络的分布式多专家协同专家系统的开发，Java语言也是值得考虑的一种语言工具。



# ES的开发语言与环境

## 二、开发环境

在国外已有一批较有影响的专家系统开发环境，如 KEE、GUGU等。

KEE是由美国加州Intellicorp公司推出的一个集成化的专家系统开发环境，它把基于框架的知识表示、基于规则的推理、逻辑表示、数据驱动的推理、面向对象的程序设计等结合在一起，可以满足各个领域开发专家系统的需求。

GUGU是由微数据公司MDBS于1985年用C语言研制的一个功能很强的混合型专家系统开发环境，它包含关系数据库系统、标准的 SQL查询语言、远程通讯、多功能程序设计等多种功能。

# ES的开发语言与环境

国内由中国科学院、浙江大学、武汉大学等七个单位联合开发，于1990年完成了一个专家系统开发环境《天马》。该系统有4部推理机，即常规推理机、规划推理机、演绎推理机及近似推理机；有3个知识获取工具，即知识库管理系统、机器学习、知识求精；有4套人机接口生成工具，即窗口、图形、菜单及自然语言；有多种知识表示模式，如框架、规则及过程。

应用《天马》环境，已经开发出了一些专家系统，如暴雨预报专家系统、长沙旅游咨询专家系统、石油测井数据专家系统等。

《天马》是一个通用型，集成式的专家系统开发环境，是国家七五攻关的一项重大成果。自一九九零年问世以来，已经运用于农业，畜牧，铁路，医学，石油，煤矿，核能，气象，农机，水利，力学，资源，电力，旅游，国防等二十多个领域，产生了重要的经济和社会效益。获1992年国家科技进步一等奖。

# ES的开发语言与环境

- 1977年美国斯坦福大学计算机科学家费根鲍姆 (E. A. Feigenballm) 在第五届国际人工智能联合会议上提出知识工程的新概念。20世纪80年代以来, 在知识工程的推动下, 涌现出了不少专家系统开发工具, 例如EMYCIN、CLIPS (OPS5, OPS83)、G2、KEE、OKPS等。
- 1977年, 基于关幼波先生的经验, 研制成功了我国第一个“中医肝病诊治专家系统”。
- 1985年10月中科院合肥智能所熊范纶建成“砂姜黑土小麦施肥专家咨询系统”, 这是我国第一个农业专家系统。
- 中科院计算所史忠植与东海水产研究所等合作, 研制了东海渔场预报专家系统。在专家系统开发工具方面,
- 中科院数学研究所研制了专家系统开发环境“天马”,
- 中科院计算所研制了面向对象专家系统开发工具“OKPS”。

---

## **3 .基于规则的专家系统**

### **3.1 基于规则的专家系统的工作模型**

### **3.2 基于规则的专家系统的结构**

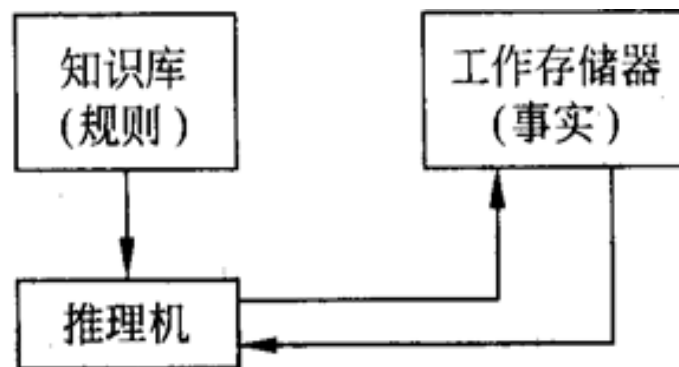
### 3. 基于规则的专家系统

#### 3.1 基于规则的专家系统的工作模型

该模型使用一套包含在知识库内的规则对工作存储器内的具体问题信息(事实)进行处理, 通过推理机推断出新的信息。通常包括以下模块:

- (1) **知识库**。以一套规则建立的长期存储器模型。
- (2) **工作存储器**。建立短期存储器模型, 存放问题事实和由规则激发而推断出的新事实。
- (3) **推理机**。借助于把存放在工作存储器内的问题事实和存放在知识库内的规则结合起来, 建立推理模型, 以推断出新的信息。

推理机作为推理模块, 并把事实与规则的先决条件进行比较, 确定哪条规则能够被激活。通过这些激活规则, 推理机把结论加进工作存储器, 并进行处理, 直到再没有其他规则的先决条件能与工作存储器内的事实相匹配为止。

