

PCS 2428 / PCS 2059  
Inteligência Artificial

Prof. Dr. Jaime Simão Sichman  
Prof. Dra. Anna Helena Reali Costa

Bases de Conhecimento e  
Ontologias

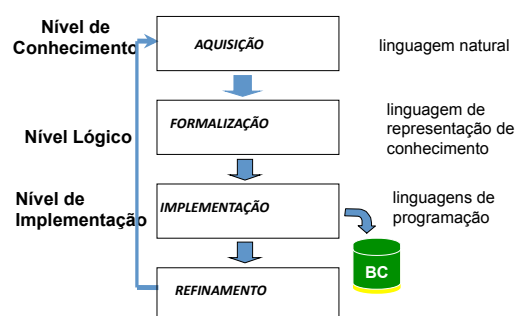
## Construindo Bases de Conhecimento

- Lógica de Primeira Ordem
  - eficiente para representar conhecimento e para raciocinar
  - porém, nada diz sobre quais  *fatos*  devem ser representados e qual  *vocabulário*  utilizar
- Engenharia do Conhecimento
  - estuda  *como*  construir uma  **Base de Conhecimento (BC)**
- Engenheiro do conhecimento
  - determina quais  *conceitos*  são importantes no domínio escolhido
  - cria  *representações*  (em alguma linguagem de RC) desses  *objetos*  e suas  *relações*  no domínio dado

## Propriedades de Bases de Conhecimento

- Toda BC tem dois “consumidores”:
  - usuários
  - procedimentos de inferência
- Uma BC deve:
  - ser clara e correta
  - representar apenas objetos e relações relevantes
  - *idealmente* , ser codificada separada do procedimento de inferência (modularidade, reusabilidade)
  - melhorar a eficiência do processo de inferência
- O processo de inferência deve fornecer a mesma resposta, independente de  *como*  a base foi codificada

## Engenharia de Conhecimento

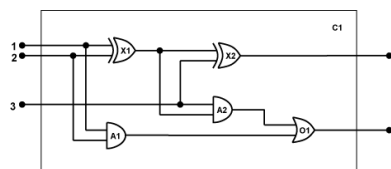


## Engenharia de Conhecimento

- 1) Decida sobre *o que falar*
- 2) Escolha o *vocabulário* de predicados, funções e constantes (**Ontologia do Domínio**)
- 3) Codifique o *conhecimento genérico* sobre o domínio (**axiomas**)
 
$$\forall x,y,z \text{ Americano}(x) \wedge \text{Arma}(y) \wedge \text{Nação}(z) \wedge \text{Hostil}(z) \wedge \text{Vende}(x,z,y) \rightarrow \text{Criminoso}(x)$$
- 4) Codifique uma *descrição* de uma instância específica do problema
 
$$\text{Nação}(\text{Cuba}), \text{Nação}(\text{USA})$$
- 5) Proponha *questões* para o procedimento de inferência e obtenha *respostas*

*West é criminoso?*

## Exemplo: Circuitos Digitais



- **Objetivo:**
  - determinar se o circuito está de acordo com sua especificação (o circuito acima é um *somador*)
  - responder a perguntas sobre o valor da corrente em qualquer ponto do circuito

## Decida sobre o que falar

- Para alcançar o *objetivo*, é relevante falar sobre
  - circuitos, terminais, sinais nos terminais, conexões entre terminais
- Para determinar quais serão esses sinais, precisamos saber sobre:
  - portas e tipos de portas: *AND, OR, XOR* e *NOT*
- Não é relevante falar sobre:
  - fios, caminhos dos fios, cor e tamanho dos fios, etc.

