



中国科学院自动化研究所
INSTITUTE OF AUTOMATION
CHINESE ACADEMY OF SCIENCES

专家系统

中国科学院自动化研究所
复杂系统管理与控制国家重点实验室

王丹力 研究员

2019/11/28



The State Key Laboratory of Management and Control for Complex Systems
复杂系统管理与控制国家重点实验室



内容

Contents

1. 专家系统概述
2. 专家系统基本原理
3. 专家系统分类
4. 专家系统设计——知识表示
5. 专家系统设计——推理机制
6. 专家系统开发工具
7. 专家系统应用实例



专家系统概述

- 专家系统 (expert system, ES) 是一类具有专门知识和经验的计算机智能系统，通过对人类专家的问题求解能力的建模，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。
- 这种基于知识的系统设计方法是以知识库和推理机为中心而展开的，即

专家系统 = 知识库 + 推理机

- 专家系统把知识从系统中与其他部分分离开来。专家系统强调的是知识而不是方法。很多问题没有基于算法的解决方案，或算法方案太复杂，采用专家系统，可以利用人类专家拥有丰富的知识，因此专家系统也称为基于知识的系统 (Knowledge-Based Systems)。

专家系统概述

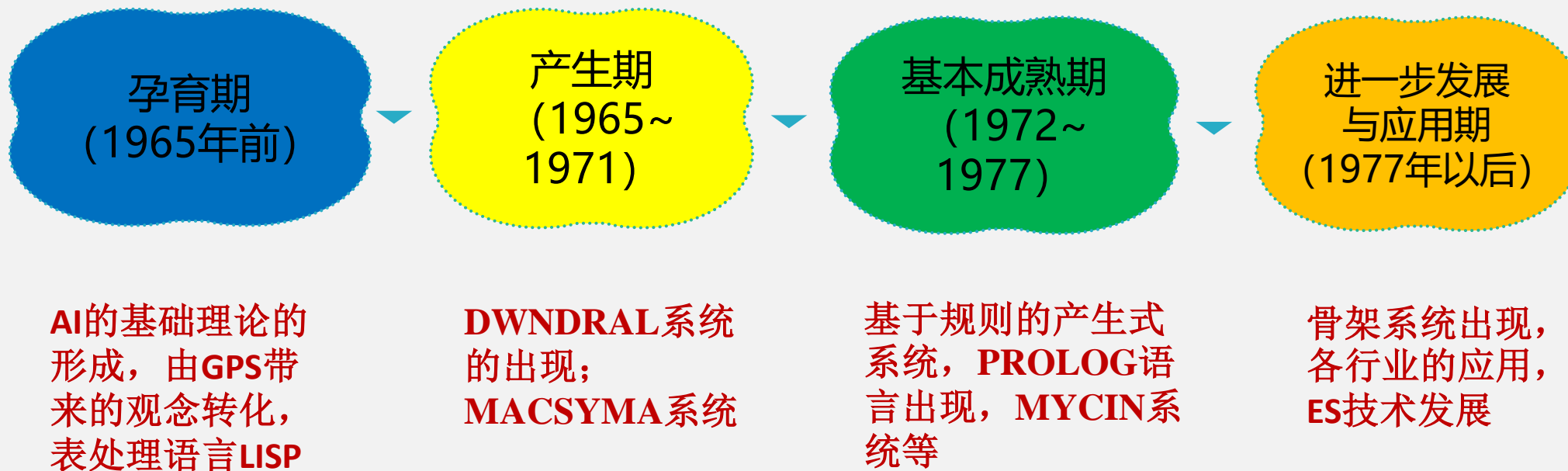
➤一个**专家系统**应该具备以下**三个要素**：

- (1) 具备某个应用领域的**专家级知识**；
- (2) 能**模拟专家的思维**；
- (3) 能达到**专家级的问题求解水平**。

➤ **“知识工程”**：建造一个专家系统的过程，它是把软件工程的思想应用于设计基于知识的系统。知识工程包括下面几个方面：

- (1) 从专家那里获取系统所用的知识（即知识获取）
- (2) 选择合适的知识表示形式（即知识表示）
- (3) 进行软件设计
- (4) 以合适的计算机编程语言实现。

专家系统产生和发展



专家系统产生和发展

➤ 第一阶段：孕育期（20世纪60年代前中期以前）

■ AI理论基础的奠定：1956年以前AI研究针对的具体问题：

- ✓ 逻辑理论机（定理证明）、西洋跳棋程序

■ 由A.Newell和H.A.Imans（1957年后）进行的通用问题求解程序（GPS），带来的AI理论成熟，研究价值：

- ✓ 发现一些通用问题求解技术和各种搜索策略；
- ✓ 认识到知识在智能行为中的地位；
- ✓ 研究开始从通用问题基于推理的模型转向专门问题基于知识的模型；

■ 表处理语言LISP出现：

- ✓ 1960年由J.McCarthy研制的表处理语言LISP，奠定了ES的开发工具的基础，它除了具有数据处理的功能外，还能方便地进行符号处理。

专家系统产生和发展

➤ 第二阶段：产生期（20世纪60年代中期— 20世纪70年代初）

- **DENDRAL系统（1968年，斯坦福大学费根鲍姆等人）——推断化学分子结构的专家系统**
- **MYCSYMA系统（1971年，麻省理工学院）——用于数学运算的数学专家系统**
- **特点：**
 - ✓ 高度的专业化。
 - ✓ 专门问题求解能力强。
 - ✓ 结构、功能不完整。
 - ✓ 移植性差。
 - ✓ 缺乏解释功能。

专家系统产生和发展

➤ 第三阶段: 成熟期（20世纪70年代中期— 20世纪80年代初）

- **MYCIN系统**（斯坦福大学）——血液感染病诊断专家系统
- **PROSPECTOR系统**（斯坦福研究所）——探矿专家系统
- **CASNET系统**（拉特格尔大学）：用于青光眼诊断与治疗。
- **AM系统**（ 1981年，斯坦福大学）：模拟人类进行概括、抽象和归纳推理，发现某些数论的概念和定理。
- **HEARSAY系统**（卡内基—梅隆大学）——语音识别专家系统

专家系统产生和发展

➤ 第四阶段：发展期（20世纪80年代至今）

- 专家配置系统**XCON**（DEC公司、卡内基—梅隆大学）：DEC公司为VAX计算机系统制订硬件配置方案的专家系统。
- 专家系统开发工具：CLIPS、JESS、OKPS等。
- 骨架系统：EMYCIN、KAS、EXPERT等。
- 通用型知识表达语言：OPS5等。
- 专家系统开发环境：AGE等。

专家系统产生和发展——中国

- 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”。
- 中科院数学研究所陆汝钐院士研制了专家系统“天鹅”。
- 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。
- 吉林大学“勘探专家系统”及“油气资源评价专家系统”。
- 浙江大学开发服装剪裁专家系统及花布图案设计专家系统
- 中科院数学研究所研制了专家系统开发环境“天马”，应用于国防和经济领域
- 中科院计算所研制了面向对象专家系统开发工具“OKPS”。

内容

Contents

1. 专家系统概述
- 2. 专家系统基本原理**
3. 专家系统分类
4. 专家系统设计——知识表示
5. 专家系统设计——推理机制
6. 专家系统开发工具
7. 专家系统应用实例



专家系统基本原理

ES的信息处理

现实中许多问题不能用算法表示，其求解是经验性的，ES对信息处理以“知识”为中心，包括：

- 知识的表示，用特定的数据结构来表示知识，而不仅是用算法表示；
- 知识的处理，以符号处理为特征；
- 知识的启发式应用，以问题求解的特定策略为基础，指导问题求解策略。

ES信息处理的主要特点是“知识”与对知识的应用相分离。

专家系统基本原理

ES的基本思想与特征

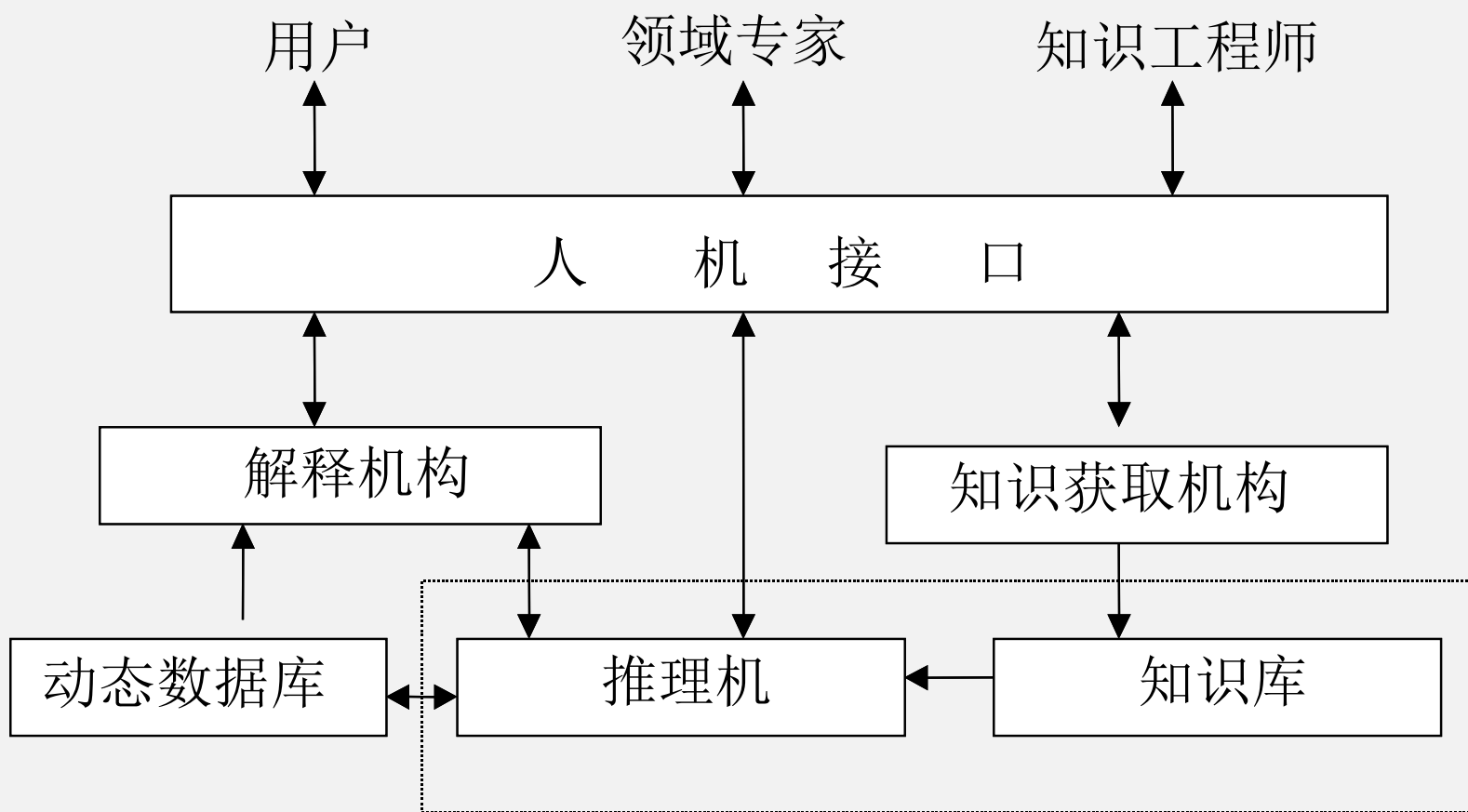
(1) 专家系统的基本思想：利用计算机存储关于某一领域的大量专门知识；有效地利用这些知识去解决问题。

(2) ES的特征：ES以知识为中心，并具有三大特征：

- 启发性（Heuristic），不仅使用逻辑知识，也使用启发性知识；
- 透明性（Transparency），能向用户解释推理；
- 灵活性（Flexibility），系统的知识便于修改和补充。

专家系统基本原理

ES的基本结构



专家系统基本原理

知识库：存放领域专家提供的求解问题的专门知识，其质量直接影响ES的性能。

如：知识库—“规则”+“事实”组成

动态数据库：反映具体问题在当前求解状态下的符号或事实的集合，包括 (或上下文) 问题的有关初始数据和系统求解期间所产生的所有问题。

如：上下文——动态事实集

推理机：在一定的控制策略下，针对上下文中的当前问题信息，识别和选取知识库中对当前问题的可用知识进行推理，以修改上下文，直到最终得到问题的解。

如：规则推理机（正向、反向、双向）

人机接口：实现用户输入和ES内部表达方式的转换。

知识获取机构：负责建立、修改和扩充知识库，是把问题求解的各种专门知识从人类专家的头脑中或其他知识源那里转换到知识库中的一个机构。

解释机构：用于对求解过程做出说明，并回答用户的提问。

内容

Contents

1. 专家系统概述
2. 专家系统基本原理
- 3. 专家系统分类**
4. 专家系统设计——知识表示
5. 专家系统设计——推理机制
6. 专家系统开发工具
7. 专家系统应用实例