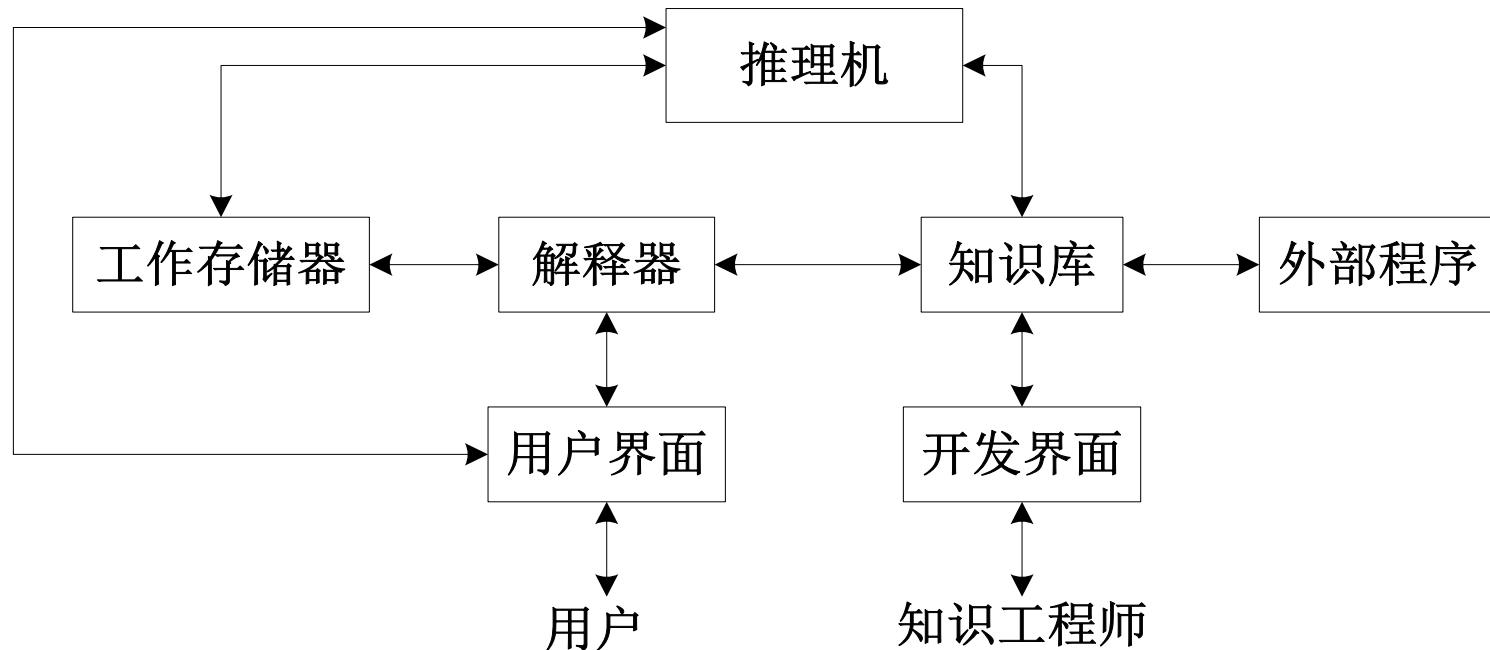


## 3.2 基于规则的专家系统的结构

一个基于规则的专家系统的完整结构组成如下：

- (1) 知识库。
- (2) 工作存储器。
- (3) 推理机。
- (4) 用户界面(接口)。用户通过该界面来观察系统, 并与系统对话(交互)。
- (5) 开发(者)界面。知识工程师通过该界面对专家系统进行开发。
- (6) 解释器。对系统的推理提供解释。

**(7) 外部程序。**如数据库、扩展盘和算法等,对专家系统的工作起支持作用。它们应易于被专家系统所访问和使用。





---

## **4 .基于框架的专家系统**

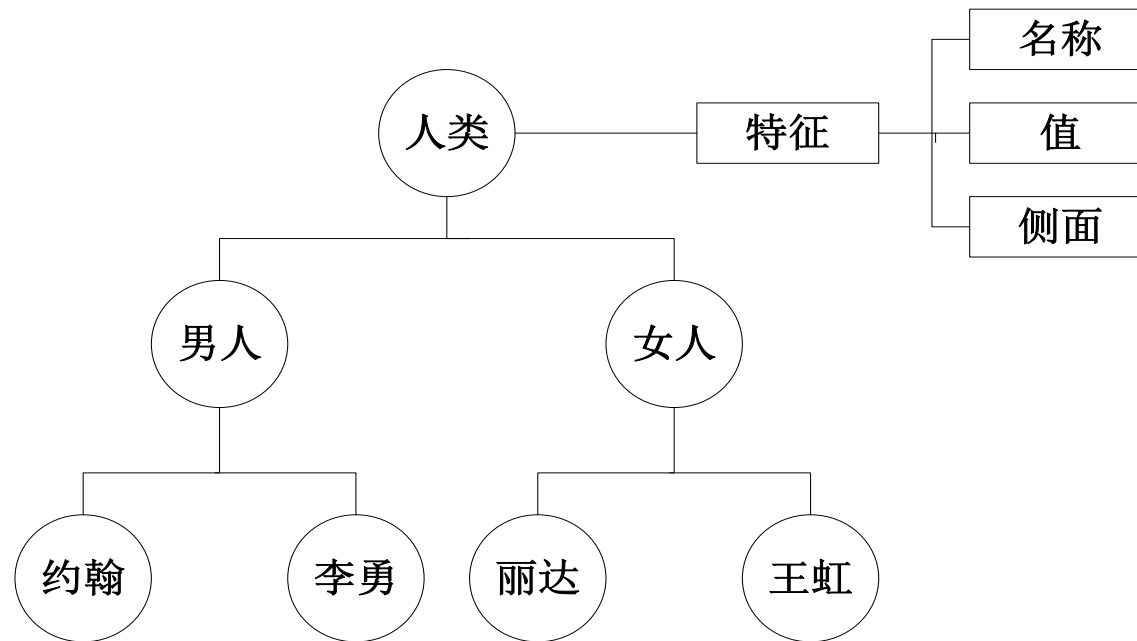
### **4.1 基于框架的专家系统的结构**

### **4.2 基于框架的专家系统的一般设计方法**

## 4. 基于框架的专家系统

### 4.1 基于框架的专家系统的结构

与基于规则的专家系统的定义类似, 基于框架的专家系统也是个计算机程序, 该程序使用一组包含在知识库内的框架对工作存储器内的具体问题信息进行处理, 通过推理机推断出新的信息。



## 4.2 基于框架的专家系统的一般设计方法

开发基于框架的专家系统的主要任务如下：

- (1) 定义问题, 包括对问题和结论的考察与综述。
- (2) 分析领域, 包括定义事物、事物特征、事件和框架结构。
- (3) 定义类及其特征。
- (4) 确定框架结构。
- (5) 确定模式匹配规则。
- (6) 规定事物通信方法。
- (7) 设计系统界面。
- (8) 对系统进行评价。
- (9) 对系统进行扩展, 深化和扩宽知识。

**基于框架的专家系统能够提供基于规则的专家系统所没有的特征,如继承、侧面、信息通信和模式匹配规则等,因而也就提供了一种更加强大的开发复杂系统的工具。也就是说,基于框架的专家系统具有比基于规则的专家系统更强的功能,适用于解决更复杂的问题。**