## Decida qual vocabulário usar

- Nomear os objetos e relações do domínio com funções, predicados e constantes
  - **constantes**
    - distinguir as portas : X1, X2...
    - distinguir os tipos de porta: AND, OR, XOR...
  - **funções e predicados**
    - tipo de uma porta:
 
$$\text{Tipo}(X1) = \text{XOR}, \text{Tipo}(X1, \text{XOR}), \text{XOR}(X1)$$
    - indicar entradas e saídas:
 
$$\text{Out}(1, X1), \text{In}(1, X2)$$
    - indicar conectividade entre portas:
 
$$\text{Conectado}(\text{Out}(1, X1), \text{In}(1, X2))$$

### Codifique regras genéricas

- (1) Dois terminais conectados têm o mesmo sinal:  
 $\forall t_1, t_2 \text{ Conectado}(t_1, t_2) \rightarrow \text{Sinal}(t_1) = \text{Sinal}(t_2)$
- (2) O sinal de um terminal é *On* ou *Off* (nunca ambos)  
 $\forall t \text{ Sinal}(t) = \text{On} \vee \text{Sinal}(t) = \text{Off}, \text{On} \neq \text{Off}$
- (3) *Conectado* é um predicado comutativo  
 $\forall t_1, t_2 \text{ Conectado}(t_1, t_2) \Leftrightarrow \text{Conectado}(t_2, t_1)$
- (4) Uma porta *OR* está *On* se qualquer das suas entradas está *On*:  
 $\forall g \text{ Tipo}(g) = \text{OR} \rightarrow \text{Sinal}(\text{Out}(1,g)) = \text{On} \Leftrightarrow \exists n \text{ Sinal}(\text{In}(n,g)) = \text{On}$
- (5) etc...

### Codifique a instância específica

- Portas:  
 $\text{Tipo}(X1) = \text{XOR} \quad \text{Tipo}(X2) = \text{XOR}$   
 $\text{Tipo}(A1) = \text{AND} \quad \text{Tipo}(A2) = \text{AND}$   
 $\text{Tipo}(O1) = \text{OR}$
- Conexões:  
 $\text{Conectado}(\text{Out}(1,X1), \text{In}(1,X2))$   
 $\text{Conectado}(\text{Out}(1,X1), \text{In}(2,A2))$   
 $\text{Conectado}(\text{Out}(1,A2), \text{In}(1,O1)) \dots$

### Formule questões ao Mecanismo de Inferência

- Que entradas causam  $\text{Out}(1,C1) = \text{Off}$  e  $\text{Out}(2,C1) = \text{On}$ ?  
 $\exists i_1, i_2, i_3 \text{ Sinal}(\text{In}(1,C1)) = i_1 \wedge \text{Sinal}(\text{In}(2,C1)) = i_2 \wedge \text{Sinal}(\text{In}(3,C1)) = i_3 \wedge \text{Sinal}(\text{Out}(1,C1)) = \text{Off} \wedge \text{Sinal}(\text{Out}(2,C1)) = \text{On}$
- Resposta:  
 $(i_1 = \text{On} \wedge i_2 = \text{On} \wedge i_3 = \text{Off}) \vee (i_1 = \text{On} \wedge i_2 = \text{Off} \wedge i_3 = \text{On}) \vee (i_1 = \text{Off} \wedge i_2 = \text{On} \wedge i_3 = \text{On})$

### Ontologias

- Em Filosofia:  
 – estudo do que existe no mundo
- Em IA:  
 – especificação das classes, objetos e relações da realidade
- Usadas para:  
 – organizar a BC, separando conceitos (objetos) das regras  
 – aumentar a reusabilidade
- Ontologias Genéricas (Gerais)  
 – usadas para representar grandes conjuntos de objetos e relações  
 – representam *senso comum* - conhecimento enciclopédico  
 – codificadas dentro de algum formalismo (ex. LPO)

## Ontologias Genéricas

- Maior tempo de construção do que ontologias especializadas, porém mais vantajosas:
  - unificam diferentes áreas de conhecimento, porque o raciocínio e a resolução de problemas pode envolver várias áreas simultaneamente.
  - aplicáveis a domínios específicos, com a adição de axiomas específicos desse domínio

→ Senso Comum (ontologia genérica)

