

课程概述

毛文吉

中国科学院自动化研究所



提 纲

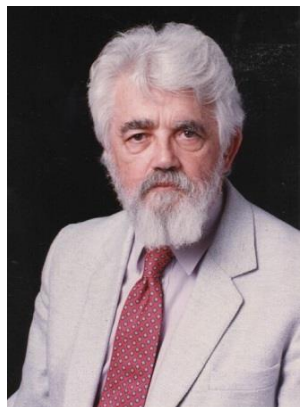
- 智能研究的发展历程
- 多智能体领域的主要进展
- 课程安排与考核方式



人工智能学科的起源



- 1956年夏美国达特茅斯学院
“**如何用机器模拟人的智能**”
- 提出“人工智能”这一概念
- 标志学科的诞生



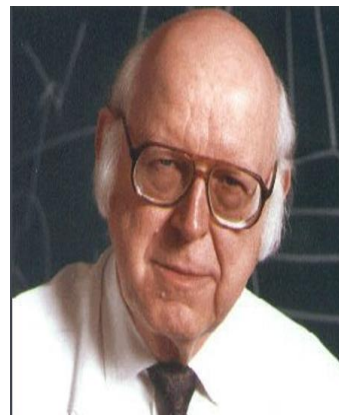
John McCarthy



Marvin Minsky



Herbert Simon

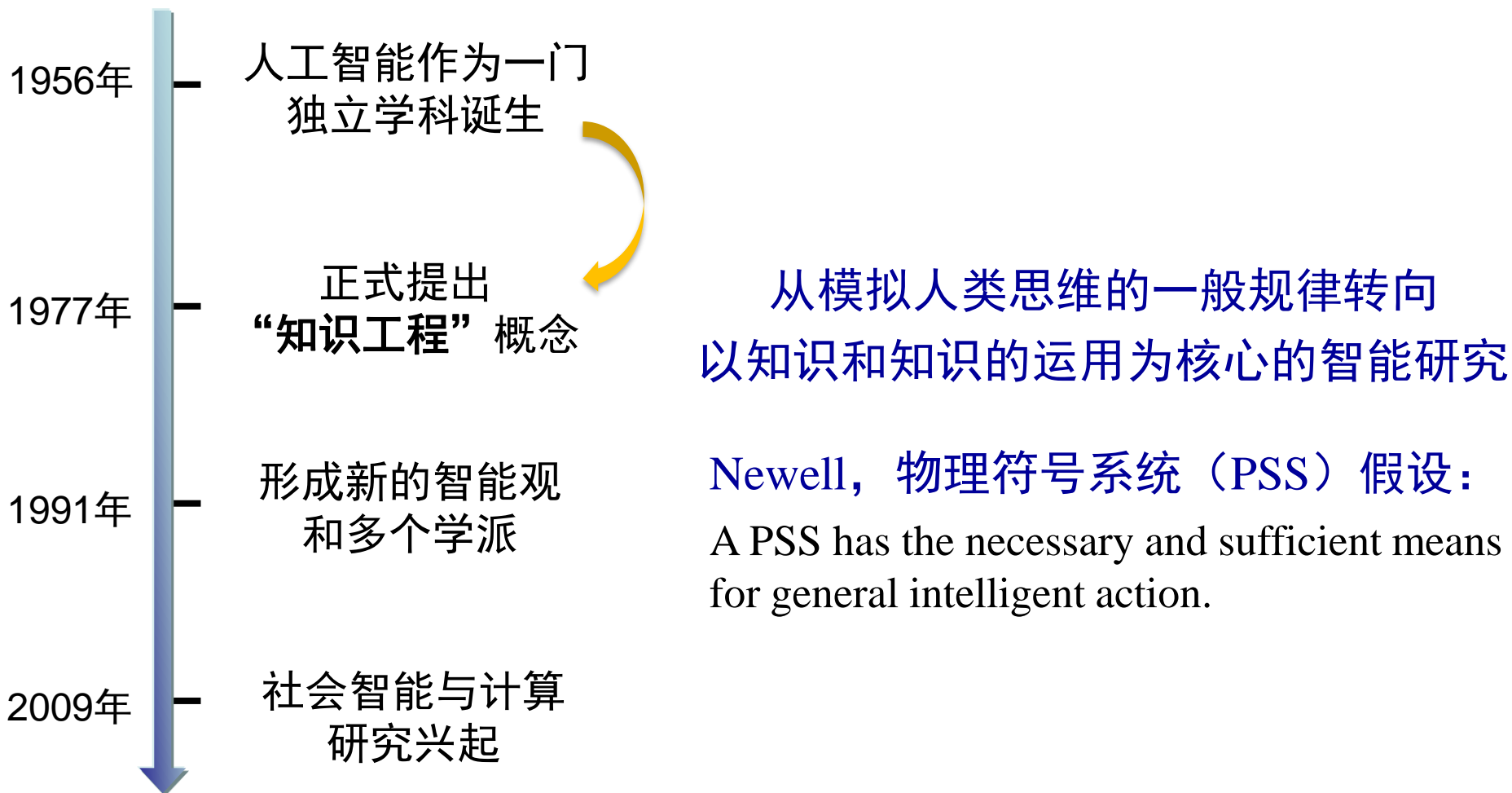


Allen Newell



Arthur Samuel

AI 发展历程



知识工程

■ 核心领域

Edward Feigenbaum



Edward Feigenbaum

Edward Albert Feigenbaum is a professor emeritus of computer science at Stanford University and a cofounder of three start-up firms in applied AI: IntelliCorp, Teknowledge, and Design Power. Often called the “father of expert systems,” he founded the Knowledge Systems Laboratory at Stanford University, where the first expert system, DENDRAL, was developed. Feigenbaum completed his BS and a PhD at Carnegie Mellon University. In his 1959 PhD thesis, carried out under the supervision of Herbert Simon, he developed EPAM, one of the first computer models of how people learn. He helped establish Stanford’s SUMEX-AIM national computer facility for applications of AI to medicine and biology. He has been a member of the Board of Regents of the National Library of Medicine and was a member of the Board of Directors of Sperry Corporation. He was the second president of the American Association for Artificial Intelligence. Feigenbaum has received the ACM Turing Award and the Feigenbaum Medal of the World Congress on Expert Systems. From 1994 to 1997, he served as US Air Force Chief Scientist, and he won a US Air Force Exceptional Civilian Service Award. He is a member of the National Academy of Engineering and American Academy of Arts and Sciences.

HALL OF FAME

Engineering Knowledge

By Alun Preece

Arguably more than anyone, Ed Feigenbaum was responsible for getting AI out of the lab and into the enterprise.

The key insight behind what he termed *expert systems* was to capture valuable problem-solving knowledge in a knowledge

base, such that a machine could perform automated inference on it. Typically, the knowledge was captured by means of rules or descriptive frames. As a result, significant advances were made in techniques for rich knowledge and information representation and on efficient methods for large-scale inference. Indeed, one of Feigenbaum’s key interests was the use of heuristic methods for solving computationally hard problems. Motivated in part to compete with Japan’s “Fifth Generation Computer” project, Feigenbaum worked to alert companies and organizations to the potential benefits of knowledge-based systems.

Feigenbaum coined the term *knowledge engineering* to refer to the incorporation of knowledge into software systems in order to perform complex problem-solving tasks. Essentially, knowledge engineering represented the first widespread industrialization of AI. The AI field was transformed during the years of the expert systems boom, with significant investment both in basic

research and application development. The limits of the technology were quickly identified, but nevertheless numerous useful systems were deployed at all levels within enterprises—some of which continue to be used today.

In 1994, at the tail-end of the expert systems boom, Feigenbaum’s work was recognized—jointly with that of Raj Reddy—by an ACM Turing Award for “pioneering the design and construction of large-scale artificial intelligence systems, demonstrating the practical importance and potential commercial impact of artificial intelligence technology.” As professor emeritus at Stanford, Feigenbaum has focused interest, as a Board of Trustees member of the Computer History Museum, on preserving the history of computer science, and with the Stanford Libraries on software for building and using digital archives.

Feigenbaum’s legacy is broad. We can recognize patterns and characteristics of

the technology in intelligent assistants and software “wizards” that step users through complex tasks. In the 1990s, there was a great deal of interest in technical aspects of knowledge sharing, by means of federated knowledge bases connected to the Internet. Organizations remained committed to the idea of capturing and leveraging their valuable knowledge, and expert systems technology became one element of the emerging sociotechnical knowledge management field. Effective heuristic problem-solving techniques remain an important tool within many enterprises, for example, for scheduling and resource-allocation tasks. We can also see components of knowledge-based systems technology and knowledge engineering methods in diverse areas such as business rules systems, database integrity management, and Semantic Web ontologies. Ideas from the expert systems field are also evident in recent high-profile launches such as IBM’s Watson system and the Wolfram Alpha computational knowledge engine. Ed Feigenbaum will always be associated with the 1980s assertion that “knowledge is power.” Thirty years later, a great many AI systems developers still agree.

Alun Preece is a professor of intelligent systems in the School of Computer Science and Informatics at Cardiff University, UK.

专家系统的弱点

- **脆弱性**

专业知识过于狭窄，在解决本专业问题时有高超的水平；一旦稍微偏离原领域，性能急剧下降

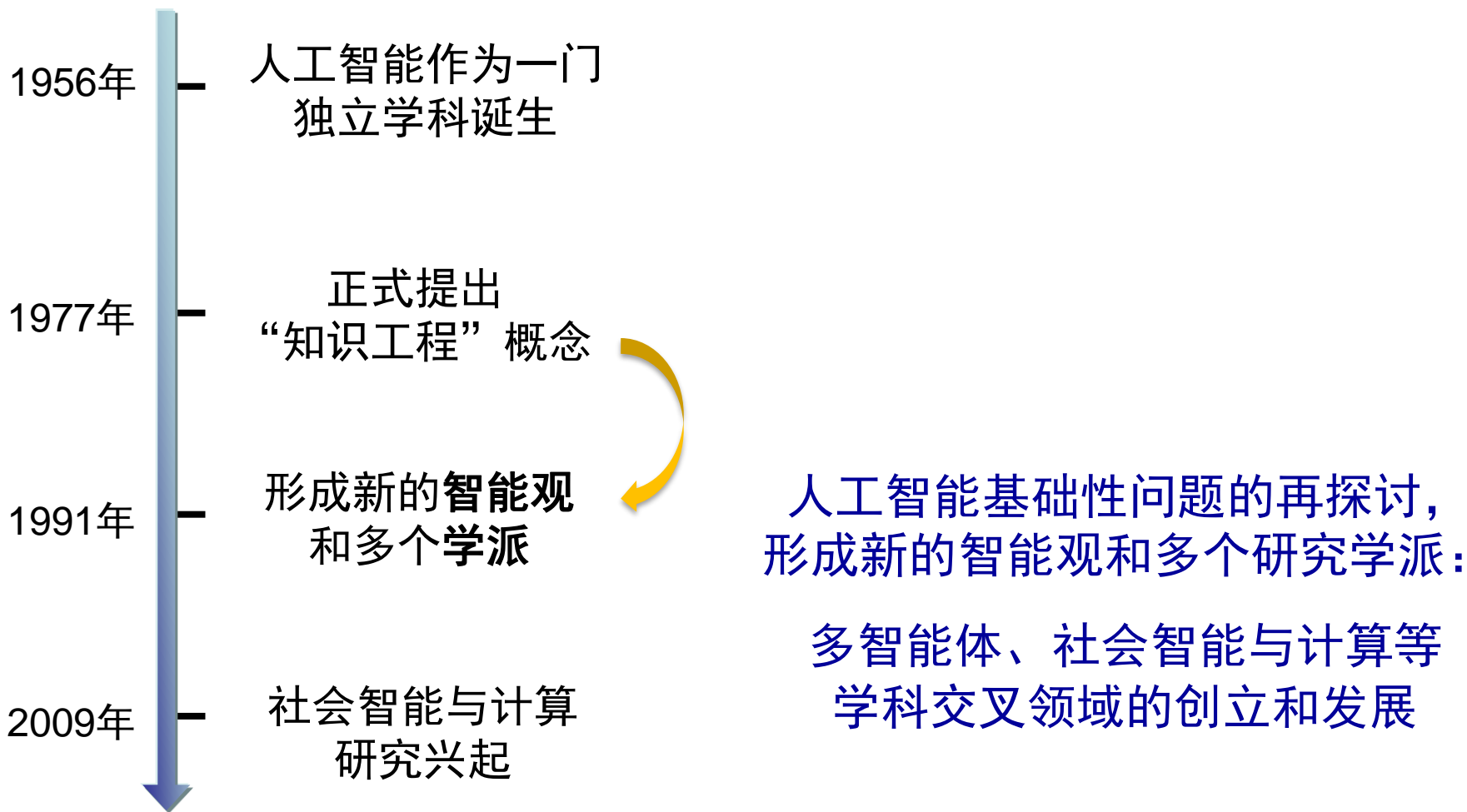
解决方法：增加知识的宽度，使系统具有常识和一般知识

- **知识获取的困难性**

知识获取作为建造知识系统的瓶颈，是一个长期存在的关键研究难点

针对知识本身的广袤性、模糊性、不确定性，以建造高性能知识系统为目标，结合多领域的研究成果，加强以知识为核心的智能研究

AI 发展历程



行为主义智能观

- 继符号主义和联接主义两大智能观，MIT 的 **Brooks** 提出了**无需表示和理性的智能**

认为人工智能研究应以复杂的现实世界为背景，先使机器具有初级智能，再在实践中逐渐增长智能

靠“感知-行动”的机器虫，没有推理、规划等高级思维能力，但感知、运动等应付环境的能力大大超过了现有的机器人

- **反对派的质疑：**今天的虫，明天能变成人吗？

逻辑学派

- **逻辑对人工智能的贡献：** 提供了严格的知识表示和有效的推理机制

代表人物 **Nilson**：主张在统一的逻辑框架内进行人工智能研究，认为各种类型的逻辑是人工智能的基础

代表人物 **McCarthy**：主张以非单调逻辑为中心的常识推理作为人工智能的基础

- **反对派的质疑：** 人类的许多智能活动，如：联想、直觉等都是非逻辑的，不可能用统一的逻辑框架来表达

N. J. Nilsson. Logic and Artificial Intelligence. Special Issue on Foundations of AI. *Artificial Intelligence*, 47(1-3):31-56, 1991.