专家系统分类

1. 按知识表示技术分为

- 基于规则的ES， 基于逻辑的ES;， 基于框架的ES， 基于模型的ES等

2. 按任务类型分为

- 解释型ES——用于分析符号等数据，进而阐明这些数据的实际意义；
- 预测型ES——根据处理对象的过去与现在情况推测未来的演变结果；
- 诊断型ES——根据输入信息找出处理对象中存在的故障、缺陷；
- 调试型ES——给出已确认故障的排除方案；
- 维修型ES——制定并实施纠正某类故障的规划；
- 规划型ES——根据给定目标拟定行动计划；
- 设计型ES——根据给定要求形成所需方案和图样；
- 监护型ES——完成实时监测任务；
- 控制型ES——完成实时控制；
- 教育型ES——诊断型和调试型的组合，用于教学和培训。

3. 新型专家系统

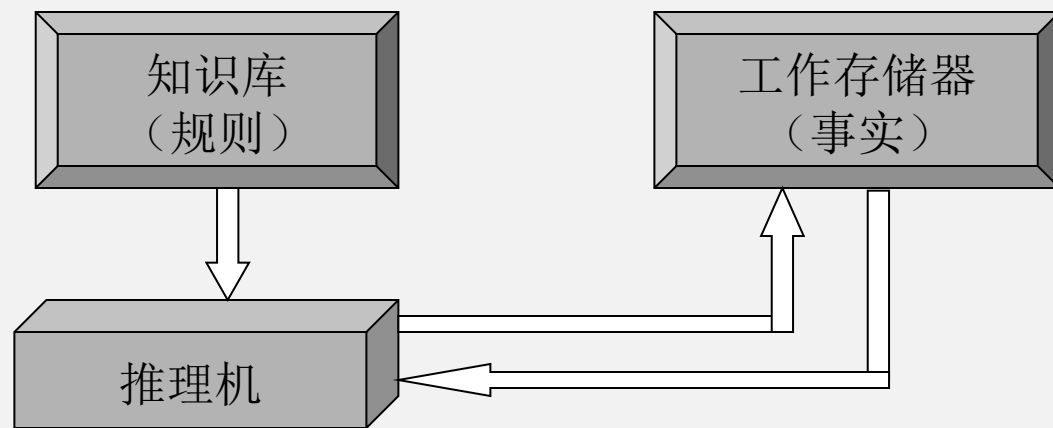
- 分布式专家系统、协同专家系统、互联网专家系统等

专家系统分类——按知识表示技术分类

1. 基于规则的专家系统

- **知识库** 以一套规则建立的长期存储器模型
- **工作存储器** 建立的短期存储器模型，存放问题事实和由规则激发而推断出的新事实
- **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，建立的推理模型，以推断出新的信息

基于规则的专家系统的工作模型



专家系统分类——按知识表示技术分类

2. 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的子句构成，这些句子合起来组成Turbo Prolog的数据库。
- 和基于规则的系统一样，基于逻辑的系统也有一个可以由输入数据流激活的规则集。该系统有一个解释程序，它可以选择和激活在该系统操作中用到的模块。解释程序在系统中的作用是：
 - (1) 系统有启发搜索和模式匹配的知识库子句。解释程序将这些子句与数据库中的数据元素匹配。
 - (2) 如果同时激活的规则多于1个，则系统使用Turbo Prolog的冲突消解功能，因此，用户/程序员不必担心这些可能产生的冲突。
 - (3) 系统可以自动得到一个处理结果，结果可以直接送到输出设备。

专家系统分类——按知识表示技术分类

3. 基于框架的专家系统

- 基于框架的专家系统建立在框架的基础之上
- 基于框架的专家系统采用面向目标编程技术
- 基于框架的设计和面向目标的编程共享许多特征
- 在设计基于框架系统时，专家系统的设计者们把目标叫做框架

专家系统分类——按知识表示技术分类

4 基于模型的专家系统

- 人工智能的一个核心
- 综合各种模型的专家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

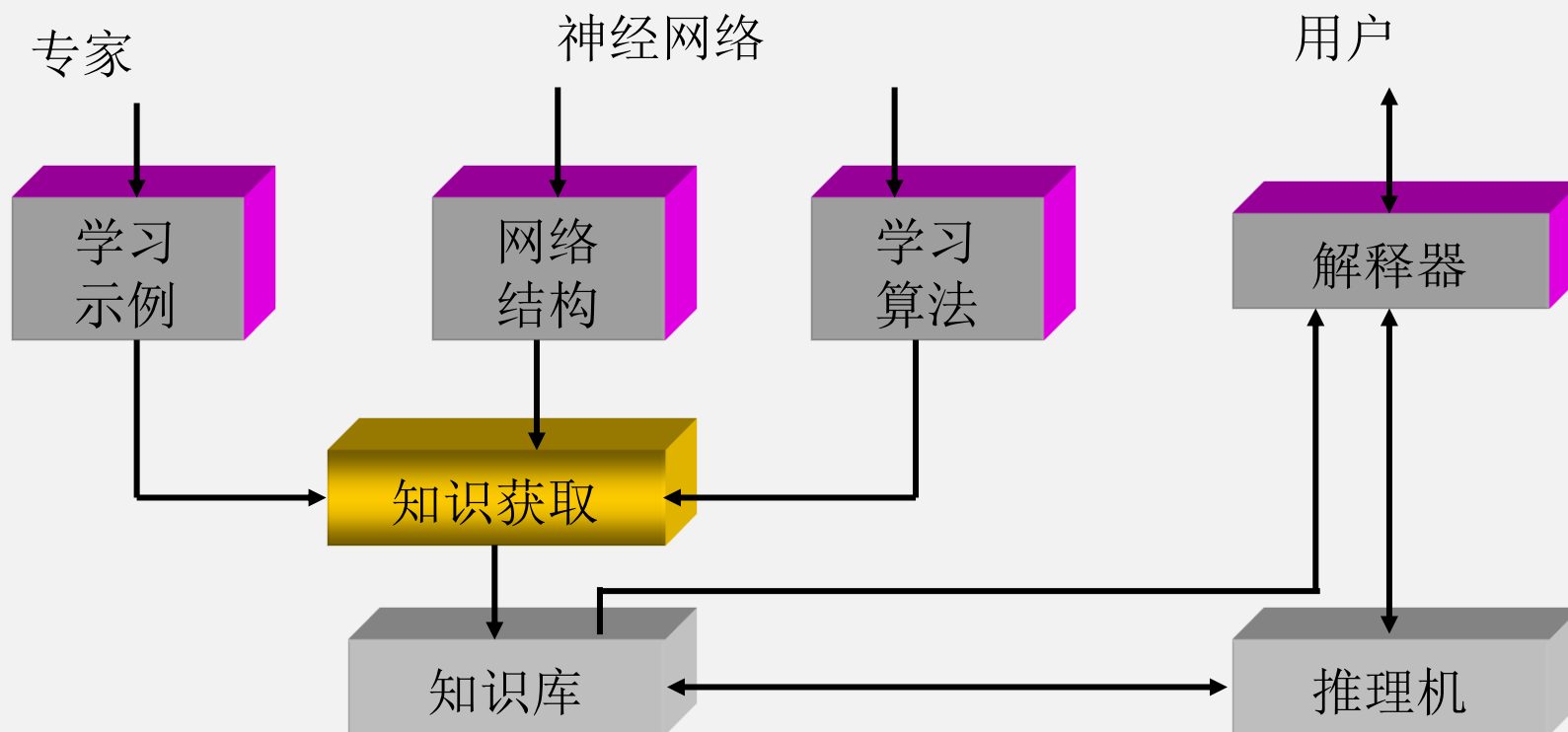
专家系统分类——按知识表示技术分类

基于神经网络的专家系统

- 神经网络模型从知识表示、推理机制到控制方式，与目前专家系统有本质的区别
- 三种神经网络模型与专家系统集成模式
 - 神经网络支持专家系统
 - 专家系统支持神经网络
 - 协同式的神经网络专家系统

专家系统分类——按知识表示技术分类

神经网络专家系统的基本结构



神经网络的知识表示是一种**隐式**表示。神经网络通过**实例学习**实现知识自动获取。神经网络的推理是个正向非线性数值计算过程，同时也是一种**并行**推理机制。

专家系统分类——按任务类型分类

- 解释专家系统
- 预测专家系统
- 诊断专家系统
- 设计专家系统
- 规划专家系统
- 监视专家系统
- 控制专家系统
- 调试专家系统
- 教学专家系统
- 维修专家系统

内容

Contents

1. 专家系统概述
2. 专家系统基本原理
3. 专家系统分类
- 4. 专家系统设计——知识表示**
5. 专家系统设计——推理机制
6. 专家系统开发工具
7. 专家系统应用实例



专家系统设计——知识表示

➤ 知识的定义

知识是人类在实践中所积累的认识和经验的总和，其中ES的知识指与领域问题求解相关的知识即“数据”和“符号”。

➤ 知识的分类

(1) 按知识的作用，将计算机处理的知识分为三类

- 描述性知识：表示对象及概念的特征及其相互关系的知识，问题求解状况的知识；
- 判断性知识：表示与领域有关的问题求解知识，如推理规则等；
- 过程性知识：表示问题的求解策略，即如何应用判断性知识等进行推理的知识；

专家系统设计——知识表示

➤ 知识的分类

(2) 按知识的作用层次分

- 对象级知识：直接描述有关领域对象的知识。
- 元级知识：描述对象级知识的知识，如关于领域知识的内容、特征、应用范围的知识，如何运用这些知识的知识。

(3) 其它分类形式

- 共性知识与个性知识；
- 静态知识与动态知识。

专家系统分类——知识表示

(3) 知识表示方法的分类（按其表示的特征分）

- **叙述性表示**：将知识与控制分开，把知识的使用方法，即控制部分留给计算机程序。
 - 它的特点是：严密性强、易于模块化、具有推理的完备性，但推理效率较低。
- **过程性表示**：将知识与控制结合起来（利弊与上述方法相反）。

两种表示方法各有利弊，对不同性质的问题应采用不同形式的表达方法。

知识的表示和推理在大部分情况下位于独立的两个模块，但推理机制强烈关联于相应的知识表示方式。

专家系统设计——知识表示

(4) 常用的知识表示方法简介

① 产生式规则表示法

- 将知识表示成“ifthen.....”的形式；
- 表示方法自然、简洁；

② 框架表示法

- 将知识表示为层状结构，一个对象或概念的所有信息均属于该层次的结构中；
- 该层次结构还可以表示对象间的关系；
- 该层次结构由一系列的“槽”和相关于“槽”的一系列“侧面”组成；

③ 语义网络表示法

采用结点和结点间的弧表示对象、概念及其相互关系。

专家系统设计——知识表示

④ 逻辑表示法

- 采用一阶谓词逻辑表示知识
- 属叙述性知识表示
- 有严格的数学基础

⑤ 其它表示法

- 状态空间法
- 与或图
- PETRI网
- 概念图

专家系统设计——知识表示——产生式规则表示法

➤ 概述

产生式最早由P.Post于1943年提出，用于构造Post机计算模型；1972年 A.Newell和H.A.Simon在研究人类的认知模型中提出了 Rule-Based 产生式方法及规则表示模式；Rule-Based 的表示法是目前应用最为普遍的一种。

产生式通常用于具有因果关系的知识。

➤ 产生式规则的逻辑结构

格式

```
if (前提1) & (前提2) & .....  
then (结论1) & (结论2) & .....
```

正规化格式

```
if (前提1) & (前提2) & .....  
then (结论1)
```

专家系统设计——知识表示——产生式规则表示法

➤ 产生式规则的基本单元

前提：符号化的事实型知识，不同前提形式构成不同产生式规则类型。

常量型前提（Tiger has Leg）；

变量型前提（Lieats \$VAR）；

约束型前提（\$VAR>=28）；

表达式计算型（\$VAR=2x+5y-16）；

结论：符号化事实型知识；

例如 规则1: if 该动物有羽毛 then 该动物是鸟

规则2: if 该动物是鸟 and 有长脖子

and 有长腿

and 不会飞

then 该动物是鸵鸟

专家系统设计——知识表示——产生式规则表示法

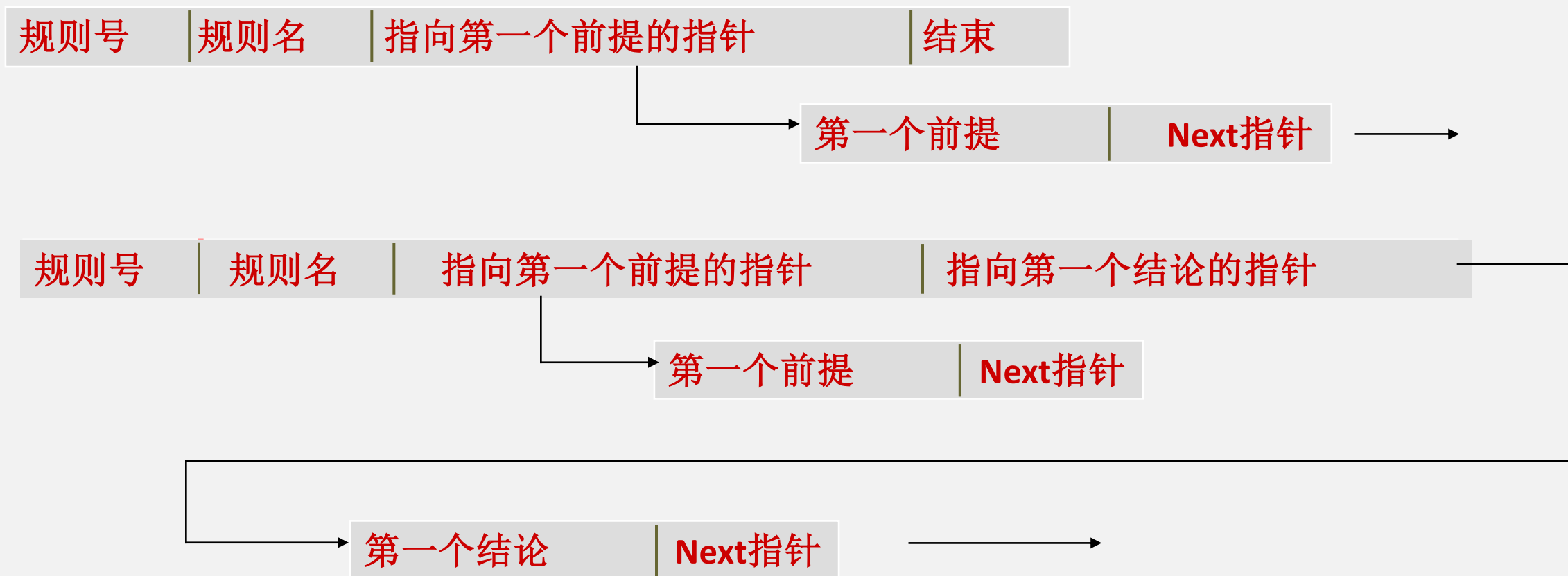
➤产生式规则的物理结构

产生式规则为变化数据结构，采用“链接表”实现。

Prolog语言的实现

QULE(integer,string,list,real)

