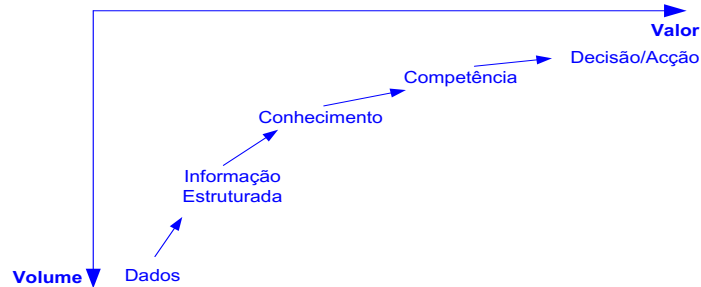


Conhecimento

Termo abstracto usado para capturar a compreensão de um indivíduo num domínio específico.

↪ área de conhecimento bem delimitada, focalizada.



Existem várias teorias que explicam como se organiza o conhecimento humano na resolução de problemas

Representação de Conhecimento

Vários Tipos de Conhecimento

1

1

Tipos de Conhecimento

- Conhecimento Procedimental
- Conhecimento Declarativo
- Meta-Conhecimento
- Conhecimento Heurístico
- Conhecimento Estrutural

Representação do Conhecimento

Método usado na codificação do conhecimento contido na Base de Conhecimento do Sistema Pericial

Não existe uma representação única ideal para todos os tipos de conhecimento

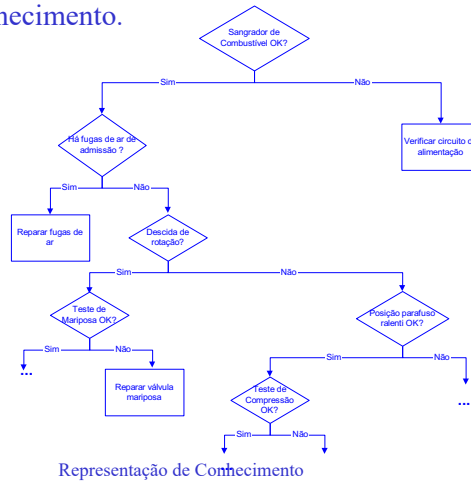
Representação de Conhecimento

2

2

Conhecimento Procedimental

- Descreve *como* um problema é resolvido ou como agir perante uma dada situação (*como fazer*).
- Regras, estratégias, agendas e procedimentos são representações típicas para este tipo de conhecimento.



3

3

Conhecimento Declarativo

- Descreve o que é conhecido acerca de um determinado problema. Inclui declarações (statements) que são assumidas como verdadeiras ou falsas e que descrevem um objecto ou conceito.
- Corresponde a uma representação descritiva.

Exemplo

Fumar pode provocar cancro no pulmão.

- Conhecimento Declarativo
 - é mais transparente – mais facilmente entendido, mais fácil de manter
- Representações procedimentais
 - são mais eficientes, mas mais difíceis de manter

Representação de Conhecimento

4

4

Meta-Conhecimento

- Conhecimento acerca do próprio conhecimento.
- É usado para aceder a conhecimento mais orientado para resolver determinado problema
- Aumenta a eficiência de resolução do problema dirigindo o raciocínio para o subconjunto de conhecimento mais adequado
- Representado através de **meta-regras** – regras que descrevem como usar outras regras

Exemplo

Se o carro não pega
E o sistema eléctrico está operacional
Então usar regras relativas ao circuito de alimentação

Representação de Conhecimento

5

5

Conhecimento Heurístico

- Reflete o conhecimento obtido com toda a experiência que se detém ao lidar com um determinado tipo de problema
- É obtido pela experiência prévia na resolução de um grande número de problemas de uma determinada especialidade, é essencialmente empírico
- Muitas vezes assume o aspecto de regras de bom senso ou de “*Rules of Thumb*”

Exemplo

Para elaborar horários considerando salas devemos começar com as salas que impõem mais restrições;
Numa máquina de pintura fazer as mudanças de tintas sempre das mais claras para as mais escuras.

Representação de Conhecimento

6

6

Conhecimento Estrutural

Descreve a estruturação do conhecimento, ou seja, o modelo mental que o perito tem na resolução de um determinado tipo de problema. Pode indicar conceitos e sub-conceitos na estruturação do conhecimento.

Exemplo com as anomalias no Betão

Tipos de Anomalias no Betão

Fissura

Excesso de Ar

Excesso de Água

Grande grau de Silicato Tricálcio no Cimento

Factores Climatéricos

Temperatura Inconstante

Chocho

Má vibração

Falta de Inertes Finos

Cofragem não estanque

Excesso de Inertes Grossos

Representação de Conhecimento

7

7

Tipos de Conhecimento

Conhecimento Procedimental	Regras Estratégias Agendas Procedimentos
Conhecimento Declarativo	Conceitos Objectos Factos
Metaconhecimento	Conhecimento acerca de outros tipos de conhecimento e como usá-lo
Conhecimento Heurístico	Regras de <i>bom-senso</i>
Conhecimento Estrutural	Conjuntos de Regras Relações entre Conceitos Relações entre Objectos

Representação de Conhecimento

8

8

Conhecimento do Perito

- não é geralmente baseado em definições claras nem em algoritmos precisos
- é composto por teorias de carácter geral, mas também por regras de dedo, estratégias e truques aprendidos com a experiência ...

→ **heurísticas**

... utilizadas para simplificar a resolução de problemas, para identificar situações comuns

- é muito dependente do domínio
- pode estar continuamente sujeito a mudança

→ **Consequências**

Separação explícita entre conhecimento e algoritmos para aplicação do conhecimento

Sistema Pericial = Conhecimento + Inferência

Representação de Conhecimento

9

9

Níveis de Conhecimento

Conhecimento Superficial

- faz uma descrição básica (superficial) do conhecimento

Exemplo

SE o depósito de gasolina está vazio ENTÃO o carro não irá funcionar

- é muito limitado, por exemplo, explicar a alguém os fenómenos que se passam
- não traz valor acrescentado por ser demasiadamente genérico e básico
- é um tipo de conhecimento que tira relações causa-efeito - “caixa preta” (no sentido de não se observar o interior dos sistemas)

Representação de Conhecimento

10

10

Níveis de Conhecimento

Conhecimento Profundo

- considera a estrutura causal e interna de um sistema e contempla a interação entre os componentes desse sistema
- É mais difícil de adquirir, representar e validar
- A representação com redes semânticas ou enquadramentos (frames) é a mais adequada.

