



# Methodologies of Knowledge Representation 知识表示方法 II



Central South University  
Department of Information  
Science and Engineering

- <http://netclass.csu.edu.cn/jpkc2003/rengongzhineng/Index.htm>





# 知识表示方法

- 1. 状态空间表示(State Space Representation)
- 2. 问题归约表示(Problem Reduction Representation) 2.3
- 谓词逻辑表示(Predicate Logic)
- 2.4 语义网络 (Semantic Network Representation) 2.5
- 框架, 剧本, 过程(Frame, Script, Procedure) 2.6 知
- 识图谱(Knowledge Graph)





# Knowledge Representation and Logic

逻辑推理是人类另一常用的问题求解技术

- 如何进行逻辑推理？
- 推理的过程怎样？
- 怎么实现自动推理？

所有这些问题的基础首先必须是怎样用适合逻辑推理的方式表示问题。

逻辑推理是指遵循逻辑规律来分析推理的思路。  
把不同排列顺序的意识进行相关性的推导就是逻辑推理。





# 马普尔小姐的推理过程

## 观察结果

- 马克是右撇子，手表戴在左手
  - 约尔是左撇子，手表戴在右手
- } **事实**

## 推理规则

- 如果手表戴在左手，那么他是马克
  - 如果手表戴在右手，那么他是约尔
- } **规则**

## 结论

- 只是穿着不同衣服的同一个人——约尔

**人类的推理可以理解语义**

**机器如何进行这样类似的推理？**

**需要将推理的过程与理解分割开，将其形式化**





# 推理的一般形式

已知：事实1，事实2， ...

如果 事实1 那么 结论1

如果 事实2 那么 结论2

....

得到：结论1，结论2， ...

- 将事实与规则等抽象出来，不涉及具体内容，借助一些符号来表示，推理过程可以被**形式化**

P：某已知事实  $P \rightarrow Q$

：如果 P 那么 Q

结论： Q

这个过程不需要直觉和解释





# 符号与形式语言



自然语言不适合计算机处理

■ 例：用红墨水写一个“蓝”字，请问，这个字是红字还是蓝字？



需要一种无歧义，方便存储和表达的**形式化符号**表征体系



**数理逻辑**

● 命题逻辑：逻辑运算符结合原子命题构建命题

● **谓词逻辑**：量词+谓词

**数理逻辑**又称符号逻辑、理论逻辑，用数学方法研究逻辑或形式逻辑。





# 谓词逻辑

个体词

谓词

比如：苹果可以吃

## 什么是谓词？

- 原子命题中刻画个体的性质或个体间关系的成分

## 谓词逻辑是一种形式语言

- 是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
- 接近自然语言，又方便存入计算机处理
- 最早应用于人工智能中表示知识
- 适合于表示事物的状态、属性、概念等，也可用来表示事物间确定的因果关系



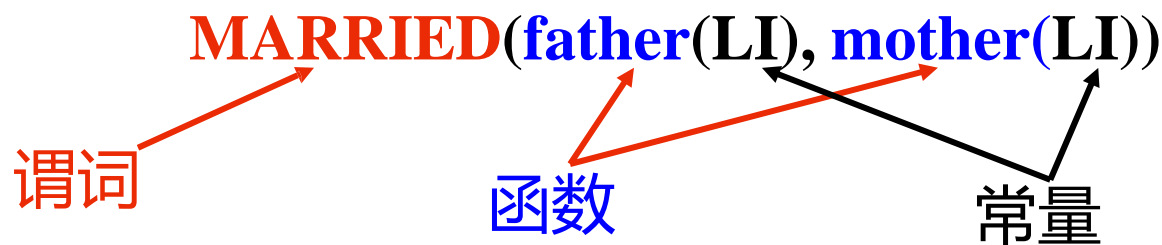


## 2.3 Predicate Logic 谓词逻辑法

### 1. 谓词演算 (Predicate Calculus)

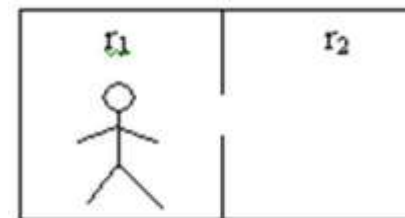
#### 语法和语义 (Syntax and Semantics)

- 基本符号：谓词符号、变量符号、函数符号、常量符号、括号和逗号隔开，以表示论域内的关系。



#### 原子公式 (Atomic Formulas)

- 原子公式 = Predicate + Terms







## 2.连词和量词 (Connective & Quantifiers)

❖ 连词( $\wedge$ ,  $\vee$ ,  $\Rightarrow$ ,  $\sim$ )

Conjunction (与及合取) : 用连词 $\wedge$  (表示复合句子) 把几个公式连接起来而构成的公式

例:  $\text{LIKE}(\text{I}, \text{MUSIC}) \wedge \text{LIKE}(\text{I}, \text{PAINTING})$

(我喜爱音乐和绘画。)

$\text{LIVES}(\text{L1}, \text{HOUSE-1}) \wedge \text{COLOR}(\text{HOUSE-1}, \text{YELLOW})$

(李住在一幢黄色的房子。)



**Disjunction (或及析取)**：用连词 $\vee$ （表示可兼有的“或”）把几个公式连接起来而构成的公式

例：

$\text{PLAYS}(\text{LILI}, \text{BASKETBALL}) \vee \text{PLAYS}(\text{LILI}, \text{FOOTBALL})$   
(李力打篮球或踢足球。)





Implication (蕴涵) : 用连词 $\Rightarrow$  (表示“如果-那么”) 把几个公式连接起来而构成的公式

例:  $\text{RUNS}(\text{LIUHUA}, \text{FASTEST}) \Rightarrow \text{WINS}(\text{LIUHUA}, \text{CHAMPION})$

如果刘华跑得快, 那么他取得冠军.

Not (非) : 符号  $\sim$  或者  $\neg$  用来否定一个公式的真值

例:  $\sim \text{INROOM}(\text{ROBOT}, r2)$

机器人不在2号房间内





Ex1: 李明是个学生, 他住的房间是白色的。

**$\text{Isa}(\text{Liming}, \text{Student}) \wedge \text{Lives}(\text{Liming}, \text{House1}) \wedge \text{Color}(\text{House1}, \text{White})$**

Ex2: 李明在学校的话, Wang也在学校。

**$\text{At}(\text{Liming}, \text{School}) \Rightarrow \text{At}(\text{Wang}, \text{School})$**





## 量词 Quantifiers

**全称量词 (Universal Quantifier)** 若一个原子公式  $P(x)$ , 对于所有可能变量  $x$  都具有T值, 则:

$$(\forall x) P(x)$$

↑ 全称量词      x 约束变量      P(x) 辖域

## 存在量词 (Existential Quantifier)

若一个原子公式  $P(x)$ , 至少有一个变元  $x$ , 可使  $P(x)$  为T值, 则:  $(\exists x) P(x)$

例:  $(\exists x) \text{INROOM}(x, r1)$  (1号房间内有个物体)

$$(\exists x) \text{INROOM}(x, r1)$$

↑ 存在量词      x 约束变量      INROOM(x, r1) 辖域





Ex3: All roads lead to Roma

$$(\forall x) [ \text{Road}(x) \Rightarrow \text{Lead}(x, \text{Roma}) ]$$

Ex4: For all variables  $x$ , if  $x$  is an integer, then  $x$  is positive or negative

$$(\forall x) [ I(x) \Rightarrow ( P(x) \vee N(x) ) ]$$

Ex4: Mary gives each person one book.

$$(\forall x) (\exists y) [ \text{Person}(x) \wedge \text{Book}(y) \wedge \text{Give}(\text{Mary}, x, y) ]$$





## ➤ 如何阅读量词公式

$(\forall x)$  : 所有的 $x$ ; 任意的 $x$ ; 一切的 $x$ ; 每一个 $x$ 等

$(\exists x)$  : 有些 $x$ ; 至少有一个 $x$ ; 存在 $x$ 等

$F(x, y)$  : “ $x$  比 $y$  飞得更快”, 所属集合无穷大

$(\forall x)(\forall y) F(x, y)$  对任一 $x$ 和任一 $y$ ,  $F(x, y)$ 为真或者 $F(x, y)$ 成立。

$(\forall x)(\exists y) F(x, y)$  对任一 $x$ , 存在 $y$ , 使得 $F(x, y)$ 为真或者 $F(x, y)$ 成立。

$(\exists x)(\forall y) F(x, y)$  存在 $x$ , 对任一 $y$ , 使得 $F(x, y)$ 为真或者 $F(x, y)$ 成立。

$(\exists x)(\exists y) F(x, y)$  存在 $x$ , 存在 $y$ , 使得 $F(x, y)$ 为真或者 $F(x, y)$ 成立。





## 量词的应用顺序

(1) 括号内部最接近原子公式的量词先被应用

$$(\exists y)(\forall x) P(x, y)$$

$$(\exists y) [(\forall x) P(x, y)]$$

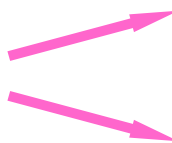


(2) 同一类型量词的位置可以互换

$$(\exists y)(\exists x)(\exists z) P(x, y, z)$$

$$(\exists x)(\exists y)(\exists z) P(x, y, z)$$

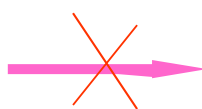
$$(\exists z)(\exists y)(\exists x) P(x, y, z)$$



(3) 不同类型量词的位置不能互换

$$(\exists y)(\forall x) P(x, y)$$

$$(\exists x) [(\forall y) P(x, y)]$$







## 变量 (Variable)

$$(\forall y) P(y) \Rightarrow (\exists x) Q(x,y)$$

$y$  是  $(\exists x)$  定义域内的自由变量, 而  $y$  是  $(\forall y)$  定义域内的约束变量.



### 3. 谓词公式 ( Predicate Formula )

#### ❖ 原子公式 (Atomic Formula)

- 原子公式或原子只是一个谓词应用于一个元组的术语; 用  $P(x_1, x_2, \dots, x_n)$  表示谓词公式, 其中  $P$  为谓词,  $x_1, x_2, \dots, x_n$  客体变量或变元.

#### ❖ 分子谓词公式 (Predicate calculus formula)

- 用逻辑**连词**把原子谓词公式组成复合谓词公式.





## 合式公式 ( Well-Formed Formulas , WFF) 在

谓词演算中**合式公式**的递归定义如下:

- (1) 原子谓词公式是**合式公式**.
- (2) 若  $A$  为**合式公式**, 则  $\sim A$  也是一个合式公式.
- (3) 若  $A$  和  $B$  都是**合式公式**, 那么  $(A \wedge B)$ ,  $(A \vee B)$ ,  $(A \Rightarrow B)$  和  $(A \Leftrightarrow B)$  也都是合式公式.
- (4) 若  $A$  是**合式公式**,  $x$  为  $A$  中的自由变元, 则  $(\forall x)A$  和  $(\exists x)A$  都是合式公式.
- (5) 只有按上述规则 (1) 至 (4) 求得的那些公式, 才是合式公式.