FACT(integer,string,list,real)



专家系统设计——知识表示——产生式规则表示法

➤ 产生式规则的组织

规则的组织是构造规则库的核心！

(1) 规则库的实质

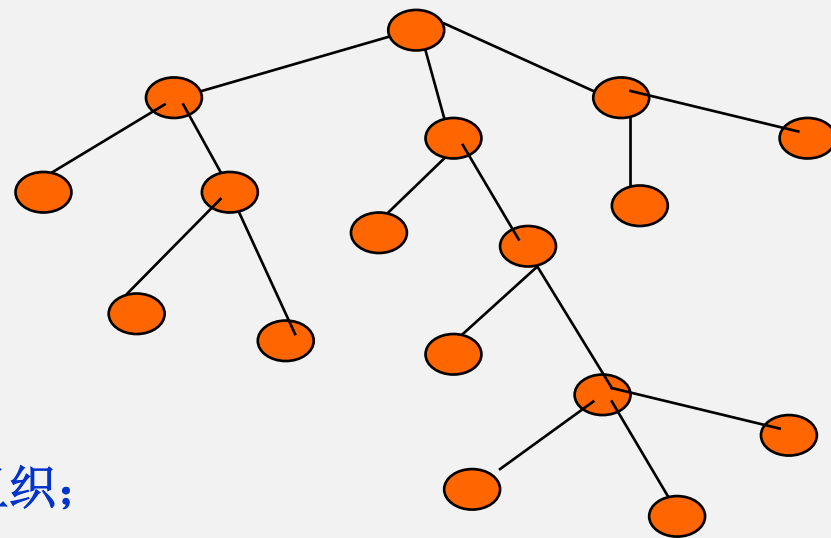
一种隐含的“与或图”，该“与或图”以“事实性知识”为节点，节点间的连接关系由规则确定，为“与”关系，规则间的关系为“或”关系。

(2) 原则上规则库的规则可以顺序存放，但存在下列矛盾；

- 规则的一致性问题应解决；
- 如何控制冲突；
- 提高规则的匹配效率；

(3) 推荐的组织形式

- 以规则树的形式组织，最高点为推理目标规则，依次按层次组织；
- 以类层次形式的“与或图”组织规则；



专家系统设计——知识表示——框架表示法

➤ 概念

- 知识的框架表示法1975年由M.Minsky提出，最早用作视觉感知、自然语言对话等问题的知识表示，目前已作为一种通用数据结构来表示知识对象(实体)。
- 框架理论认为，人们对现实世界中各种事物的认识都是以一种类似于框架的结构存储在记忆中的，当面临一种新事物时，就从记忆中找出一个合适的框架并根据实际情况对其细节加以修改、补充，从而形成对当前事物的认识。

专家系统设计——知识表示——框架表示法

➤ 框架的定义

- (1) 框架是用于描述具有固定的静态对象的通用数据结构，该对象用“对象...属性...属性值”表示。
- (2) 框架由若干个槽（Slot）组成，槽用于描述属性。
- (3) 槽又可由若干个侧面组成。侧面用于描述相应属性的一个方面。
- (4) 侧面又可由一个或多个侧面值组成。
- (5) 框架实质上是一个层次的嵌套链接表！

➤ 框架实例1

框架名： < 假冒伪劣商品 >

商品名称：

生产厂家：

出售商店：

处 罚： 处理方式：

处罚依据：

处罚时间： 单位（年、月、日）

经办部门：

说明：在这个框架中，有4个槽，其中，“处罚”槽有4个侧面，侧面“处罚时间”用“单位”指出了填值时的标准限制。

专家系统设计——知识表示——框架表示法

➤ 框架实例2

框架名： < 教师 >

姓名： 单位（姓、名）

年龄： 单位（岁）

性别： 范围（男、女）

缺省（男）

职称： 范围（教授、副教授、讲师、助教）

缺省（讲师）

部门： 单位（系、教研室）

住址： <住址框架>

工资： <工资框架>

开始工作时间： 单位（年、月）

专家系统设计——知识表示——框架表示法

➤ 框架表示法的逻辑结构

(Frame名

(Slot名

{{(Value侧面 Value)}} .and.

{{(Default侧面 Value)} .or.

(Range侧面 Value) .or.

(Type侧面 Value) .or.

(if_needed侧面 过程) .or.

(if_added 侧面 过程) .or.

(if_removed侧面 过程) }} .or.

(Slot名

{{(Value Value)}} .and.

{{(Default Value)} .or.

(Range Value) .or.

(Type Value) .or.

(if_needed侧面 过程) .or.

(if_added 侧面 过程) .or.

(if_removed侧面 过程) }} .or.

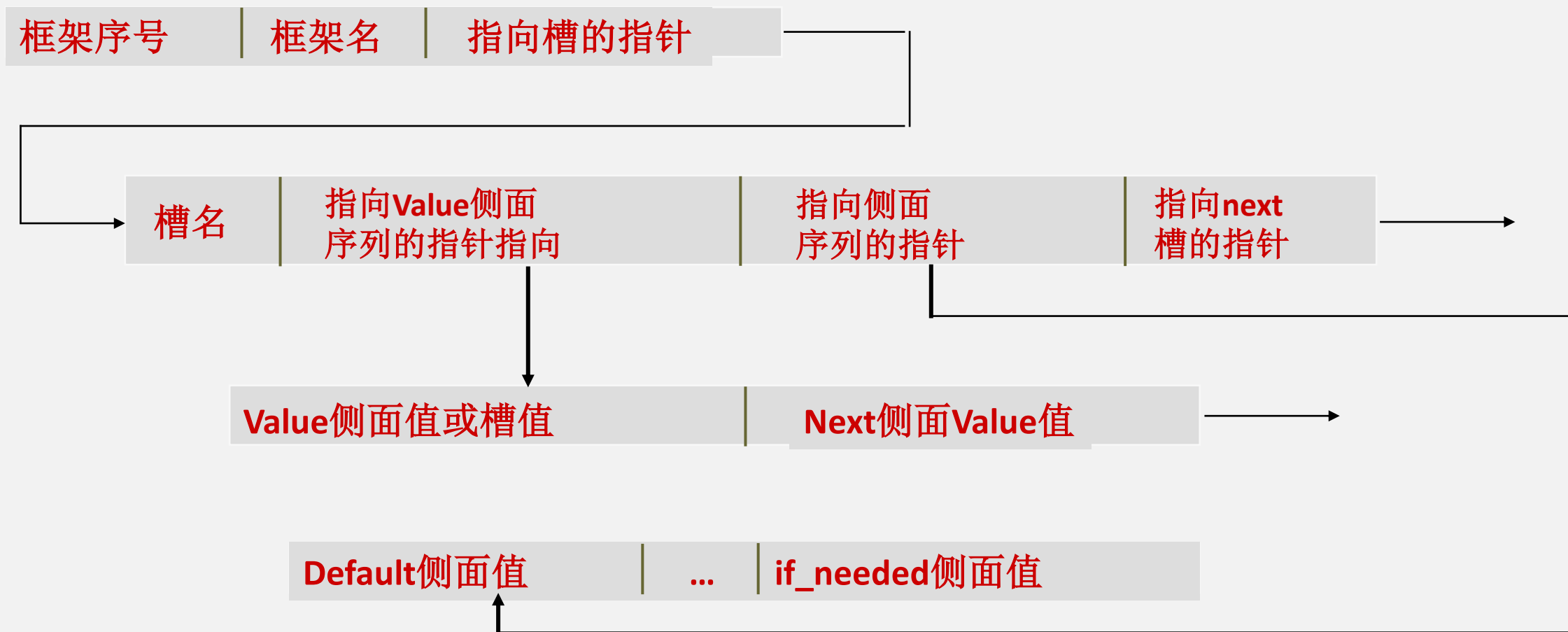
附加过程一般有三种类型：

- 当需要槽值，但初始值不存在且缺省值也未设定时，执行 **if_needed** 过程。
- 当将槽值加入槽中时，将运行 **if_added** 过程。
- 当要从槽中删除槽值时，运行 **if_removed** 过程。

专家系统设计——知识表示——框架表示法

➤ 框架表示法的物理结构

框架也采用链表表示。



专家系统设计——知识表示——框架表示法

➤C语言实现

```
struct values{ char valu[80];  
               struct valus *next_value;}  
  
struct other_facets{ char default[80];  
                   char range[80];  
                   char types[80];  
                   char if_needed[80]; }  
  
struct solts{ char slotname[80];  
             struct values *first_value;  
             struct otherfacets *first;  
             struct slots *next_slot; }  
  
struct frames{ int frame_no;  
              char frame_name[80];  
              struct slots *first_slot; }
```


专家系统设计——知识表示——框架表示法

➤ 知识的框架表示法的特点

(1) 框架可以进行结构化深层知识表示:

- 框架可为实体、属性关系和默认值等提供显示表示；其中提供默认值相当于经验预测。
- 适合表示常识性知识；
- 表示实体固有的因果模型，便于知识的解释；

(2) 容易附加过程信息

框架主要描述静态知识，它的 `if_needed`，`if_added`，`if_removed` 侧面可进行附加。

(3) 框架之间的层次结构提供了继承特性

一个框架的属性及附加过程可从高层次的框架继承下来。

(4) 框架间的组织结构化

框架可组织成层状，每个框架形成了一个独立的知识单元，可扩展、模块化。

专家系统设计——知识表示——逻辑表示法

➤ 概述

- 命题逻辑和谓词逻辑是最先应用于人工智能的两种逻辑，谓词逻辑是在命题逻辑基础上发展起来的。

(1) 命题：命题是具有真假意义的语句。

- 命题代表人们进行思维时的一种判断。若命题的意义为真，则称它的真值为真。记作T；若命题的意义为假，则称它的真值为假，记作F。一个命题不能同时既为真又为假，但可以在一定条件下为真，在另一条件下为假。

例如：“北京是中华人民共和国的首都”，“ $3 < 5$ ” 都是真值为T的命题。

“太阳从西边升起”，“煤球是白的”都是真值为F的命题。

- 在命题逻辑中，命题通常用大写的英文字母表示：

例如，可以用P表示“西安是个古老的城市”。

- 命题这种表示法有较大的局限性，它无法描述的客观事物的结构及逻辑特征，也不能把不同事物的共同特征描述出来。

专家系统设计——知识表示——逻辑表示法

(2) 谓词

I. 在谓词逻辑中，命题用谓词表示，一个谓词可分为谓词名和个体这两个部分。

个体：表示某个独立存在的事物或者某个抽象的概念；

谓词名：用于刻画个体的性质、状态或个体间的关系。

例如：对于“老张是教师”这个命题，用谓词可表示为 **Teacher(zhang)**。

其中，**Teacher**是谓词名，**zhang**是个体，“**Teacher**”刻画了“**zhang**”的职业是教师这一特征。

“**5>3**”可用谓词表示为：**Greater(5,3)**；这里，**Greater**刻画了5与3之间的“大于”关系。

II. 谓词的一般形式是

$P(x_1, x_2, \dots, x_n)$

其中，**P**是谓词名， **x_1, x_2, \dots, x_n** 是个体。在谓词中，个体可以是常量、变元、也可以是函数。

III. 一阶谓词

在谓词 **$P(x_1, x_2, \dots, x_n)$** 中，若 **x_i** 都是个体常量、变元或函数，则称它为一阶谓词。

专家系统设计——知识表示——逻辑表示法

➤ 一阶谓词逻辑的几个概念

无论是命题逻辑还是谓词逻辑，均可用连接词把一些简单命题连接起来，构成一个复合命题，以表示一个比较复杂的含义。

谓词逻辑的合式公式wff (wall_formed_formula)