Representação de Conhecimento

11

11

Representação do Conhecimento

Necessário formalismos:

- Utilizáveis em computador
- Com forma próxima da do conhecimento do perito
- Que facilitem as operações de recolha, organização, manutenção, validação
.... transparentes

Representação de Conhecimento

12

12

Representação do Conhecimento

Alguns formalismos de representação do conhecimento mais vocacionados para o desenvolvimento de Sistemas Periciais :

- Tripletos Objecto-Atributo-Valor e Listas de Propriedades
- Relações de Classificação e Pertença (IS-A e IS-PART)
- Redes Semânticas
- Enquadramentos (Frames)
- Guiões
- Regras
- Lógica

Representação de Conhecimento

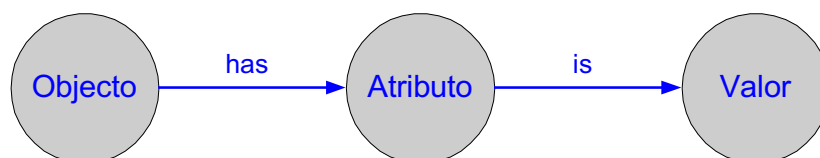
13

13

Tripletos Objecto-Atributo-Valor (O-A-V)

Caracterizam os valores de determinados atributos de um dado objecto

O objecto pode ser uma entidade física (carro) ou uma entidade conceptual (empréstimo)



Representação de Conhecimento

14

14

Tripletos Objecto-Atributo-Valor (O-A-V)

Exemplo

Tripletos associados a um carro

carro-marca-opel
carro-modelo-astra
carro-cilindrada-1400
carro-nºportas-4
carro-côr-verde
...

Os tripletos podem vir afectados de valores numéricos que expressam a certeza, ou incerteza, que se tem no conhecimento em causa.

Exemplo

Previsão do tempo é de chuva com 60% de certeza
(previsão- tempo-chuva – CF = 0.6)

Representação de Conhecimento

15

15

Listas de Propriedades

No exemplo

carro-marca-opel
carro-modelo-astra
carro-cilindrada-1400
carro-nºportas-4
carro-côr-verde
...

O nome do objecto aparece muitas vezes, visto que temos muitos atributos para o mesmo objecto.

Nessas situações podemos usar **listas de propriedades**, nas quais para um dado objecto temos uma lista de pares atributo-valor.

Lista de propriedades para o carro:

carro-[marca-opel, modelo-astra, cilindrada-1400, nºportas-4, côr-verde,...].

Representação de Conhecimento

16

16

Limitações

- Os tripletos e as listas de propriedades têm limitações quando se pretende representar conhecimento declarativo sobre atributos de objectos que estejam em modificação.
- Nessas situações o conhecimento é dinâmico e temos que modificar o valor de um atributo.

Exemplo

tripleto dinâmico que se referem ao estado de um disjuntor (aberto ou fechado) e ao modo de operação de uma linha (manual ou automático):

disjuntor_D - estado-aberto.

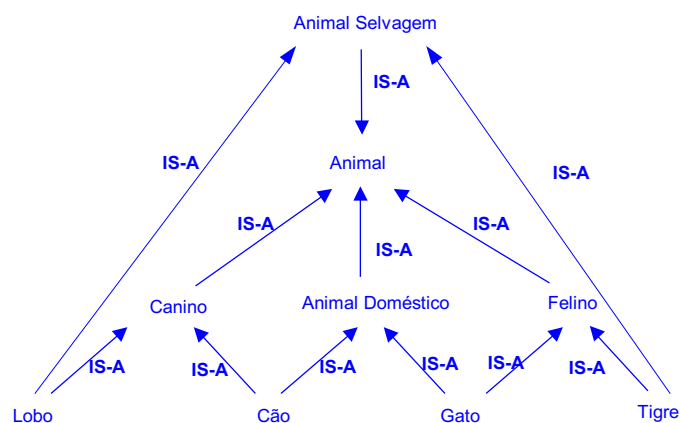
linha_L - modo_operação - manual.

Representação de Conhecimento

17

17

Relações de Classificação (IS-A)



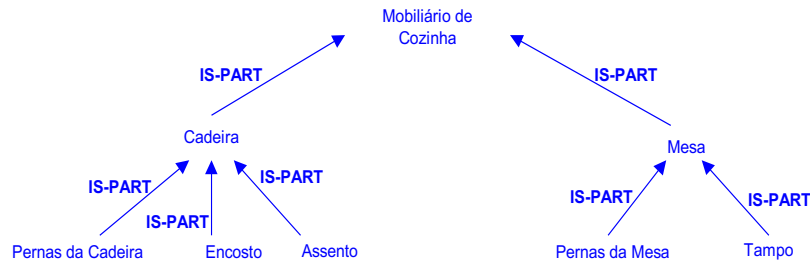
Representação de Conhecimento

18

18

Relações de Pertença (IS-PART)

As relações de pertença (IS-PART) organizam o conhecimento através da composição ou decomposição de componentes.



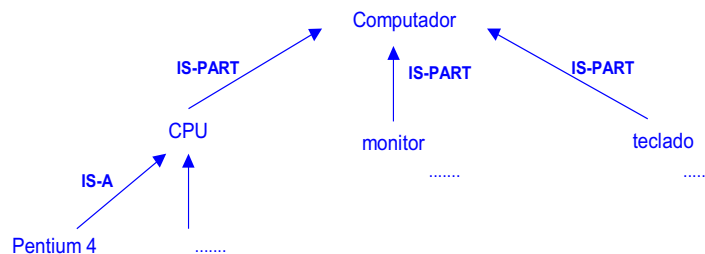
Representação de Conhecimento

19

19

Relações de Classificação (IS-A) e de Pertença (IS-PART)

As relações IS-A e IS-PART podem ser combinadas na mesma representação.