## 合式公式的性质

### 真值表

真值表 (两个合式公式P和Q组成的复合表达式)

P	Q	$P \vee Q$	$P \wedge Q$	$P \Rightarrow Q$	$\sim P$
T	T	T	T	T	F
F	T	T	F	T	T
F	F	F	F	T	T
T	F	T	F	F	F

### 等价关系

如果两个合式公式, 无论如何解释, 其真值表都是相同的, 那么就称此二者是等价的.





(1) 否定之否定:  $\sim(\sim P) = P$

(2)  $P \vee Q = \sim P \Rightarrow Q$

(3) 狄摩根定理 (反演律):  $\sim(P \vee Q) = \sim P \wedge \sim Q$

$\sim(P \wedge Q) = \sim P \vee \sim Q$

(4) 分配律:  $P \wedge (Q \vee R) = (P \wedge Q) \vee (P \wedge R)$

$P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$

(5) 交换律:  $P \wedge Q = Q \wedge P$

$P \vee Q = Q \vee P$

(6) 结合律:  $(P \wedge Q) \wedge R = P \wedge (Q \wedge R)$

$(P \vee Q) \vee R = P \vee (Q \vee R)$

(7) 逆否律 (连词化归律):  $P \Rightarrow Q = \sim Q \Rightarrow \sim P$



**(8)量词转换律**  $\sim (\exists x)P(x) = (\forall x) [\sim P(x)]$

$$\sim (\forall x)P(x) = (\exists x) [\sim P(x)]$$

**(9)量词分配律**

$$(\forall x) [P(x) \wedge Q(x)] = (\forall x)P(x) \wedge (\forall x)Q(x)$$

$$(\forall x) [P(x) \vee Q(x)] = (\forall x)P(x) \vee (\forall x)Q(x)$$

$$(10) (\forall x)P(x) = (\forall y)P(y)$$

$$(\exists x)P(x) = (\exists y)P(y)$$



## 4. 置换与合一 (Substitution and Unification)

### 置换 (Substitution)

#### 定义

假元推理

$$\left. \begin{array}{l} W_1 \Rightarrow W_2 \\ W_1 \end{array} \right\} \rightarrow \text{产生 } W_2$$

全称化推理

$$(\forall x) W(x) \rightarrow W(A)$$

└─ 约束变量
└─ 自由变量

综合推理

$$\left. \begin{array}{l} (\forall x) [W_1(x) \rightarrow W_2(x)] \\ W_1(A) \end{array} \right\} \rightarrow \text{产生 } W_2(A)$$



## 定义:

- 用项(A)替换函数表达式中的变量(x), 记为**Es**, 即表示一个表达式E(Expression)用一个置换S(Substitution)而得到的表达式的**置换**.

## 性质 (Characters)

- 可结合律 (Associative)

$$(E s1) s2 = E (s1 s2)$$

$$(s1 s2) s3 = s1 (s2 s3)$$

E 为表达式,  $s1, s2, s3$  是表达实例

- 不可交换 (Noncommutative)

$$s1 s2 \neq s2 s1$$







例子:

$P[x, f(y), B]$  的4个置换

$s1 = \{z/x, w/y\}$  then  $P[x, f(y), B]s1 = P[z, f(w), B]$

$s2 = \{A/y\}$  then  $P[x, f(y), B]s2 = P[x, f(A), B]$

$s3 = (q(z)/x, A/y)$  then  $P[x, f(y), B]s3 = P[q(z), f(A), B]$

$s4 = (c/x, A/y)$  then  $P[x, f(y), B]s4 = P[c, f(A), B]$



## 合一 (Unification)

### ■ 定义

- 寻找项对变量的置换，以使两表达式一致，叫做**合一** (unification)。合一是人工智能中很重要的过程。

### ■ 可合一 (Unifiable)

- 如果一个置换 $s$ 作用于表达式集 $\{E_i\}$ 的每个元素，则我们用 $\{E_i\}s$ 来表示置换例的集。我们称表达式集 $\{E_i\}$ 是可合一的，如果存在一个置换 $s$ 使得：

$$E_1s = E_2s = E_3s = \dots$$

那么称此 $s$ 为 $\{E_i\}$ 的**合一者** (unifier)。





例子:

表达式集 $\{P[x, f(y), B], P[x, f(B), B]\}$ 的合一者为

$$s = \{A/x, B/y\}$$

因为

$$\left. \begin{array}{l} P[x, f(y), B] \text{ } s = P[A, f(B), B] \\ P[x, f(B), B] \text{ } s = P[A, f(B), B] \end{array} \right\} \text{是一样的}$$

所以,  $s = \{A/x, B/y\}$  是  $\{P[x, f(y), B], P[x, f(B), B]\}$  的合一者,  $s = \{B/y\}$  是  $\{P[x, f(y), B], P[x, f(B), B]\}$  最简单的合一者.





## 2.4 Semantic Network Representation 语义网络

- J.R.Quillian在其博士论文中作为人类联想记忆的一个显式心理学模型最先提出
- 1972年，Simmons首先将语义网络表示法用于自然语言理解系统
- 结构化手段，高效和集中地描述特定地事物和事物之间的关系，强大而直观。





## 语义网络

- 通过概念及其语义关系来表达知识的一种**网  
络图**
- 带标志的有向图
- 语义网络是**知识**的一种**结构化**表示方法
- 由**节点**(node)与有向**弧**(arc)组成
  - 节点表示概念、事物、事件、情况等
  - 弧表示语义关系



# 语义网络的结构

## ■ 组成部分

### ● 词法

- 词汇表中允许有哪些符号，它涉及各个节点和弧

### ● 结构

- 符号排列的约束条件，指定各弧线连接的节点对

### ● 过程

- 访问过程，这些过程能用来建立和修正描述，以及回答相关问题

### ● 语义

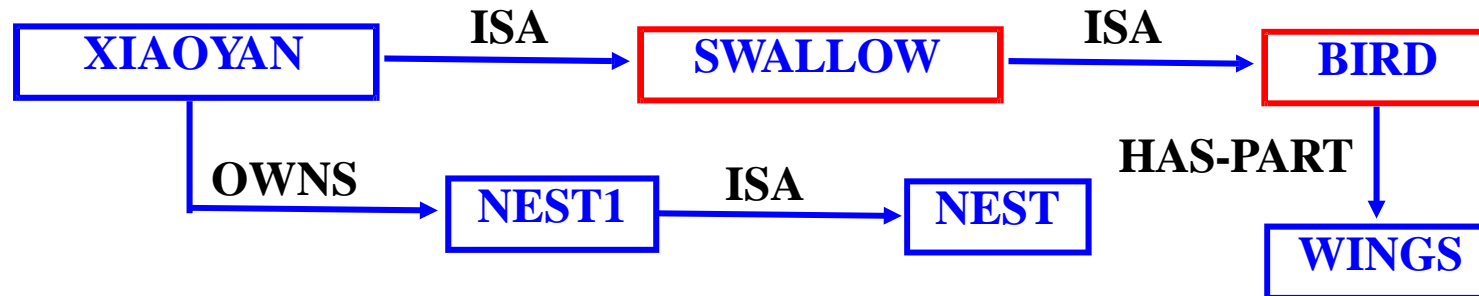
- 确定有关节点的排列及其占有物和对应弧线





## 二元语义网络的表示

Ex1: All swallows are birds



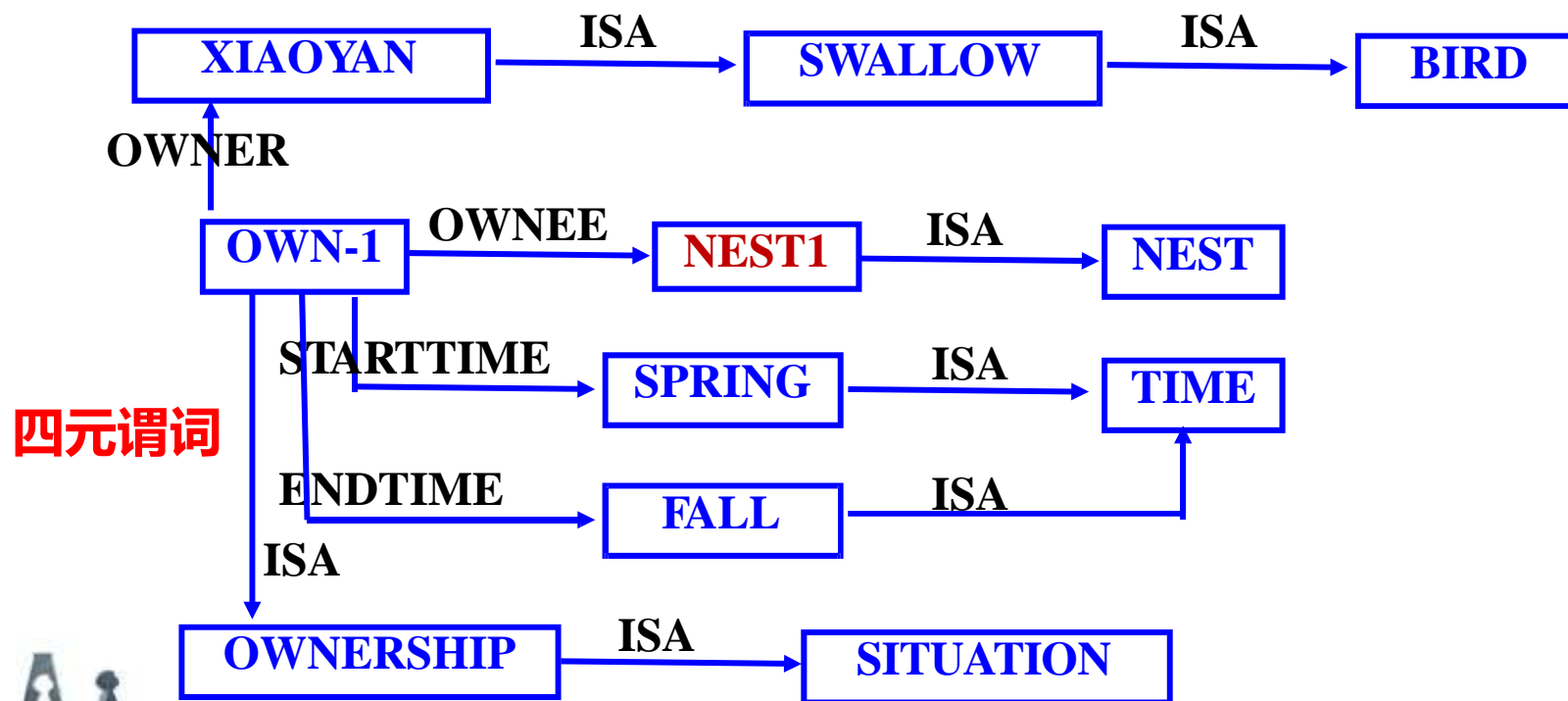
**XIAOYAN is a swallow.**

**Bird has wings.**

**XIAOYAN owns a nest.**



Ex2: XIAOYAN owns a nest from spring to autumn.







## 选择语义基元

寻找基本概念和基本弧

My car is tan.