合式公式是由 $\left\{ \begin{array}{l} \text{原子公式} \\ \text{连接词} \\ \text{量词} \end{array} \right.$ 组成

I. 原子公式：是最基本的合式公式，它由谓词、括号和扩号中的项组成，其中项可以是常量、变量和函数。一个事实可用不同形式的原子公式表示。

专家系统设计——知识表示——逻辑表示法

II. 连接词:

\wedge 表示合取: 表示被它连接的两个命题具有“与”的关系。

设 P, Q 为合式公式, $P \wedge Q$ 表示 P 与 Q 的合取, P, Q 称合取项, $P \wedge Q$ 合取的真值表:

当 P, Q 都为真时, $P \wedge Q$ 为“真True”,

否则为“假False”。

\vee 表示析取: 表示被它连接的两个命题具有“或”的关系。

设 P, Q 为合式公式, $P \vee Q$ 表示 P 与 Q 的析取, P, Q 称析取项, $P \vee Q$ 析取的真值表:

当 P, Q 至少有一个为真时, $P \vee Q$ 为“真True”,

否则为“假False”。

专家系统设计——知识表示——逻辑表示法

→ 表示蕴涵

设 P , Q 为合式公式, $P \rightarrow Q$ 称为蕴涵, P 为前项, Q 为后项

$P \rightarrow Q$ 常用于表示 IF P THEN Q

真值表:

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

专家系统设计——知识表示——逻辑表示法

¬ 表示否定

¬P 真值表: P为真时, ¬P 为假;
P为假时, ¬P 为真;

III. 量词:

指在合式公式中出现“量化”时, 在其前面可加“量词”说明的范围, 这种说明称为量化。

- 全称量词 \forall , $\forall x$ 表示对所有的 x ;

例: “所有的大象都是灰色的” 可以表示为:

$(\forall x)[\text{Elephant}(x) \rightarrow \text{Color}(x, \text{Gray})]$

- 存在量词 \exists , $\exists x$ 表示至少存在一个 x ;

专家系统设计——知识表示——逻辑表示法

➤知识的一阶谓词表示法

- ✓ 谓词逻辑是一种形式语言，也是到目前为止能够表达人类思维活动规律的一种最精确的语言，它与人们的自然语言比较接近，又可方便地存储到计算机中去，并被精确地处理。
- ✓ 谓词逻辑适合于表示事物的状态、属性、概念等事实性的知识，也可以用来表示事物间确定的因果关系，即规则。
- ✓ 事实通常用合式公式的“与/或”形表示（用合取符号 \wedge 及析取符号 \vee 连接起来的公式）。
- ✓ 规则通常用蕴涵式 \rightarrow 表示。
- ✓ 用谓词公式（合式公式）表示知识时，需要首先定义谓词，指出每个谓词的确切含义，然后再用连接词把有关的谓词连接起来，形成一个谓词公式表达一个完整的含义。

专家系统设计——知识表示——逻辑表示法

例1 有下列知识：

刘欢比他父亲出名。

高扬是计算机系的一名学生，但他不喜欢编程序。

人人爱劳动。

为了用谓词公式表示上述知识，首先需要定义谓词：

Bigger(x,y): x 比 y 出名。

Computer(x): x 是计算机系的学生。

Like(x,y): x 喜欢 y 。

Love(x,y): x 热爱 y。

Man(x): x 是人。

然后用谓词公式把上述知识表示为：

Bigger (Liuhong , father(Liuhong))

Computer(Gaoyang) \wedge \neg Like(Gaoyang , programing)

$(\forall x) (Man(x) \rightarrow Love(x, labour))$

专家系统设计——知识表示——逻辑表示法

➤逻辑表示法的特点

优点：

自然性：符合人类对问题的直觉理解；

描述性：表示与知识分离；

精确性：只有“真与假”的值；

严密性：谓词逻辑具有严格的形式定义以及推理规则；

易实现：用谓词逻辑表示的知识可以比较容易地转换为计算机内部形式，易于模块化，便于对知识的增加、修改、删除；

缺点：

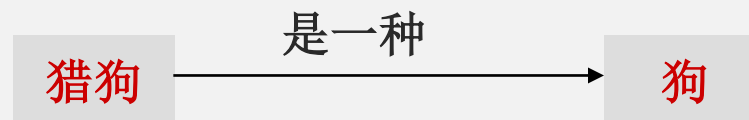
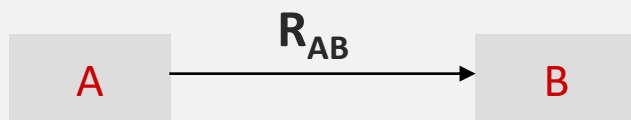
不能表示不确定的知识，组合爆炸，效率低；

专家系统设计——知识表示——语义网络表示法

➤概述

语义网络1968年由J.R.Quillian提出，开始是作为人类联想记忆的一个显式心理学模型提出，随后在AI中用于自然语言理解，表示命题信息（具有逻辑真的事实），在ES中，语义网络首先由PROSPECTOR实现，用于语义网络的定义。

- 语义网络用于描述物体概念与状态及其间的关系；
- 语义网络由节点和节点间的弧组成；
 - ✓ 节点：表示物理实体、概念或状态；
 - ✓ 弧：表示它们间的相互关系。
- 语义网络在数学上是一个有向图。
 - ✓ 一个最简单的语义网络是如下一个三元组：（节点1，弧，节点2）
 - ✓ 它可用图表示，称为一个基本网元。



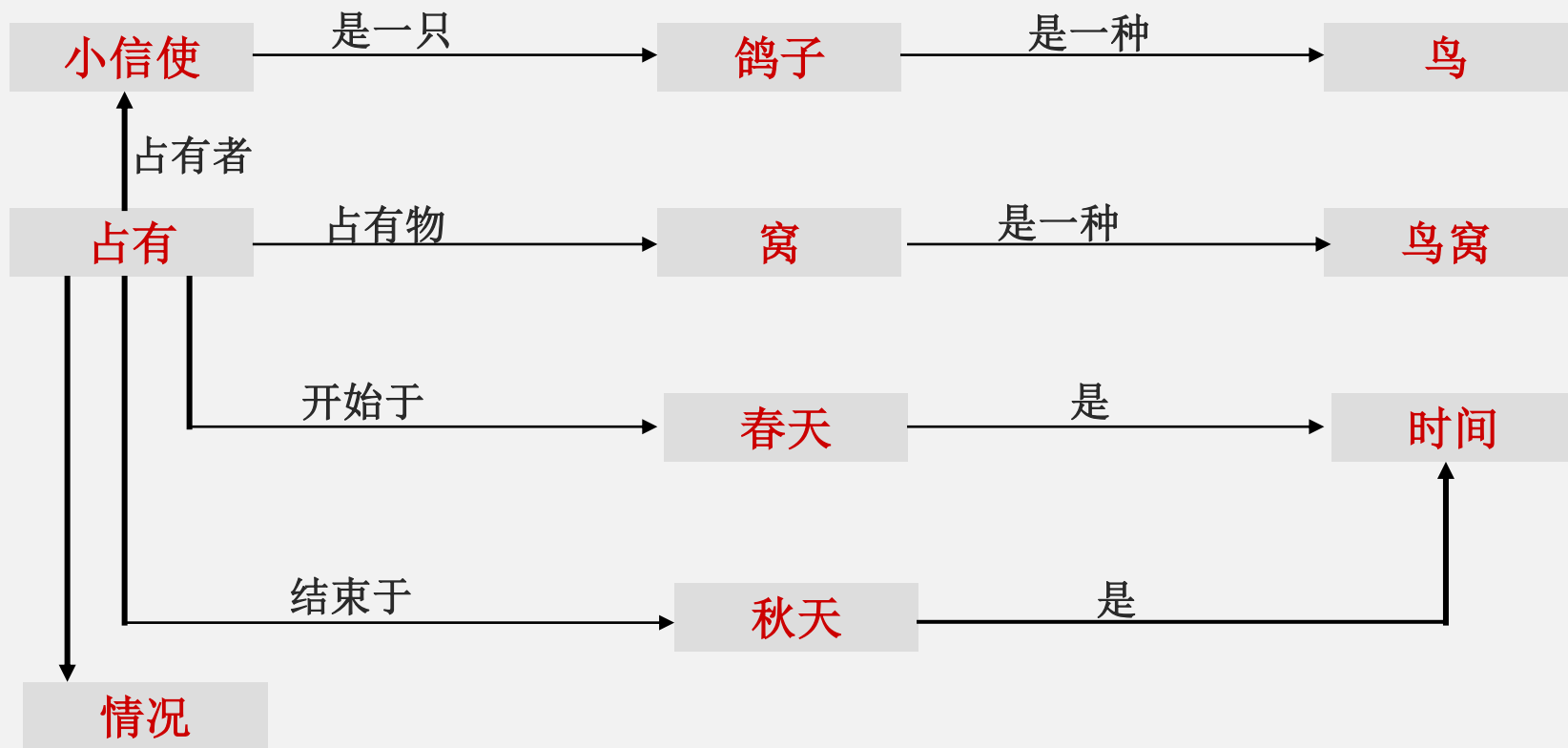
专家系统设计——知识表示——语义网络表示法

➤ 语义网络表示

语义网络可以表示事实性的知识，也可以表示有关事实性知识之间的复杂联系。

(1) 用语义网络表示事实

例如有下述事实：“小信使”这只鸽子从春天到秋天占有一个窝。



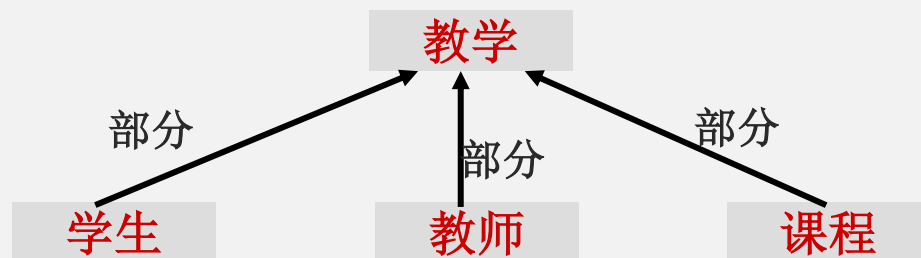
- ✓ 节点可以表示一个事物者一个具体概念，也可以表示某一情况、某一事件或者某个动作。
- ✓ 在一些稍复杂的事实性知识中，语义网络中可通过增设合取节点及析取节点来表示。

专家系统设计——知识表示——语义网络表示法

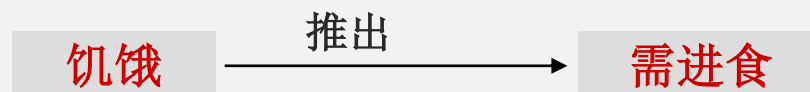
(2) 用语义网络表示有关事实间的关系

语义网络可以描述事物间多种复杂的语义关系，下面是常用的几种：

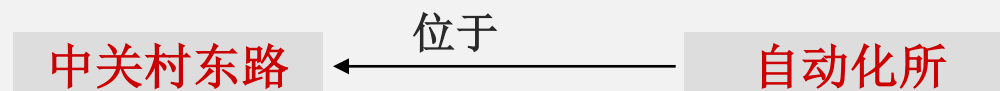
- 分类关系：指事物间的类属关系。如“是一种”等。
- 聚集关系：如果下层概念是其上层概念的一方面或者一个部分，则称他们是聚集关系。



- 推论关系：如果一个概念可由另一个概念推出，则称它们之间推论关系。



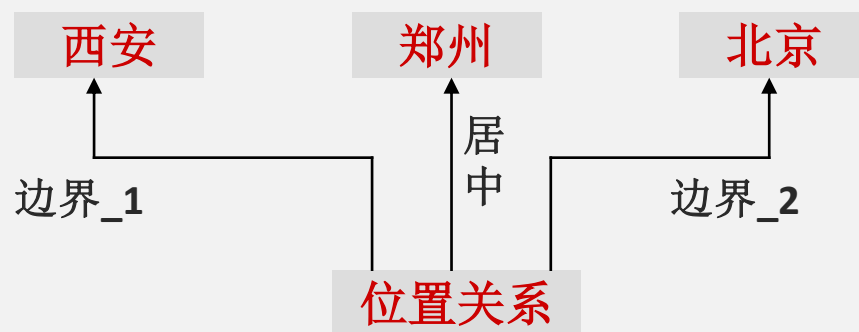
- 时间、位置关系：



专家系统设计——知识表示——语义网络表示法

- 多元关系：在语义网络中，一条弧只能从一个节点指向另一个节点，适合于表示一个二元关系。但在许多情况下需要用一种关系把几个事物联系起来。例如对于如下事实：

郑州位于西安和北京之间。



为了在语义网络中描述多元关系，可以用节点来表示关系。

专家系统设计——知识表示——过程表示法

➤概述

过程性表示方法着重于对知识的利用，它把与问题有关的知识以及如何运用这些知识求解问题的控制策略都表述为一个或多个求解问题的过程，每一个过程一段程序，用于完成对一个具体事件或情况的处理。，这类知识的表示方法就是过程性表示法，或过程表示法。