分布式人工智能学派

- Hewitt主张用“分布式”的方法，在开放系统中研究人工智能问题

认为智能系统是一个开放系统

智能系统的行为不仅体现在单个个体的活动，而应当体现在多个体之间既相互依存、又相互冲突的活动中

最终作出的决策是通过在多个个体之间进行协调的分布式方法

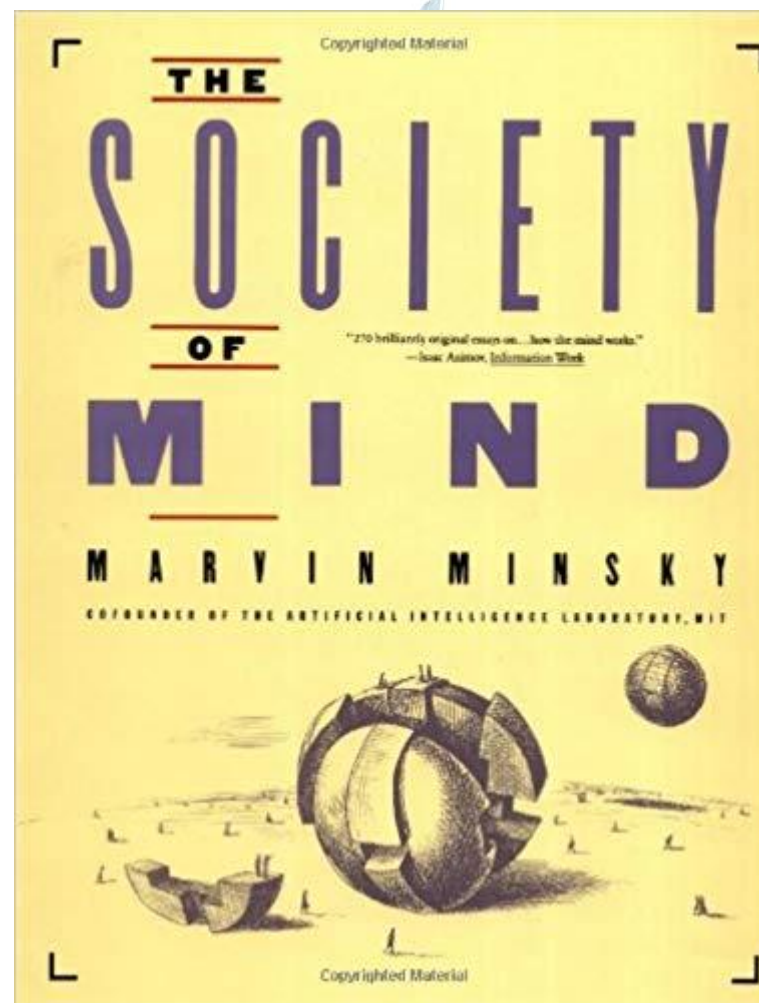
- 对多智能体研究的出现和发展起到了积极的推动作用

C. Hewitt. Open Information-Systems Semantics for Distributed Artificial Intelligence. Special Issue on Foundations of AI. *Artificial Intelligence*, 47(1-3):79-106, 1991.

The Society of Mind (Minsky 86)

- 认为人类智能并不存在统一的理论，而是由大量具有某种思维能力的单元组成的复杂社会（即“思维的社会”）

by Marvin Minsky



多智能体领域的形成

- 1980s后期:

从分布式人工智能领域开始区分协作型和自私型智能体, 研究建立在博弈论和经济学概念之上的自私智能体交互

- 1990s初期:

- 1) 智能体的出现促进了软件构件理念的更新: Agent-Oriented Programming (AOP)
- 2) 智能体通信语言的标准化: KQML/KIF → FIPA

- 1990s后期:

进一步拓宽多智能体系统的现实应用领域: 机器人世界杯 (RoboCup) 和RoboCup营救

多智能体领域的形成

- 国际会议的推动：

- 1) International Workshop on Agent Theories, Architectures, and languages (ATAL, 1994-)
- 2) International Conference on Multi-Agent Systems (ICMAS, 1995-)
- 3) International Conference on Autonomous Agents (AA, 1999-)

- 专刊：Communications of the ACM “Intelligent Agents” 1994



- 国际智能体及多智能体系统协会 (IFAAMAS)

多智能体领域的发展

- 2000–2010:

- 1) 电子商务应用：交易智能体在谈判和竞拍等领域的发展、基于博弈论的拍卖和机制设计研究
- 2) 与认知和社会科学领域交叉的多智能体模拟技术，以及分布式传感网络、自动驾驶车辆和机器人系统、自主计算等

- 2010–2020:

- 1) 基于多智能体的大规模棋牌类游戏应用
- 2) 多智能体学习（特别强化学习）、算法博弈论（算法机制设计、计算社会选择理论等）、分布式问题求解/分布式规划

领域研究热点与挑战

- 超大规模分布式人工智能系统
 - 超大规模系统应用面临可能出现的状态和采取的决策随系统规模呈指数级增长
- 多智能体系统的鲁棒性和安全性
 - 以深度学习和深度神经网络为基础的算法鲁棒性问题
- 多智能体决策的可解释性
 - 基于深度模型的多智能体学习算法带来的“黑箱”问题
- 传统方法与深度模型的结合
 - 深度神经网络与传统符号计算、传统控制理论的结合

主要期刊和会议

- **国际会议：**

- 1) **International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS, 2002-)**
- 2) The AAI Conference on Artificial Intelligence (AAAI, 1980-)
- 3) International Joint Conference on Artificial Intelligence (IJCAI, 1969-)

- **国际期刊：**

- 1) **Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS, 1998-)**
- 2) Journal of Artificial Intelligence Research (Open access, 1993-)
- 3) Artificial Intelligence Journal (Elsevier, 1970-)
- 4) ACM Computing Surveys (1969-)

国内情况

- 国内会议

- 1) 全国Agent理论与应用学术会议（2006-2012，两年一次）
- 2) 2013年起，并入“中国计算机学会人工智能会议”（CCF-AI，两年一次）
- 3) 2014年起，举办“中国智能体及多智能体系统”研讨会（2014-2022）

- 国内学会

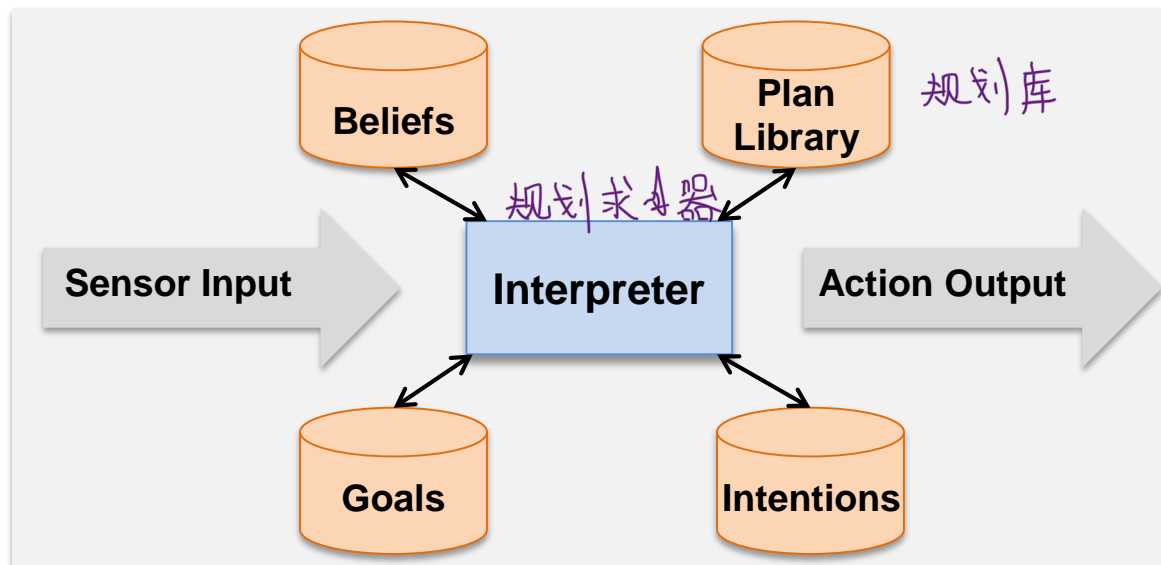
- 1) 2014年，成立中国计算机学会人工智能与模式识别专委会“多智能体与智能系统学组”
- 2) 2022年12月，成立中国人工智能学会心智计算专委会

什么是智能体（Agent）？

特性	含义
Situated	Sense and act in dynamic/uncertain environments
Flexible 灵活的	^{反应性} Reactive (responds to changes in the environment) ^{主动性} Pro-active (acting ahead of time)
Autonomous 自主的	Exercises control over its own actions
^{目标导向的} Goal-oriented	Purposeful
^{持久的} Persistent	Continuously running process
^{社交的} Social	Interacts with other agents/people
^{学习性} Learning	Adaptive ^{自适应的}
^{可移动的} Mobile	Able to transport itself
^{具有个性的} Personality	Character, Emotional state