---

## **5.基于模型的专家系统**

### **5.1基于模型的专家系统的提出**

### **5.2基于神经网络的专家系统**

## 5. 基于模型的专家系统

### 5.1 基于模型的专家系统的提出

前面讨论过的**基于规则的专家系统**和**基于框架的专家系统**都是采用规则逻辑或框架逻辑,并以逻辑作为描述启发式知识的工具而建立的计算机程序系统。

综合各种模型的专家系统无论在知识表示、知识获取还是知识应用上都化那些基于逻辑心理模型的系统具有更强的功能,从而有可能显著改进专家系统的设计。

在诸多模型中,人工神经网络模型的应用最为广泛。早在1988年,就有人把神经网络应用于专家系统,使传统的专家系统得以发展。

## 5.2基于神经网络的专家系统

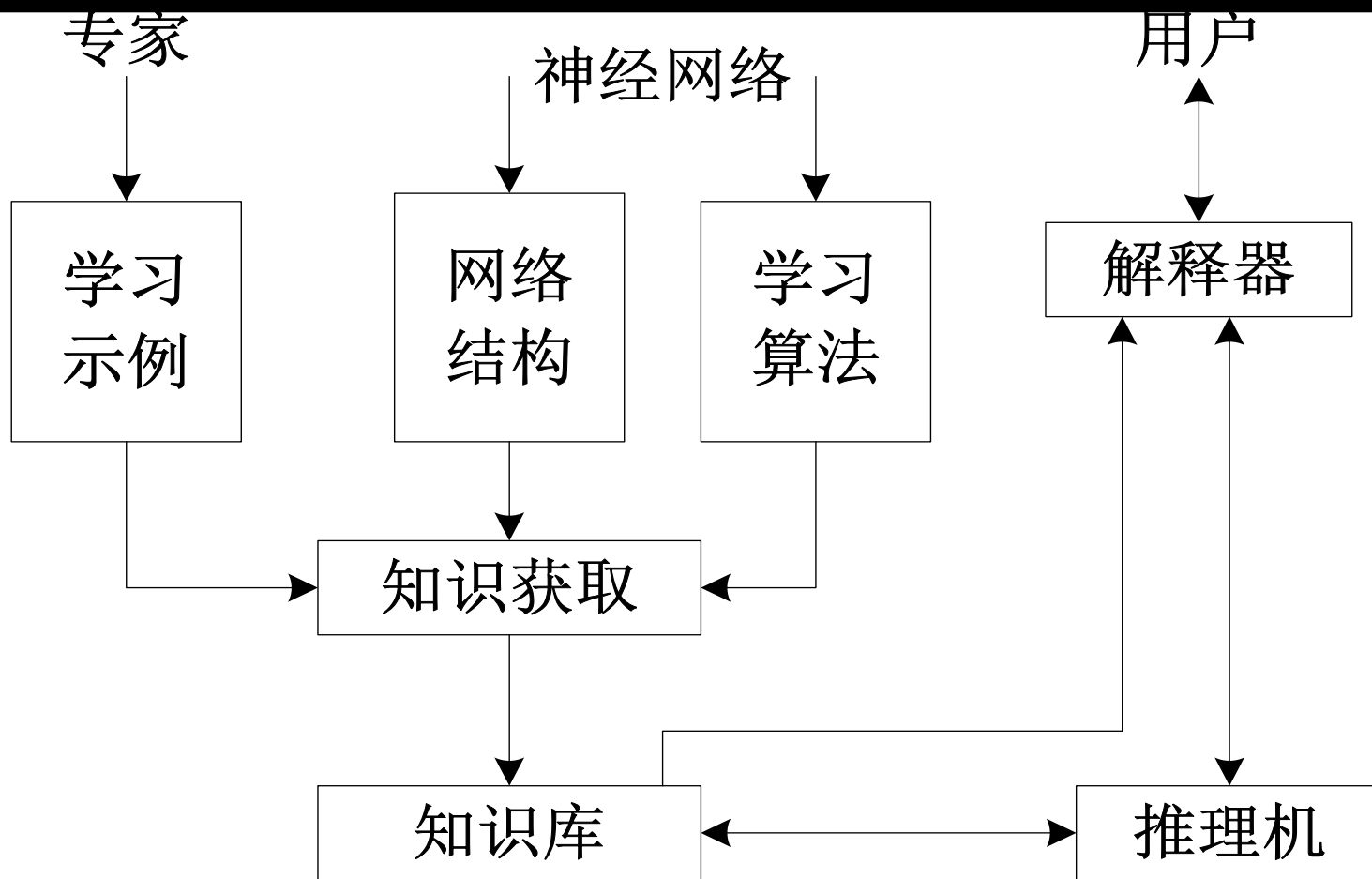
知识库由一些知识元构成, 知识元可以是一个神经网络模块, 也可以是一组规则或框架的逻辑模块。只要对神经网络的输入转换规则和输出解释规则给予形式化表达, 使之与外界接口及系统所用的知识表达结构相似, 则传统的推理机制和调度机制就都可以直接应用到专家系统中去。神经网络与传统专家系统的集成, 协同工作, 优势互补。根据侧重点的不同, 其集成有三种模式:

**(1) 神经网络支持专家系统。**以传统的专家系统为主, 以神经网络的有关技术为辅。例如对专家提供的知识和样例, 通过神经网络自动获取知识。又如运用神经网络的并行推理技术以提高推理效率。

**(2) 专家系统支持神经网络。**以神经网络的有关技术为核心, 建立相应领域的专家系统, 采用专家系统的相关技术完成解释等方面的工作。

**(3) 协同式的神经网络专家系统。**针对较大的复杂问题, 将其分解为若干子问题, 针对每个子问题的特点, 选择用神经网络或专家系统加以实现, 在神经网络和专家系统之间建立一种藕合关系。





# ES实例

## 5.4.1 动物识别专家系统

### 1. 动物识别专家系统的知识库

用产生式系统鉴别动物，需要一种演绎机制，利用已知事实的集合做出新的结论，一种方法是替动物园中的每一个动物作一个产生式，使用者首先收集所有可利用的事实，然后在产生式的表中进行扫描，寻找一个在条件部分能与之匹配的产生式。

一般要经过多步并生成和利用一些中间事实才能从基本事实推出结论，这样做，所包含的产生式可以比较小，容易理解，容易使用和容易产生。

# ES实例

## 1. 知识库中10条规则用Prolog语言表示

后面是前件

It-is (“哺乳动物”) : - positive (“产奶”).

It-is (“哺乳动物”) : - positive (“有毛发”).

前件：肯定的，事实，证据

It-is (“食肉动物”) : - positive (“吃肉”).

It-is (“食肉动物”) : - positive (“有犬齿”), positive (“有爪”), positive (“眼前视”).

It-is (“有蹄动物”) : - positive (“有蹄子”), it\_is (“哺乳动物”).

It-is (“有蹄动物”) : - positive (“反刍食物”), it\_is (“哺乳动物”).

animal\_is (“老虎”) : - it\_is (“哺乳动物”), it\_is (“食肉动物”), positive (“黄褐色”), positive (“有黑条纹”).

animal\_is (“金钱豹”) : - it\_is (“哺乳动物”), it\_is (“食肉动物”), positive (“黄褐色”), positive (“有暗斑点”).

animal\_is (“斑马”) : - it\_is (“有蹄动物”), positive (“有黑条纹”).

animal\_is (“长颈鹿”) : - it\_is (“有蹄动物”), positive (“有长脖子”), positive (“有长腿”), positive (“有暗斑点”).