➤知识表示方法

例如，有如下知识：

如果 x 与 y 是兄弟，且 x 是 z 的父亲，
则 y 是 z 的叔父。

若用产生式规则（说明性知识表示法）表示：

```
if brother(  $x$ ,  $y$  ) and father( $x$ ,  $z$ )  
then uncle( $y$ ,  $z$ )
```

专家系统设计——知识表示——过程表示法

用过程性知识（过程表示法有多种表示形式，在此选用过程规则表示）：

```
BR(Uncle ? y ? z)
  GOAL( brother ? x y )
  GOAL( father x z )
  INSERT(Uncle y z)
  RETURN
```

其中，**BR** 是后向推理的标志；**GOAL** 表示求解子目标，即进行过程调用；**INSTER** 表示对数据库实施插入操作；**RETURN** 表示该过程规则结束。

上述过程规则的含义：

按后向推理方式进行推理，为了求解（Uncle ? y ? z），首先应通过过程调用求解（Brother ? x y），得到 x 的值，然后将得到的 x 值传递给（Father x z），并求解它，如果这些都成功，就将（Uncle y z）插入到数据库中，并将控制权返回给调用者。

专家系统设计——知识表示——过程表示法

一般来说，一个过程规则包括激发条件、演绎操作、状态转换及返回 四个部分：

(1) 激发条件：

由两部分组成，即推理方向和调用模式。推理方向指出其推理是前向推理（FR）还是后向推理（BR）。

(2) 演绎操作：

演绎操作由一系列的子目标构成，当上面的激发条件被满足时，将执行这里列出的演绎操作，

如 GOAL(brother ? x y) ， GOAL(father x z)。

(3) 状态转换：

状态转换操作用于对数据库进行增、删、改，分别用 INSTER、DELETE、MODIFY 语句实现。

(4) 返回：

过程规则的最后一个语句是 RETURN ，用于将控制权返回到调用该过程规则的上级过程规则那里去。

激发条件

推理方向
调用模式

演绎操作

子目标序列

状态转换

数据库刷新

返回

RETURN

过程规则的结构

内容

Contents

1. 专家系统概述
2. 专家系统基本原理
3. 专家系统分类
4. 专家系统设计——知识表示
- 5. 专家系统设计——推理机制**
6. 专家系统开发工具
7. 专家系统应用实例



专家系统设计——知识推理

➤ 概述

什么是推理

推理——从已知事实出发，通过运用已掌握的知识，找出其中蕴含的事实，或归结出新的事实，这一过程称为推理。

推理机——在专家系统中，推理是由程序实现的，称为推理机。

专家系统设计——知识推理——分类

分类方法	类 型
推理方式不同	演绎推理、归纳推理、默认推理
知识的精确性	确定性推理、不确定性推理
推理过程	单调推理、非单调推理
运用知识	启发式推理、非启发式推理
知识获取	统计推理、直觉推理

专家系统设计——知识推理——分类

演绎推理

- 演绎推理——从全称判断推导出特称判断或单称判断的过程，即由一般性知识推出适合于某一具体情况的结论。这是一种从一般到个别的推理。

- ✓ 举例

- 1) 足球运动员的身体都是强壮的；
- 2) 高波是一名足球运动员；
- 3) 所以，高波的身体是强壮的。

- 三段论式演绎推理

- ✓ 演绎推理有多种形式，经常用的是三段论式，它包括：

- 1) 大前提，这是已知的一般性知识或假设；
- 2) 小前提，这是关于所研究的具体情况或个别事实的判断；
- 3) 结论，这是由大前提推出的适合于小前提所示情况的新判断。

专家系统设计——知识推理——分类

归纳推理

- 归纳推理——归纳推理是从足够多的事例中归结出一般性结论的推理过程，是一种从个别到一般的推理。归纳推理又分为完全归纳和不完全归纳两种。

默认推理

- 默认推理——又称缺省推理，它是在知识不完全的情况下假设某些条件已经具备所进行的推理。在默认推理过程中，如果到某一时刻发现原先所作的默认不正确，则就要撤销所作默认，以及由此默认推出的所有结论，重新按新情况进行推理。

专家系统设计——知识推理——分类

➤ 确定性推理

指推理时所用的知识都是精确的，推出的结论也是确定的，其真值或为“真”，或为“假”，没有第三种情况出现。

下面将要讨论的经典逻辑推理就属于这一类。

➤ 不确定性推理

指推理时所用的知识不都是精确的，推出的结论也不完全是肯定的，其真值位于“真”和“假”之间，命题的外延模糊不清。

专家系统设计——知识推理——分类

➤ 单调推理

指在推理过程中随着推理的向前推进及新知识的加入，推出的结论越来越接近最终目标，在推理过程中不会出现反复的情况，既不会由于新知识的加入而否定了前面推出的结论，从而使推理又退回到前面的一步。

➤ 非单调推理

指在推理过程中由于新知识的加入，不仅没有加强已推出的结论，反而要否定它，使得推理退回到前面的某一步，重新开始。

专家系统设计——知识推理——分类

启发式推理、非启发式推理

启发式推理——推理中运用与问题有关的启发性知识，即解决问题的策略、技巧、窍门，对解的特性及规律的估计等实践经验和知识，以加快推理过程、提高搜索效率，这种推理称为启发式推理。

统计推理、直觉推理

统计推理——根据对某事物的数据统计进行的推理。

直觉推理——又称常识性推理，是根据常识进行的推理。

专家系统设计——知识推理——控制策略

推理的控制策略

- 推理过程是一个思维过程，即求解问题的过程，求解问题的质量和效率不仅依赖于所采用的求解方法，而且还依赖于求解问题的策略，即推理的控制策略。推理的控制策略主要包括推理方向、模式匹配、搜索策略、冲突消解策略等。
- 按照推理的方向，推理可分为：
 - ✓ 正向推理
 - ✓ 反向推理
 - ✓ 混合推理
 - ✓ 双向推理

专家系统设计——知识推理——控制策略——推理方向

正向推理

正向推理是以已知事实作为出发点的一种推理，又称数据驱动推理、前向链推理及前件推理等。

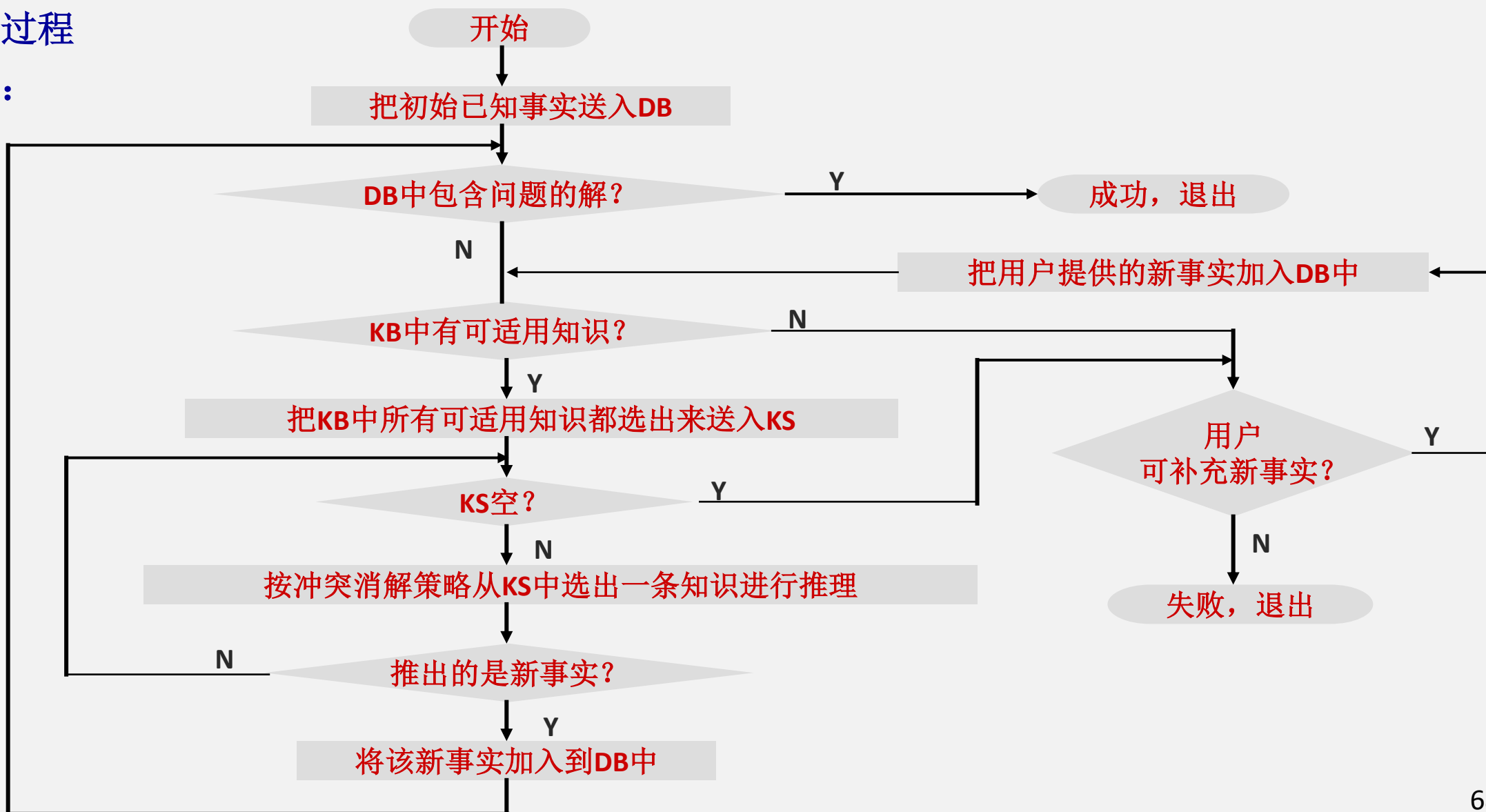
➤ 正向推理的基本思想：

从用户提供的初始已知事实出发，在知识库**KB**中找出当前可适用的知识，构成可适用知识集**KS**，然后按某种冲突消解策略从**KS**中选出一条知识进行推理，并将推出的新事实加入到数据库中作为下一步推理的已知事实，在此之后再在知识库中选取可适用的知识进行推理，如此重复，直到求得了所要求的解，或者知识库中再无可适用的知识为止。