PRS: 一个BDI智能体的实现

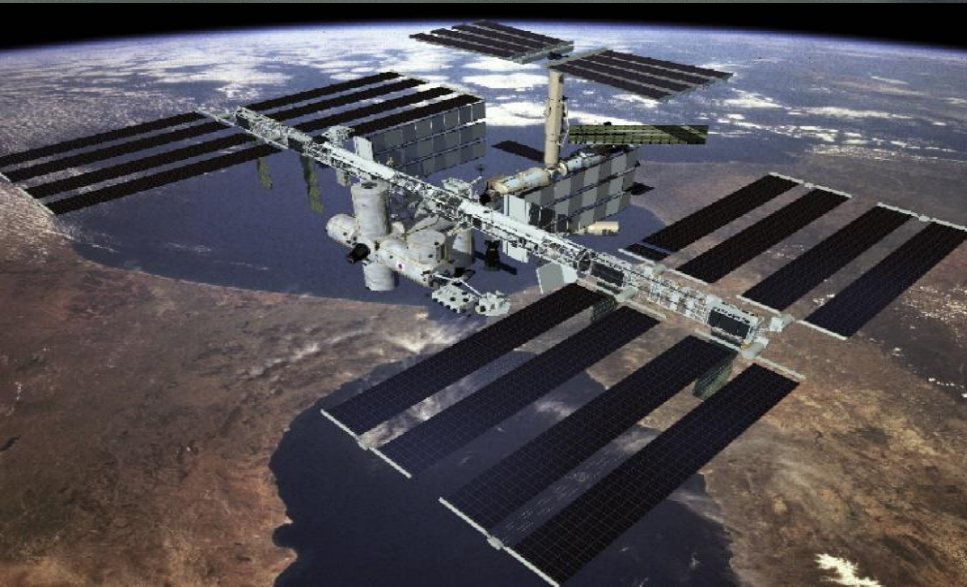
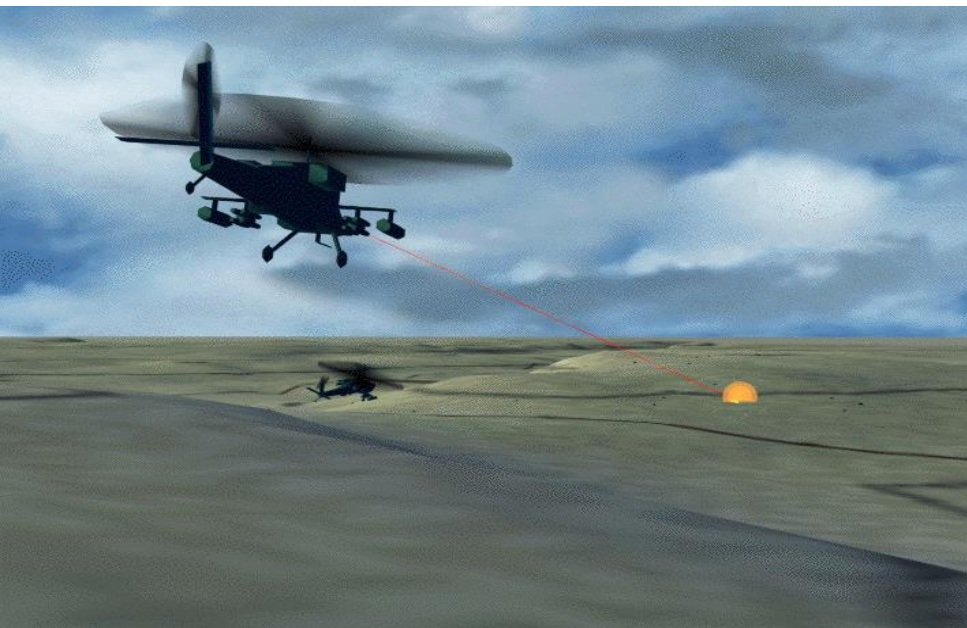
- BDI (Belief-Desire-Intention) 智能体系统结构



- 多智能体系统的应用

- *Synthetic agents*: Education, training, entertainment ...
- *Software agents*: Information gathering, logistics planning
- *Robotic agents*: Spacecraft constellations, space-station, rovers...
- *Assistants*: Support human collaboration, groupware...

多智能体（Multi-Agent）系统



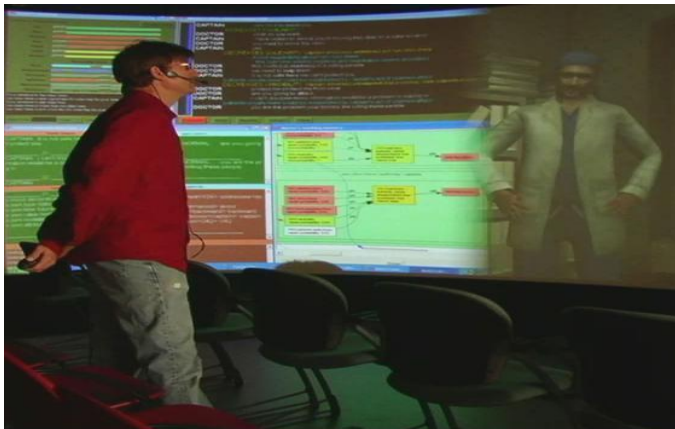
虚拟智能体与智能交互环境



Mission Rehearsal Exercises



Carmen's Bright IDEAS



SASO Simulation & Training



Tactical Language Training

课程计划安排

第1讲 (28/2) :

概述、自主智能体与多智能体系统

第2-3讲 (7, 14/3) :

智能体心智模型、规划问题求解

第4讲 (21/3) :

意图识别、团队合作与智能体通信

第5讲 (28/3) :

面向智能体的认知与社会模拟

课程计划安排

第6-7讲 (4, 11/4) :

多智能体博弈专题 (李林静)

第8-10讲 (18, 23, 25/4) :

多智能体学习专题 (张启超)

第11-12讲 (9, 16/5) :

多智能体协作与协商

第13讲 (23/5) :

多智能体系统结构



考核方式

- **教学目的：** 相互学习、共同进步
- **考核方式：**
 - 1) 课堂学习（包括考勤）： 10%
 - 2) 平时作业（2次左右）： 20%
 - 3) 课堂报告（Optional）： 0或30%
 - 4) 期末考试（开卷形式）： 70或40%
- **期末考试安排：**
 - 1) 时间： 5月30号上课时间（随堂考试）
 - 2) 内容： 本课程范围内需要理解和掌握的知识
（本课程涉及的基础和预备知识不作考察）

考核方式（续）

- 关于课堂报告：

1) 时间： 安排在每次课最后半小时； 如果合作最多2人

2) 邮件： 发给我和助教（张睿珂）

wenji.mao@ia.ac.cn

zhangruike2020@ia.ac.cn

内容： 希望报告的日期、报告的论文及出处

(3月13号之前发邮件)

考核方式（续）

- 课堂报告说明：

- 1) 报告论文的内容应与多智能体相关
- 2) 论文选择根据个人兴趣，但需由老师审核
- 3) 报告PPT应包括论文的主要内容和创新点
（概述、研究贡献、主要方法、实验验证及结论）
- 4) 课程结束前提交最终版（可根据反馈意见修改）
- 5) 只需要上传报告的PPT文件，不需要写读书报告

主讲教师：李林静



中国科学院自动化研究所研究员，硕士生导师。主要从事博弈论和人工智能的交叉研究，通过对博弈均衡及其特性的研究为博弈和智能的交叉协同及增强提供支撑，包括人工智能增强博弈和博弈增强人工智能两个方向。主持和参与多项中国科学院重点部署项目和先导专项，科技部重点研发计划，国家自然科学基金重大、重点和面上项目。已发表各类期刊和会议论文70余篇，其中两篇文章获得国际会议最佳论文提名奖，出版学术专著1部。研发的全球媒体云系统获得王选新闻科学技术奖一等奖。

主讲教师：张启超



中国科学院自动化研究所副研究员，硕士生导师。主要从事深度强化学习和智能驾驶方向研究，在IEEE汇刊等国际期刊会议上发表论文20余篇。主持和参与多项中国科学院先导专项、国家重点研发计划、国家自然科学基金重点和青年基金、北京市科技计划等项目。先后获得北京市优秀毕业生、中科院院长优秀奖、IEEE计算智能学会研究资助奖等奖项，并获RoboMaster AI挑战赛全球一等奖、国科大杯创新创业大赛新一代信息技术分项赛一等奖等。

内容回顾： 知识表示与推理



关于知识表示

- 什么是知识表示？

知识表示是用计算机能够接受并进行处理的方式来表示人类的知识

- 知识表示的目的

通过知识的有效表示，使人工智能程序能够利用这些知识求解问题



人工智能所涉及的知识

- **事实**

关于问题环境的一些客观事物的知识，通常形式：“…是…”

- **规则**

关于事物之间因果关系的知识，通常形式：“如果…那么…”