# ES实例

## 2. 数据库

在计算机中留出一些存储区间，以存放反映系统当前状态的事实，存放用户回答的事实、已知的事实和由推理而得的事实，即由已知事实推导出的假设成立时，也作为事实。

如用下列语句可通过问答建立动物识别专家系统动态数据库：

**positive(X): – ask(X).**

**ask(X): – write(X, ” 吗 ? ” ), readln(Reply), remember(X, Reply).**

当内部目标和 **animal\_js(“老虎”)** 匹配时，首先要满足条件 **it\_is(“哺乳动物”)**，也即要满足 **positive(“产奶”)**，这时搜索数据库没有发现这个事实，系统立即转入询问，通过谓词 **readln(Reply)** 从键盘输入 “yes” 或 “no”，用自定义谓词 **remember(X, Reply)** 来肯定产奶或否定产奶，从而达到增加某个新数据或否定某个数据的目的。

# ES实例

## 3. 知识表示

知识的表达形式应该自然简单、精确，又容易存取。

在Prolog专家系统中，知识通常用以下两种方法之一来表示：

- 把事实和数据(事实性知识)分类，放到Prolog规则库中。这种表示实用于基于规则的专家系统。在基于规则的专家系统中，肯定的结果是一个产生式规则的动作，这些产生式规则由输入数据激活。
- 把事实和数据构成子句，形成子句知识库。这种子句表示实用于基于逻辑的专家系统，在基于逻辑的专家系统中，知识库是由说明事实的谓词逻辑子句组成，有一个可以由输入数据流激活的规则集，如上例的动物识别专家系统的四种动物可用如下Prolog语句表示：

# ES实例

rule (1, “老虎”, [1, 2, 3, 4] ).  
rule (2, “金钱豹”, [1, 2, 3, 5] ).  
rule (3, “斑马”, [4, 6] ).  
rule (4, “长颈鹿”, [5, 6, 7, 8] ).  
condition (1, “哺乳动物” ).  
condition (2, “食肉动物” ).  
condition (3, “黄褐色” ).  
condition (4, “有黑条纹” ).  
condition (5, “有暗斑点” ).  
condition (6, “有蹄动物” ).  
condition (7, “有长脖子” ).  
condition (8, “长腿” ).

**Condition: 条件**

# ES实例

## 4. 推理机

推理机是一组函数，本例既有正向推理机又有反向推理机，都是用精确推理。

### 1)正向推理

用户首先提供一批事实，存放到数据库中，然后推理机进行工作。

① 推理机用这批事实与知识库中规则的前提进行匹配。

② 把匹配成功的规则的结论部分作为新的事实加到数据库中去。再用更新后的数据库中的所有事实，重复上述①、②二步，如此反复进行，直到得出结论(答案)或不再有新的事实加到数据库为止。

例如，用户输入一批事实：动物有暗斑点、长脖子、长腿、产奶、有蹄子，要求系统判断这是一个什么动物？

# ES实例

推理机利用这批事实来匹配规则, 方法一般是将第一条规则的前提部分取出来, 验证一下这些前提是否在数据库中, 若都在, 匹配成功, 不然就取第二条规则进行匹配, 如此下去。

首先, 成功地匹配了第一条规则`it_is(“哺乳动物”)`, 于是把该规则的结论“哺乳动物”加到数据库中去;

接着又成功地匹配了规则`it_is(“有蹄动物”)`, 因为数据库中有: 哺乳动物及有蹄子, 满足该规则的前提, 于是把该规则的结论“有蹄动物”加到数据库中去;

接着又成功地匹配了规则`animal_is(“长颈鹿”)`, 于是得出结论:

这个动物是长颈鹿。

当一条规则匹配成功之后, 不仅将其结论部分加入到数据库, 而且还要记录下这条规则, 当推理结束时, 以备解释显示调用规则的轨迹。



# 正向推理

设计一个正向推理机就是设计一组程序，使其至少具有以下功能：

① 能用数据库中的事实去匹配规则的前提，若匹配不成功，能自动地进行下一条规则的匹配。

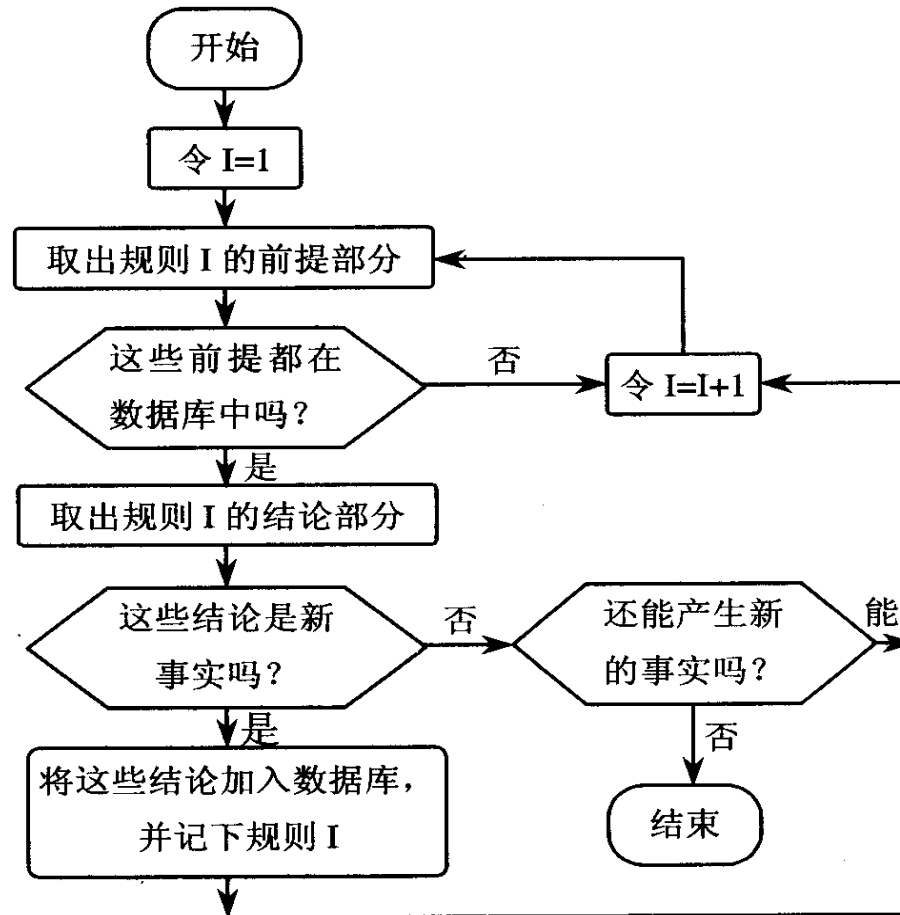
如何匹配最为合适，是设计专家系统者根据专业特点和知识表示等情况，需要很好考虑的问题，也就是在匹配时到底用什么策略等问题都需要考虑周全；