Representação de Conhecimento

20

20

Redes Semânticas

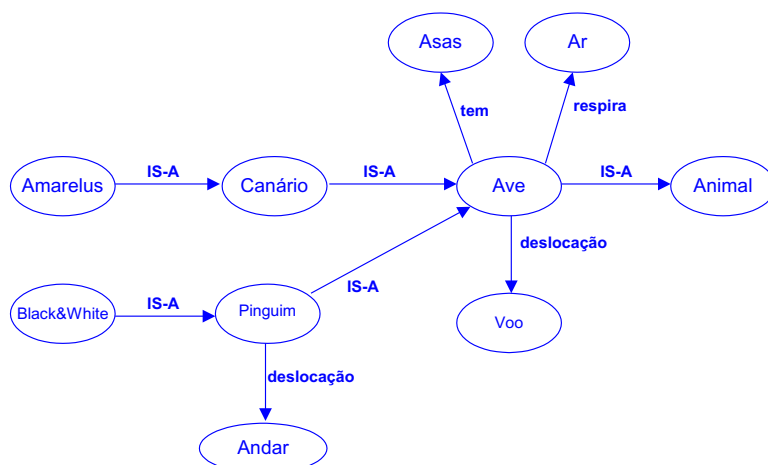
- São um método de representação do conhecimento através de um grafo directo composto por nós e arcos
- Os nós representam objectos (físicos ou abstractos), as suas propriedades e valores
- Os arcos representam relações entre os nós
- As relações IS-A e IS-PART são vulgarmente usadas como etiquetas dos arcos, podem ser usadas outras etiquetas à nossa escolha (tem, desloca-se, respira, etc) que capturam conhecimento

Representação de Conhecimento

21

21

Redes Semânticas



Representação de Conhecimento

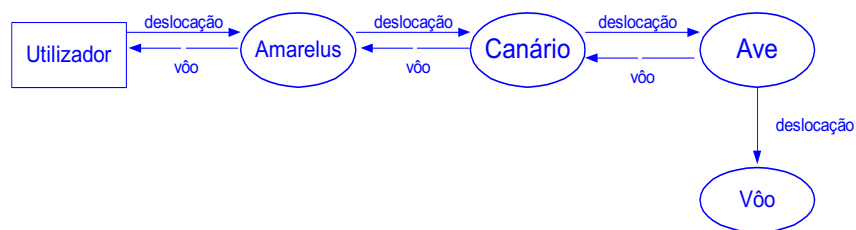
22

22

Modo de Operação

Quando se coloca uma questão a um Nó,
este procura nos seus arcos locais por uma etiqueta que coincida
com a questão
se não existir procura a resposta via as suas ligações IS_A, ou seja,
passa a questão até um Nó que contenha um arco com a resposta.

deslocação Amarelus ?



Representação de Conhecimento

23

23

Inferência sobre Redes Semânticas

Para descrever o processo de inferência nas Redes Semânticas
vamos usar lógica:
Cada ligação



É traduzido para

relação (Obj1,Obj2)

Deslocação Amarelus ?

Is_a (Amarelus,Canário).

Is_a (Canário, Ave).

deslocação(Ave,Voo).

Representação de Conhecimento

24

24

Inferência sobre Redes Semânticas

O tratamento de excepções no mecanismo de herança faz-se impondo uma restrição ao mecanismo de herança

↳ algo dito explicitamente sobrepõe-se aos factos herdados

deslocação Black&White?

```
Is_a (Black&White,Pinguim).  
deslocação(Pinguim,andar).  
Is_a (Pinguim, Ave).  
Is_a (Ave,animal).  
respira(Ave,Ar).
```

↳ herda todas as características da superclasse
excepto aquelas explícitas no próprio Nó

Representação de Conhecimento

25

25

Inferência sobre Redes Semânticas

- Um exemplo em Prolog:

Base de conhecimento:

```
is_a(amarelo,canario).  
is_a(canario,ave).  
is_a('black&white',pinguim).  
deslocacao(pinguim,andar).  
is_a(pinguim,ave).  
deslocacao(ave,voo).  
tem(ave,asas).  
respira(ave,ar).  
is_a(ave,animal).
```

Representação de Conhecimento

26

26

Inferência sobre Redes Semânticas

- Um exemplo em Prolog:

Motor de inferência:

```
inferencia(Relacao, Objecto, Valor) :-  
    Predicado=..[Relacao, Objecto, Valor],  
    call(Predicado), !.  
inferencia(Relacao, Objecto, Valor) :-  
    is_a(Objecto, Objecto1),  
    inferencia(Relacao, Objecto1, Valor).
```

Interrogações:

```
?- inferencia(respira, canario, X).  
X = ar  
?- inferencia(tem, amarelus, X).  
X = asas  
?- inferencia(deslocacao, 'black&white', X).  
X = andar.
```

27

27

Características Redes Semânticas

- simplicidade de representação - devido às características de herança
 - Amarelus herda todas as propriedades de Aves
- as Redes Semânticas estão na origem da Programação Orientada a Objectos
- permitem uma redução no tempo de pesquisa, visto que os nós estão directamente ligados aos nós vizinhos com interesse

Desvantagens:

- podem permitir inferências inválidas
- não têm uma norma de interpretação - a interpretação depende dos programas que a manipulam.

Representação de Conhecimento

28

28

Evolução das redes semânticas

- Evolução na motivação, a partir da modelação cognitiva de processos para a consideração de aspectos computacionais
- Evolução na representação de objectivos, passando-se de uma abordagem “toda a memória humana” para a consideração de certos tipos de conhecimento separadamente
- Semântica das ligações torna-se menos intuitiva mas formalmente melhor definida
- Evolução dos mecanismos de raciocínio de forma a torná-los mais adequados ao tratamento da definição de primitivas
- Possível impacto na WWW

Representação de Conhecimento

29

29

Redes semânticas e a WWW

- Web Semântica
- Tratamento de identificadores WWW (URI's) como nós
- Criação de um repositório contendo descrições acerca dos nós:
 - Meta-dados típicos, tais como autor, data de criação
 - Outros tipos de meta-dados, tais como sumário e conteúdo
- Utilização da rede para recuperar recursos Web com base na sua semântica
 - Padrões W3C têm evoluído com este propósito:
 - RDF (Resource Description Format), sintaxe XML

Representação de Conhecimento

30

30

Ontologia versus Bases de Conhecimento?

- O que é uma Ontologia?
 - Uma especificação formal, explícita de uma conceptualização partilhada
 - Um vocabulário partilhado que pode ser usado para modelar um domínio, ou seja, os objectos e/ou conceitos que existam, as suas propriedades e relações
 - Imposição de um conjunto específico de conceptualizações num domínio de interesse
 - Conhecimento acerca de electrónica digital
 - Conhecimento acerca de electrónica analógica
 - Definições de terminologia
 - e restrições entre termos

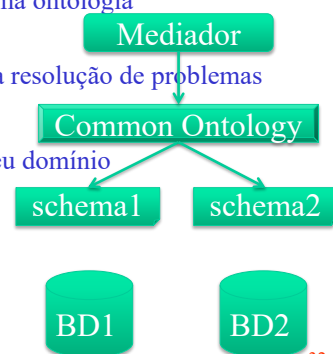
Representação de Conhecimento

31

31

Ontologia versus Bases de Conhecimento?