- **控制知识**

关于做一件事的技巧性、策略性的知识

- **元知识**

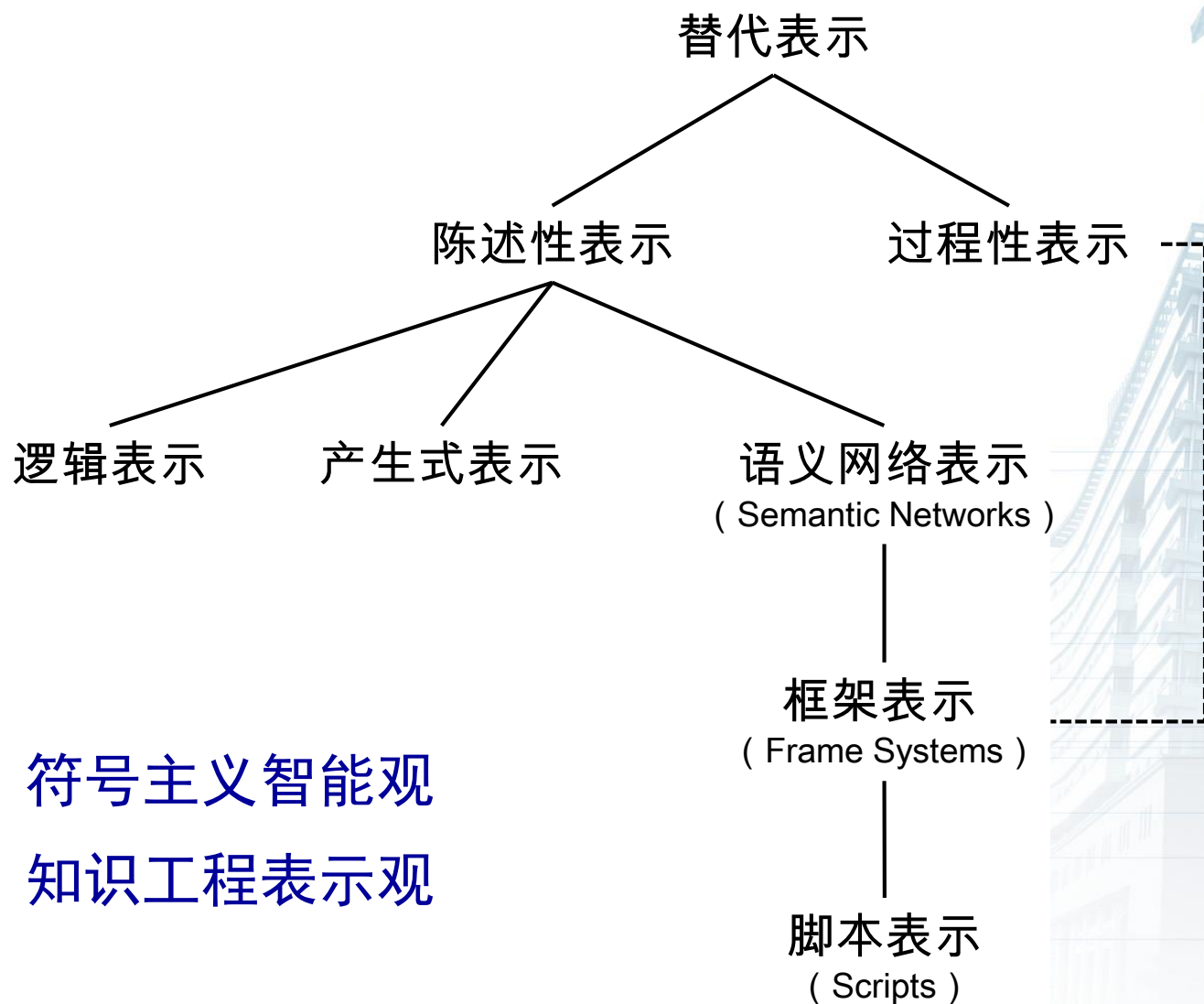
关于知识的知识，有两种：包括说明知识和运用知识的元知识

如何衡量知识表示方法

- 知识表示的范围和程度（广度、精度）
- 是否适于计算机处理
- 是否适合推理
- 是否易于用户使用

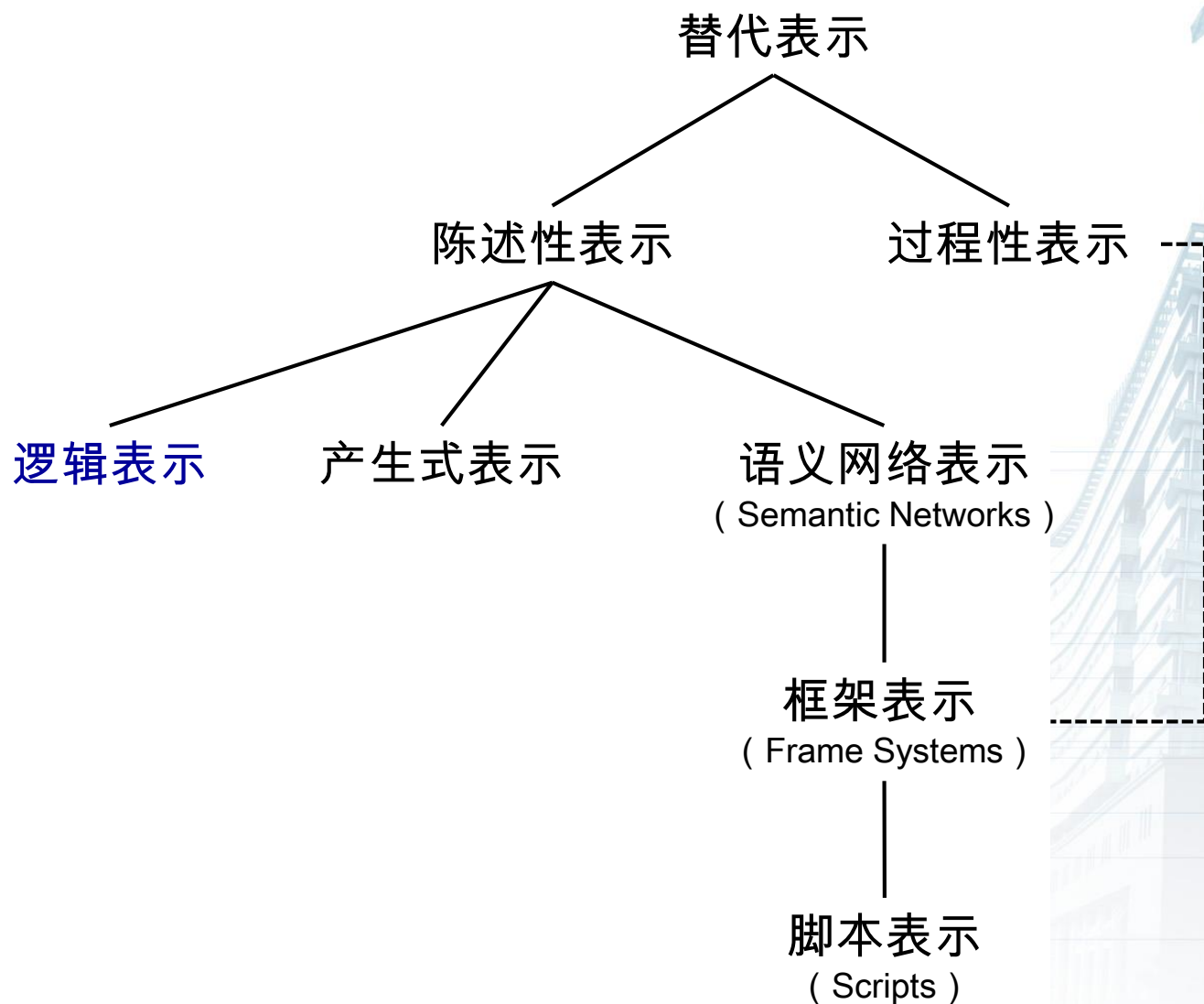
陈述性表示	过程性表示
知识的静态描述	知识的动态描述
表示与推理分开处理	表示与推理相结合
较严格，模块性较好	不够严格，知识间有交互
有时效率较低	求解效率高

知识表示方法体系



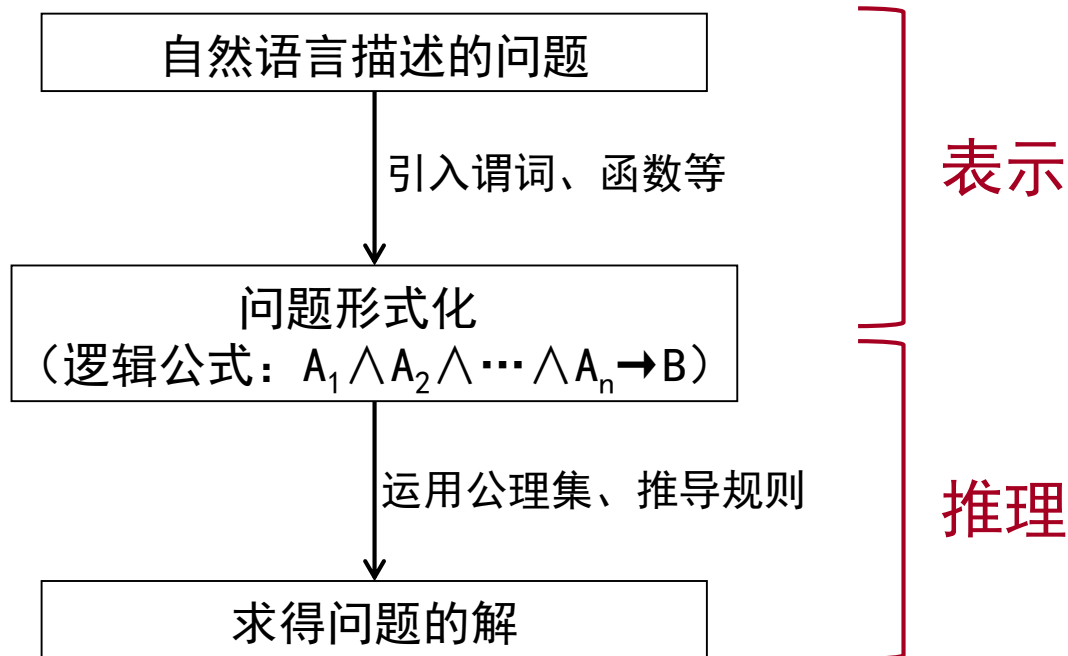
- 符号主义智能观
- 知识工程表示观

知识表示方法体系



逻辑表示法

- 逻辑法求解问题的一般步骤：



- 优点：自然性、严格性、模块性
- 弱点：难以表达结构性、多层次关联知识；推理低效性

命题逻辑

- **命题** (proposition):

有真假意义的一句话

- **逻辑连接符:**

\neg : 否定 (表示 “非”)

\wedge : 合取 (表示 “与”)

\vee : 析取 (表示 “或”)

\rightarrow : 蕴涵 (表示 “隐含”)

\leftrightarrow : 等值 (表示 “等价”)

- **逻辑公式:** $P \vee \neg P$ $P \wedge \neg P$ $(P \vee \neg Q) \rightarrow R$



一阶谓词逻辑

- 谓词 (predicate):

$\text{Man}(x), \text{Friend}(x, y)$

- 函数:

$\text{father}(x), \text{son}(y)$

- 量词:

\forall : 全称量词

$\forall x$: 表示对所有 x

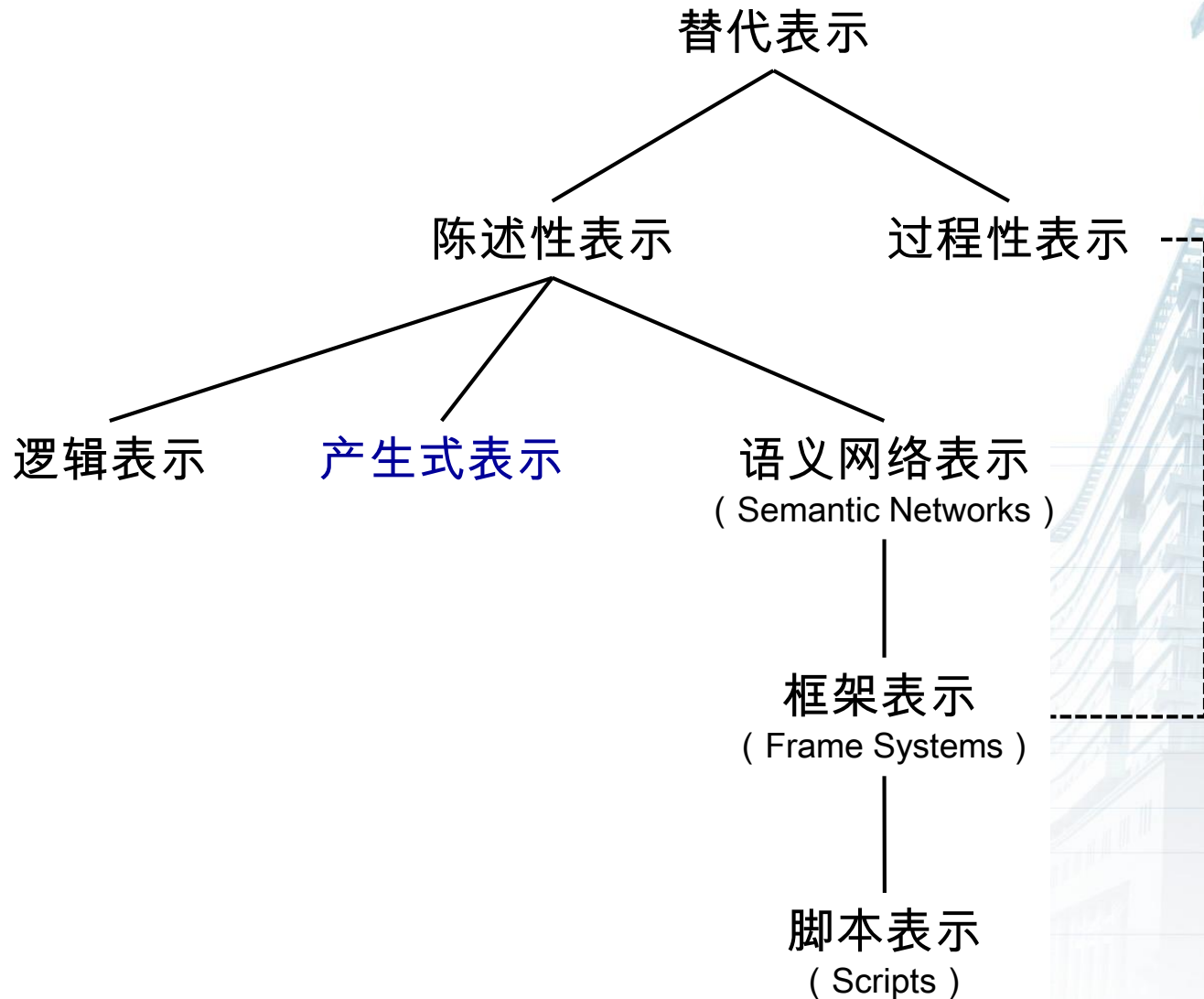
\exists : 存在量词

$\exists x$: 表示存在一个 x

- 逻辑公式: $\forall x (\text{Man}(x) \rightarrow \text{Mortal}(x))$



知识表示方法体系



产生式表示法

- 事实

使用三元组（对象, 属性, 值）或（关系, 对象1, 对象2）表示

↑
单个语言变量

↖ ↗
多个语言变量

- 规则

采用 “if <前件> then <后件>” 的形式

↑
条件部分, 事实 A_i 的合取

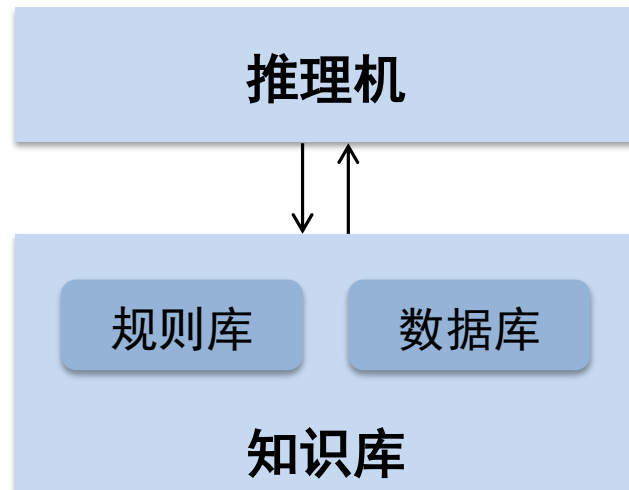
↑
结论部分, 某一事实B

产生式系统的结构

- **知识库**

规则库：以产生式规则形式存放领域知识的存储区

数据库：存放初始事实、中间结果及最后结果的工作区



- **推理机：** 包含推理方式和控制策略，工作过程：

- (1) 数据库中的数据与规则库中的规则相匹配，得匹配规则集
 - (2) 从匹配规则集中选择一条规则作为使用规则
 - (3) 根据规则，执行相应的操作
- 重复这个过程，直到满足结束条件。

产生式系统示例

- 一个识别食物的简单产生式系统

规则库中有5条规则：

R1: if 绿色 then 农产品

R2: if 装在小容器里 then 精美食品

R3: if 冷冻食品 or 农产品 then 易腐烂食品

R4: if 非精美食品 and 不易腐烂食品 then 大众食品

R5: if 重10斤 and 农产品 then 西瓜

推理方式：正向推理

控制策略：① 不产生冗余数据，② 按规则排序

结束条件：无可规则

数据库：（绿色，重10斤）

产生式系统示例

第1次循环：初始事实（绿色，重10斤）

1. 匹配规则集 {R1}
2. 选择 R1 作为使用规则
3. 数据库：（绿色，重10斤，农产品）

第2次循环：1. 匹配规则集 {R1, R3, R5}

2. 选择 R3 作为使用规则
3. 数据库：（绿色，重10斤，农产品，易腐烂食品）

第3次循环：1. 匹配规则集 {R1, R3, R5}

2. 选择 R5 作为使用规则
3. 数据库：（绿色，重10斤，农产品，易腐烂食品，西瓜）

第4次循环：无可规则，结束。

产生式系统的推理

- 正向推理（数据驱动、自底向上方式）

(1) 规则前件与已知数据匹配

... ..