② 若某条规则匹配成功了，系统能将此规则的结论部分自动加入数据库；

③ 能判断何时应结束推理；

④ 能将匹配成功的规则记录下来；

# 正向推理



正向推理机示意图

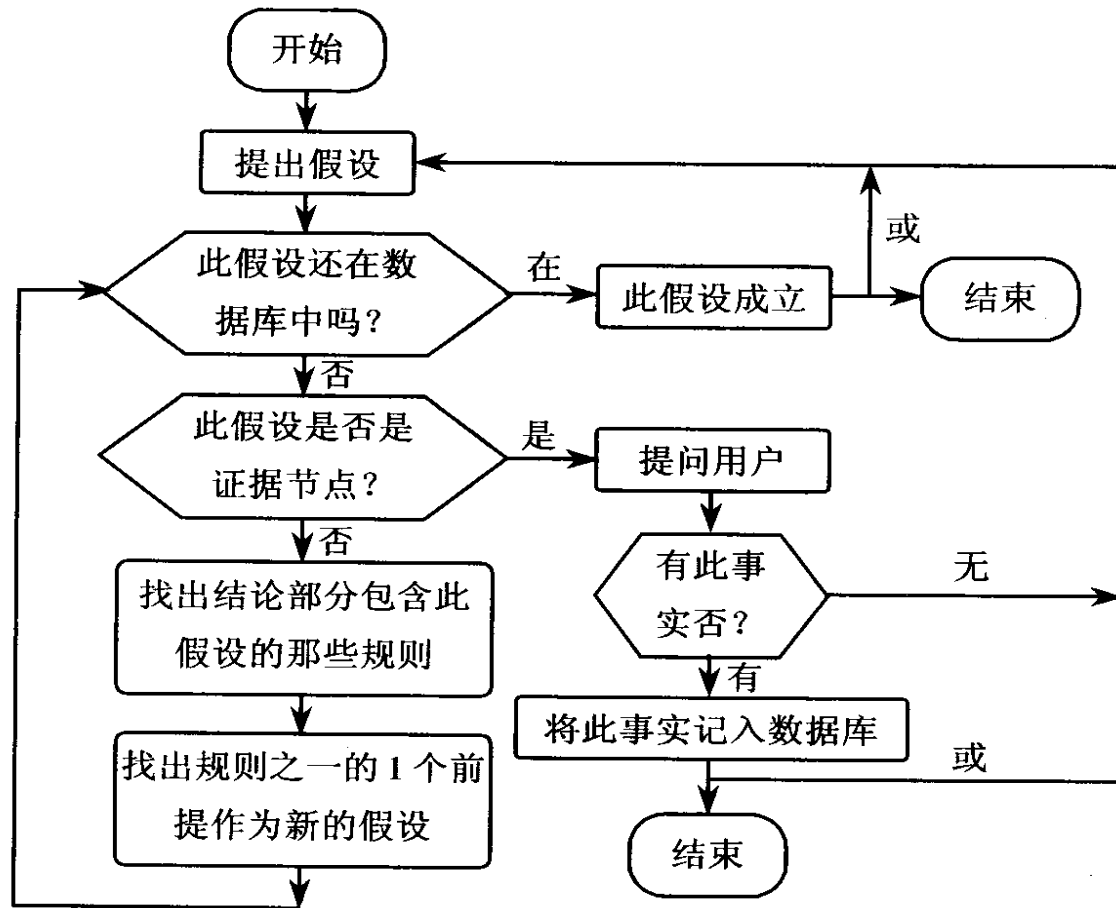
# 反向推理

## 2)反向推理

由用户或系统首先提出一批假设，然后系统逐一验证这些假设的真假性，方法是：

- ① 看假设是否在数据库中，若在，则假设成立，推理结束或进行下一个假设的验证，否则，进行下一步。
- ② 判断这些假设是否是证据节点，若是，系统提问用户，否则进行下一步。
- ③ 找出结论部分包含此假设的那些规则，把这些规则的所有前提作为新的假设。
- ④ 重复①、②、③步。

# 反向推理



反向推理机示意图

# 反向推理

例如，用户要求系统验证一下该动物是否是老虎(此时数据库中什么也没有)。

系统首先检查数据库，看其中有没有老虎这个事实，现在没有，即不满足第一步；

其次检查此假设“老虎”是否是证据节点，得出结论：不是，即不满足第二步；

进行第三步，在知识库中找出结论部分含有“老虎”的规则，结果找出规则 `animal_is(“老虎”)`。

第四步：将该规则的四个前提条件：**1.哺乳动物、2.食肉动物、3.黄褐色和4.有黑条纹**都作为新的假设，逐一验证这些新假设。

# 反向推理

首先验证“**1.哺乳动物**”是否为真。前二步都不满足，即数据库中既没有“哺乳动物”，又不是证据节点。

那么进行第三步，系统查出：它是如下规则的结论部分。

it\_is(“哺乳动物”): - positive(“产奶”)

it\_is(“哺乳动物”): - positive(“有毛发”)

于是将该两条规则的前提：“有毛发”和“产奶”作为新的假设进行验证。

先验证“有毛发”是否为真，它不满足第一步。

当系统进行第二步时，查出“有毛发”是证据节点，于是系统就提问用户“此动物有毛发吗？”

# 反向推理

若用户回答“有”，则系统自动地将事实“动物有毛发”加入数据库，于是系统证实了此动物是哺乳动物，所以将“哺乳动物”这一事实自动加入数据库，并记录匹配成功的前一条规则(由于这两条规则都能推出是哺乳动物，若前一条规则为真，即已证实此动物是哺乳动物，这时系统能自动地不考察后一条规则的真假，否则，系统还要考察后一条规则)。