ISA

CAR

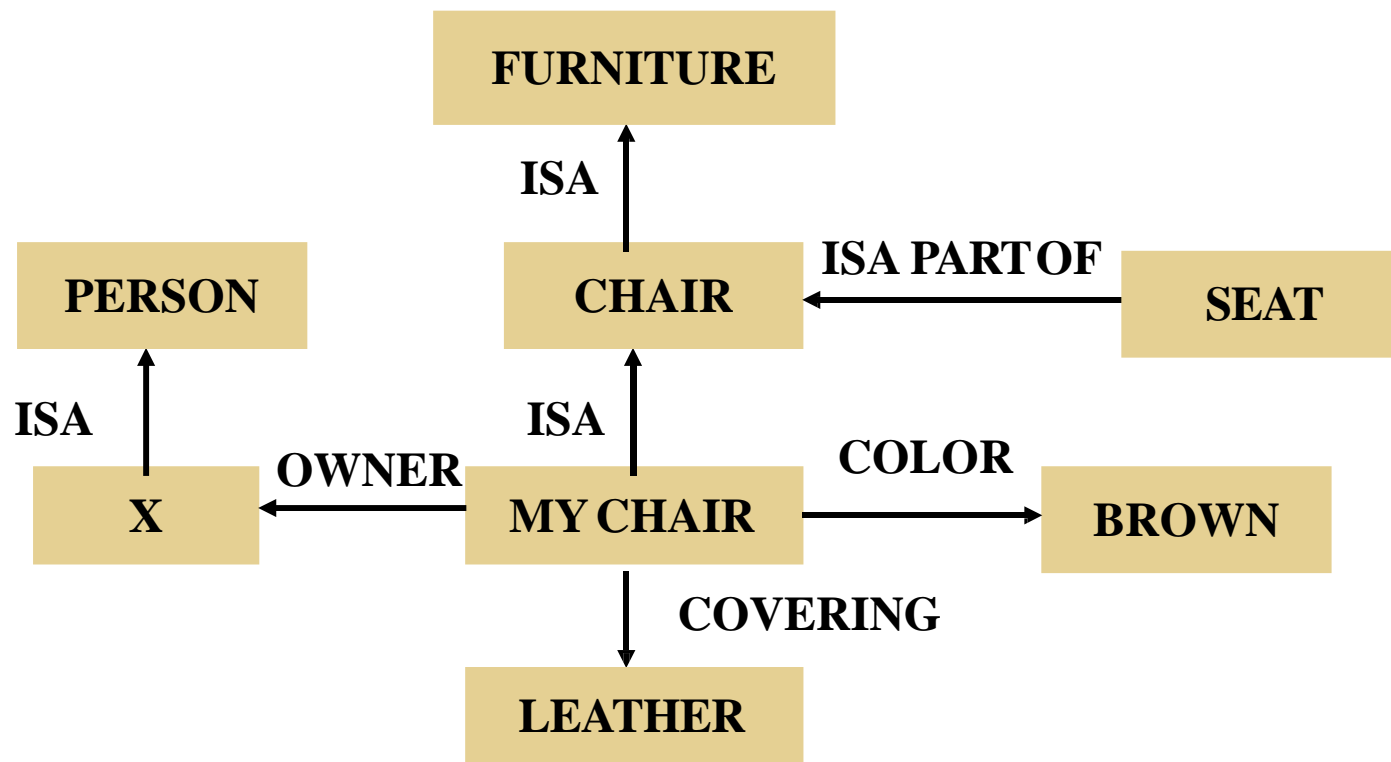
ISA

Lihua's car is green





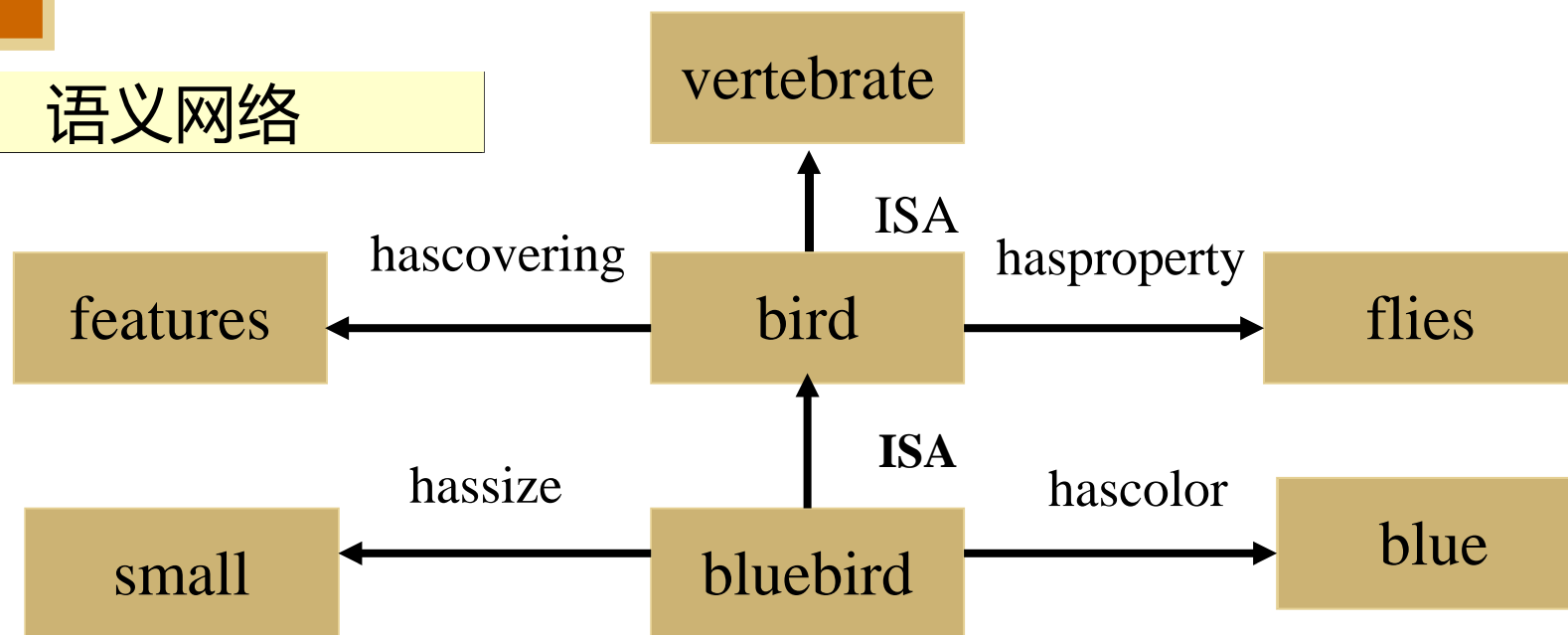
Example 3. The color of my chair is brown; the covering of the chair is leather; chair is one kind of furniture; chair is part of seat; chair's owner is X; X is a person.





例 蓝鸲是一只体型很小 蓝色羽毛会飞的卵生脊椎动物

语义网络



谓词演算

hassize(bluebird, small)  
hascovering(bird, features)  
hascolor(bluebird, blue)  
hasproperty(bird, flies)  
isa(bluebird, bird)  
isa(bird, vertebrate)





## 多元语义网络的表示

谓词逻辑与语义网络等效

(谓词逻辑)

ISA (LIMING, MAN) 或 MAN (LIMING)

LIMING

ISA



MAN

(语义网络)



## 多元语义网络表示的实质

- 把多元关系转化为一组二元关系的组合，或二元关系的合取。

$$R(X_1, X_2, \dots, X_n)$$

$$R_{12}(X_1, X_2) \wedge R_{13}(X_1, X_3) \wedge \dots \wedge R_{1n}(X_1, X_n)$$

.....

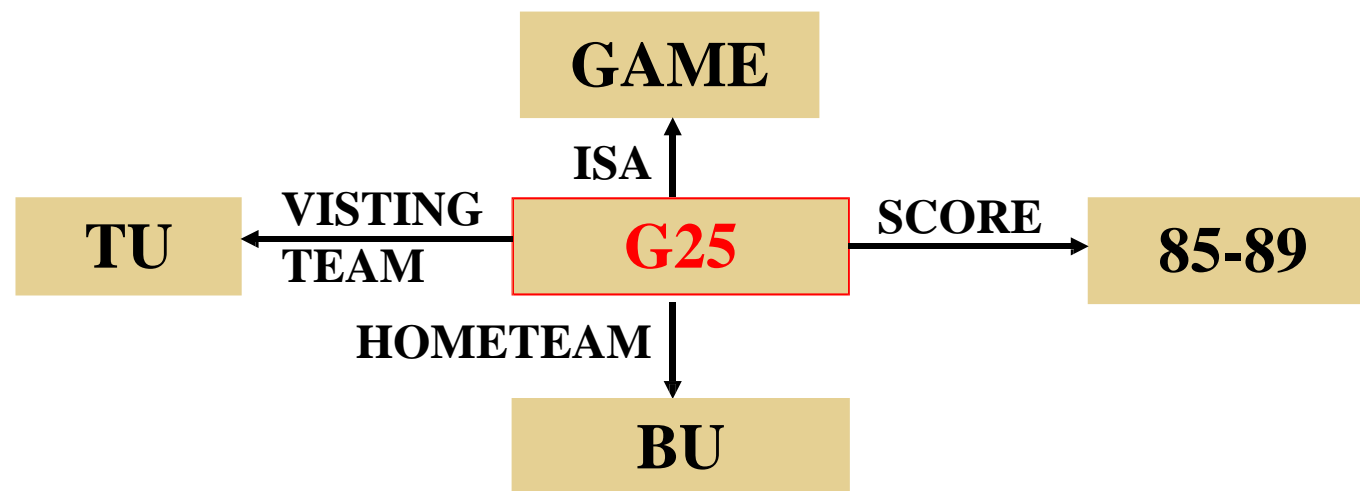
$$R_{n-1\ n}(X_{n-1}, X_n)$$



Ex4: 要表达北京大学(BEIJING University,简称BU) 和清华大学(TSINGHUA University,简称TU)两校篮球队在北大进行的一场比赛的比分是85比89.

**谓词逻辑:**  $\text{SCORE}(\text{BU}, \text{TU}, (85:89))$

**语义网络:** 三元关系转换成一组二元关系的组合或合取, 可以建立一个**G25**节点来表示这场特定的球赛。



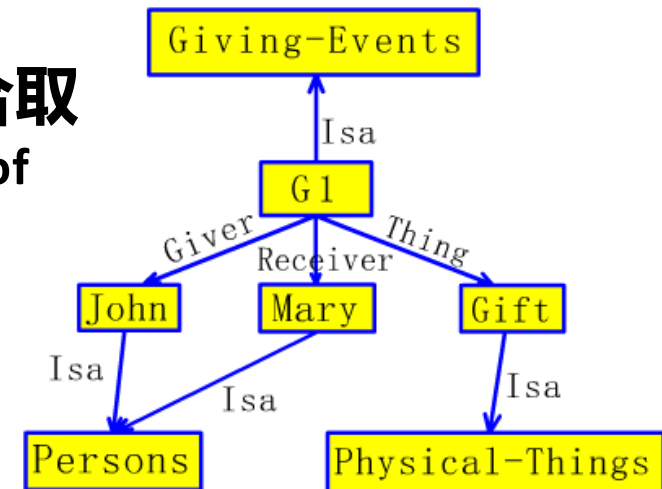


Ex5: John gives Mary a gift.