- É possível pensar numa Ontologia como um tipo de Base de Conhecimento, mas:
 - Uma Ontologia ser propósitos diferentes:
 - precisa apenas de descrever vocabulário, axiomas
 - por exemplo, um esquema de uma BD é uma ontologia
 - Uma Base de Conhecimento inclui:
 - Conhecimento específico necessário para a resolução de problemas
- Motivações
 - Qualquer KBS é baseado num Ontologia do seu domínio
 - Bases de Dados distribuídas
 - Aplicações distribuídas
 - Comunidades de agentes
 - Web semântica



Representação de Conhecimento

32

32

Exemplo

- Uma ave é um animal.
- A maneira normal de movimentação das aves é voar.
- Uma ave está activa durante o dia.
- Um albatroz é uma ave.
- Uma albatroz é preto e branco.
- O tamanho normal do albatroz é 115 cm.
- O Alberto é um albatroz.
- O tamanho do Alberto é 120 cm.
- Um pinguim é uma ave.
- Um pinguim é branco e preto.
- A maneira normal de movimentação dos pinguins é andar.
- O Tweety é um pinguim.

Questões:

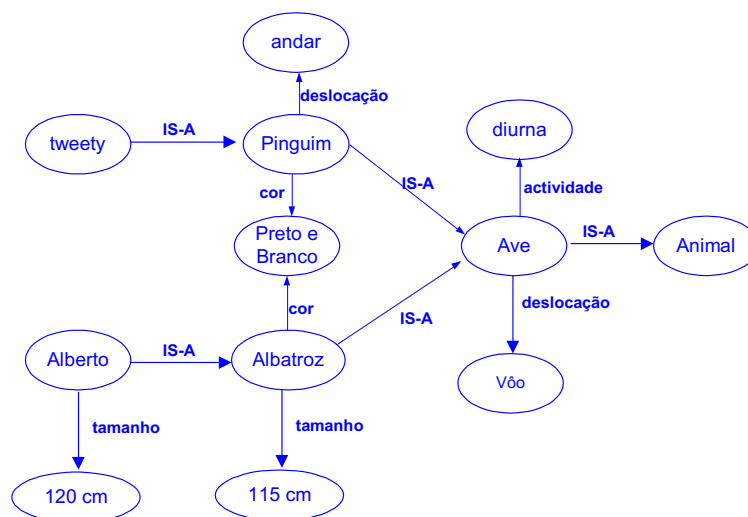
- Qual o método de movimentação do Alberto ?
- Qual o método de movimentação do Tweety ?

Representação de Conhecimento

33

33

Exemplo



Representação de Conhecimento

34

34

Enquadramentos (Frames)

- Introduzido em 1975 por Marvin Minsky
- Permitem representar conhecimento de um conceito ou objecto
- Um enquadramento é uma versão enriquecida de um registo ou de um objecto:
 - Propriedades
 - Herança: características, comportamentos
- É possível criar Instâncias dos Enquadramentos
- Adequados para sistemas complexos, de larga escala, envolvendo valores por defeito e quantidades elevadas de dados conhecidos a-priori

Representação de Conhecimento

35

35

Frames - Campos

- **Identificador**
- **Slots**
 - são representados em gavetas (slots)
 - correspondem aos atributos
 - Cada slot tem
 - identificação
 - valor que toma por defeito (quando nenhum valor foi ainda atribuído)
 - valor actual (espaço onde são guardadas as alterações ao atributo)

Nome do frame		
Nome do slot 1	Valor por defeito - slot 1	Valor actual slot 1
Nome do slot 2	Valor por defeito - slot 2	Valor actual slot 2

Representação de Conhecimento

36

36

Definição de Frames em LPA Flex

```
frame <nome_do_frame>
  [{is a|is an|is a kind of} <frame_pai>] [,<outro_frame_pai>,...] [;]
  [ [ default <slot> {is|is a|are} <valor> ]
  [ and default <outro_slot> {is|is a|are} <outro_valor> [and ... ] ] [;]
  [ inherit <slot> from <nome_do_frame_herda>
    [ and do not inherit <slot> [ and ... ] ] ] ].
```

- nome_do_frame é o nome do *frame* (atômico)
- frame_pai, outro_frame_pai e nome_do_frame_herda são nomes de *frames* (com o mesmo formato) definidos noutro local
- slot e outro_slot são identificações dos *slots* pertencentes ao *frame* nome_do_frame.
- valor e outro_valor são valores por defeito que os *slots* tomam respectivamente.

Representação de Conhecimento

37

37

Relações entre Frames

Permitem implementar características de herança entre Frames

Tipos de Relações

- “is a” ou “is an” : relação de dependência hierárquica entre *frames*
- “inherit” : herança de *slot(s)* de *frames* que não estão na mesma linha de hierarquia
- “do not inherit” : ausência de herança de *slot(s)* de *frame(s)* hierarquicamente superiores
- um *slot* ser ele mesmo um *frame*

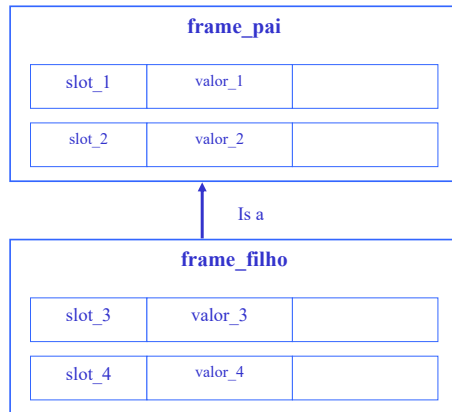
Representação de Conhecimento

38

38

“Is a” Simple

O frame_filho herda todos os *slots* de frame_pai



frame frame_pai
default slot_1 is valor_1 and
default slot_2 is valor_2.

frame frame_filho is a frame_pai
default slot_3 is valor_3 and
default slot_4 is valor_4.

Representação de Conhecimento

39

39

“Is a” Múltiplo

Exemplo



frame mamifero
default pele is pelo and
default habitat is terra and
default movimento are {anda,nada}.

frame felino is a mamifero
default cauda is longa and
default pernas are 4.