(3) 使用规则的后件 → 数据库

... .. 结束条件：数据库中包含目标

S（包含目标）

⋮

↑
R₃

S₂

↑
R₂

S₁

↑
R₁

S₀（初始事实）

产生式系统的推理

- 反向推理（目标驱动、自顶向下方式）

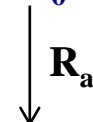
（1）规则后件与假设目标匹配

... ..

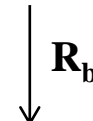
（3）使用规则的前件 → 子目标

... .. 结束条件：各子目标均为已知事实

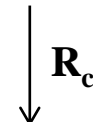
T_0 （初始目标）



T_1



T_2



\vdots

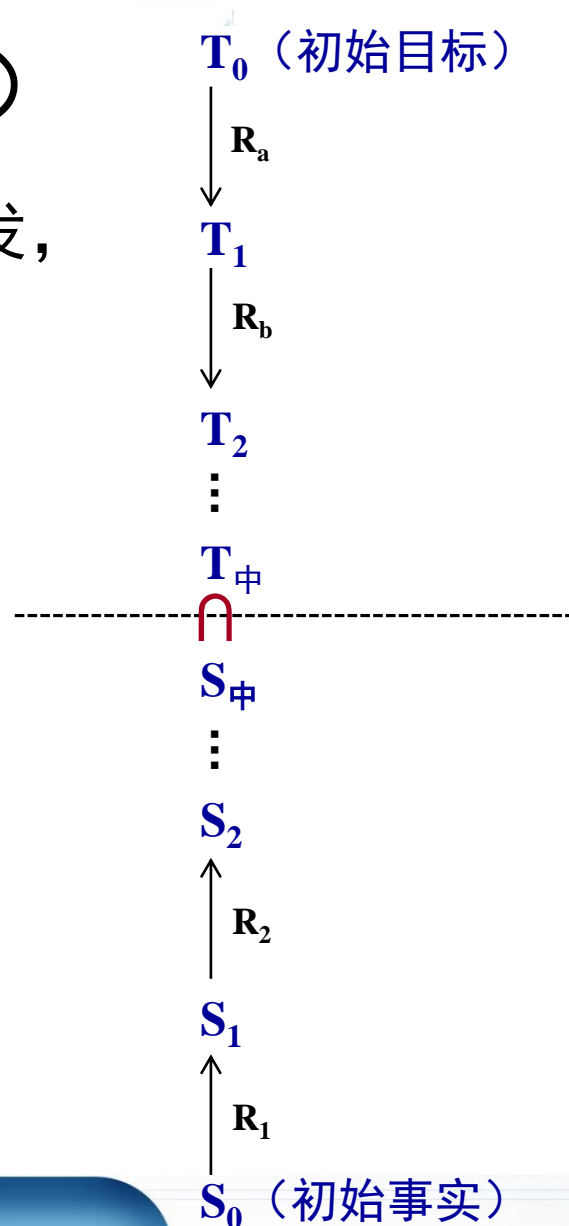
T （已知事实）

产生式系统的推理

- 双向推理（既自底向上又自顶向下）

既从已知事实出发，又从假设的目标出发，
达到某个中间界面

结束条件：各子目标都在当前数据库中



产生式系统示例

- 一个识别食物的简单产生式系统

规则库中有5条规则：

R1: if 绿色 then 农产品

R2: if 装在小容器里 then 精美食品

R3: if 冷冻食品 or 农产品 then 易腐烂食品

R4: if 非精美食品 and 不易腐烂食品 then 大众食品

R5: if 重10斤 and 农产品 then 西瓜

R6: if 重10斤 and 易腐烂食品 then 鸡肉

控制策略：① 不产生冗余数据，② 按规则排序

结束条件：无可用规则

数据库：（绿色，重10斤）

产生式系统示例

第1次循环：初始事实（绿色，重10斤）

1. 匹配规则集 {R1}
2. 选择 R1 作为使用规则
3. 数据库：（绿色，重10斤，农产品）

第2次循环：

1. 匹配规则集 {R1, R3, R5}
2. 选择 R3 作为使用规则
3. 数据库：（绿色，重10斤，农产品，易腐烂食品）

第3次循环：

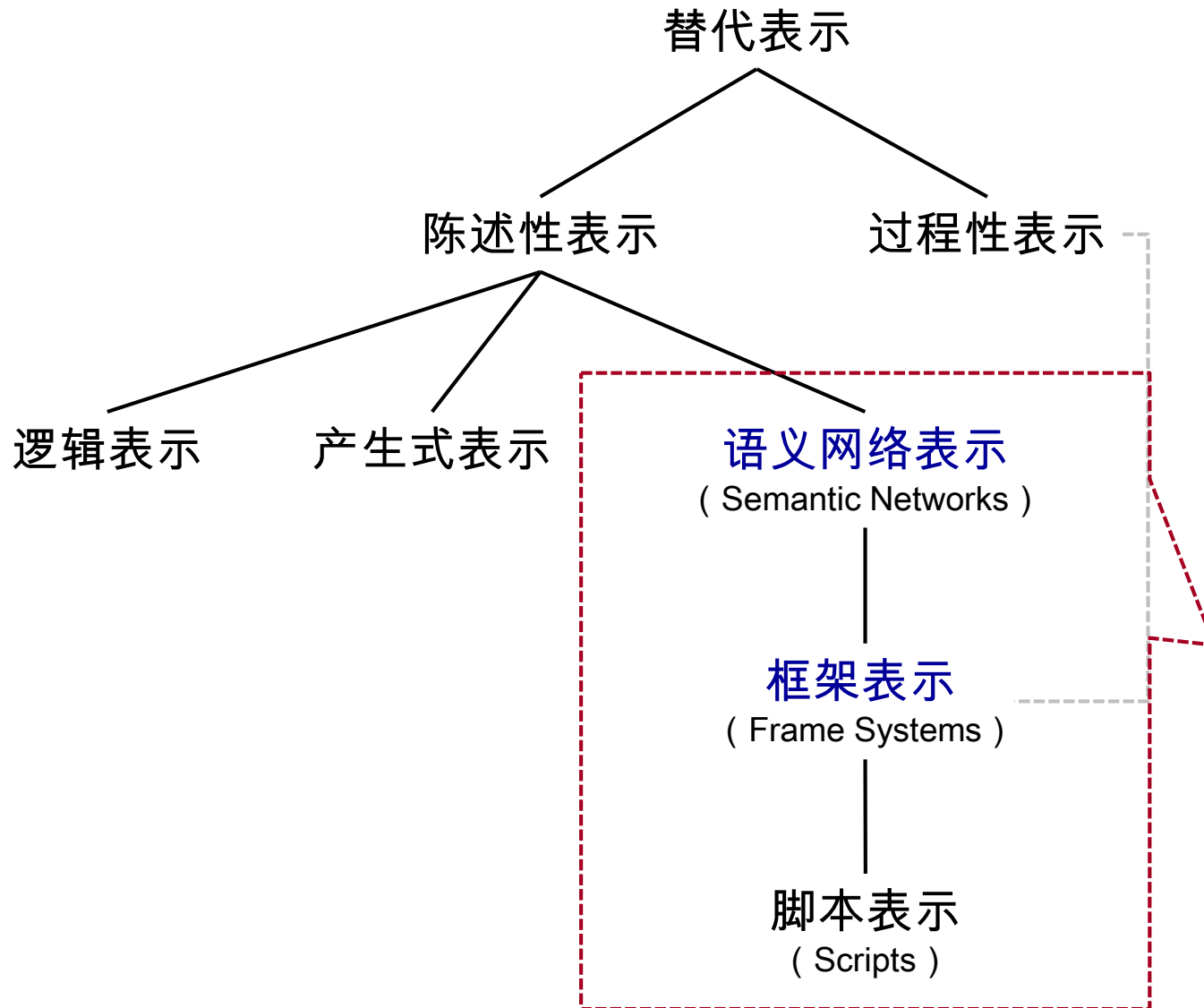
1. 匹配规则集 {R1, R3, R5, R6}
2. 选择 R5 作为使用规则
3. 数据库：（绿色，重10斤，农产品，易腐烂食品，西瓜）

第4次循环：使用规则 R6 → 鸡肉！

产生式表示法的特点

- 产生式表示格式固定，形式单一，规则间相互较为独立，使知识库的建立较为容易；同时知识库与推理机分离，易于对知识库进行增删、修改；推理方式单纯，对推理也容易作出解释；因此常作为建造知识系统的首选知识表示方法，尤其在一些较简单的知识系统中使用得最多
- 优点：自然性、模块性、有效性
- 弱点：
 - 推理过程的低效率，容易引起组合爆炸
 - 不能表达结构性的知识以及知识间的联系

结构化知识表示方法



结构化知识表示

语义网络

- 结构

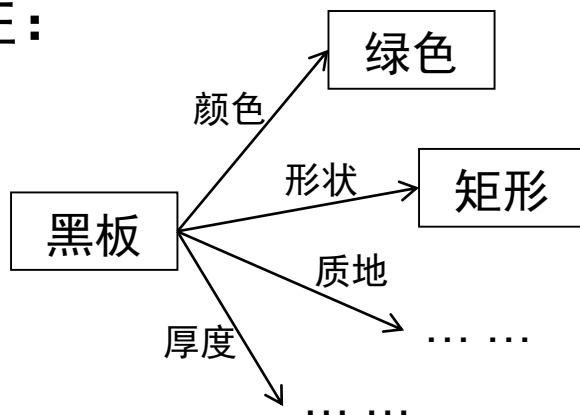
采用有向图表示知识：

结点：事物、概念、属性值、事件、情形等

弧：结点1的属性或结点之间的关系；方向表示主次

- 表示

(1) 结点的属性：

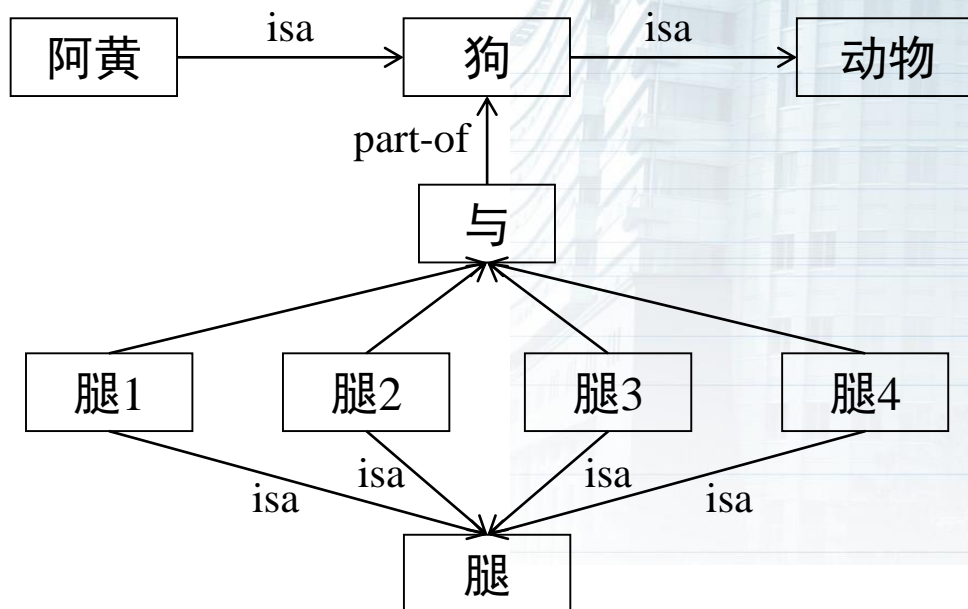
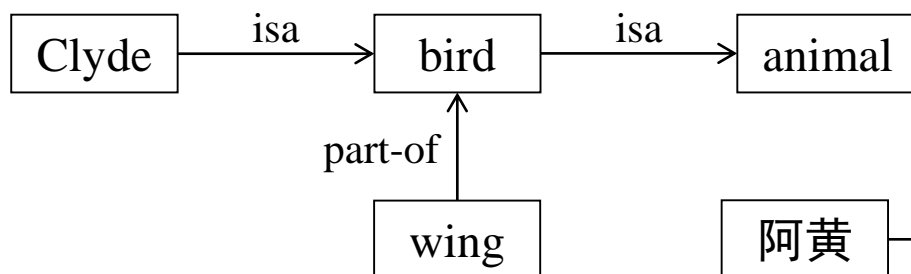