**Gives**(John, Mary, Gift)

### 三元关系可转换成一组二元关系的合取

- The whole sentence describes a instance of Giving-Event case, as **G1**.
- John is the Giver in G1. Mary
- is the Receiver in G1. Gift is
- a thing in G1.



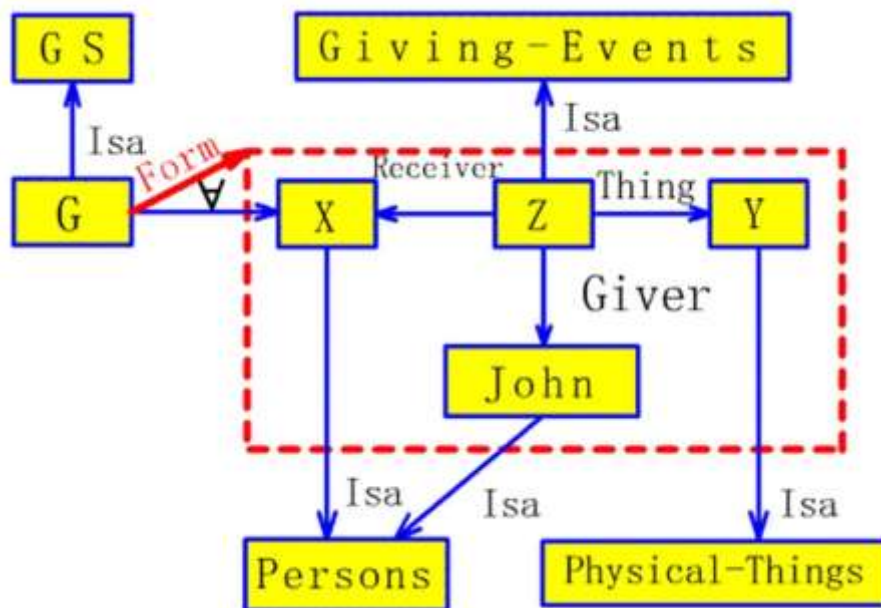
$\text{Isa}(\text{G1}, \text{Givig-Event}) \wedge \text{Giver}(\text{G1}, \text{John}) \wedge \text{Receiver}(\text{G1}, \text{Mary})$   
 $\wedge \text{Thing}(\text{G1}, \text{Gift})$



Ex6: John gave each person a gift.

Ex5:  $\text{Isa}(\text{G1}, \text{Givig-Event}) \wedge \text{Giver}(\text{G1}, \text{John}) \wedge \text{Receiver}(\text{G1}, \text{Mary})$   
 $\wedge \text{Thing}(\text{G1}, \text{Gift})$

$(\forall x)(\exists y)(\exists z) [\text{Isa}(\mathbf{z}, \text{Giving-Event}) \wedge \text{Giver}(\mathbf{z}, \text{John}) \wedge \text{Receiver}(\mathbf{z}, \mathbf{x})$   
 $\wedge \text{Thing}(\mathbf{z}, \mathbf{y})]$  **合适公式**



显示全称量词，从实例节点G引出两种关系弧：

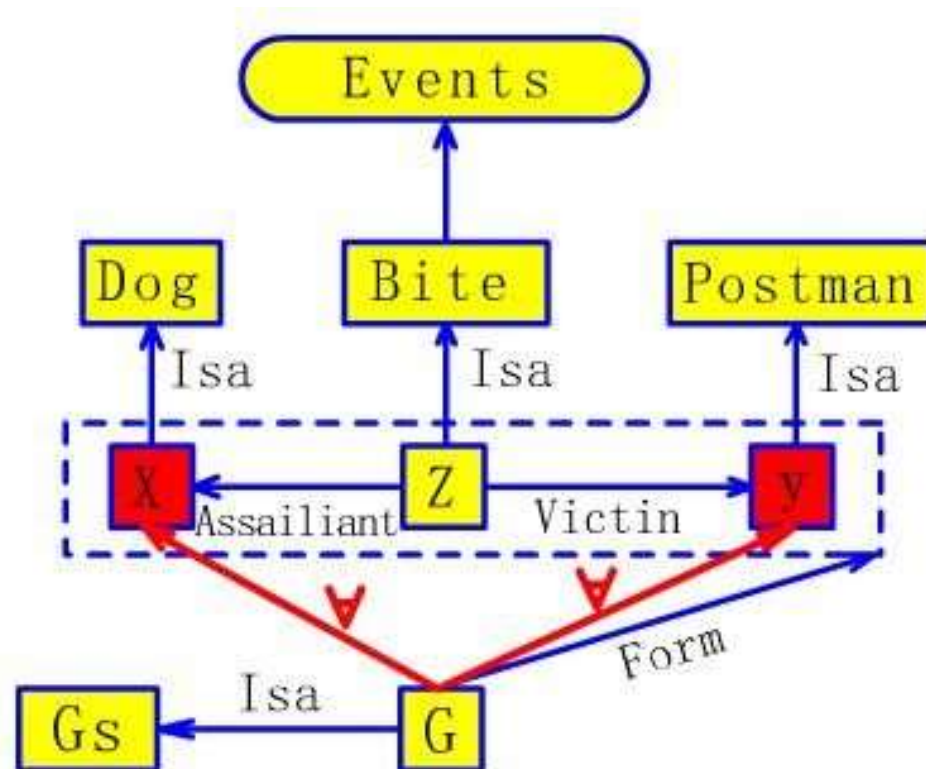
- 带标签**Form**，指向全称量词的辖域
- 带标签**∇**，指向全称量词的约束变量（可多个）





Ex7: Every dog has bitten every postman

合适公式?

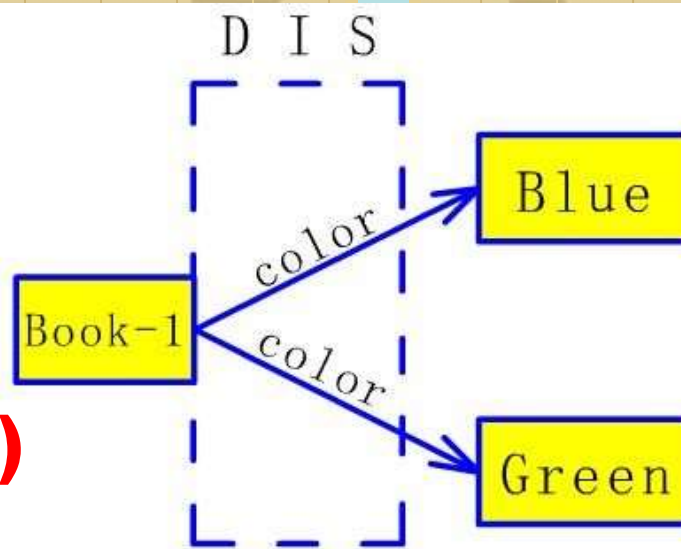


- 包含多个全称量词约束变量 (e.g.  $\forall x$ 和 $\forall y$ )



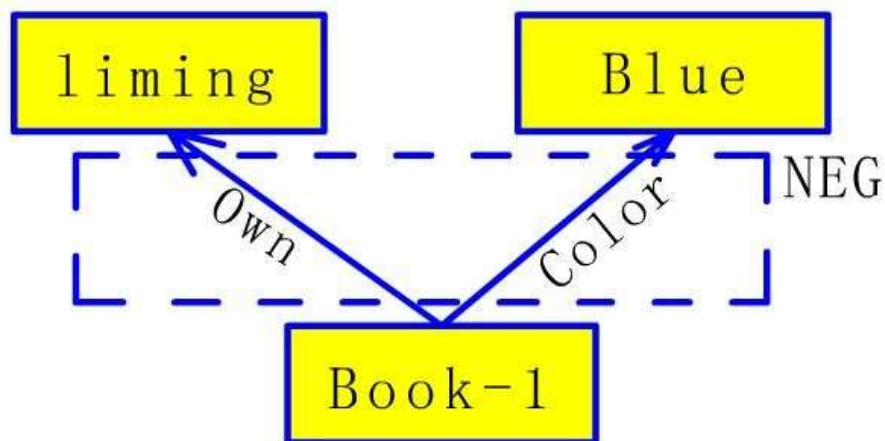
语义网络中逻辑“或”

**Disjunction (DIS)**



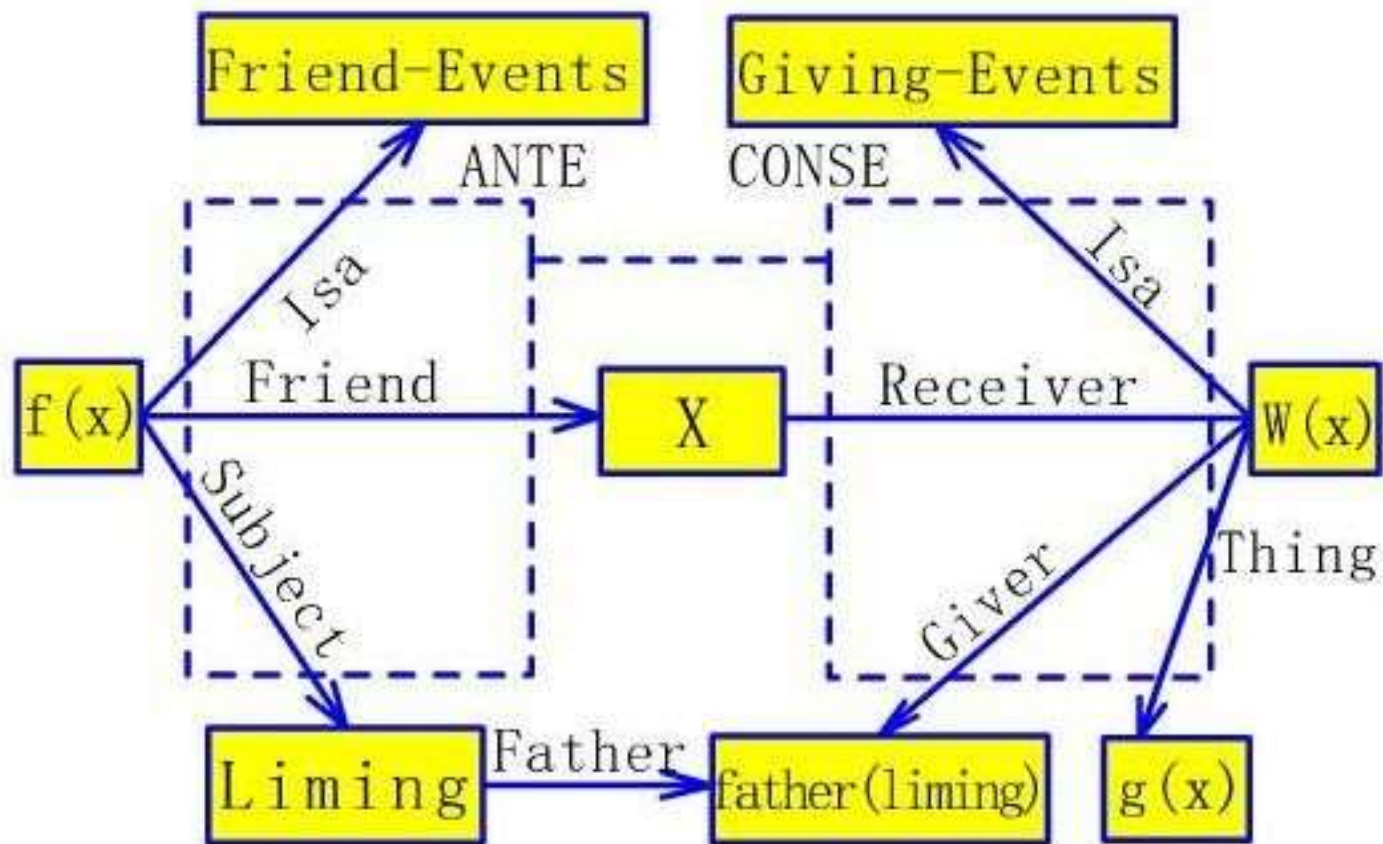
语义网络中逻辑“非”