其次验证“2.食肉动物”是否为真。

前两步都不满足，进行第三步时，系统查出它是如下规则的结论部分，

`it_is(“食肉动物”): - positive(“吃肉” )`

`it_is(“食肉动物”): - positive(“有犬齿” ),positive(“有爪” ),  
positive(“眼前视” )`

于是将该两条规则的前提部分“吃肉”和“有犬齿”、“有爪”、“眼前视”都作为新的假设来进行逐一验证；

# 反向推理

先验证前一规则的前提“吃肉”是否为真，由于它是证据节点，系统问用户：“此动物吃肉吗？”，若用户回答“是”，至此，系统已证实了此动物是“食肉动物”，于是系统将此事实自动加入数据库，并记录匹配成功的前一规则。

再验证“3. 黄褐色”是否为真。因为“黄褐色”是证据节点，所以系统问用户：“该动物是黄褐色的吗？”若用户回答“是”，则系统自动地将此事实“黄褐色”加入数据库。



# 反向推理

最后验证“4.有黑条纹”是否为真。因为“有黑条纹”是证据节点，所以系统问用户：“该动物有黑条纹吗？”用户回答“有”，则系统将此事实“有黑条纹”自动地加入数据库。

至此，系统证实了此动物是“老虎”。此时，系统将此事实“老虎”自动地加入数据库，并记录此匹配成功的规则`animal_is(“老虎”)`。

如果上述假设“老虎”被否定，那么系统能提出下一个假设，比方是“金钱豹”来进行验证。

直至某一个假设成立，系统就结束推理。

若所有假设都验证了且都不成立，此时系统回答：“不认识此动物”。

# 反向推理

从上面例子可知，要设计一个反向推理机，至少要具有以下功能：

- ① 能根据用户要求或情况提出假设；
- ② 能验证此假设是否是在数据库中；
- ③ 能把知识库中将结论部分包含此假设的规则都找出来；
- ④ 能将找出来的规则的前提部分取出，并作为新的假设逐条验证；
- ⑤ 能判断假设是否是证据节点，若是，能向用户提出相应的问题，并记录结果；
- ⑥ 能将匹配成功的规则记录下来；
- ⑦ 能判断何时应结束推理。

# 反向推理

## 3)反向推理机的实现

positive(X): - xpositive(X), ! . 肯定的动态数据库

positive(X): - not(negative(X)), ! , ask(X).

negative(X): - xnegative(X), ! . 否定的动态数据库

ask(X): - write("该动物是" , X, " 吗 ? " ), readln(Reply),

加入谓词

remember(X, Reply).

remember(X, yes): - asserta(xpositive(X)).

remember(X, no): - asserta(xnegative(X)), fail.

clear\_facts: - retract(xpositive(\_)), fail.

clear\_facts: - retract(xnegative(\_)), fail.

取消谓词

推理机必须有管理用户输入的规则，并用产生式规则与其匹配，记录yes和no的回答，规则positive和negative用于将用户输入数据和产生式规则中的相应部分进行匹配。

# 反向推理

规则**remember**根据yes和no的回答生成断言，供匹配使用。

**clear\_facts**规则在识别-动作循环末端除去知识库项。这里**xpositive**和**xnegative**分别是肯定和否定的动态数据库，

**asserta**和**retract**分别是插入和删除动态数据库的内部谓词。

当系统搜索到没有某个事实，就会启动上述推理机(语句)，通过人机对话对事实进行肯定或否定，并把肯定或否定结果放入动态数据库，以保证问过的问题不会再问。

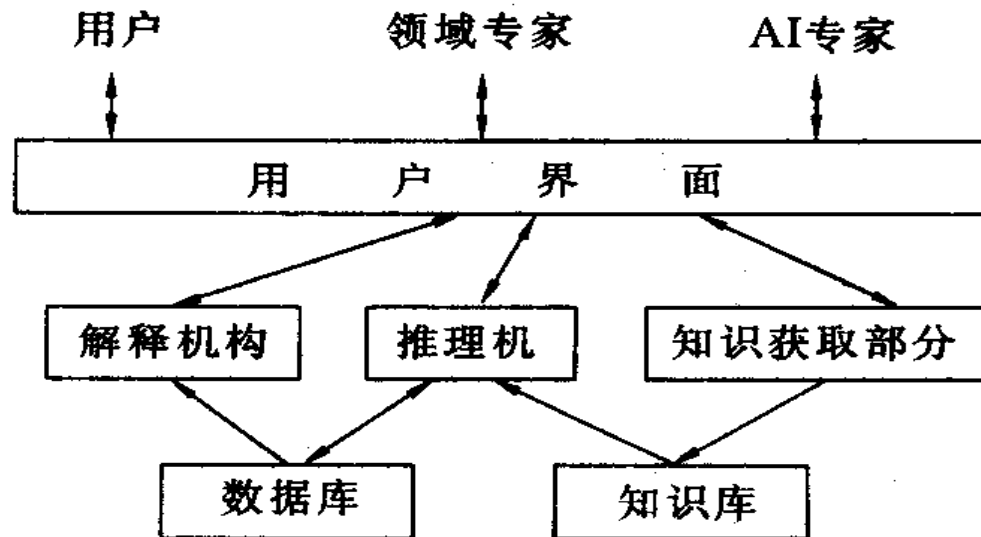
不管是正向推理机还是反向推理机，都是根据情况，用某种程序语言实现的。

# 回忆----ES的基本结构

## 6.2 ES的基本结构

ES的结构是指ES各组成部分的构造方法和组织形式。

一个最基本的ES应由知识库、数据库、推理机、解释机构、知识获取机构和用户界面6个部分所组成。



专家系统的基本结构

# ES实例

## 5. 解释部分

本例中仅能回答下列两个问题。

### (1) 怎么得出这个结论的？

系统回答这个问题是通过打印(或屏幕显示)展示推导这个结论所用的规则。

例如若用户问：“为什么说该动物是金钱豹呢？”系统回答：“由于你告诉动物有毛发，它是哺乳动物；又由于你告诉它吃肉，所以它又是食肉动物；由于它是哺乳动物、食肉动物，且你告知是黄褐色的，又有暗斑点，所以它是金钱豹。”

### (2) 为什么问我这个问题？

系统根据推理过程，解释问这个问题的原因。

若用户提出：“为什么问我：动物吃肉？”系统就会回答：“我为了推出该动物是金钱豹，我首先看它是否是哺乳动物，根据你的回答，这个结论是对的，于是我要考虑它是否是食肉动物，因为只要它吃肉，它就是食肉动物，所以我要问你这个问题。”