专家系统设计——知识推理——控制策略——推理方向

正向推理过程

算法描述：



专家系统设计——知识推理——控制策略——推理方向

逆向推理

逆向推理是以某个假设目标作为出发点的一种推理，又称为目标驱动推理、逆向链推理及后件推理等。

➤ 逆向推理的基本思想：

首先选定一个假设目标，然后寻找支持该假设的证据，若所需的证据都能找到，则说明原假设成立；若无论如何都找不到所需证据，说明原假设不成立，此时需要另作新的假设。

➤ 推理过程算法描述（图示）

➤ 逆向推理的主要优缺点

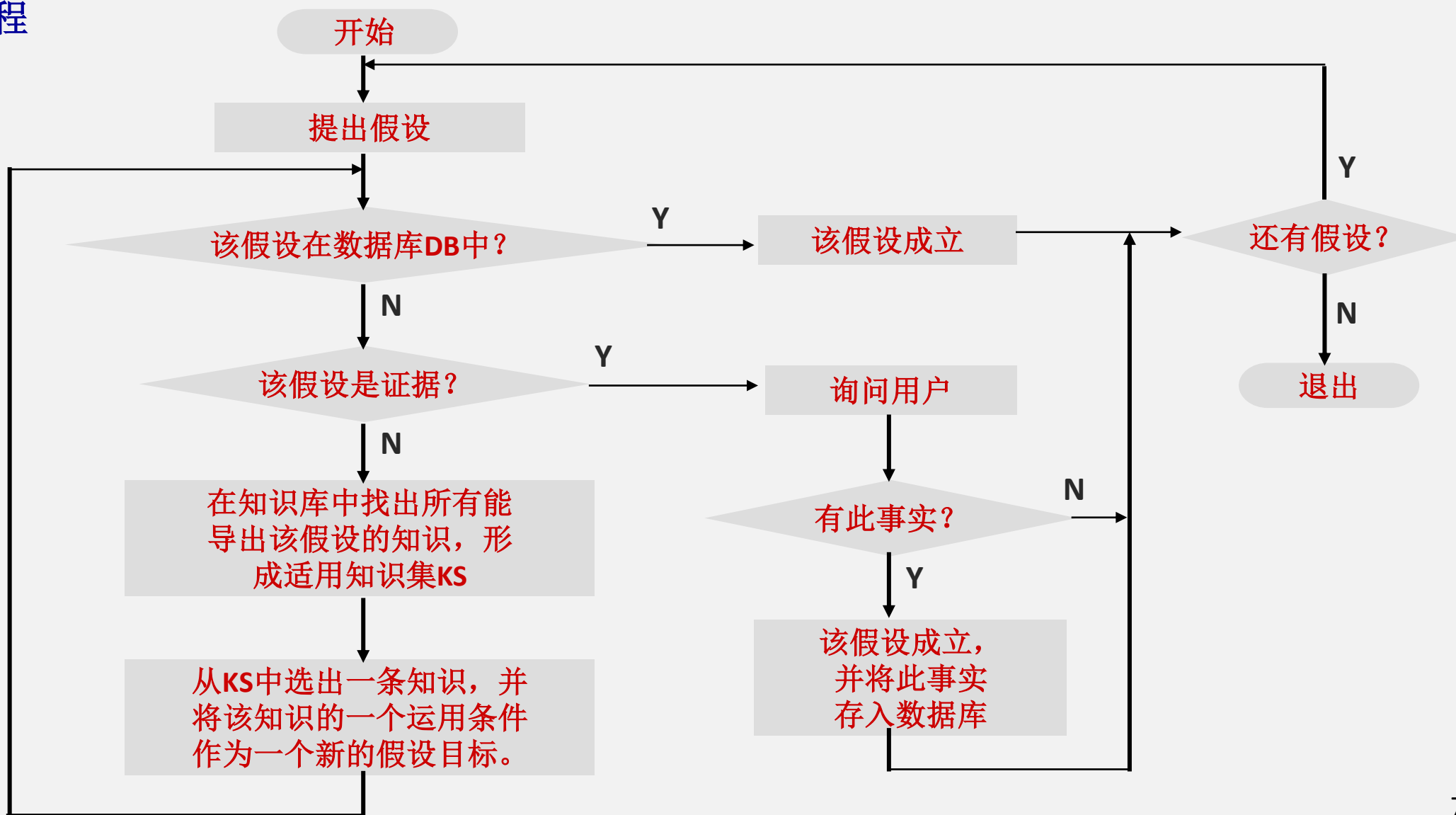
主语优点：不必使用与目标无关的知识，目的性强，同时还有利于向用户提供解释。

主要缺点：初始目标的选择有盲目性，若不符合实际，就要多次提出假设，影响到系统效率。

专家系统设计——知识推理——控制策略——推理方向

逆向推理过程

算法描述



专家系统设计——知识推理——控制策略——推理方向

混合推理

➤ 正、逆向推理存在的缺陷

正向推理——具有盲目、效率低等缺点；

逆向推理——若提出的假设目标不符合事实，也会降低系统效率。

为解决这些问题，把正向推理与逆向推理结合起来，这样既有正向又有逆向的推理称为混合推理。

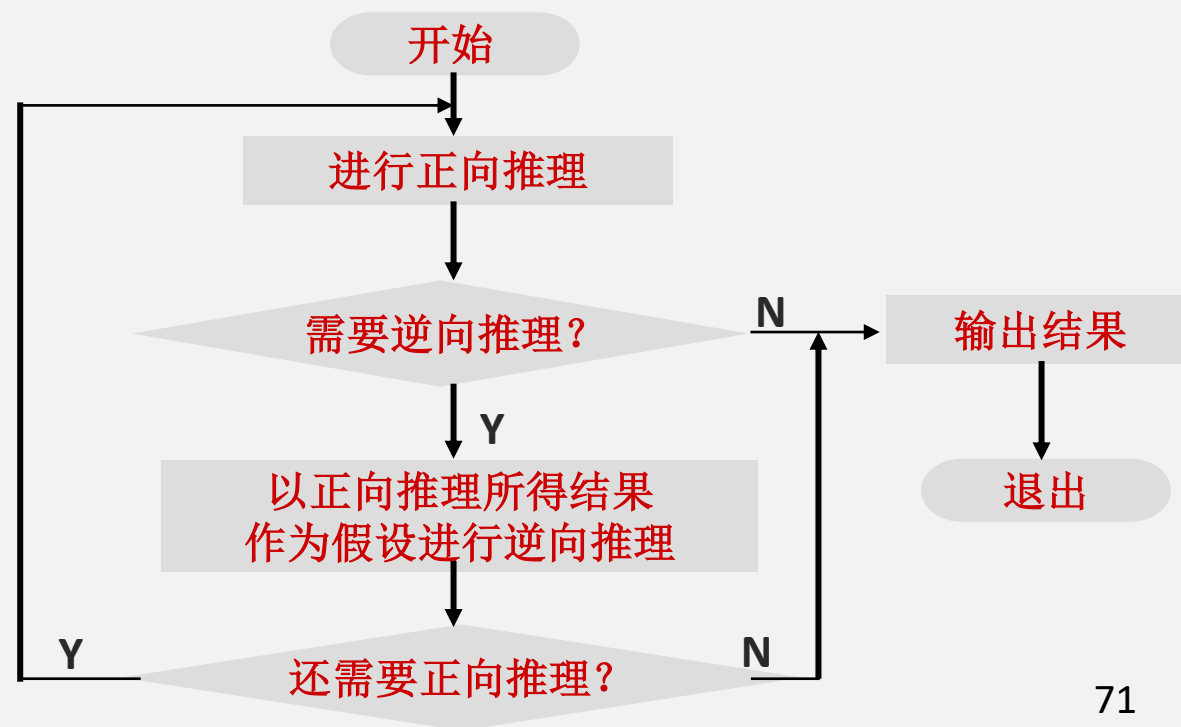
➤ 混合推理的两种情况

✓ 先正向再逆向

先进行正向推理，帮助选择某个目标，即从已知事实演绎出部分结果，然后再用逆向推理证实该目标或提高其可信度。

✓ 先逆向再正向

先假设一个目标进行逆向推理，然后再利用逆向推理中得到的信息进行正向推理，以推出更多的结论。



专家系统设计——知识推理——控制策略——推理方向

双向推理

双向推理是指正向推理与逆向推理同时进行，且在推理过程中某一步骤上“碰头”的一种推理方式。

➤ 基本思想

一方面根据已知事实进行正向推理，但并不推到最终目标；另一方面从某假设目标出发进行逆向推理，但并不推至原始事实，而是让它们在中途相遇，即由正向推理所得的中间结论恰好是逆向推理此时所需要的证据，这时推理就可结束，逆向推理时所做的假设就是推理的最终结论。

专家系统设计——知识推理——控制策略——模式匹配

➤ 基本概念

所谓模式匹配是指对两个知识模式的比较与耦合，即 检查这两个知识模式是否完全一致或近似一致。如果两者完全一致，或虽不完全一致但其相似程度落在指定的限度内，就称它们是可匹配的，否则为不可匹配。

模式匹配是推理中必须进行的一项重要工作，因为只有经过模式匹配才能从知识库中选出当前适用的知识，才能进行推理。

➤ 分类

若按匹配时两个知识模式的相似程度分，可分为确定性匹配与不确定性匹配两种。

专家系统设计——知识推理——控制策略——模式匹配

➤ 确定性匹配

指两个知识模式完全一致，或经过变量代换后变的完全一致。

例如，设有两个知识模式：

P_1 : father(李四, 李小四) and man(李小四)

P_2 : father(x, y) and man(y)

若用“李四”代换变量 x ，用“李小四”代换 y ，则 P_1 ， P_2 就变得完全一致，若用这两个知识模式进行匹配，则称它们是确定性匹配。

➤ 不确定性匹配

指两个知识模式不完全一致，但从总体上看，其相似程度落在指定的限度内。

无论是确定性匹配或不确定性匹配，在进行匹配时一般都需要进行变量的代换，有关概念和方法请参阅有关书籍。

专家系统设计——知识推理——控制策略——冲突消解策略

➤ 概念

在推理过程中，系统要不断地用当前已知的事实与知识库中的知识进行匹配，此时可能发生如下三种情况：

1. 已知事实不能与知识库中的任何知识匹配成功；
2. 已知事实恰好只与知识库中的一个知识匹配成功；
3. 已知事实可以与知识库中的多个知识匹配成功；或者有多个（组）已知事实都可与知识库中的一个知识匹配成功；或者有多个（组）已知事实可与知识库中的多个知识匹配成功。

第三种情况为发生了冲突，此时需要按一定的策略解决冲突，以便从中挑选一个知识用于当前的推理，称这一解决冲突的过程称为**冲突消解**。解决冲突时所用的方法称为**冲突消解策略**。

专家系统设计——知识推理——控制策略——冲突消解策略

➤以产生式系统为例进行较详细说明

✓产生式系统冲突

在产生式系统中，若出现下列情况就认为发生了冲突：

- **对正向推理而言**，如果有多条产生式规则的前件都和已知事实匹配成功；或者有多组不同的已知事实都与同一条产生式规则的前件匹配成功；或者以上两种情况同时出现。
- **对逆向推理而言**，如果有多条产生式规则的后件都和同一个假设匹配成功；或者有多条产生式规则的后件可与多个假设匹配成功。

✓冲突消解

冲突消解的任务是解决冲突。

- **对正向推理来说**，它将决定选择哪一组已知事实来激活哪一条产生式规则，使它用于当前的推理，产生其后件指出的结论或执行相应的操作。
- **对逆向推理来说**，它将决定用哪一个假设与哪一个产生式规则的后件进行匹配，从而推出相应的前件，作为新的假设。

专家系统设计——知识推理——控制策略——冲突消解策略

➤ 冲突消解策略

目前已有多种消解冲突的策略，其基本思想都是对知识进行排序，常用的有以下几种：

- **按针对性排序**——本策略是优先选用针对性较强的产生式规则。
- **按已知事实的新鲜性排序**——把数据库中后生成的事实称为新鲜的事实。
- **按匹配度排序**——在不确定性匹配中，为了确定两个知识模式是否匹配，需要计算这两个模式的相似程度，当其达到某个预先规定的值时，则认为它们是可匹配的。相似度又称为匹配度。
- **根据领域问题的特点排序**——某些领域问题，预先可知道它的某些特点，此时可根据这些特点把知识排成固定的顺序。
- **按上下文限制排序**——把产生式规则按它们所描述的上下文分成若干组，在不同的条件下，只能从相应的组中选取有关的产生式规则。这样，不仅可以减少冲突的发生，而且由于搜索范围小，也提高了推理效率。
- **按冗余限制排序**——如果一条产生式规则被应用后将产生冗余知识，则降低了它被应用的优先级。
- **按条件个数排序**——如果有多条产生式规则生成的结论相同，则要求条件少的产生式规则被优先应用，因为要求条件少的规则匹配时花费的时间较少。

专家系统设计——知识推理——控制策略——搜索策略

➤ 什么是搜索

人工智能所要解决的大部分问题是结构不良或非结构化的问题，对这样的问题一般不存在成熟的求解算法可供利用，而只能是利用已有的知识一步步摸索着前进。

在此过程中，存在着如何寻找可用知识的问题，即如何确定推理路线，使其付出的代价尽可能的少，而问题又能得到较好的解决。

如：在正向推理中，

- 对已知的初始事实，需要在知识库中寻找可使用的知识，这就存在按何种线路进行寻找。
- 另外，可能存在多条线路都可实现对问题的求解，这就又出现按哪一条线路进行求解以获得较高的运行效率的问题。

像这样根据问题的实际情况不断寻找可利用的知识，从而构造一条代价较少的推理路线，使问题得到圆满解决的过程称为搜索。

专家系统设计——知识推理——控制策略——搜索策略

➤ 搜索分类

搜索分为盲目搜索和启发式搜索。

盲目搜索——按预定的控制策略进行搜索，在搜索过程中获得的中间信息不用来改进控制策略。这种搜索具有盲目性，效率不高，不便于复杂问题的求解。

启发式搜索——在搜索中加入了与问题有关的启发性信息，用以指导搜索朝着最有希望的方向前进，加速问题的求解过程并找到最优解。

➤ 搜索策略

1. 状态空间的搜索策略

基本思想：首先把问题的初始状态作为当前状态，选择适用的算符对其进行操作，生成一组子状态，然后检查目标状态是否在其中出现。若出现，则搜索成功，找到了问题的解；若不出现，则按某种搜索策略从已生成的状态中再选一个状态作为当前状态。重复上述过程，直到目标状态出现或者不再有可供操作的状态及算符时为止。

专家系统设计——知识推理——控制策略——搜索策略

搜索策略包括:

- 广度优先搜索
- 深度优先搜索
- 有界深度优先搜索
- 代价树的广度优先搜索
- 代价树的深度优先搜索

(以上属于盲目搜索策略)

- 局部择优搜索
- 全局择优搜索

(以上搜索属于启发式搜索)

专家系统设计——知识推理——控制策略——搜索策略

➤搜索策略

2. 与/或树的搜索策略

基本思想：把原始问题作为初始节点，并把它作为当前节点；应用分解或等价变换算符对当前节点进行扩展；为每个子节点设置指向父节点的指针；选择合适的子节点作为当前节点，反复执行上述步骤，直至初始节点被标示为可解节点或不可解节点为止。

整个搜索过程所形成的节点和指针结构称为搜索树。

搜索策略包括：

- 与/或树的广度优先搜索
- 与/或树的深度优先搜索
(以上属于盲目搜索策略)
- 与/或树的有序搜索
- 博弈树的启发式搜索
(以上搜索属于启发式搜索)

专家系统设计——知识推理——逻辑推理方法

➤经典逻辑推理

经典逻辑推理是根据经典逻辑（命题逻辑和一阶谓词逻辑）的逻辑规则进行的一种推理，又称机械—自动定理证明。主要推理方法有自然演绎推理、归结演绎推理和与/或形演绎推理（只讲前两个方法）等。这种推理是基于经典逻辑的，其真值只有“真”与“假”两种，因此它是一种精确推理，或称为确定性推理。