**Negative (NEG)**





Ex7: Liming's father gave Liming's each friend a gift





## 节点Node

- 概念节点

- 用以表示基本概念

- 实例节点

- 用以表示具体事物或属性





弧

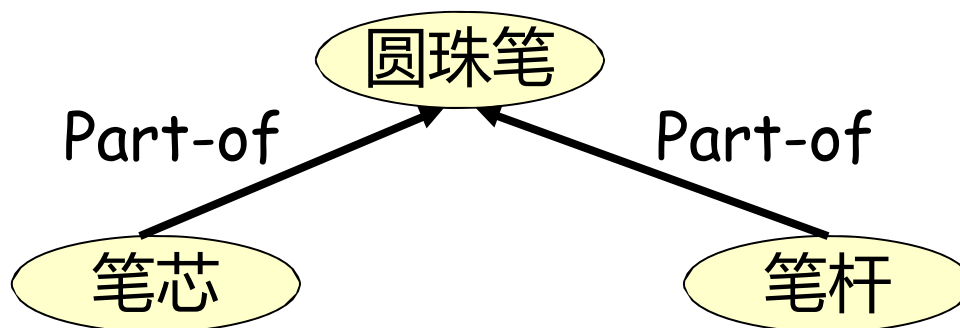
以个体为中心（对应节点一般是名词性个体或概念）

实例联系（ISA）

泛化联系，子类（A Kind Of, AKO）



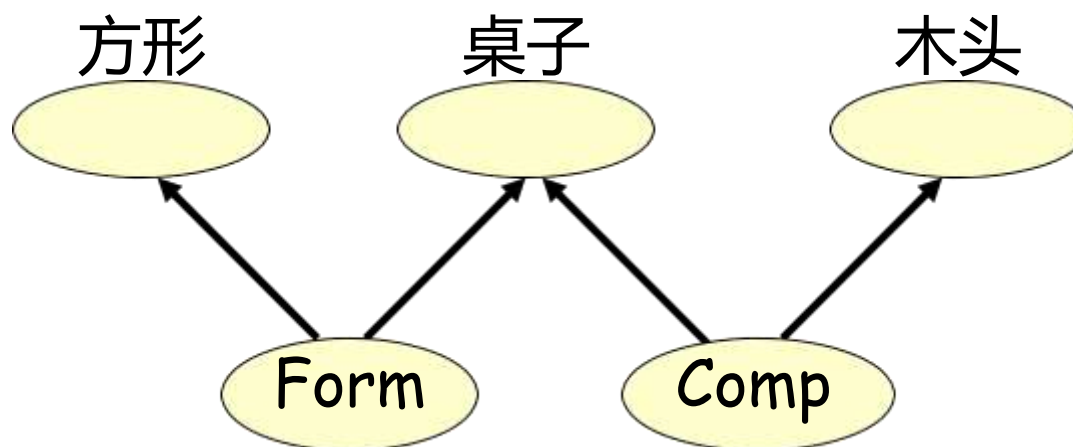
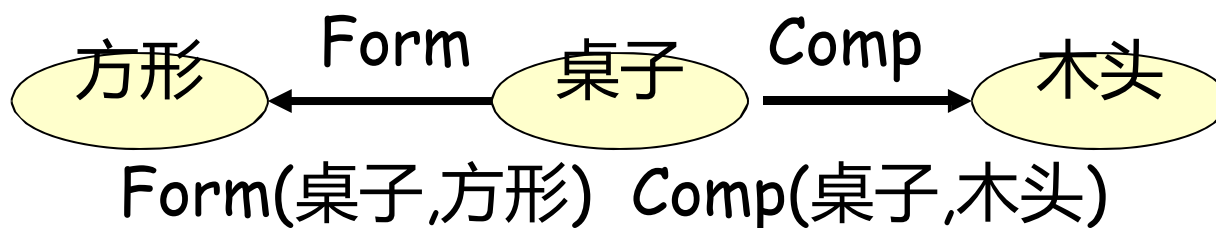
聚集联系（Part-of）

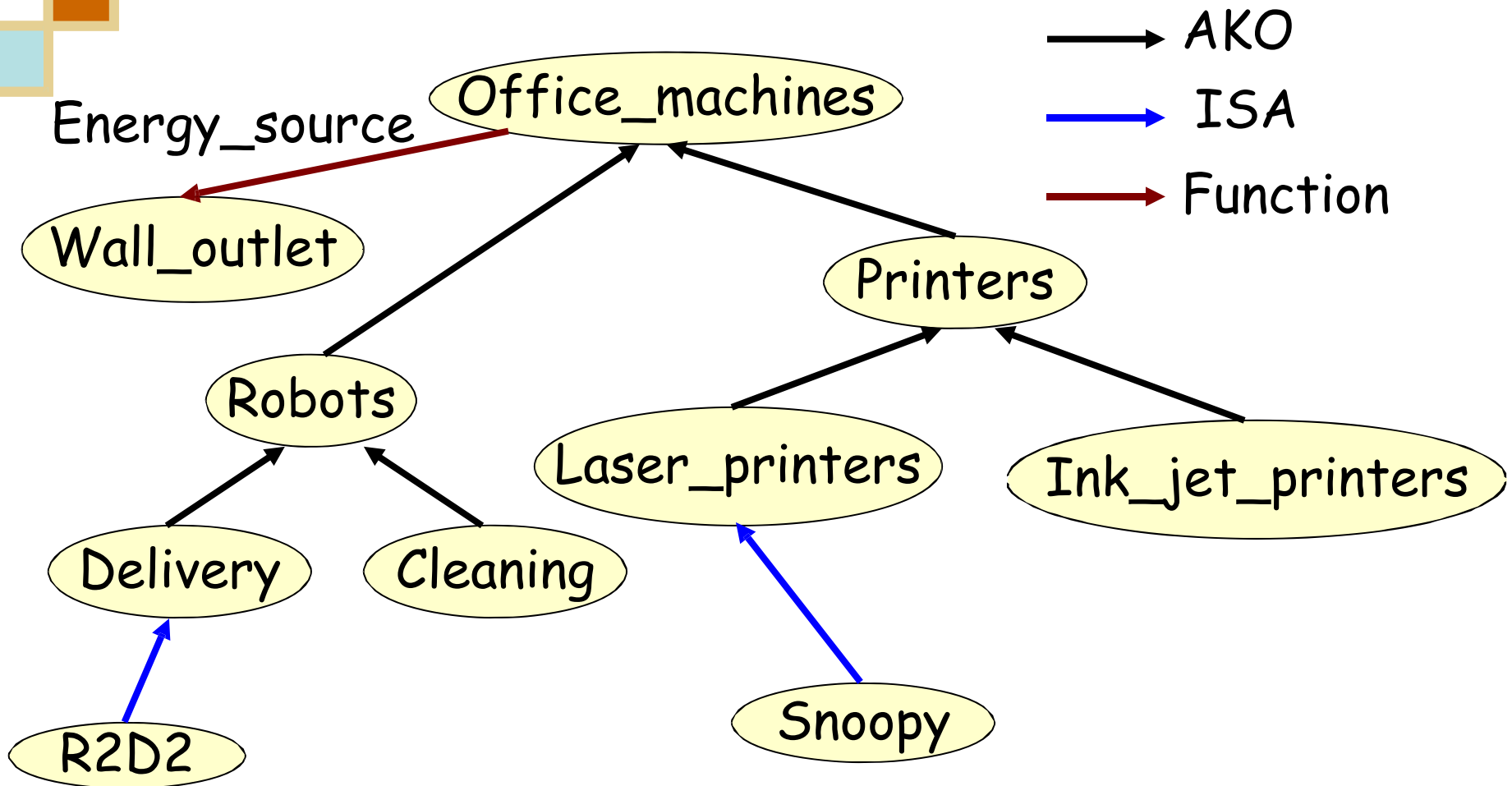




## 以谓词或关系为中心

例：有一张木头做的方桌







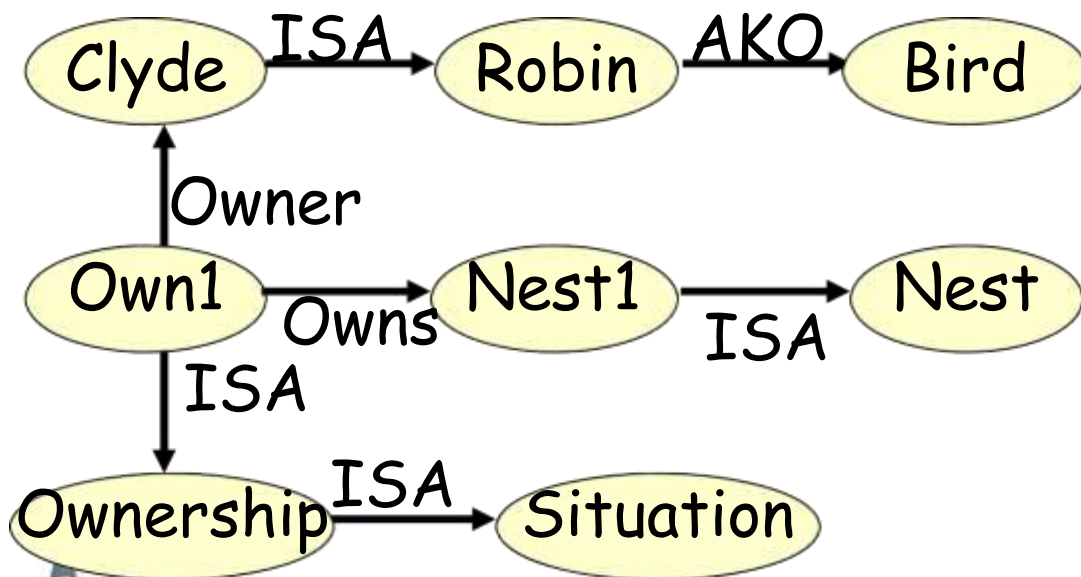


# 语义网络的推理

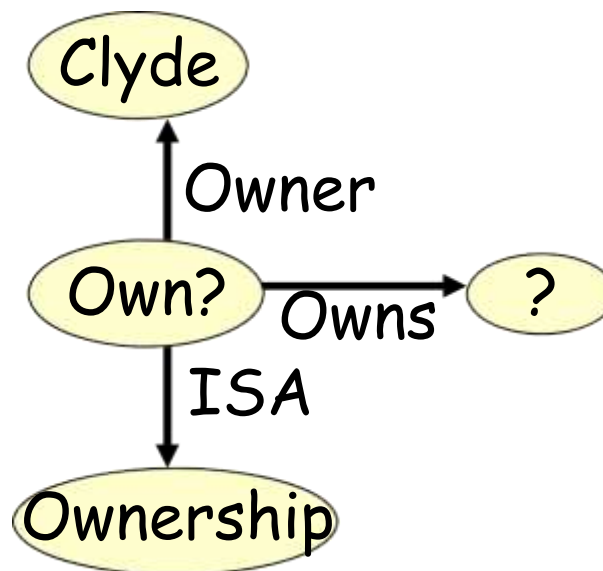
两种推理机制：匹配、继承

匹配(包括节点和弧的匹配)

- 构造目标网络块
- 在事实网络中寻找匹配



What does Clyde own?