# ES实例

对于一般的专家系统，它到底应该能回答哪些问题？在设计专家系统时，就应该与领域专家商量好。因为到目前为止，计算机对自然语言的理解(即能听懂人说的话)，还是很有局限性的。因此计算机只能回答你预先设计好的问题。

专家系统的解释部分，至少应有以下功能：

- ① 能分辨出用户问的是什么问题；
- ② 能从推理记录中找出答案来。

## 6. 知识获取部分

知识获取部分也称学习功能，这是为了向专家获取新规则或修改已有的不适用或有错误的规则。通过这个手段，可以不断增加知识库中的知识，使系统能做更多、更复杂的事，且做得更好。

# ES实例

## 7. 动物识别专家系统的Prolog程序

/\*该程序是动物识别专家系统,程序名为 DWSB.PRO \*/

database

    xpositive(symbol)

    xnegative(symbol)

predicates

    run

    animal\_is(symbol)

    it\_is(symbol)

    positive(symbol)

    negative(symbol)

    clear\_facts

    remember(symbol,symbol)

    ask(symbol)

goal

    run.

clauses



# ES实例

## 7. 动物识别专家系统的Prolog程序(续1)

```
run:-makewindow(1,7,7,"动物识别专家系统",1,5,22,60),animal_is(X),!,write("您要找的动物是:",X),nl,nl,clear_facts,clearwindow,exit.  
run:- write("不认识此动物"),clear_facts.  
positive(X):- xpositive(X),!.  
positive(X):- not(negative(X)),!,ask(X).  
negative(X):- xnegative(X),!.  
ask(X):- write("该动物",X,"吗?"),readln(Reply),remember(X,Reply).  
remember(X,yes):- asserta(xpositive(X)).  
remember(X,no):- asserta(xnegative(X)),fail.  
clear_facts:- retract(xpositive(_)),fail.  
clear_facts:- retract(xnegative(_)),fail.  
clear_facts:- write("按任意键退出"),readchar(_).  
it_is(“哺乳动物”):- positive(“产奶”).  
it_is(“哺乳动物”):- positive(“有毛发”).  
it_is(“食肉动物”):- positive(“吃肉”).
```

# ES实例

## 7. 动物识别专家系统的Prolog程序(续2)

it\_is(“食肉动物”):- positive(“有犬齿”), positive(“有爪”), positive(“眼前视”).

it\_is(“有蹄动物”):- positive(“有蹄子”), it\_is(“哺乳动物”).

it\_is(“有蹄动物”):- positive(“反刍食物”), it\_is(“哺乳动物”).

animal\_is(“老虎”):- it\_is(“哺乳动物”), it\_is(“食肉动物”), positive(“黄褐色”), positive(“有黑条纹”).

animal\_is(“金钱豹”):- it\_is(“哺乳动物”), it\_is(“食肉动物”), positive(“黄褐色”), positive(“有暗斑点”).

animal\_is(“斑马”):- it\_is(“有蹄动物”), positive(“有黑条纹”).

animal\_is(“长颈鹿”):- it\_is(“有蹄动物”), positive(“有长脖子”), positive(“有长腿”), positive(“有暗斑点”).

# ES实例

## 5.4.2 飞机结构布局设计专家系统

假设飞机总体确定了机翼的平面形状为三角翼。机翼展弦比为2.35，机翼相对厚度平均为4.5%。机翼外形由其4个顶点的(X,Z)坐标定义，坐标的单位为m。这4个顶点为

(1.219,0.9) (6.602,4.875) (7.352,4.875) (7.526,0.9)

机翼上安装有3个活动翼面，包括1个前缘襟翼和2个后缘副翼。活动翼面的外形也由顶点坐标定义，分别为

襟翼：(3.133,2.315), (6.229,4.6), (6.428,4.35), (3.898,2.315)。

内侧副翼：(2,6.644),(0.9,6.674),(2.602,7.451),(2.602,7.526,0.9)。

外侧副翼：(3,6.674,2.608),(6.713,4.875),(7.352,4.875),(7.451,2.608)。

目前的有限源分析软件大多具有这些功能

# ES实例

原始设计数据如下：

展弦比  $BWLB=2.35$ ;

平面形状  $BPTY=$ 三角翼;

翼身相对位置  $BWFP=$ 下单翼;

翼身连接方式  $BWFF=$ 集中连接;

平面轮廓  $BOUT=1.219,0.9,6.602,4.875,7.352,4.875,7.526,0.9$ ;

展长  $BWEL=9.75$ ;

后掠角  $BARL=53.56$ ;

根肋位置  $BRRD=0.9$ ;

根弦长  $BRRL=6.415$ ;

翼尖弦长  $BRTL=0.75$ ;

机翼面积  $BWSQ=40.46$ ;

最大使用过载  $BBGN=9$ ;

压心位置X  $BPXX=5.4$ ;

压心位置Z  $BPZZ=2.3$ ;

翼载  $BWPN=500$ ;