→ Domínios Especializados

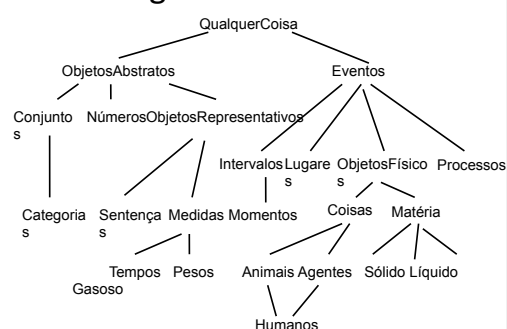
## Estratégias de Construção de Ontologias

- Abordagem *Top-Down*:
  - começa definindo os conceitos do *sensu comum*, depois agrega o conhecimento especializado.
- Abordagem *Bottom-Up*:
  - começa com N bases especializadas, depois integra os conceitos mais gerais.
- Qualquer ontologia específica pode ser modificada a fim de se tornar mais genérica.

## Ontologias Genéricas: Categorias

- Também chamadas de classes, relações, tipos ...
  - conjuntos de objetos com propriedades comuns
  - organiza e simplifica a base de conhecimento.
- Exemplos de simplificação:
  - comprar(Maçã<sub>123</sub>) x comprar(Maçã) - **classe-instância**
  - Todo mamífero bebe leite - **herança**
- Taxonomia:
  - tipo particular de ontologia:
    - relações hierárquicas entre classe e sub-classes em forma de árvores
  - propriedades discriminantes
    - ex. biologia sistemática

## Ontologia do Universo



## Representando Categorias

- Predicados unários:
  - e.g.,  $Humano(x)$  é verdade apenas quando  $x$  for uma pessoa.
- **Reificação (Coisificação):**
  - transforma um predicado ou uma função em um objeto (termo) da linguagem
  - e.g.,  $Humanos$  é um objeto que representa o conjunto de todas as pessoas
  - $População(Humanos) = 9.000.000$

## Exemplos de Categorias

- Um objeto é membro de uma categoria:  
 $Maçã_{12} \in Maçãs$
- Uma categoria pode ser subclasse de outra categoria:  
 $Maçãs \subset Frutas$
- Todos os membros de uma categoria têm alguma propriedade em comum:  
 $\forall x. x \in Maçãs \rightarrow Vermelho(x) \wedge Arredondado(x)$
- Membros de uma categoria podem ser reconhecidos por algumas propriedades:  
 $\forall x. Vermelho(Interior(x)) \wedge Verde(Exterior(x)) \wedge x \in Melões \rightarrow x \in Melancias$

## Relações entre Categorias

- Disjunção: **não há interseção entre as categorias**  
 $\forall s$  Disjunção(s)  $\Leftrightarrow$   
 $(\forall c_1, c_2. c_1 \in s \wedge c_2 \in s \wedge c_1 \neq c_2 \rightarrow c_1 \cap c_2 = \emptyset)$   
ex. Disjunção({Animais, Vegetais})
- Decomposição exaustiva  
 $\forall s, c$  DecomposiçãoExaustiva(s,c)  $\Leftrightarrow$   
 $(\forall i. i \in c \Leftrightarrow \exists c_2. c_2 \in s \wedge i \in c_2)$   
ex. DecomposiçãoExaustiva({Americano, Canadense, Mexicano}, Norte-Americano)
- Partição: **decomposição exaustiva disjunta**  
 $\forall s, c$  Partição(s,c)  $\Leftrightarrow$  Disjunção(s) DecomposiçãoExaustiva(s,c)  
ex.. Partição({macho, fêmea}), animal)

## Ontologias Genéricas

- Medidas
  - Valores atribuídos às propriedades dos objetos do mundo real: peso, comprimento, altura, etc...
- Objetos compostos
  - formados por partes que também são objetos: relação “parte-de”.
- Mudanças com eventos
  - Cálculo de eventos: um fato é verdade em um *intervalo* de tempo.