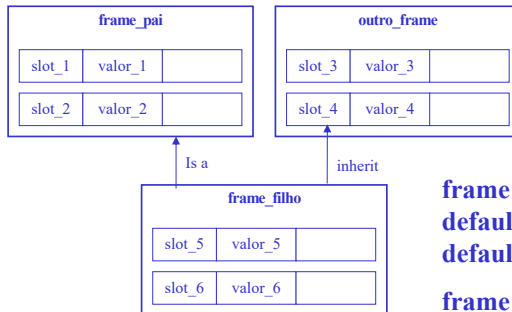
Representação de Conhecimento

40

40

“Inherit”

Herança de *slot(s)* de *frames* que não estão na mesma linha de hierarquia



frame frame_pai
default slot_1 is valor_1 and
default slot_2 is valor_2.

frame outro_frame
default slot_3 is valor_3 and
default slot_4 is valor_4.

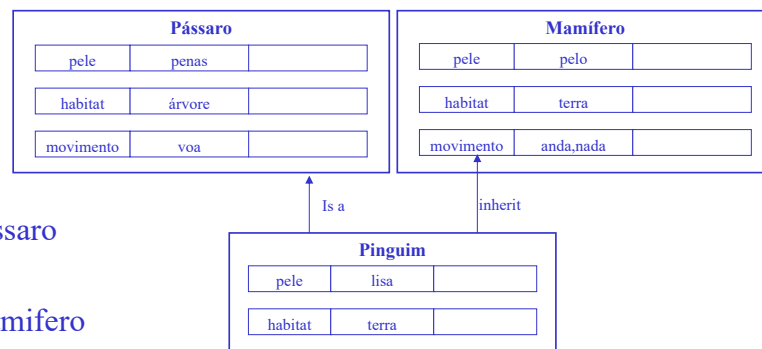
frame frame_filho is a frame_pai
default slot_5 is valor_5 and
default slot_6 is valor_6 ;
inherit slot_4 from outro_frame.

Representação de Conhecimento

41

41

“Inherit” - Exemplo



frame passaro

....

frame mamifero

.....

frame pinguim is a passaro
default pele is lisa and
default habitat is terra ;
inherit movimento from mamifero.

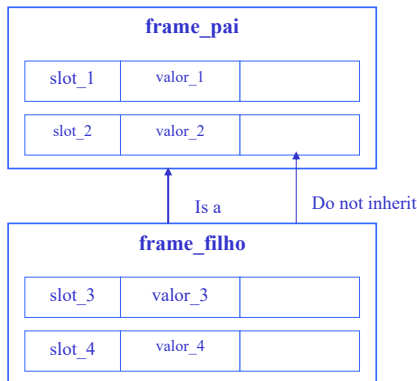
Representação de Conhecimento

42

42

“do not inherit”

Ausência de herança de *slot(s)* de *frame(s)* hierarquicamente superiores



frame frame_pai
default slot_1 is valor_1 and
default slot_2 is valor_2.

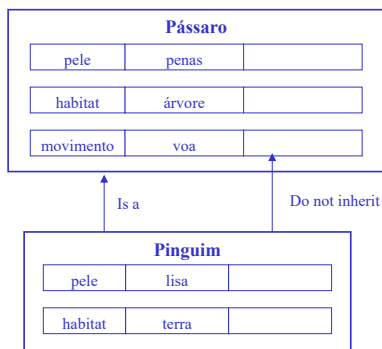
frame frame_filho is a frame_pai
default slot_3 is valor_3 and
default slot_4 is valor_4 ;
do not inherit slot_2.

Representação de Conhecimento

43

43

“do not Inherit” - Exemplo



frame passaro
default pele is penas
default habitat is arvore and
default movimento is voa;

frame pinguim is a passaro
default pele is lisa and
default habitat is terra ;
do not inherit movimento.

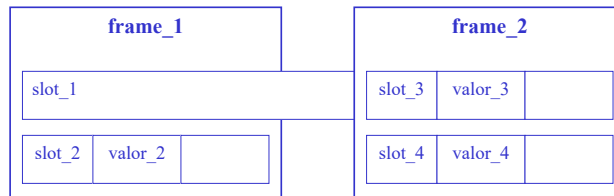
Representação de Conhecimento

44

44

Um *slot* como *Frame*

O slot tem como valor por defeito o nome de um *frame* que pode depois ser acedido como tal



frame frame_1
default slot_1 is frame_2 and
default slot_2 is valor_2.

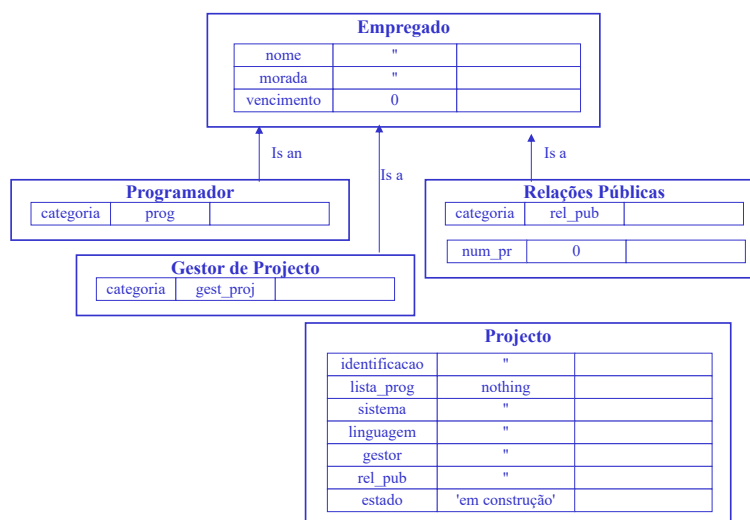
frame frame_2
default slot_3 is valor_3 and
default slot_4 is valor_4.

Representação de Conhecimento

45

45

Exercício Definir em termos de *frames* em LPA-*flex* a estrutura



Representação de Conhecimento

46

46

frame empregado

default nome is " and

default morada is " and

default vencimento is 0.

frame programador is an empregado

default categoria is prog.

frame 'gestor de projecto' is an empregado

default categoria is gest_proj.

frame 'relacoes publicas' is an empregado

default categoria is rel_pub and

default num_pr is 0.

frame projecto

default identificacao is " and

default lista_prog is nothing and

default sistema is " and

default linguagem is " and

default gestor is " and

default rel_pub is " and

default estado is 'em construcao'.