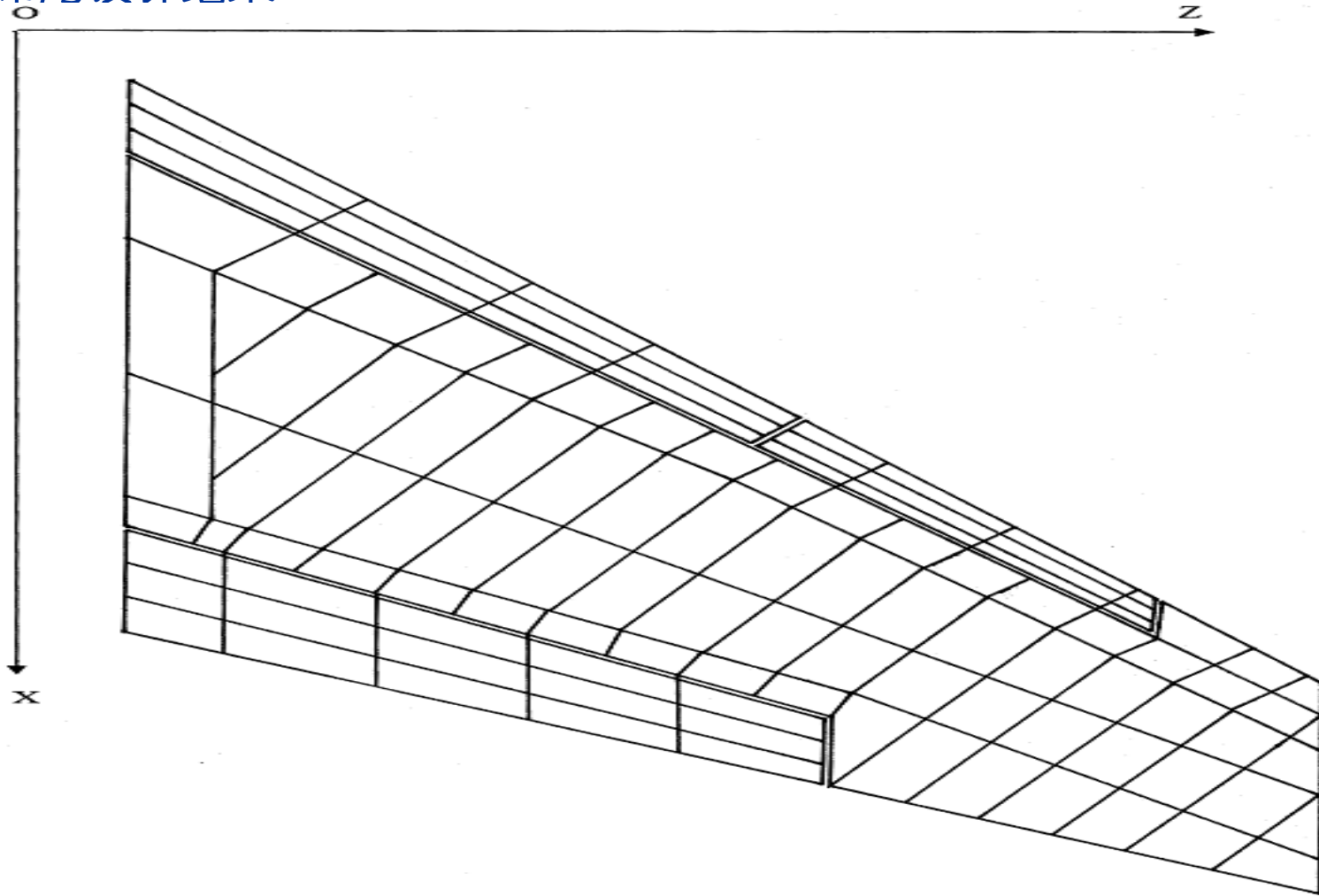
平均相对厚度  $BBRT=0.045$ ;

# ES实例

全机重心位置 BWCE=4.546,-0.081,0;  
翼型布置位置 BWID=0.90,101,1.112,7.529,  
活动翼面有无 BAWN=1;  
活动翼面个数 BAAN=3;  
机翼起落架有无 BWGR=0;  
翼内整体油箱有无 BWWT=1;  
翼内整体油箱个数 BTNN=1;  
装载要求有无 BWTT=1;  
操纵要求有无 BWST=0;  
活动翼面参数 BFLB=1,3.133,2.315,6.229,4.6,6.428,4.35,3.898,2.315,  
2,6.644,0.9,6.674,2.602,7.451,2.602,7.526,0.9,  
3,6.674,2.608,6.713,4.875,7.352,4.875,7.451,2.608;  
安装角 BWEA= 0;  
下反角 BBAA= 0;  
单位面积翼重 BUWW= 30;  
颤振速度 BWVV= 680;  
滚转效率 BREF=0.3;

# ES实例

布局设计结果



# ES实例

## 医疗诊断系统

- 专家系统的一个重要应用领域就是医疗诊断系统。早在一九七一年就由斯坦福大学的E. H. Shortiffe等研制了血液感染病医疗诊断系统MYCIN，它已成为成功的专家系统的一个典型。此外，世界上比较著名的医疗诊断系统还有青光眼医疗诊断系统CASNET，内科病医疗诊断系统INTERNIST，肾病医疗诊断系统PIP，处理精神病的系统PARRY等。我国的研究者根据我国的特点，在中医专家系统方面做了大量的工作，有一些已投入实际应用。

# ES实例

---

- MYCIN系统研制发起人E. H. Shortiffe(爱德华持·肖持利夫)是哈佛大学数学系毕业生，他获得了斯坦福大学面向医学的计算机应用方面的奖学金，到斯坦福大学当研究生。他在计算机科学和医学之间的边缘领域——医疗诊断的研究中，进行了开创性的工作。当他在一九七一年完成MYCIN系统时，他只是一名研究生，到了一九七九年成为斯坦福大学的内科副教授。一九八零年召开第二届医疗中的人工智能(AIM)学术会议时，他成为大会的组织委员会主席。



# ES实例

---

- MYCIN是有关传染病诊断和治疗的咨询系统。它能教会不擅长诊治传染病的医生，怎样从患者症状出发，确定病的种类及相应的治疗方法。
- 我们知道，传染病种类繁多，与其相应的抗生素种类也不少。要在限定的时间内确定病症，选择出恰当的治疗方法，决非易事。似乎这就是开发MYCIN系统的着眼点。

# ES实例

- MYCIN系统存放有大量传染病专家长期积累的知识，它们是肖特利夫与许多著名的传染病专家交谈，推理和总结得到的，他把这些知识归纳成200多条规则(后扩充至500多条)存放在计算机中，这些规则具有“如果…那么…”这种形式，称为产生式规则。这是目前专家系统使用得最广泛的推理方式之一。
- 当系统获得一个数据且与某个“如果…”相一致时(称为匹配)，则相应的“那末…”就代替了该数据，再继续搜寻是否存在与这个新数据匹配的“如果…”，这样一个过程含有“产生”、“做出”的含义，因此获得“产生式”的名子。
- 当使用MYCIN进行医疗诊断时，医生通过计算机的人机交互接口，将病人数据送入计算机，MYCIN系统将外来数据不断与内部知识进行匹配，直到获得最终结果。

# ES实例

## 训练数据例子分析

- **假定：**1000次行车记录，其中快车有200次，常速800次，共肇事10次，肇事记录中有七次是快车。
- 专家系统：先请教一位交通管理专家，将（历史）经验表达到知识库。 历史经验：“十次肇事九次快”，得出下列规则：
- 专家系统结论：
- **IF 肇事 THEN 开了快车 置信度90%**

# 课后习题

---

- 1 什么是专家系统？它有哪些基本特征？
  - 2 专家系统有哪些基本部分？每一部分的基本功能是什么？
  - 3 专家系统的开发过程是什么？
- 
- 考试时间：11月21日