## 继承

把对事物的描述从概念节点或类节点传递到实例节点

### 三类继承性

- ❖ 直接传递(pass)——子节点直接继承父节点的属性
- ❖ 附加传递(add)——子节点把父节点的属性和自己的属性相综合
- ❖ 排斥传递(exclude)——子节点与父节点的属性不相容, 抑止传递

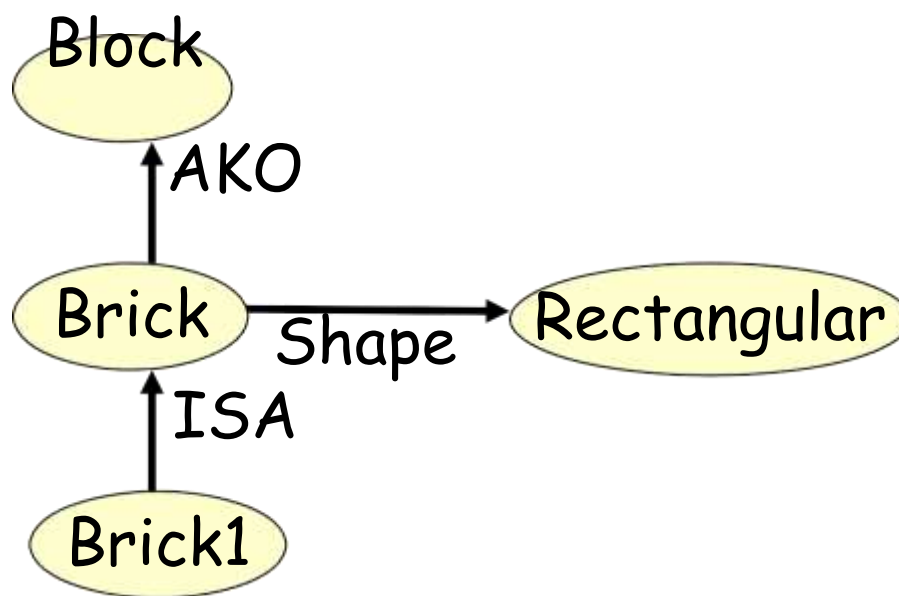


## 三种继承方式

### 值继承

is-a, AKO(a kind of)

求Brick1的形状

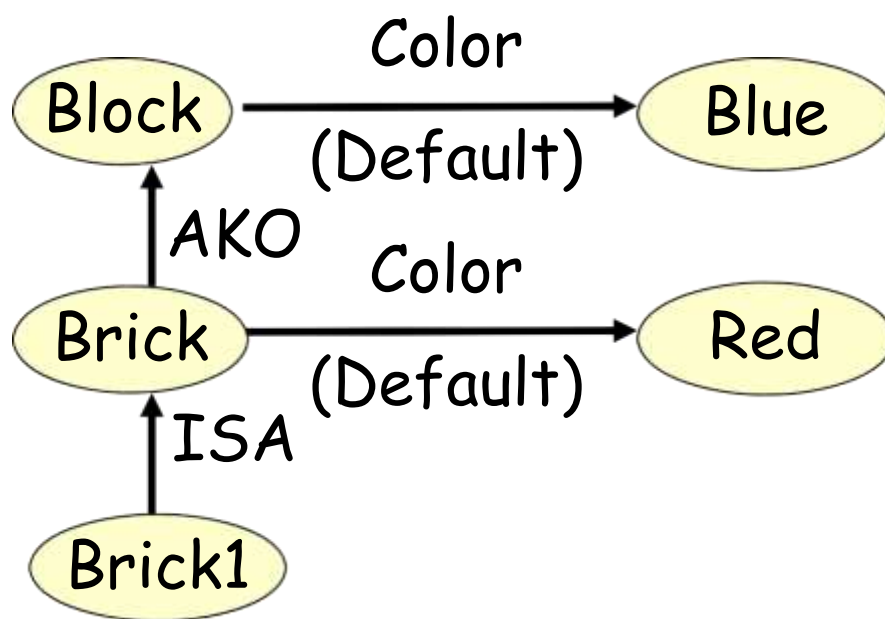




## 默认继承 (Default)

默认值——具有相当程度的真实性但不能十分肯定的值

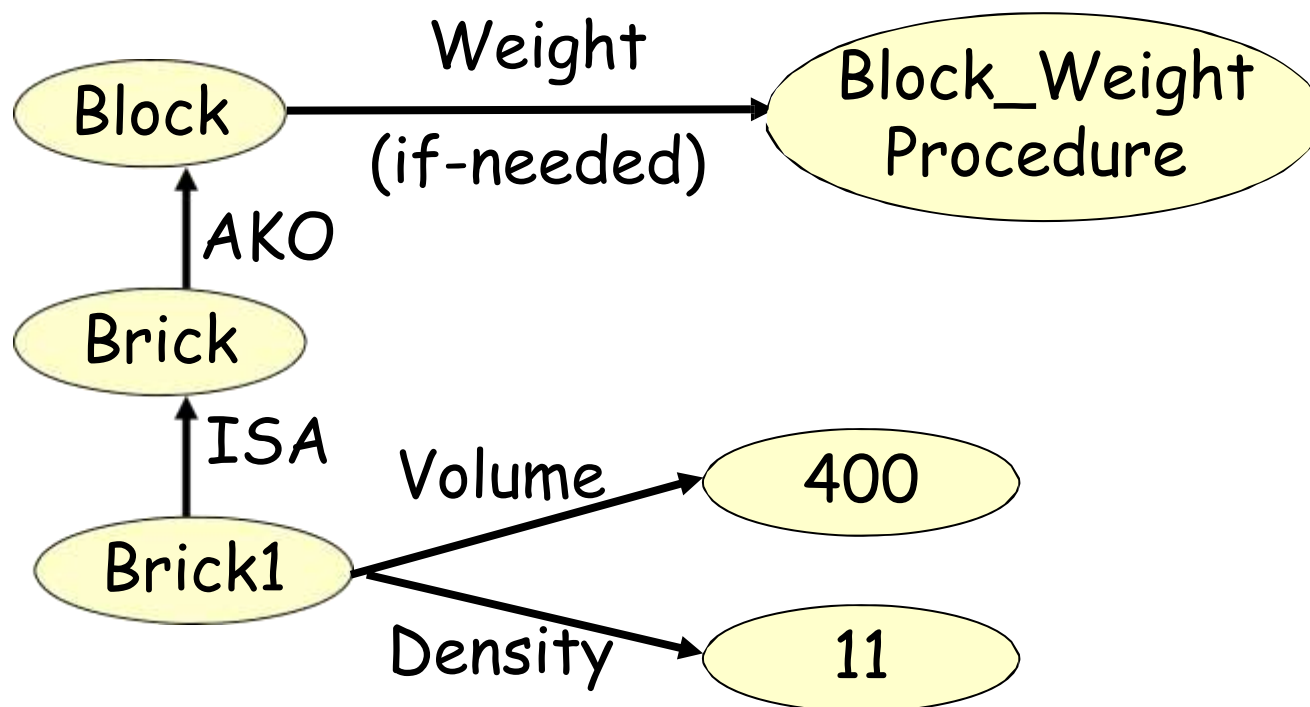
如：鸟会飞；头疼可能是感冒





■ if-needed (“如果需要”继承, “附加过程”继承)

某些情况下, 对事物的描述不能直接从概念节点或类节点继承得到, 但可利用已知信息来计算, 这种计算程序称为if-needed程序





# 语义网络表示的特点与不足

- ✿ 具有结构性  
适于表示分类学型知识和事物特性的知识
- ✿ 灵活性  
可以任意定义新的节点与弧
- ✿ 继承性，表达能力强
- ✿ 语义网络上的继承推理具有非单调性
- ✿ 处理上的复杂性
- ✿ 非严格性
- ✿ 多重继承冲突问题是重要的研究课题

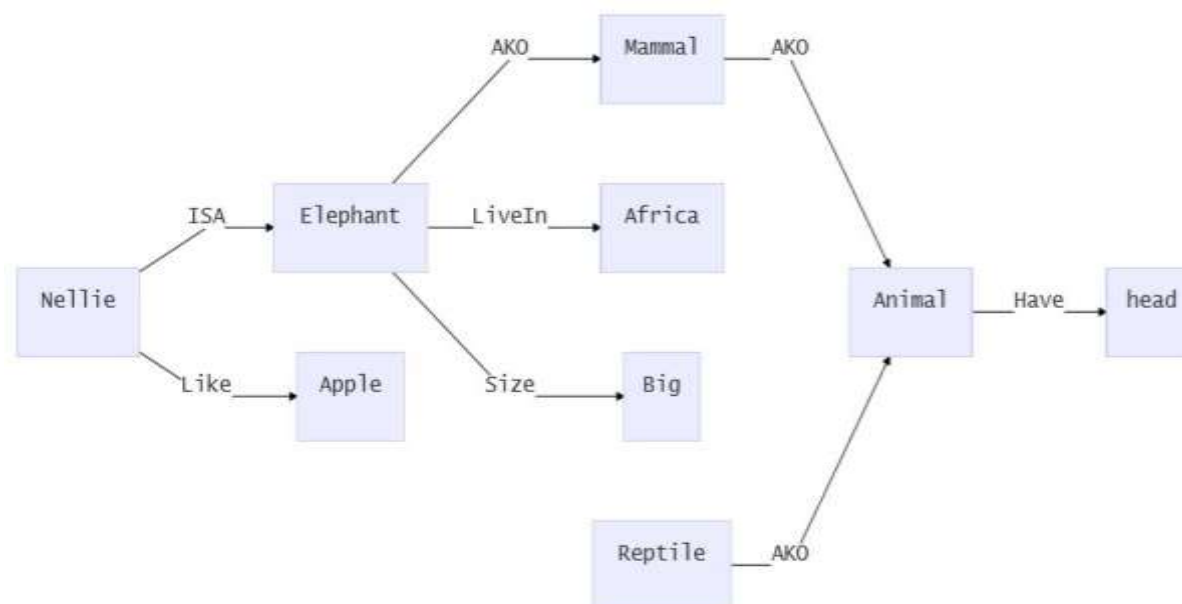




## 思考题

使用语义网络来表示:

- Nellie is an elephant, he likes apples. Elephants are a kind of mammals, **they** live in Africa, and **they** are big animals. Mammals and reptiles are both animals, all animals have head.