语义网络表示

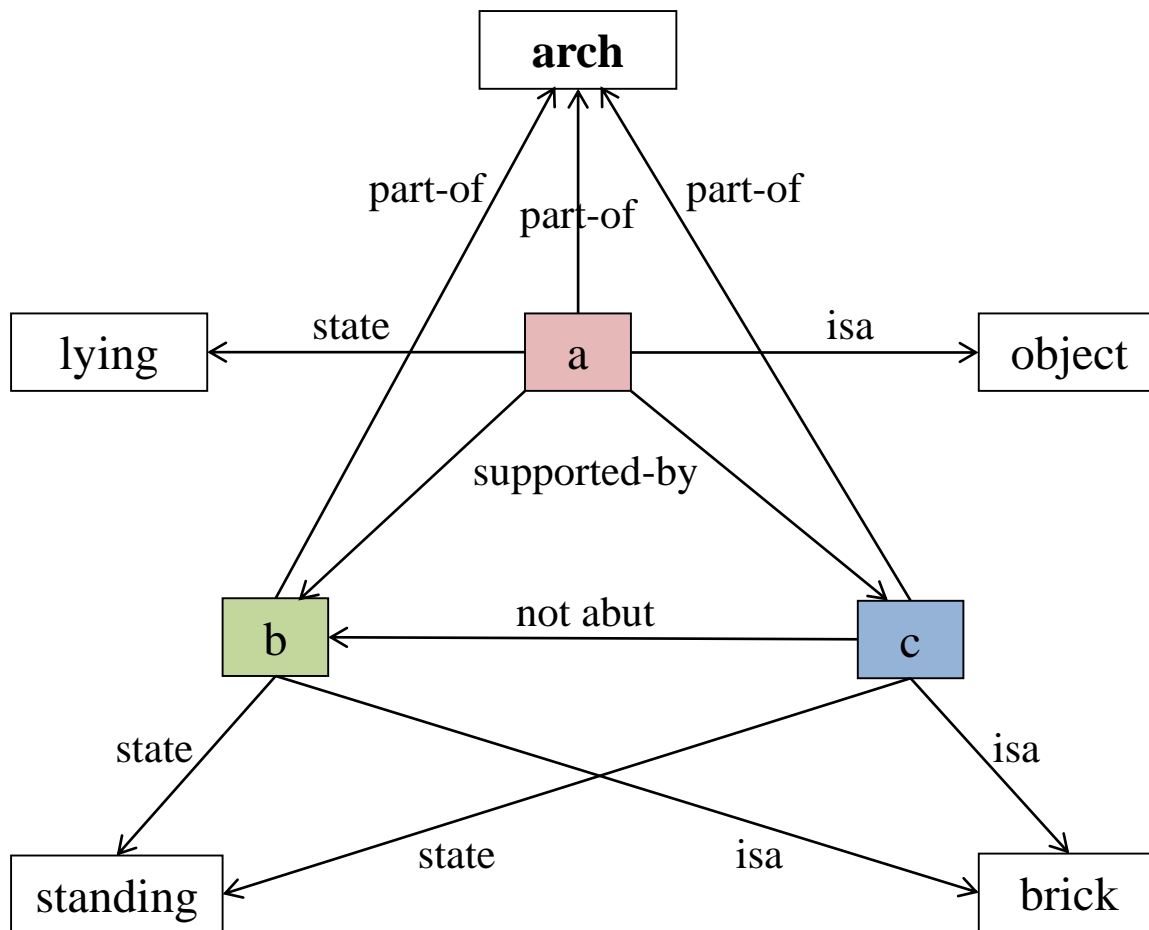
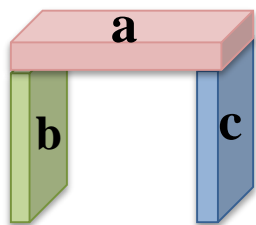
(2) 结点间典型关系:

isa: 个体对类、子类对类的隶属关系

part-of: 全体对部分的包含关系



Winston 学习系统中对“拱”的概念进行学习时的语义网络：



语义网络表示

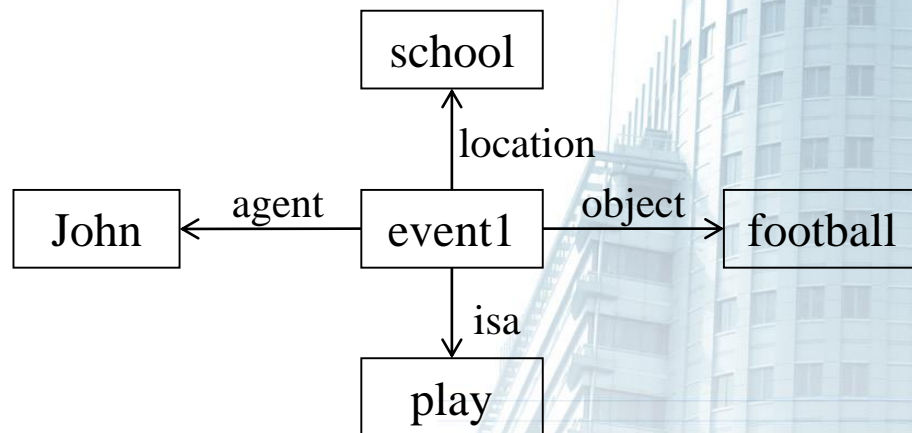
(3) event 结点：表示一个动作或事件

agent: 动作的施主

object: 动作的对象

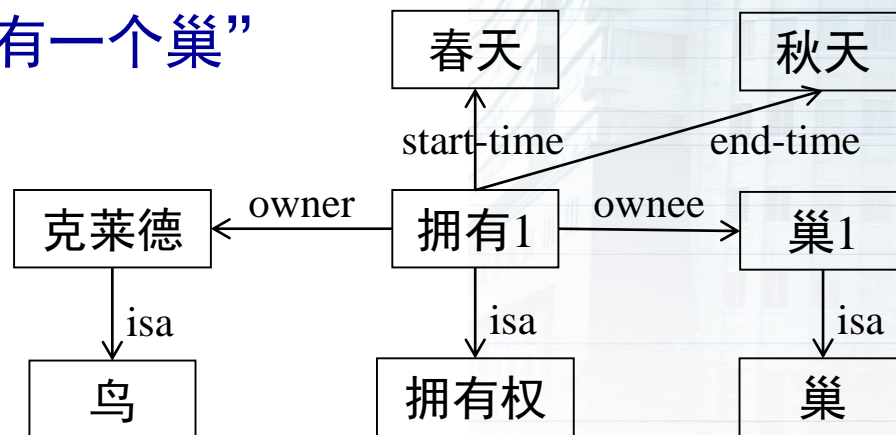
location: 动作的位置

time: 动作的时间等

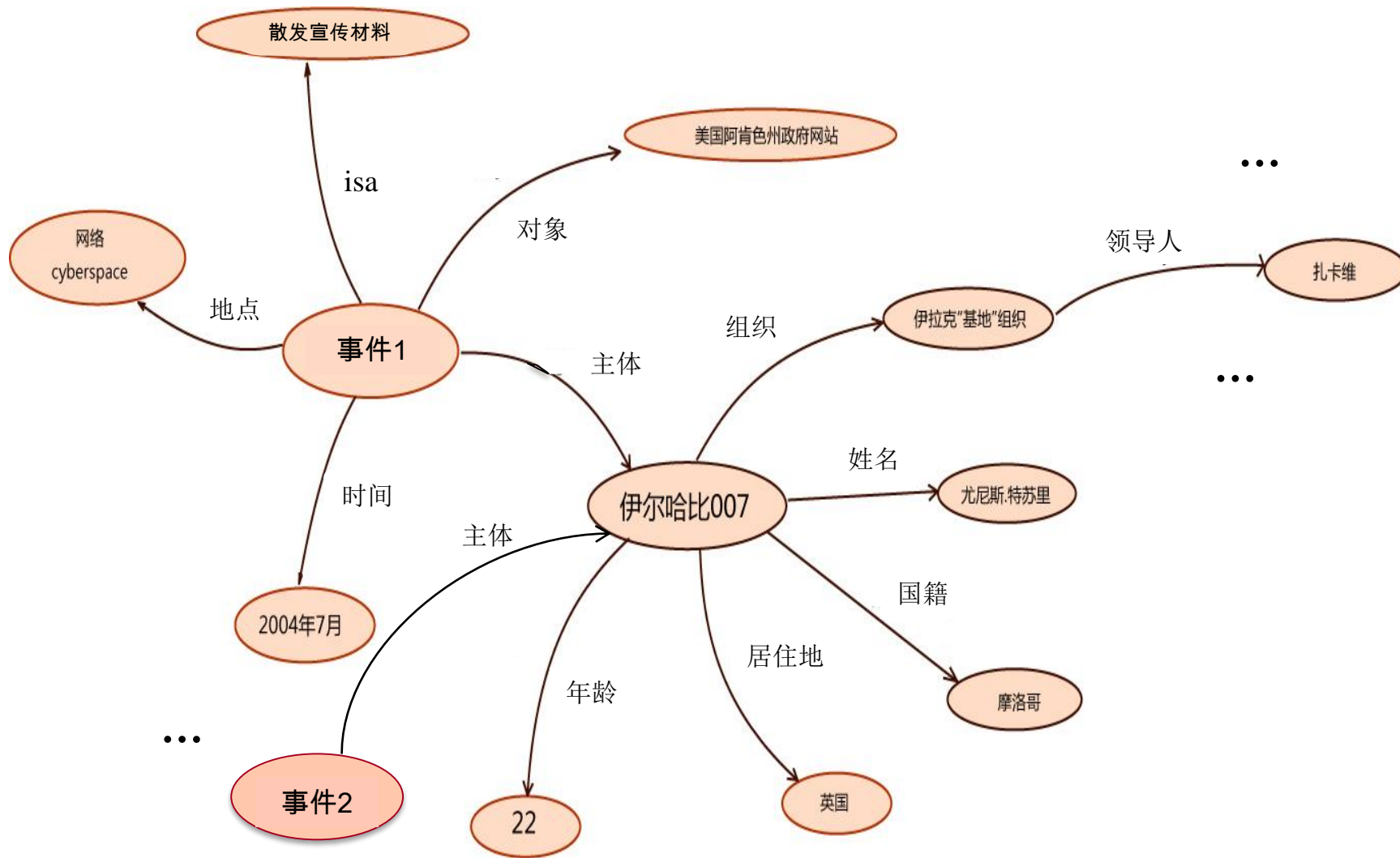


(4) situation 结点：表示一个情形或状况

“克莱德从春天到秋天拥有一个巢”



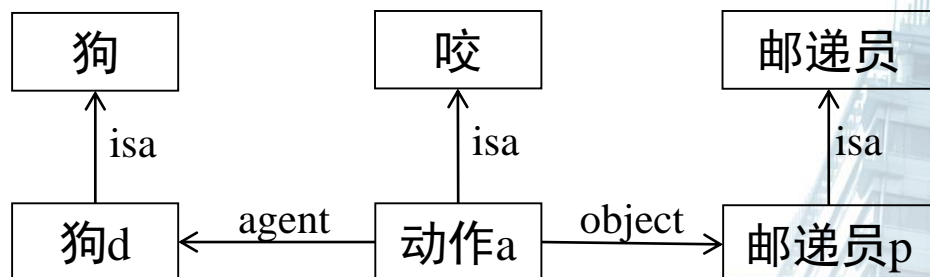
语义网络应用示例



语义网络表示

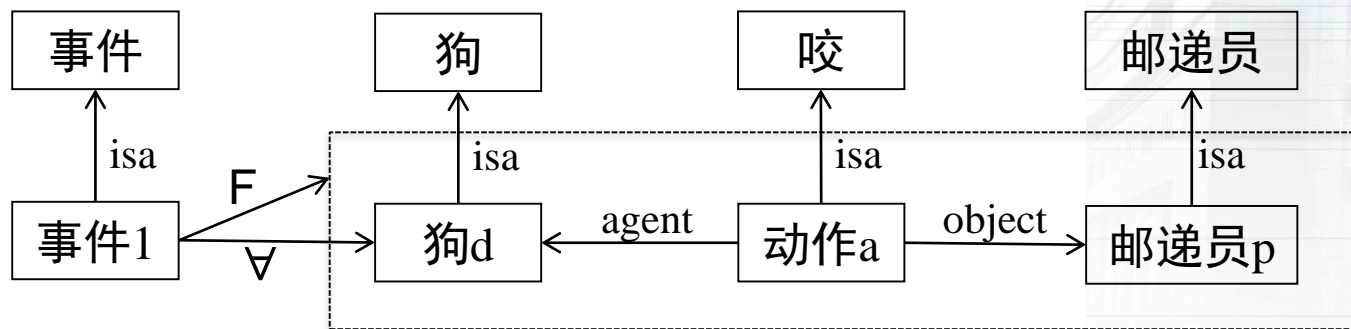
(5) 逻辑联结词和量词的表示

“狗咬了邮递员”