Representação de Conhecimento

47

47

Instância de um Frame

- é um elemento do conjunto que o *frame* representa, ou uma particularização da classe
- podem ser vistas como *frames* cujos *slots* estão restringidos apenas ao valor actual
- as instâncias, contrariamente aos *frames*, só podem ter um pai
- é possível introduzir novos atributos a instâncias

Frame

Nome do Enquadramento	Ave	
Propriedades	Nº Asas	2
	Voa	Verd
	Actividade	Desc

Instância

Nome do Enquadramento	Tweety	
Classe	Ave	
Propriedades	Cor	Branco
	Nº Asas	2
	Voa	Falso
	Actividade	Desc

Representação de Conhecimento

48

48

Definição de Instância

As instâncias podem ser definidas:

- no programa
- em execução (através de ações)

Definição no programa

```
instance <nome_da_instancia>
[ is a | is an | is a kind of | is an instance of <nome_do_frame> ] [;]
[ [ <slot> is|is a|are <valor> ]
[ and <outro_slot> is|is a|are <outro_valor> [and ... ] ] [;]
[ inherit <slot> from <nome_do_frame_herda>
[ and do not inherit <slot> [ and ... ] ] ]].
```

Exemplo

```
instance Tweety is a passaro;
movimento are {voa, nada};
cor is branco&preto.
```

Cada vez que se tenha de criar uma nova instância é necessário:

- acrescentar a instância ao programa
- reinicia-lo → **ineficiente!**

Representação de Conhecimento

49

49

Frames

- facilitam o processamento orientado pelas expectativas, através do uso de *procedimentos geridos por dados*, ficam num estado de espera até serem realmente necessários
- permitem uma boa organização do conhecimento
- são auto-guiáveis, ou seja, eles são capazes, por si só, de determinar quando devem ser aplicados, se não forem aplicáveis podem sugerir outros Enquadramentos que o sejam.
- permitem guardar valores dinâmicos

Desvantagens

- são pouco adequados a novas situações
- a explicitação de conhecimento heurístico, comum em regras, é complexa

Representação de Conhecimento

50

50

Guiões (Scripts)

Especificam uma sequência estereotipada de acontecimentos que normalmente acontecem e que se seguem para representar conhecimento procedimental.

Contêm um conjunto de “slots” dos seguintes tipos:

- **Condições de entrada** – que devem ser atendidas para que os eventos descritos no guião possam ocorrer
- **Resultados** – que irão ser verdadeiros após a ocorrência dos eventos descritos no guião
- **Objectos** – representando objectos envolvidos nos eventos do guião
- **Participantes** – representando entidades que estão envolvidas nos eventos do guião
- **Cenas** – sequências de eventos que ocorrem

São adequados:

- para os casos em que temos sequências tipificadas (por exemplo, as fases da análise de um incidente)
- para descrever planos que devem ser seguidos (por exemplo, os tratamentos a seguir para a cura de uma doença).

Representação de Conhecimento

51

51

Guiões (Scripts)

Participantes: cliente, empregado, dono,...

Objectos: mesa, cadeira, refeição,

Condições de entrada: cliente com apetite e dinheiro, mesa/vaga disponível,...

Condições de saída: cliente satisfeito, cliente com menos dinheiro, dono com mais dinheiro,...

Cena 1: cliente entra no restaurante, aguarda por lugar, senta-se

Cena 2: cliente chama empregado, pede menu, escolhe,...

....

Cena N: cliente chama empregado, pede conta, paga, sai.

Representação de Conhecimento

52

52

Regras

Modelo psicológico do comportamento humano (regras de produção):

Estímulos → **Acções**

É uma forma de conhecimento procedimental

↳ associa informação dada com alguma acção

Neste modelo, um processador cognitivo tenta disparar as regras activadas pelos estímulos adequados

↳ É o papel do **motor de inferência**

Representação de Conhecimento

53

53

Forma Geral

Se Antecedente Então Consequente1
Senão Consequente2

Ou

Se Condição1 E ... E CondiçãoN
Então Conclusão11 E ... E Conclusão 1M
Senão Conclusão21 E ... E Conclusão 2M

As condições do antecedente podem também estar ligadas por OU:

Condição1 E ... E CondiçãoN
OU CondiçãoN1 E ...

Representação de Conhecimento

54

54

Forma Geral

Regra 1

Se SensorA tem valor > 50
Então temperatura da água é muito alta
Senão temperatura da água é normal

Alternativa (de mais fácil validação):

Regra 1a: Se SensorA tem valor > 50
Então temperatura da água é muito alta

Regra 1b: Se SensorA tem valor ≤ 50
Então temperatura da água é normal

Representação de Conhecimento

55

55

Antecedentes Disjuntivos

Se $A = x$ OU $B = y$ Então $C = k$

Pode converter-se em

Se $A = x$ Então $C = k$

Se $B = y$ Então $C = k$

Vantagem

Mais fácil o acompanhamento da inferência

Representação de Conhecimento

56

56

Forma Geral

Formato das regras

regra Identificador : se LHS então RHS!

Exemplo

Regra r1: SE Bot_1=actuado E Bot_2=actuado ENTÃO Sistema_A=activado

LHS **RHS**

Representação de Conhecimento

57

57

Regras

LHS (Left-Handed Side)

- contempla as condições que terão que ser atendidas para que a regra seja aplicável
- pode envolver a conjunção, disjunção e negação de termos
- os termos podem ser factos básicos ou termos gerados pelas conclusões de outras regras - hipóteses ou conclusões intermédias

Representação de Conhecimento

58

58

Regras

RHS (Right-Handed Side)

- corresponde às conclusões ou acções que se podem obter caso as condições sejam verdadeiras
- alguns sistemas só permitem uma conclusão por regra (Cláusulas de Horn)
- outros permitem mais do que uma conclusão
- as conclusões podem ser intermédias, se entrarem no LHS de outras regras (também recebendo o nome de hipóteses), ou conclusões definitivas

Representação de Conhecimento

59

59

Regras vs. Triplos OAV

Regra 1a: Se SensorA tem valor > 50

Então temperatura da água é muito alta

	Se	Então
Objecto	Sensor A	Água
Atributo	tem valor	temperatura
Valor	> 50	muito alta

Representação de Conhecimento

60

60

Regras vs. Triplos OAV

- As condições e as conclusões serão genericamente designadas por **átomos**
- Os átomos podem ser:
 - Triplos Objecto|Atributo|Valor
 - Pares Atributo/Valor (objectos implícitos)