## 2.5 Others 其它方法

### 框架 ( Frame )

- 框架是一种结构化表示法，通常采用语义网络中的节点-槽-值表示结构。

### 剧本 ( Script )

- 剧本是框架的一种特殊形式，它用一组槽来描述某些事件的发生序列。

### 过程 ( Procedure )

- 过程式表示就是将有关某一问题领域的知识，连同如何使用这些知识的方法，均隐式地表达为一个求解问题的过程。





# Frame (框架)

- 75年由美国人工智能学者明斯基提出
- 认为人们对现实世界的认识都是以一种类似于框架的结构存储在记忆中
- 信息检索面临新形势
  - 信息存储在带槽的帧中
  - 一些插槽触发操作，导致新的情况
- 框架是一种描述所论对象属性的数据结构
  - 框架是在某种情况下填充的模板
- 在框架理论中，框架是知识表示的一个基本单位
- 也是一种语义网络
- 继承性是其重要特性
  - 也非常类似于OOP中的对象







# 框架的结构

- **框架**描述一类物体，由**框架名**与一些描述物体各个方面的**槽(slot)**组成
- 每个槽可分为多个**侧面**(facet)
- 每个侧面可有一个或多个**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, ...)

...

约束: 约束条件1, 约束条件2...





<假冒伪劣商品>

商品名称:

生产厂家:

出售商店:

处 罚: 处理方式:

处理依据:

处罚时间: 单位 (年、月、日)

经办部门:





## 槽（侧面）类型

- 值
- 缺省
- 继承
- 如果需要 (if-needed)

<CHAIR>

Specialization-of:FURNITURE

Number-of-Legs:DEFAULT 4

Style-of-Back: Straight, Cushioned

Number-of-Arms: 0, 1 or 2

<JOHN'S-CHAIR>

Specialization-of:CHAIR

Style-of-Back: Cushioned

Number-of-Arms: 0



## 四种侧面填写方式

- 由已知情况或物体属性提供
- 通过默认隐含
- 由继承获得
- 对附加过程侧面通过执行附加过程实现

● 框架中的槽与侧面可任意定义

槽与侧面也可以是另一框架，形成框架网络





# 框架网络

- 框架间的联系由槽名指明

- 常用槽

  - ISA槽

<ATHLETE>

Name:

Age:

Sex: range: (male, female)  
default: male

<CHESS PLAYER>

ISA: ATHLETE

brain: excellent





■ Subclass槽

■ AKO槽

■ Instance槽

是AKO槽的逆关系

<ATHLETE>

Instance: <Chess player>,  
<football player>,  
<basketball player>

Name:

Age:

Sex: range: (male, female)

default: male



Part-of槽

Infer槽

指出两个框架间的逻辑推理关系  
可表示相应的产生式规则

Possible-Reason槽

与Infer槽相反

<诊断规则>

症状1：咳嗽

症状2：发烧

症状3：流涕

Infer：<结论>

可信度：0.8

<结论>

病名：感冒

治疗方法：服用感冒胶囊，  
一日3次，每次2-3粒

注意事项：多喝开水

Possible-Reason：<诊断规则>



## 对框架及侧面进行合理组织

- 减少重复性信息
- 尽量将不同框架描述的相同属性取出构成上层框架  
如：用框架描述鸽子、啄木鸟、布谷鸟、燕子及鹦鹉  
五种动物

<鸟>

体表覆盖物：羽毛  
移动方式：飞，走  
生殖方式：卵生

<鸽子>

AKO：鸟

羽毛颜色：白，灰，花

<燕子>

AKO：鸟

羽毛颜色：黑白





# 框架推理

两种推理活动：匹配、填槽

## 匹配

- 将待解问题用框架表示
- 匹配通过对相应槽的槽名和值逐个比较实现
- 比较往往牵涉到其它框架
- 问题的随机性也使匹配复杂化

## 填槽

四种填槽方式

- 查询：中间结果或用户输入
- 默认
- 继承
- 附加过程计算





<师生员工>

姓名:

年龄:

性别:

range:男,女

default:男

健康状况:

range:健康,

一般,差

default:一般

住址: <住址>

<教职工>

AKO: <师生员工>

工作类别:

range:教师,干部,语种:

工人

default:教师

开始工作时间:

截止工作时间:

default:现在离

退休状况:

range:离休,退休

default:退休

<教师>

AKO: <教职工>

部门: 单位(系,教研室)

range:英,法,德,日,俄

default:英语

外语水平:

range:优,良,中,差

default:良

职称:

range:教授,副教授,

讲师,助教

default:讲师

研究方向:

<教师1>

AKO: <教师>

姓名: 孙林

年龄: 28

健康状况: 健康

部门:

计算机系软件教研室

语种: 德语

开始工作时间:

1985.9

...

<教师x>

AKO: <教师>

姓名:

年龄: <30

性别: 男

健康状况: 健康

职称: 讲师

**例: 在关于师生员工的框架网络种找满足  
如下条件的教师:**

**男性, 30岁以下, 身体健康, 讲师**

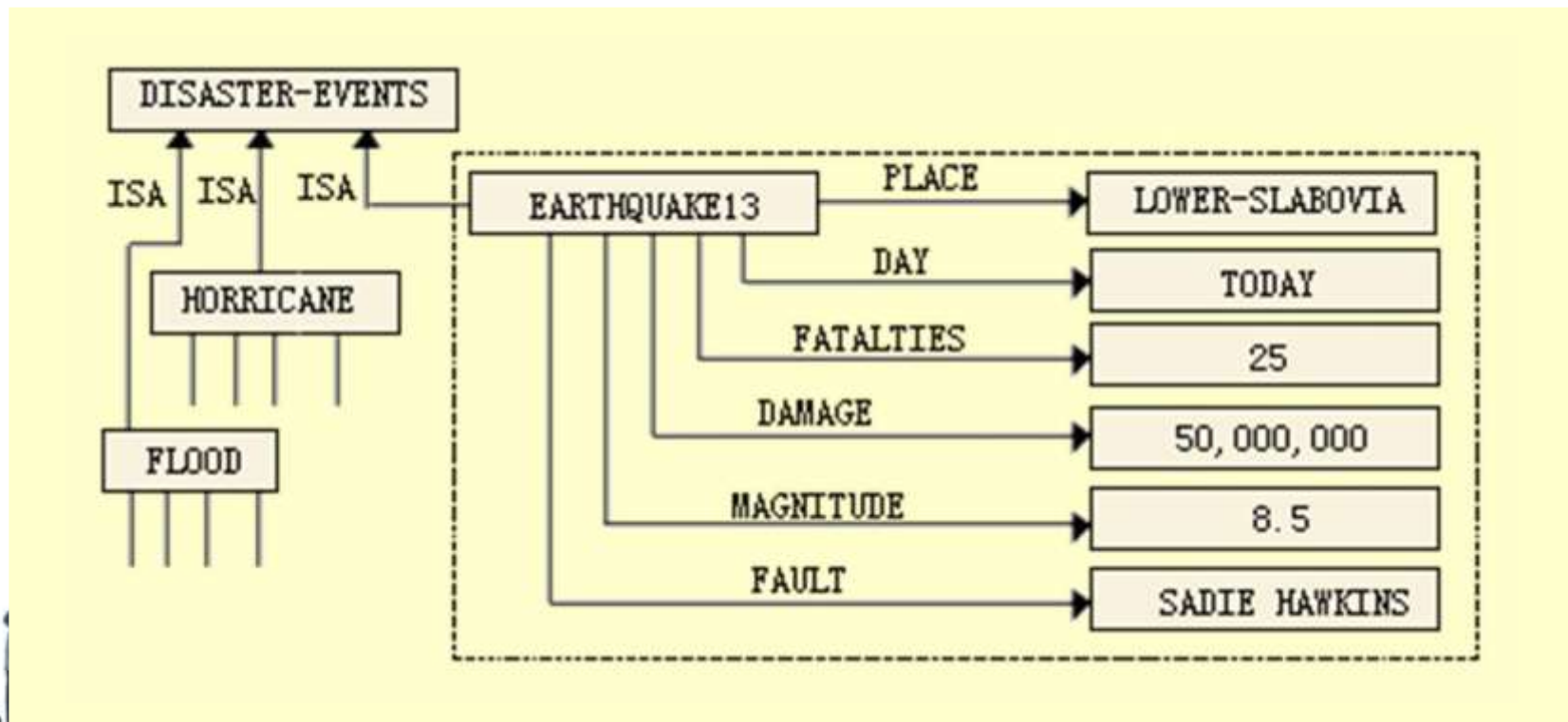


# 框架表示的特点与不足

- 结构性
- 继承性
- 自然性
- 模块性
- 没有形成完整的理论体系
- 缺乏清晰的语义
- 多重继承的冲突



Example: Today, a magnitude-8.5 earthquake struck the area of Lower-Slovenia, resulting in 25 deaths and 500 million U.S. dollars in property damage. The Lower-Slovenian President said close to the Sa Dihao Kings hit has been a dangerous area over the years.





# Script (脚本表示法)

- 依据概念依赖理论提出的
- 表示特定领域内事件的发生序列
- 脚本的组成
  - 进入条件
  - 角色
  - 道具
  - 场景
  - 结局





脚本：餐厅

进入条件：顾客饿了，需要进餐；顾客有钱

角色：顾客、服务员、厨师、老板

道具：食品、桌子、菜单、钱

场景：

第一场：进入餐厅

顾客走入餐厅

顾客寻找桌子

在桌旁坐下

第二场：点菜

服务员给顾客菜单

顾客点菜

顾客把菜单给服务员

服务员告诉厨师所要食品

厨师做菜





第三场：上菜进餐

厨师把菜给服务员

服务员把菜送给顾客

顾客吃菜

第四场：顾客离开

顾客告诉服务员结帐

服务员拿来帐单

顾客付钱

顾客离开餐厅

结局：顾客吃了饭；

顾客花了钱；

老板挣了钱；

餐厅食品少了

在脚本适用于给定事件时，可通过它预测没有明显提及的事实





## 2.6 知识图谱 Knowledge Graph

### 知识图谱的概念与分类

- ❖ 它背后的思想可以追溯到上个世纪五六十年代所提出的一种知识表示形式——语义网络。
- ❖ 知识图谱 (Knowledge Graph) 于2012年由谷歌提出并成功应用于搜索引擎当中。它以**结构化的形式描述客观世界中概念、实体及其之间的关系**，将**互联网的信息**表达成更接近人类认知世界的形式，提供了一种更好地组织、管理和理解互联网海量信息的能力。
- ❖ 知识图谱的**分类**方式很多，例如可以通过知识种类、构建方法等划分。从领域上来说，知识图谱通常分为两种：通用知识图谱、特定领域知识图谱。