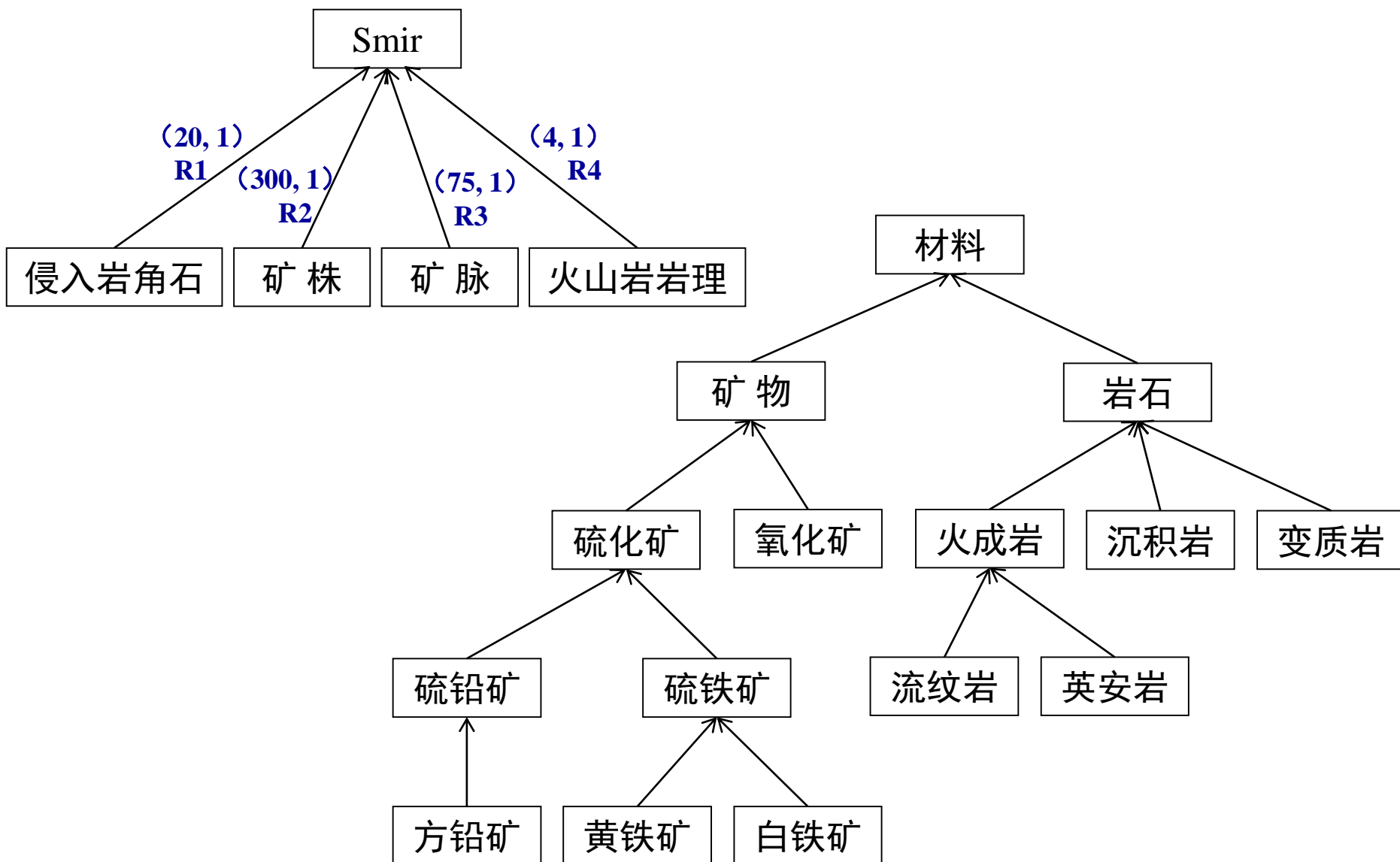


网络分块技术 (Hendrix) :

“所有的狗都咬过邮递员”



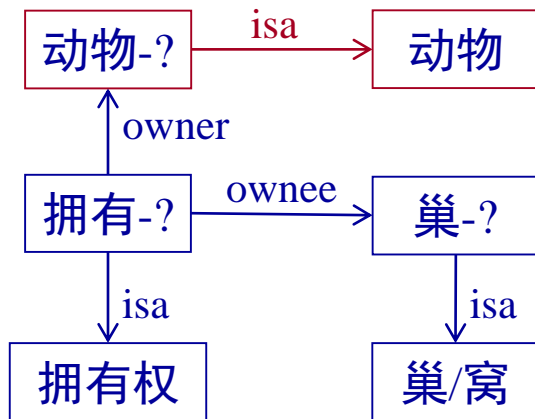
Prospector 系统中，用语义网络表示规则和分类知识：



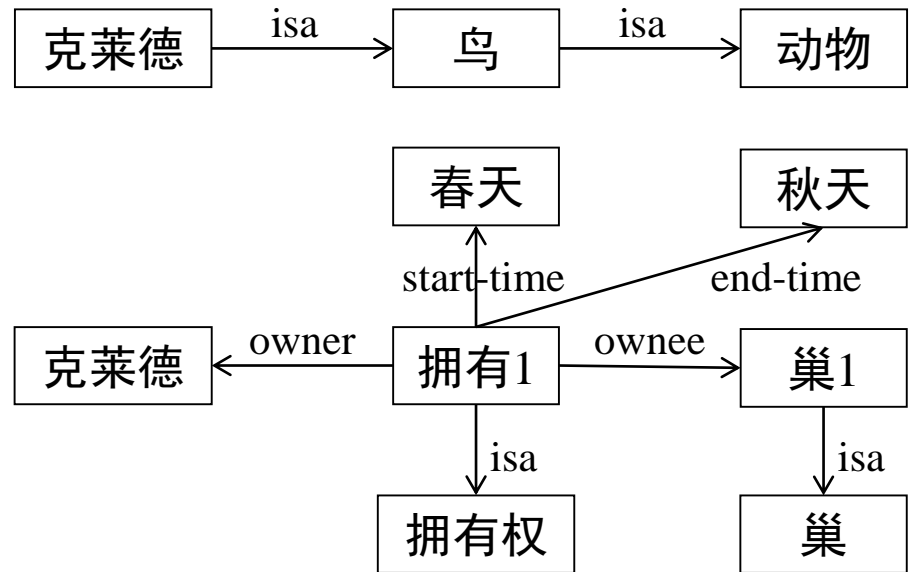
语义网络的推理

- **匹配：** 语义网络知识库与待求解问题的语义网络模式

问题：“是否有一只动物，它拥有一个窝？”



语义网络知识库：



- **继承**

隶属关系下的属性继承、直接继承等

语义网络表示法的特点

- 优点

- 自然、直观，结构化
- 联想性
- 推理效率较高

- 主要弱点

- 缺乏严格的理论基础

缺乏严格的形式定义；对于网络结构，缺乏明确的语义解释

- 继承推理的非有效性

复杂情形下继承推理的冲突；非单调性、常识推理研究



框架表示法

- **框架结构：** 由框架名和一些槽组成，每个槽有单值或多值

〈框架名〉

〈槽名₁〉 〈侧面₁₁〉 (值₁₁₁, 值₁₁₂, ...)

〈侧面₁₂〉 (值₁₂₁, 值₁₂₂, ...)

...

〈槽名₂〉 〈侧面₂₁〉 (值₂₁₁, 值₂₁₂, ...)

〈侧面₂₂〉 (值₂₂₁, 值₂₂₂, ...)

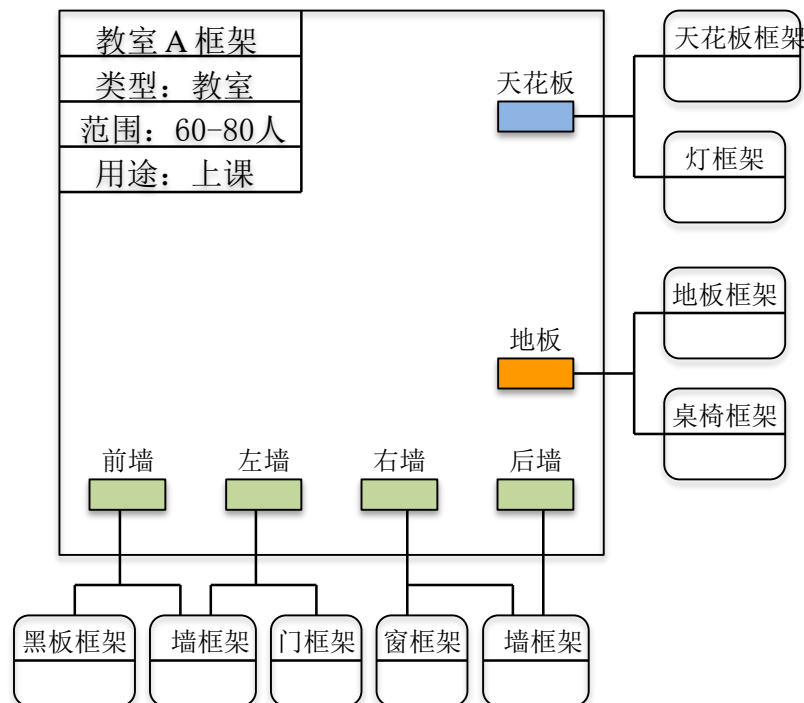
...

if needed (过程₁)

...

if added (过程₂)

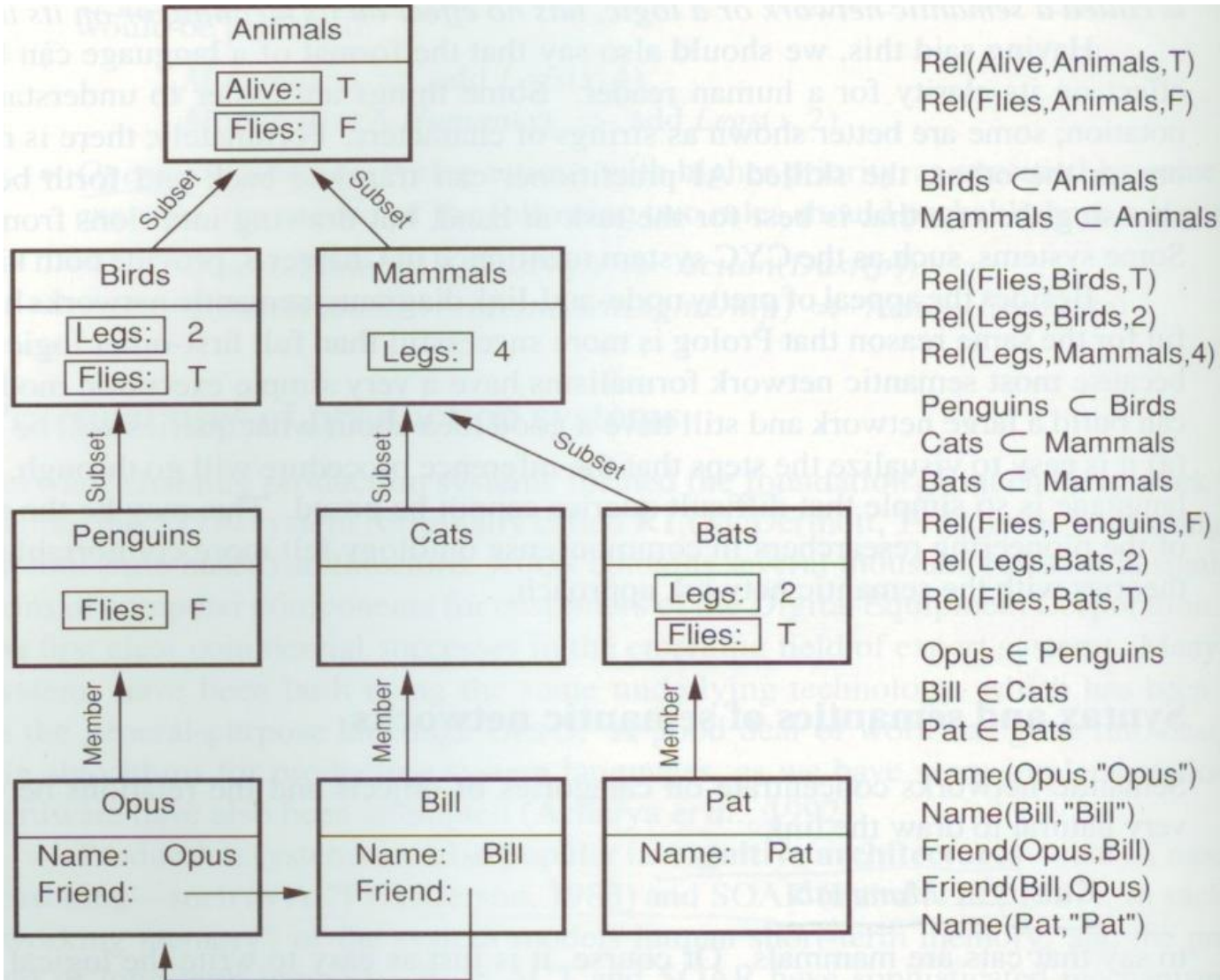
...