## Medidas

- Valores atribuídos aos objetos do mundo real: servem para descrever objetos
  - ex. peso, comprimento, altura, diâmetro, ...
- Medidas quantitativas são fáceis de representar
  - ex. Tamanho(L<sub>1</sub>) = Polegadas(1,5) = Centímetros (3,81)
- Medidas qualitativas são mais complicadas
  - ex. beleza de um poema, dificuldade de um exercício
  - O importante é **ordenar**

$$\forall e_1, e_2 \in \text{Exercícios} \wedge e_2 \in \text{Exercícios} \wedge \text{Elabora}(\text{João}, e_1) \wedge \text{Elabora}(\text{Pedro}, e_2) \rightarrow \text{Dificuldade}(e_1) < \text{Dificuldade}(e_2)$$

## Objetos Compostos

- Objetos formados por partes que também são objetos
  - São caracterizados pela estrutura dos objetos que os compõem
  - ex. massa de um carro é a soma das massas de suas partes(carroceria, motor, pneu, ...)
- Para representá-los, usamos a relação *ParteDe*
  - e.g., ParteDe(motor, Carro), ParteDe(pneu, Carro)
- Exemplo:
 
$$\forall a \text{ Bipede}(a) \rightarrow$$

$$\exists I1, I2, b \text{ Perna}(I1) \wedge \text{Perna}(I2) \wedge \text{Corpo}(b) \wedge$$

$$\text{ParteDe}(I1, a) \wedge \text{ParteDe}(I2, a) \wedge \text{ParteDe}(b, a) \wedge$$

$$\text{Ligado}(I1, b) \wedge \text{Ligado}(I2, b) \wedge I1 \neq I2 \wedge$$

$$\forall I3 \text{ Perna}(I3) \wedge \text{ParteDe}(I3, a) \rightarrow (I3 = I1 \vee I3 = I2)$$

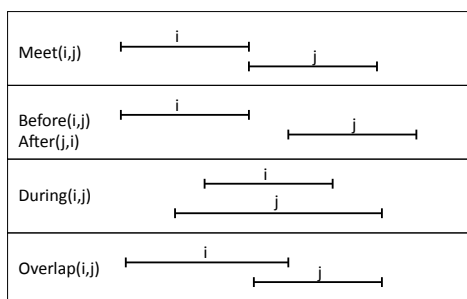
## Objetos Compostos

- *ParteDe* também serve para descrever estrutura de eventos: Script ou Schema
  - ex. comer no restaurante
- Quando se está interessado apenas nas características do conjunto: BunchOf
  - ex. peso do saco de Maçãs
  - BunchOf(Maçãs) define um objeto composto formado pelas Maçãs do saco.

## Representando Mudanças com Eventos

- *Cálculo de situações*:
  - **adequado** quando temos um único agente realizando ações discretas e instantâneas (uma ação por situação).
  - **inadequado** quando:
    - existem vários agentes no mundo.
    - o mundo pode mudar espontaneamente.
    - mudanças ocorrem continuamente.
- *Cálculo de eventos: versão contínua do cálculo de situações*
  - No cálculo de situações, um fato é verdade em uma **situação**
  - No cálculo de eventos, uma coisa é verdade num **intervalo de tempo**
    - ex. SubEvento(BatalhaDaNormandia, SegundaGuerraMundial)
    - SubEvento(SegundaGuerraMundial, SéculoXX)

## Tempos, Intervalos e Ações



## Cálculo de Eventos: Exemplos

- Se duas pessoas ficaram noivas, então em algum intervalo no futuro elas irão se casar ou acabar o noivado:

$$\forall x,y,i_0 T(\text{Noivado}(x,y),i_0) \rightarrow$$

$$\exists i_1 (\text{Encontra}(i_0,i_1) \cup \text{Depois}(i_1,i_0)) \wedge$$

$$T(\text{Casamento}(x,y) \cup \text{FimNoivado}(x,y), i_1)$$

- A segunda guerra de Canudos ocorreu na Bahia no século XIX

$$\exists g g \in \text{Guerras} \wedge \text{SubEvento}(g,\text{SéculoXIX}) \wedge \text{ParteDe}(\text{Localização}(g),\text{Bahia})$$

## Referências Bibliográficas

- S. Russel and P. Norvig. Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River, USA. 2<sup>nd</sup>. Edition, 2003. Chapter 8 and 9.
- G. Bittencourt. Inteligência Artificial: Ferramentas e Teorias. Editora da UFSC, Florianópolis. 2a. Edição, 2001. Cap. 3.