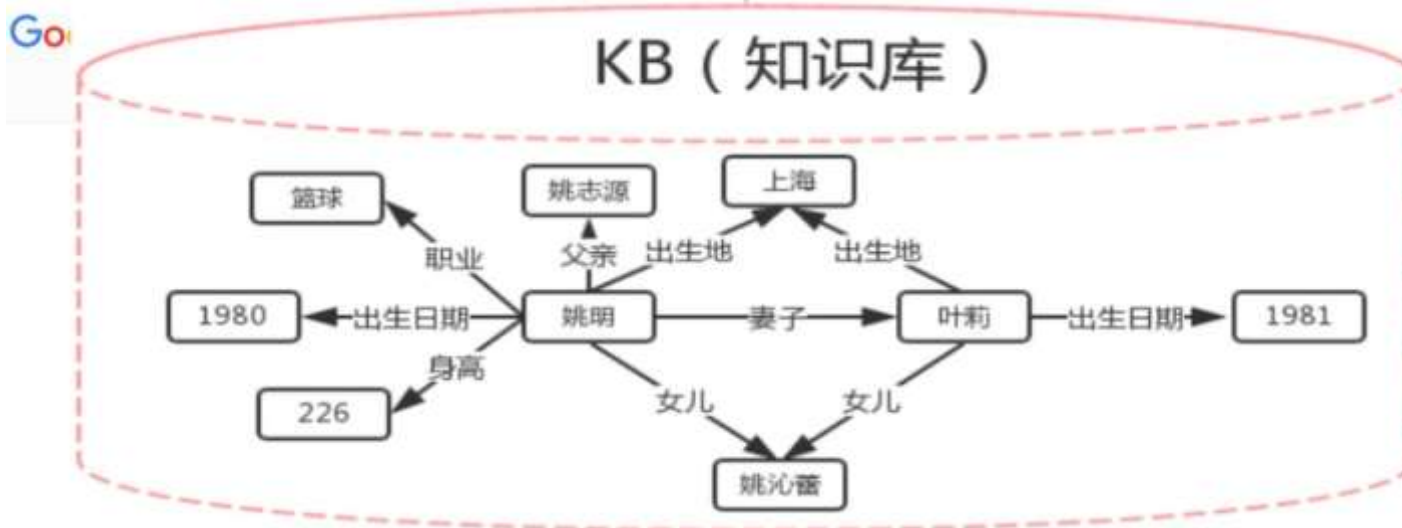






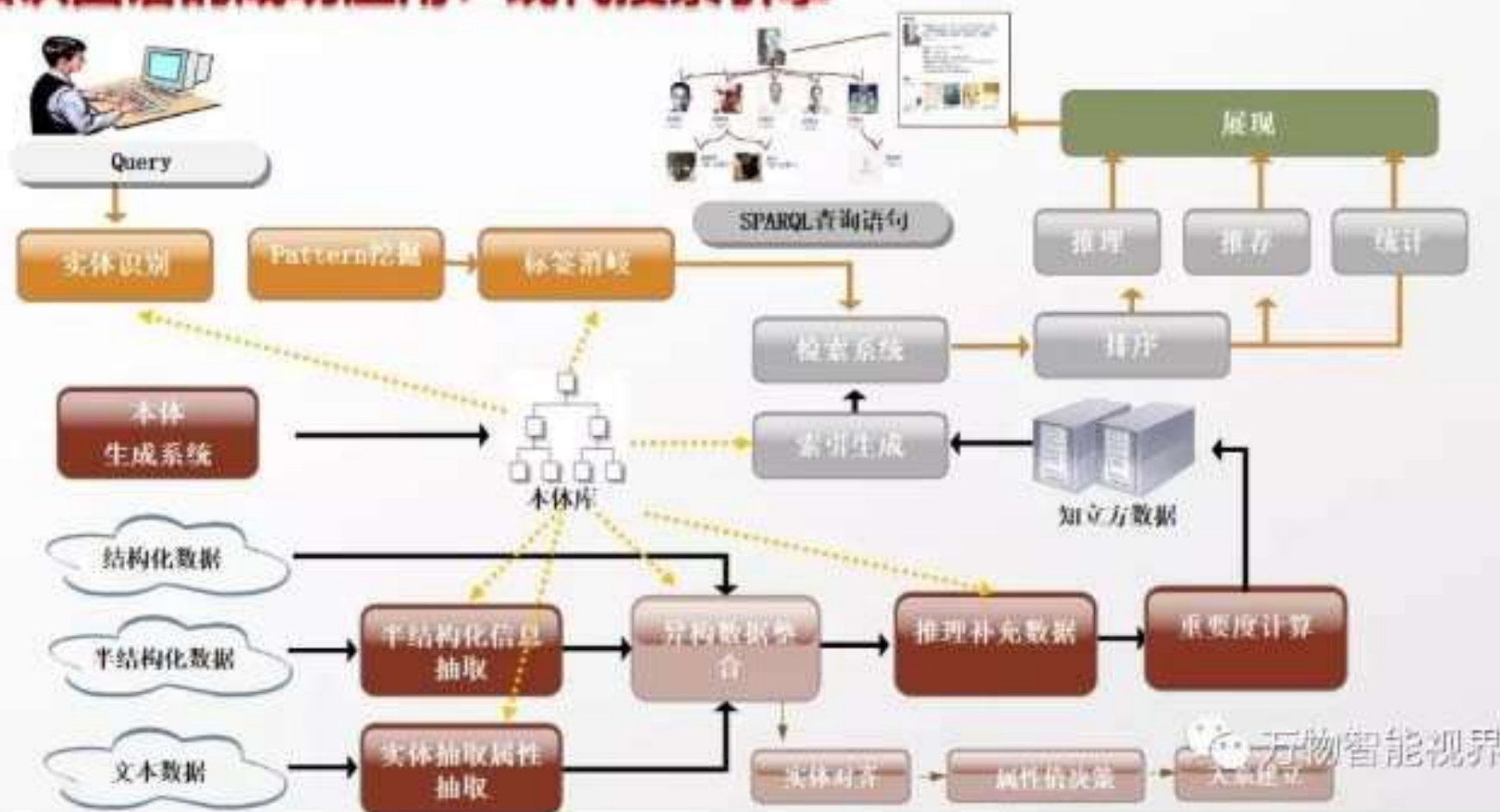
## 知识图谱

- 知识图谱旨在描述真实世界中存在的各种实体或概念及其关系，一般用**三元组**表示。
- 可看成一张巨大的**图**，节点表示实体或概念，边表示属性或关系。
- 知识图谱的体系架构是其指构建模式结构。





## 知识图谱的成功应用：现代搜索引擎



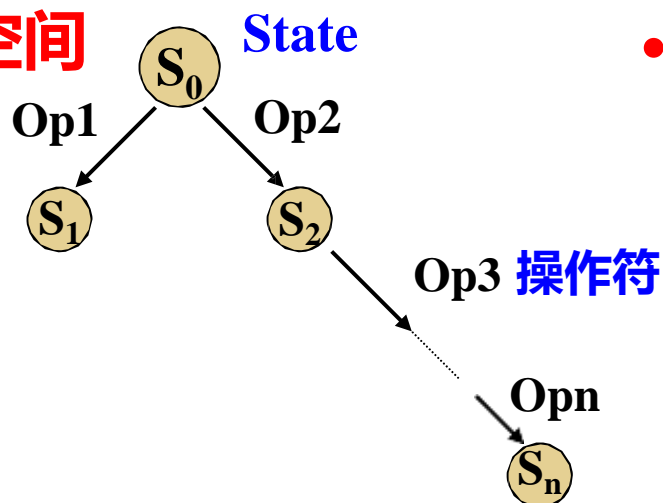


# Summary

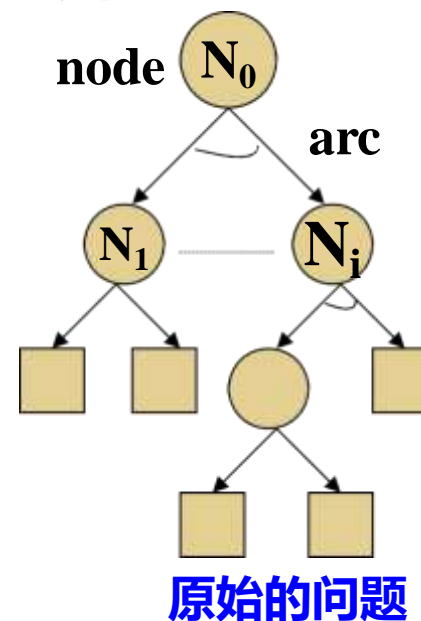
- 本章所讨论的**知识表示问题**是人工智能研究的核心问题之一。
- ❖ 对于同一问题可以有許多不同的表示方法。不过对于特定问题有的表示方法比较有效，其它表示方法可能不大适用，或者不是好的表示方法。
- ❖ 在表示和求解比较复杂的问题时，采用单一的知识表示方法是远远不够的。往往必须采用多种方法混合表示。



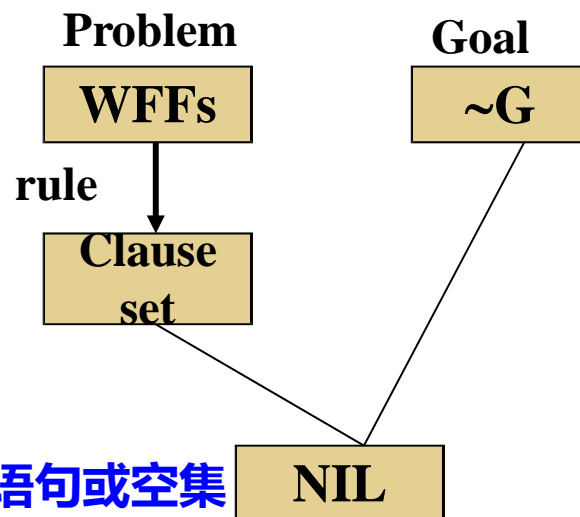
## • 状态空间



## • 问题归约



## • 谓词逻辑



## • 语义网络





# Summary

方法	Problem	Operation	Goal	Solution
状态空间	state	operator	Goal state	Solution path
问题归约	node	arc	terminal nodes	Solution tree
谓词逻辑	wffs	rule	root of tree	answer statement
语义网络	node	link	network	network



## 四种表现方式之间的关系



方法	初始问题	算符	目标	结果
状态空间	状态	算符	目标状态	解答路径 (path)
问题归约	节点	弧线	节点	解答树 (tree)
谓词逻辑	合适公式	子句集 置换合一 消解反演	根节点	NIL
语义网络	节点	链	目标网络	语义网络





作业三 (20分一题, 5题)

1. 下列说话正确的是 ()

- A 置换可以交换                      B 公式集可以合一  
C 语义网络是知识的图解表示      D “时间”是春天的实例

2. 假设P为真, Q为假, 下列公式为真的是 ()

- A  $P \vee Q$    B  $P \wedge Q$    C  $P \rightarrow Q$    D  $\sim P$

3. 下列等价关系不成立的是 ()

- A  $\sim(\sim P)$  等价于P                      B  $P \vee Q$  等价于  $\sim P \rightarrow Q$   
C  $\sim(P \vee Q)$  等价于  $\sim P \wedge \sim Q$       D  $P \rightarrow Q$  等价于  $\sim P \rightarrow \sim Q$





### 作业三 (20分一题, 5题)

#### 4. 用谓词演算公式表示下列句子:

- (1) 长沙是个美丽的城市, 但它不是一个沿海城市.
- (2) A computer system is intelligent if it can perform a task which, if performed by a human, requires intelligence.

注意: 多用而不是省用不同谓词和项。例如不要用单一的谓词字母来表示每个句子。

#### 5. 用语言网络表示下列信息:

- 1 李三是思源公司的经理, 他35岁, 住在飞天胡同68号.
- 2 All branch managers of DEC participate in a profit-sharing plan.