Representação de Conhecimento

61

61

Pares Atributo/Valor vs. Triplos OAV

Se classificação > 14
Então admitir ao Mestrado

Objectos implícitos

	Se	Então
Atributo	Classificação	Admissão Mestrado
Valor	> 14	Sim

• Melhor

Se classificação do aluno > 14
Então Aluno Estado de Admissão Mestrado = Sim

Objectos explícitos

	Se	Então
Objecto	Aluno	Aluno
Atributo	Classificação	Admissão Mestrado
Valor	> 14	Sim

Representação de Conhecimento

62

62

Tipos de Regras

Relação

Se bateria descarregada
Então o automóvel não arrancará

Recomendação

Se o automóvel não arranca
Então arranjar cabos

Orientação

Se o automóvel não arranca
E sistema de alimentação = Ok
Então verificar sistema eléctrico

Heurística

Se o automóvel não arranca
E carro modelo Ford, 1975
Então Verificar Circuito de Alimentação

Estratégia

Se o automóvel não arranca
Então 1º verificar sistema de alimentação
em seguida o sistema eléctrico

Representação de Conhecimento

63

63

Classificação das Regras quanto à Incerteza

Regras de Validade absoluta

Regra da implicação usada na Lógica
Ex: mortal (X) SE humano (X)
 má (ideia) SE
 mau-humor (patrão) AND
 pedir (patrão, aumento)

Regras de Validade Incerta

Regras com grau de certeza
Ex: Se Inflação Elevada
 Então Taxas de Juro Elevadas FC=0,8
Regras com probabilidades associadas

Representação de Conhecimento

64

64

Regras Variáveis

Se ?X é Empregado
E ?X Idade > 65
Então ?X pode reformar-se.

O motor de inferência processa a memória de trabalho de modo a encontrar factos que verifiquem ambas as condições

João é Empregado.
João Idade = 75.
Maria é Empregado.
Maria Idade = 35.

João pode reformar-se.

Representação de Conhecimento

65

65

Regras Causais vs. Regras de Diagnóstico

Uma regra de Produção numa Base de Conhecimento pode estar em uma das seguintes formas:

- Causas influenciam a verosimilhança de sintomas (efeitos), ou
- Sintomas observados afectam a verosimilhança de todas as suas causas

Regras Causais

R1 SE torniquete funcionou ENTÃO relva húmida (FC)
R2 SE choveu ENTÃO relva húmida (FC)

Regra Tipo Diagnóstico

R3 SE relva húmida ENTÃO choveu (FC)

Representação de Conhecimento

66

66

Regras Causais vs. Regras de Diagnóstico

- A mistura numa mesma Base de Conhecimentos dos dois tipos de Regras (que aliás representam conhecimento verdadeiro) pode ser desastrosa.

No exemplo anterior:

A aplicação da Regra R1 seguida de R3

↳ levaria a concluir que **choveu porque o torniquete funcionou**

- As Bases de Conhecimento devem ser consistentes e incorporar regras apenas um dos tipos de regras
- São preferíveis Regras do tipo Causal porque assim representa-se o modelo teórico de causa a efeito.
- Bases de Conhecimento com Regras Causais

são ditas **Baseadas em Modelos**
Representação de Conhecimento

67

67

Principais Propriedades da Representação de Conhecimento com Regras

Modularidade

Cada regra define uma pequena parte do conhecimento relativamente independente das outras regras

Incrementalidade

Novas regras podem ser acrescentadas à Base de Conhecimento de maneira relativamente independente

Possibilidade de Modificação

Regras podem ser modificadas de maneira relativamente independente

Transparência

Permite explicações de raciocínio (questões Como ? E Porquê ?)

Representação de Conhecimento

68

68

Problemas das Representações por Regras

$\text{Ave}(X) \rightarrow \text{Deslocação}(X, \text{Voar})$

Não está totalmente correcta porque existem excepções

Uma possível solução é a introdução de excepções e o uso da negação por falha

$\text{Ave}(X) \wedge \neg \text{pinguim}(X) \wedge \neg \text{avestruz}(X) \rightarrow \text{Deslocação}(X, \text{Voar})$

↪ Isto torna difícil gerir todas as excepções

Tratamento de excepções através de regras:

- tem a desvantagem de implicar não só acrescentar novas regras (ex. sobre aves que não voam) como também modificar outras regras (regras gerais que descrevem as aves)

Representação de Conhecimento

69

69

Representação de Conhecimento com Regras

Exemplo

Base de Conhecimento para análise de um problema de inundação num andar

Conceptualização

- **Sintomas possíveis**

cozinha Ok / Com água, hall Ok / Com água, WC Ok / Com água

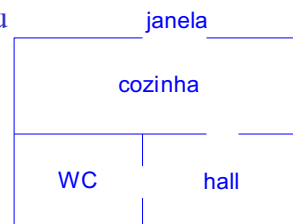
- **Outros parâmetros**

janela aberta/fechada, choveu/não choveu

- **Causas (diagnósticos) possíveis**

choveu

fuga de água na cozinha / no WC



Representação de Conhecimento

70

70

Relacionamento dos sintomas com as causas

- R1:** Se hall com água
E cozinha Ok
Então fuga de água no WC
- R2:** Se hall com água
E WC Ok
Então problema na cozinha
- R3:** Se janela da cozinha fechada
ou não choveu
Então não entrou água de fora
- R4:** Se não entrou água de fora
E problema na cozinha
Então fuga de água na cozinha
- Representação de Conhecimento

71

71

Lógica

Lógica - forma de representação de conhecimento mais usada ao nível de computação.

Exemplo

Na compreensão de texto em Língua Natural é usual transformar frases em representações em lógica

Lógicas mais usuais:

- Lógica Proposicional
- Lógica de Predicados (ou Cálculo de Predicados)

Outras Lógicas:

- Lógica Modal
- Lógica Temporal

Representação de Conhecimento

72

72

Lógica

Em Lógica os símbolos representam conhecimento e os operadores são aplicados aos símbolos para dar origem a raciocínios lógicos

Operador	Símbolos dos operadores
Conjunção (AND)	$\wedge, \&, \cap, .$
Disjunção (OR)	$\vee, \cup, +$
Negação (NOT)	\neg, \sim
Implicação	\rightarrow, \supset
Equivalência	\equiv

Representação de Conhecimento

73

73

Lógica Proposicional

Usa proposições para a representação do conhecimento e raciocínio sobre esse mesmo conhecimento.