专家系统设计——知识推理

1. 自然演绎推理

从一组已知为真的事实出发，直接运用经典逻辑的推理规则推出结论的过程称为自然演绎推理。基本的推理规则是 **P规则**、**T规则**、假言推理、拒取式推理等。

谓词公式的永真蕴含

对于谓词公式 **P** 和 **Q**，如果 $P \rightarrow Q$ 永真，则称 **P** 永真蕴含 **Q**。且称 **Q** 为 **P** 的逻辑结论，称 **P** 为 **Q** 的前提，记为： $P \Rightarrow Q$ 。

- **P规则**：在推理的任何步骤上都可以引入前提；
- **T规则**：推理时，如果前面的步骤中有一个或多个公式永真蕴含公式 **S**，则把 **S** 引入推理中
- 假言推理 $P, P \rightarrow Q \Rightarrow Q$ 由 $P \rightarrow Q$ 及 **P** 为真，可推出 **Q** 为真。
- 拒取式推理 $\neg Q, P \rightarrow Q \Rightarrow \neg P$ 由 $P \rightarrow Q$ 及 **Q** 为假，可推出 **P** 为假。

专家系统设计——知识推理

例 已知如下事实：

(1) 凡是容易的课程小王（Wang）都喜欢；

(2) C班的课程都是容易的；

(3) ds是C班的一门课程；

求证：小王喜欢ds这门课。

证明：首先定义谓词：

EASY(x): x 是容易的；

LIKE(x, y): x 喜欢 y；

C(x): x 是 C 班的一门课。

专家系统设计——知识推理

把上述已知事实及待求证的问题用谓词公式表示出来：

$EASY(x) \rightarrow LIKE(Wang, x)$ 凡是容易的课程小王（Wang）都喜欢；

$(\forall x) (C(x) \rightarrow EASY(x))$ C班的课程都是容易的；

$C(ds)$ ds是C班的一门课程；

$LIKE(Wang, ds)$ 小王喜欢ds这门课（待求证问题）。

应用推理规则进行推理：

因为： $(\forall x) (C(x) \rightarrow EASY(x))$

所以： $C(y) \rightarrow EASY(y)$

因为： $C(ds), C(y) \rightarrow EASY(y) \Rightarrow EASY(ds)$ P规则及假言推理

所以： $EASY(ds), EASY(x) \rightarrow LIKE(Wang, x)$

$\Rightarrow LIKE(Wang, ds)$ T规则及假言推理

即：小王喜欢ds这门课程。

专家系统设计——知识推理

2. 归结演绎推理

应用归结原理证明定理的过程称为归结反演。

(1) 定理证明的过程

I. 问题描述：设有公式集 F ，希望从 F 证明某个目标 Q ；

II. 方法：把 F 、 Q 组成一个新公式，用归结法推出一个归结式为 NIL ；

III. 过程：

设 F 为已知前提的公式集， Q 为目标公式（结论）：

- 否定 Q ，得到 $\neg Q$ ；
- 把 $\neg Q$ 并入到公式集 F ，得到 $\{F, \neg Q\}$ ；
- 把公式集 $\{F, \neg Q\}$ 化为子句集 S ；

• 应用归结原理对子句集中的子句进行归结，并把每次归结得到的归结式都并入 S 中。如此反复进行，若出现了空子句，则停止归结，此时就证明了 Q 为真。

专家系统设计——知识推理

基础知识：子句生成

I. 子句的定义：由文字的析取组成的公式；

II. 一个谓词公式化为子句集的方法：

- 消去蕴含符号 (\rightarrow)：

如： $P \rightarrow Q \Leftrightarrow \neg P \vee Q$ (连接词化规律)

- 把否定符号移到每个谓词符号的前面：

如： $\neg(\forall x)P \Leftrightarrow (\exists x)(\neg P)$ (量词转换律)

$\neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$ (德.摩根律)

专家系统设计——知识推理

基础知识：归结 (Resolution)

I. 定义：设有两个子句，且两者有一个互补的文字，从这两个子句推断出一个新子句的过程称为归结。

(互补文字：若 P 是原子谓词公式，则称 P 与 $\neg P$ 为互补文字。)

II. 归结原理：

归结原理的基本思想：假设系统有两个子句 $P \vee Q$ 和 $\neg P \vee R$ 同时出现在推理的前提中，则前提为真的条件是这两子句必须同时为真。因为 P 和 $\neg P$ 不可能同时为真，从而我们可以推得前提为真的条件是 $Q \vee R$ 为真，即我们能够由子句 $P \vee Q$ 和 $\neg P \vee R$ 消去其中的 P 和 $\neg P$ 而得到 $Q \vee R$ 。所以，归结原理也称为“消解原理”。

专家系统设计——知识推理

例：某公司招聘工作人员，A、B、C三人应试，经面试后公司表示如下想法：

- (1) 三人中至少录取一人；
- (2) 如果录取A而不录取B，则一定录取C；
- (3) 如果录取B，则一定录取C；

求证：公司一定录取C。

专家系统设计——知识推理

证明： 设用 $P(x)$ 表示录取 x 。

把公司的想法用谓词公式表示如下：

$$(1) P(A) \vee P(B) \vee P(C)$$

$$(2) P(A) \wedge \neg P(B) \rightarrow P(C)$$

$$(3) P(B) \rightarrow P(C)$$

把要求证的问题否定，并用谓词公式表示出来：

$$(4) \neg P(C)$$

把上述公式化成子句集：

$$(1) P(A) \vee P(B) \vee P(C)$$

$$(2) \neg P(A) \vee P(B) \vee P(C)$$

$$(3) \neg P(B) \vee P(C)$$

$$(4) \neg P(C)$$

应用归结原理进行归结：

$$(5) P(B) \vee P(C)$$

(1) 与 (2) 归结

$$(6) P(C)$$

(3) 与 (5) 归结

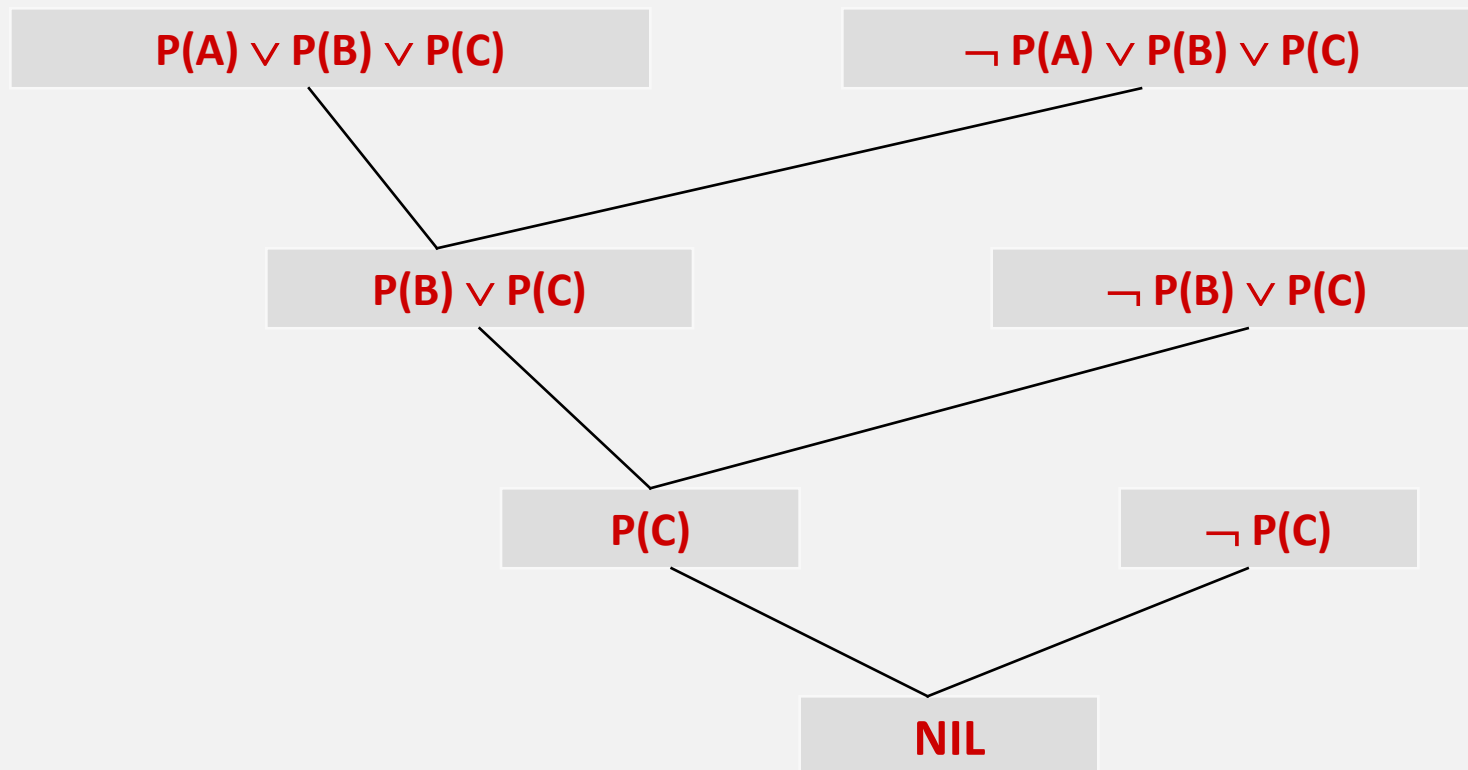
$$(7) \text{NIL}$$

(4) 与 (6) 归结

\therefore 公司一定录取B。

专家系统设计——知识推理

上述归结过程可用归结树表示。



内容

Contents

1. 专家系统概述
2. 专家系统基本原理
3. 专家系统分类
4. 专家系统设计——知识表示
5. 专家系统设计——推理机制
- 6. 专家系统开发工具**
7. 专家系统应用实例



专家系统开发工具

➤ 概况

专家系统开发工具是年代中期开始发展的，它比一般的计算机高级语言：**FORTRAN**、**PASCAL**、**C**、**LISP**和**PROLOG**等具有更强的功能。也就是说，专家系统工具是一种更高级的计算机程序设计语言。现有的专家系统工具，主要分为语言型工具、骨架型工具又称外壳、构造辅助工具和支撑环境等4类。

专家系统开发工具

➤专家系统开发语言

(1) LISP语言

基于表处理和函数，经典的LISP语言只有一种结构“表”。LISP语言与其它语言的最大区别是，它以表为对象处理符号，而不仅仅是数据。

LISP语言的主要特点：

- I. 函数式程序设计语言，LISP语言不同于传统语言，LISP程序实质上被描述为以一组接近数学形式定义的函数。
- II. 程序与数据等价，LISP的程序和数据具有相同的表示形式“表”，因此，LISP语言中程序与数据无严格界限。
- III. 递归控制结构，LISP函数多以递归定义，任何函数可直接或间接调用 本身。
- IV. 交互式解释执行。

专家系统开发工具

(2) Prolog 语言

Prolog 是一种逻辑程序设计语言，基于一阶谓词逻辑，是典型的叙述型语言（Declaration Language）。

Prolog语言的特点：

- (1) 建立在一阶谓词逻辑和归结原理基础上，有自动推理功能。
- (2) 用Prolog设计应用程序时，仅需指明领域中各对象间的关系和决策规则，而应用这些知识的推理由Prolog完成。
- (3) Prolog应用程序，由数据库和规则库组成。
- (4) Prolog程序设计要做三件事，即说明事实、定义规则、提出问题。

(3) 采用高级语言及面向对象语言

- (1) 首先必须用高级语言完成推理机，知识库建立等工具。
- (2) 在此基础上建立知识库、完成推理。

专家系统开发工具

➤专家系统外壳

专家系统外壳又称**骨架系统**，它是由一些已经开发成功，并且在实际使用中被证明为行之有效的专家系统演变而来的，即抽去这些专家系统中具体的知识，保留它的体系结构和功能，再把领域专用的界面改为通用的界面，就得到了相应的专家系统外壳。

显然，在专家系统外壳中，**知识表示模式、推理机制等都是确定的**。当使用这些外壳建造专家系统时，只需把相应领域的知识用外壳规定的模式表示出来装入到知识库中就行了。

重要的专家系统外壳有：EMYCIN，KAS，EXPERT 等。

专家系统开发工具

➤专家系统工具CLIPS

- 早期的专家系统工具大都用LISP、Prolog等编程语言开发， 它们的共同问题是运行速度慢，可移植性差， 解决复杂问题的能力差。
- 1984年美国航空航天局约翰逊空间中心（NASA' s Johnson Space Center）推出 CLIPS（C Language Integrated Production System）。
- CLIPS是一个基于Rete算法的前向推理语言，用标准C语言编写。它具有高移植性、高扩展性、强大的知识表达能力和编程方式以及低成本等特点。