槽/侧面取值：逻辑值、数值、字符、默认值、子框架

if needed、if added 等槽的槽值是附加过程

框架系统示例



框架表示下的推理

- 匹配和继承

填槽方式：

- (1) 继承：下层框架继承上层框架的属性
- (2) 默认：如果没有相反的证据，槽值取默认值
- (3) 附加过程：
 - if needed 槽：用于说明如何计算得到槽值
 - if added 槽：用于说明填槽值时需要完成哪些操作

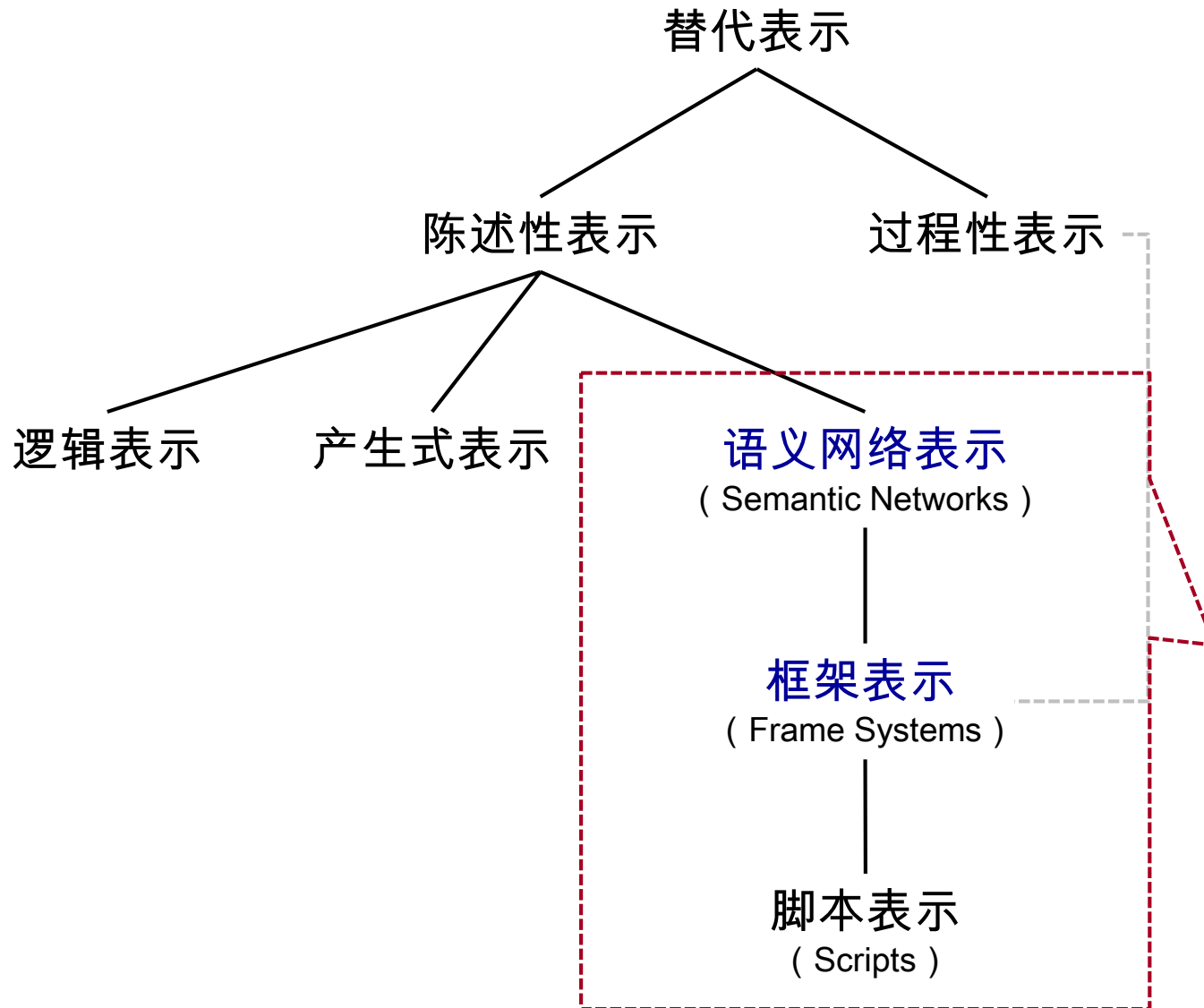
从框架库中寻找适合当前问题的最佳候选框架。

框架表示法的特点

- 优点：结构化知识、推理效率较高、陈述性与过程性表示结合
- 弱点：
 - 严格性不足、继承推理的非有效性
 - 许多实际情况与原型不符，对新的情况不易适应

	产生式系统	框架系统
知识表示单位	相互独立的规则	框架结构
推理机制	固定、与知识库分离	可变、与知识库成一体
知识库构建	较容易	困难
系统通用性	较低	高
应用问题领域	简单问题	复杂问题
用户程度	初学者	专家

结构化知识表示方法



结构化知识表示

知识表示观

	认识论表示观	本体论表示观	知识工程表示观
核心观点	表示是对自然世界的描述	表示是对自然世界的近似描述	表示是对自然世界描述的计算机模型
	表示的唯一作用就是携带知识	承认表示的携带知识作用, 但认为不唯一, 强调一般知识	承认表示的携带知识作用, 但认为不唯一, 强调领域知识
	表示研究与启发式研究无关	启发式研究是表示研究的一部分	启发式研究至关重要
主要任务	对智能现象抽象与简洁的刻画, 常识的形式化和非单调推理	强调“聚焦”和本体论约定, 不依赖具体领域的一般性知识库	研究知识的存储与对其有效地使用(推理与搜索)
面临困难	推理过于复杂, 问题求解的非有效性	本体约定的相对性: 站在不同角度将导致不同的本体论约定	缺乏常识和一般知识, 脆弱性

- 知识表示：客观事物到推理内部的一种映射
- 应与问题领域的知识特性相吻合，以最大程度地减少失真

R. Davis, et al. What Is a Knowledge Representation? *AI Magazine*, 14(1): 17-33, 1993.

关于推理

- **推理：** 依据一定的原则从前提推出结论的过程，从逻辑上分为
 - **演绎推理：** 由一组前提必然地推导出某个结论的过程
核心：三段论；保真性；没有增加新知识
 - **归纳推理：** 以某个命题为前提，推出与其有归纳关系的其它命题的过程
简单枚举法、类比法等；增加了新知识；不保真
 - **溯因推理：** 解释观察到的一个结果的推理过程（用于产生解释）
需要背景理论来构造并检查生成的解释；不保真

推理形式示例

演绎： { 规则—这个袋中的豆子都是白色的
事例—这个豆子来自这个袋子
⇒ 结果—这个豆子是白色的

归纳： { 事例—这些豆子来自这个袋子
结果—这些豆子是白色的
⇒ 规则—这个袋中的豆子都是白色的

溯因： { 规则—这个袋中的豆子都是白色的
结果—这个豆子是白色的
⇒ 事例—这个豆子来自这个袋子



经典与非经典逻辑推理

- 经典逻辑推理

标准逻辑下的演绎推理是人工智能最早采用的经典推理方法，
归结推理方法是其最重要的形式化方法

- 非经典逻辑与非经典推理

精确→不精确： $\left\{ \begin{array}{l} \text{多值化：多值逻辑} \\ \text{模糊化：模糊逻辑} \end{array} \right.$

静态→动态： $\left\{ \begin{array}{l} \text{可能性：模态逻辑} \\ \text{时态性：时态逻辑} \end{array} \right.$

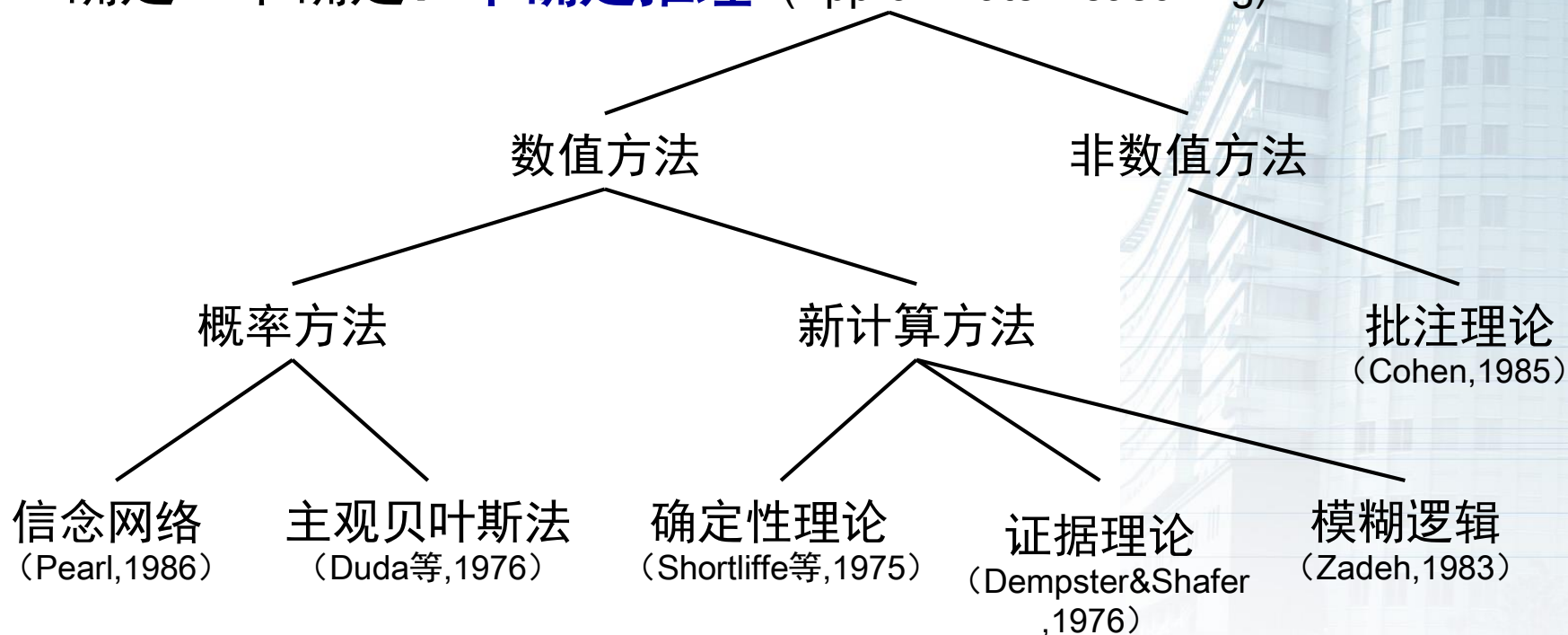
单调→非单调：非单调逻辑、非单调推理 (Nonmonotonic Reasoning)
知识不一致、不完全情形下的推理

基于知识的推理

- 基于知识的推理

基于知识的推理→演绎推理：确定、严格、一致

确定→不确定：**不确定推理** (Approximate Reasoning)



- 基于模型的推理 (**定性推理**, Qualitative Reasoning)

比较：不确定推理方法

	确定性理论	主观Bayes方法	证据理论	可能性理论
理论基础	较弱	较强	较强	中等
适于处理的不确定类型	概率	概率	概率、模糊	模糊
不确定性的给定方法	主观	主观、客观	主观	主观
能否区分不确定/不知道	难以	难以	可以	可以
易于使用	容易	容易	困难	一般

1. Kanal & Lemmer (Eds.). *Uncertainty in Artificial Intelligence*. Elsevier, 1986.
2. Genesereth & Nilsson. *Logical Foundations of Artificial Intelligence*. Elsevier, 1987.

End.