Exemplo

Representar através de A a proposição que indica:

hoje está a chover

$\rightarrow A =$ hoje está a chover.

Se quisermos provar a verdade da proposição **hoje está a chover**

É necessário provar a verdade da variável A.

Representação de Conhecimento

74

74

Lógica Proposicional

Tabela de Verdade da Conjunção

A	B	$C = A \wedge B$
F	F	F
F	V	F
V	F	F
V	V	V

Tabela de Verdade da Negação

A	$C = \neg A$
F	V
V	F

Tabela de Verdade da Disjunção

A	B	$C = A \vee B$
F	F	F
F	V	V
V	F	V
V	V	V

Tabela de Verdade da Equivalência

A	B	$A \equiv B$
F	F	V
F	V	F
V	F	F
V	V	V

Representação de Conhecimento

75

75

Lógica Proposicional

Tabela de Verdade da Implicação

$$A \rightarrow B \equiv \neg A \vee B$$

A	B	$C = A \rightarrow B$
F	F	V
F	V	V
V	F	F
V	V	V

Exemplo

*SE a bateria está em baixo
ENTÃO o carro não vai arrancar*

Representação de Conhecimento

76

76

Algumas Equivalências na Lógica Proposicional

Nome da Propriedade	Equivalência
Idempotência	$A \rightarrow B \equiv \neg A \vee B$ $A \wedge \neg A \equiv F$ $A \vee \neg A \equiv V$
Propriedade Comutativa	$A \wedge B \equiv B \wedge A$ $A \vee B \equiv B \vee A$
Propriedade Distributiva	$A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$ $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
Propriedade Associativa	$A \wedge (B \wedge C) \equiv (A \wedge B) \wedge C$ $A \vee (B \vee C) \equiv (A \vee B) \vee C$
Absorção	$A \vee (A \wedge B) \equiv A$ $A \wedge (A \vee B) \equiv A$
DeMorgan	$\neg (A \wedge B) \equiv \neg A \vee \neg B$ $\neg (A \vee B) \equiv \neg A \wedge \neg B$

Representação de Conhecimento

77

77

Limitação da Lógica Proposicional

Dificuldade em expressar conhecimento que necessite de quantificadores (todos, alguns).

Exemplo

“todos os seres humanos são mortais”

e sabendo que

“alguém é um ser humano”

inferir que esse

“alguém é mortal”

$$\forall C [\text{ser_humano}(C) \rightarrow \text{mortal}(C)]$$

Representação de Conhecimento

78

78

Lógica de Predicados

- Baseada no trabalho de Gottfried Frege
- Opera com predicados e argumentos
- As frases estabelecem relações entre objectos - pessoas, entidades físicas, conceitos
- Os predicados têm a ver com as relações
- Os argumentos têm a ver com os objectos
- Inclui os quantificadores:
universal (\forall - qualquer) e existencial (\exists - existe)

Exemplo

Em vez de representar uma proposição através de um símbolo

P = Portugal é um país europeu

fica(portugal, europa) → predicado

Representação de Conhecimento

79

79

Símbolos da Lógica de Predicados

Constantes - usadas para designar objectos ou propriedades dos predicados

Ex. **temperatura, maria, ...**

Predicados - dividem-se em duas partes: o predicado (asserção acerca do objecto) e o argumento (representa o(s) objecto(s) da proposição)

Ex: **gosta** (joão, maria)

Variáveis – usadas para representar classes gerais de objectos ou propriedades

Ex: **gosta** (X, Y)

Funções - permitem relacionar entidades de um conjunto com um elemento único de um outro conjunto.

Ex: pai (joão) = antónio mãe (joana) = maria

amigos(pai (joão), mãe (joana)) = amigos (antónio, maria)

Estes símbolos são manipulados usando os

operadores da Lógica Proposicional

Representação de Conhecimento

80

80

Representação de Frases em Lógica de Predicados

Todas as crianças gostam de gelados

$$\forall C [\text{criança}(C) \rightarrow \text{gosta}(C, \text{gelado})]$$

Há um oceano que banha Portugal

$$\exists O [\text{oceano}(O) \rightarrow \text{banha}(O, \text{portugal})]$$

Algumas aves migram

$$\exists A [\text{ave}(A) \rightarrow \text{migra}(A)]$$

Representação de Conhecimento

81

81

Raciocínio usando a Lógica de Predicados

- Requer a capacidade de inferir conclusões a partir dos factos disponíveis
 - ↪ Através do uso dos operadores do Cálculo de Predicados

Inferência

SE p é verdadeiro

E $(p \rightarrow q)$ também é verdadeiro

ENTÃO q é verdadeiro

Nova informação é **inferida** com base em informação conhecida usando regras e/ou implicações

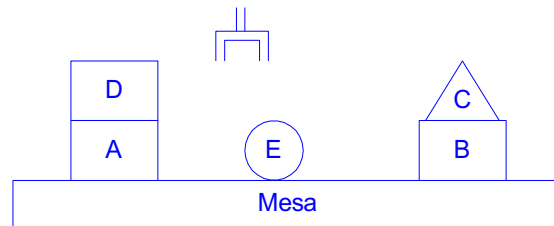
↪ “*modus ponens*”

Representação de Conhecimento

82

82

Exemplo



Objectos

cubo(a) cubo(b) cubo(d) triângulo(c) esfera(e) pega(pega) mesa(mesa)

Relações

em_cima(a, mesa) em_cima(b, mesa) em_cima(e, mesa) em_cima(d, a)
em_cima(c, b) estado(pega, livre)

Escrever as implicações que permitam

- colocar um bloco em cima do outro
- verificar se um bloco está livre
- verificar que blocos estão por debaixo

Representação de Conhecimento

83

83

Exemplo - Blocos

Colocar um bloco em cima do outro

Condições:

Pega tem de segurar bloco a colocar em cima

Tem de haver pelo menos um bloco livre para se colocar bloco em cima

$$\forall x \exists y [\text{pega_segura}(x) \wedge \text{livre}(y) \rightarrow \text{coloca_em_cima}(x, y)]$$

verificar se um bloco está livre

$$\forall x (\neg \exists y \text{ em_cima}(y, x) \rightarrow \text{livre}(x))$$

verificar que blocos estão por debaixo

$$\forall x \forall y (\text{em_cima}(y, x) \rightarrow \text{em_baixo}(x, y))$$

Representação de Conhecimento

84

84