专家系统开发工具

➤专家系统开发环境

- 可为专家系统的开发提供多种方便的构件，例如知识获取的辅助工具、适用各种不同知识结构的知识表示模式、各种不同的不确定推理机制、知识库管理系统等。
- AGE (attempt to generalize): 一种典型的模块组合式开发工具。
- 通过AGE构造专家系统的途径:
 - (1) 用户使用AGE现有的各种组件作为构造材料，方便地组合设计所需系统。
 - (2) 用户通过AGE的工具界面，定义和设计各种所需的组成部件，以构成自己的专家系统。

内容

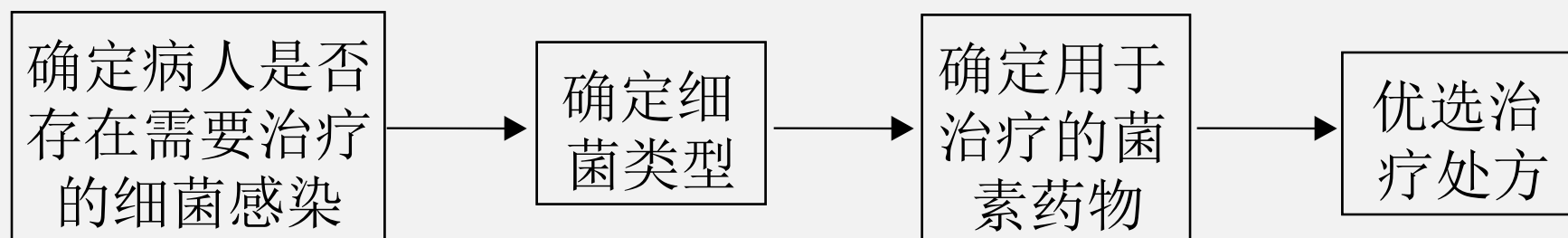
Contents

1. 专家系统概述
2. 专家系统基本原理
3. 专家系统分类
4. 专家系统设计——知识表示
5. 专家系统设计——推理机制
6. 专家系统开发工具
- 7. 专家系统应用实例**



专家系统应用案例——专家系统MYCIN

- MYCIN系统是著名的医学领域的专家协调，是由Stanford大学建立开发的对细菌感染疾病的诊断和治疗提供咨询的系统。
- 医生可以向系统输入病人信息，MYCIN系统对其进行诊断，并给出诊断结果和处方。系统使用LISP语言编写。
- 知识库库200多条规则，可是被51种病菌，正确树立23种抗生素可协统助医生诊断、治疗细菌感染性血液病，为患者提供最佳处方，成功地处理了数百个病例。

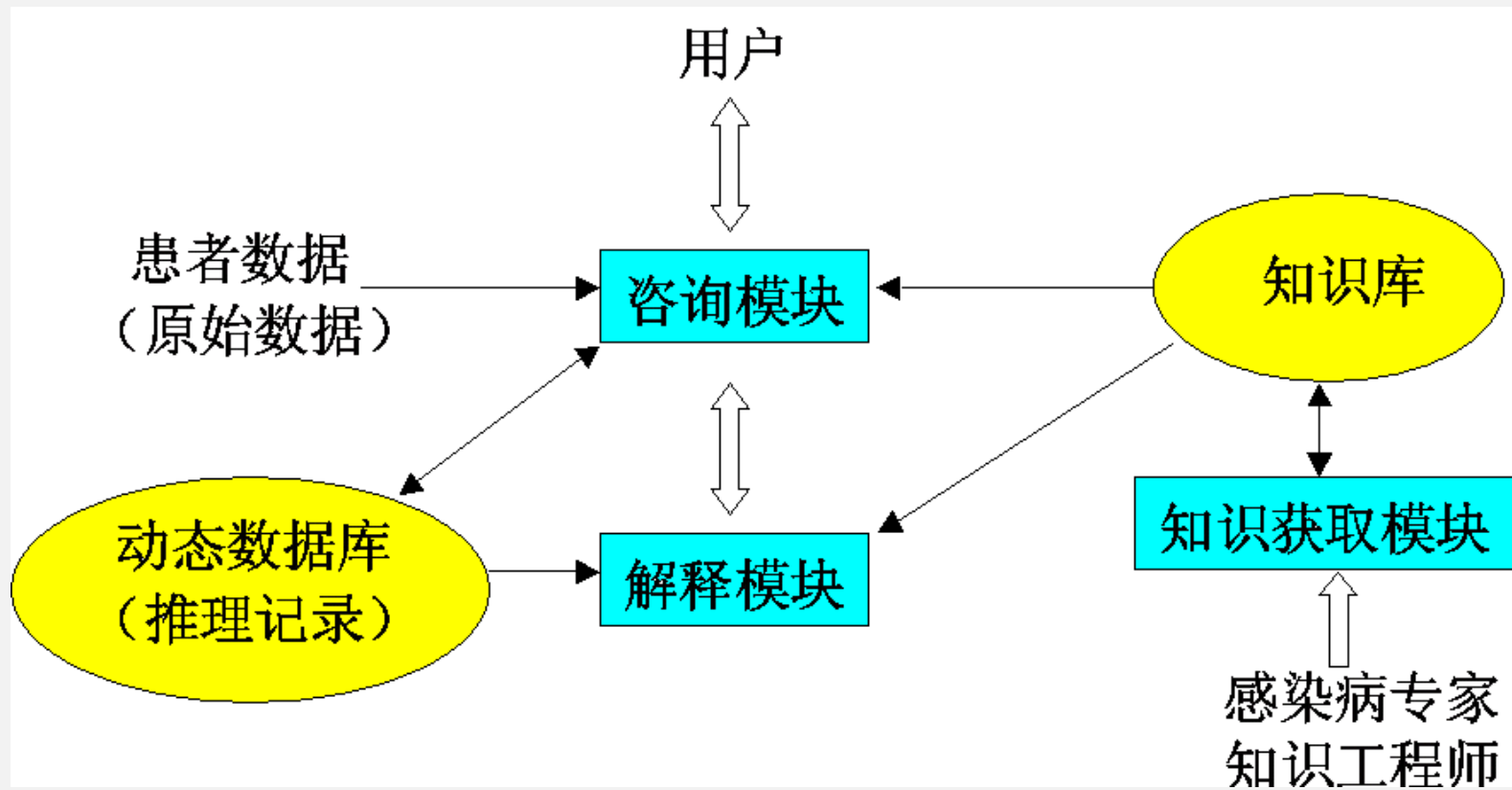


MYCIN系统的咨询过程

专家系统应用案例——专家系统MYCIN

系统结构

- 咨询开始时，先启动咨询系统，进入人一机对话状态。当结束咨询时，系统自动地转入解释子系统。解释子系统回答用户的问题，并解释推理过程。规则获取子系统只由建立系统的知识工程师所使用。当发现有规则被遗漏或不完善时，知识工程师可以利用这个系统来增加和修改规则。



MYCIN系统结构图

专家系统应用案例——专家系统MYCIN

知识表示

(1) 领域知识的表示：产生式规则。

■ **RULE 064** 如果：有机体染色是革兰氏阳性，

且 是有机形态是球状的，

且 有机体的生长结构呈链状，

则：存在证据表明该有机体为链球菌类，可信度为0.7。

■ **RULE 064**

PREMISE: (\$ AND (SAME CNTXT STALN GRAMPOS)

(SAME CNTXT MORPH COCCUS)

(SAME CNTXT CONFORM CHAINS))

ACTION: (CONCLUDE CNTXT IDENT STREPTO COCCUS TALLY 0.7)

专家系统应用案例——专家系统MYCIN

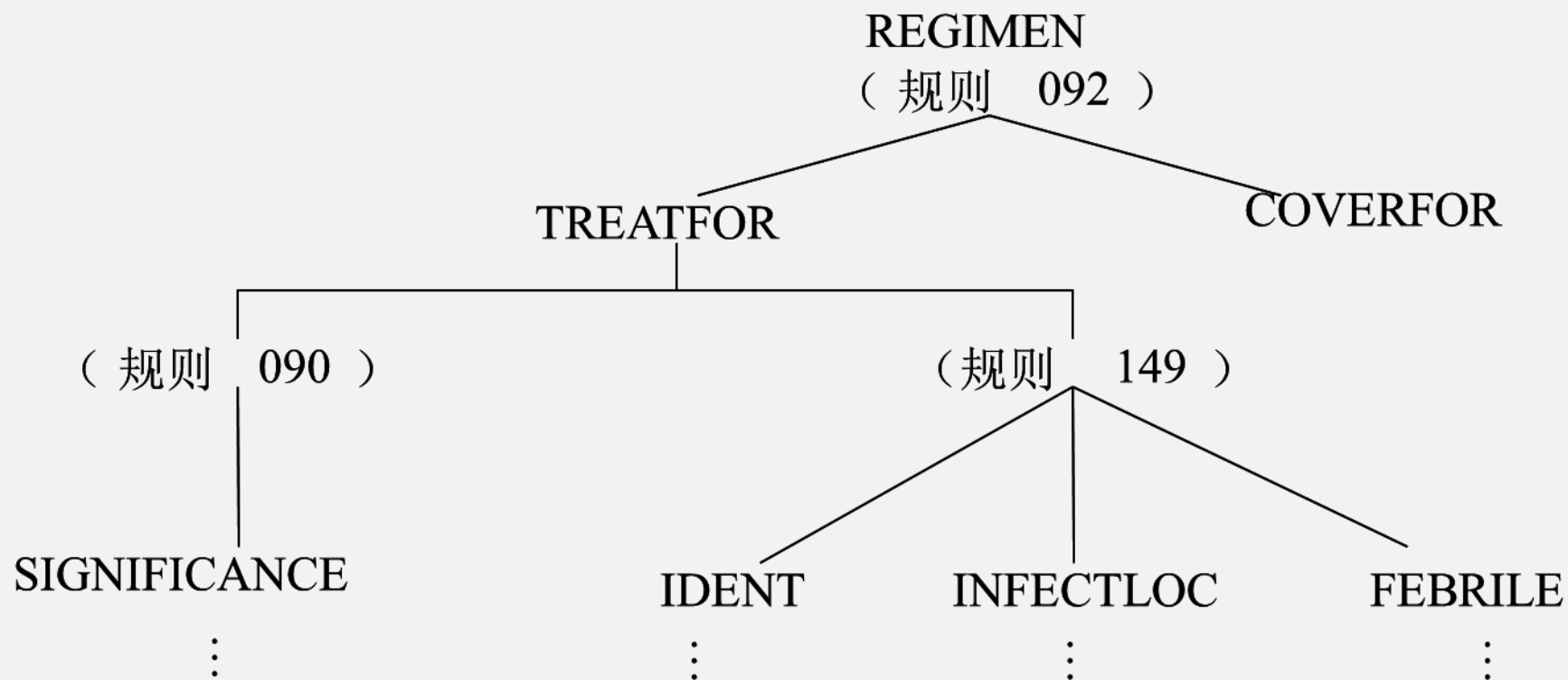
知识表示

(2) 临床参数的表示

- 临床参数：三元组（上下文树、属性、值）
- 例：三元组（机体-1，形态，杆状）
三元组（机体-1，染色体，革兰氏阴性）
- 临床数据：单值、是非值、多值。
- MYCIN系统有65个临床参数，按照其相对应的上下文分类。

专家系统应用案例——专家系统MYCIN

推理策略：反向推理、深度优先搜索策略



关于病人的上下文树

专家系统应用案例——专家系统MYCIN

推理策略

- MYCIN系统：通过两个子程序MONITOR和FINDOUT完成整个咨询和推理过程。
 - **MONITOR**：分析规则的前提条件是否满足，以决定拒绝该规则还是采用该规则，并将每次鉴定一个前提后的结果记录在动态数据库中。
 - **FINDOUT**：检查MONITOR所需要的参数，它可能已在动态数据库中，也可以通过用户提问获取。

专家系统应用案例——专家系统MYCIN

治疗方案选择

(1) 生成可能的“治疗方案表”

例如： IF 细菌的特征是 *Pseudomonas*

THEN 建议在下列药物中选择治疗：

colistin (0.98)

polynyxin (0.96)

gentamicin (0.96)

carbenicillin (0.96)

sulfisoxazole (0.96)

专家系统应用案例——专家系统MYCIN

治疗方案选择

(2) 选择用药配方

- 该药物对细菌治疗的有效性。
- 该药物是否已用过。
- 该药物的副作用。

专家系统应用案例——专家系统MYCIN

知识获取

- (1) 告诉专家新建立的规则的名字（规则序号）。
- (2) 逐条获取前提，并从英文翻译成LISP表达。
- (3) 逐条获取结论动作，也从英文翻译为LISP表达。
- (4) 用LISP-english子程序将规则翻译成语言描述，显示给专家。
- (5) 提问专家是否同意这条翻译的规则；如果规则不正确，
专家进行修改并回到步骤(4)。
- (6) 检查新规则与其他旧规则之间的矛盾。
- (7) 如果有必要，可调用辅助分类规则对新规则分类。
- (8) 把规则加入LOOKHEAD表。
- (9) 把规则加入CONTAIED-IN表、UPDATED-BY表。
- (10) 告诉专家系统新规则已是规则库中的一部分了。



中国科学院自动化研究所
复杂系统管理与控制国家重点实验室

感谢聆听



The State Key Laboratory of Management and Control for Complex Systems
复杂系统管理与控制国家重点实验室