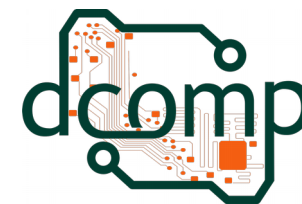




Universidade Federal do Espírito Santo
Centro de Ciências Agrárias – CCA UFES
Departamento de Computação



Sistemas Baseados em Conhecimento – SBC

Inteligência Artificial

Site: <http://jeiks.net>

E-mail: jacsonrcsilva@gmail.com

Sistemas Baseados em Conhecimento

- Tem sido utilizados por mais de 20 anos.
- São um importante avanço tecnológico na resolução computacional de problemas que antes só eram resolvidos por seres humanos.
- Surgiram com a apreciação do **conhecimento específico** do domínio como **requisito** indispensável na resolução de problemas complexos.
- Aproveitam das vantagens computacionais.
- Devem ser utilizados:
 - Quando a formulação genérica do problema a ser resolvido computacionalmente é complexa;
 - Quando existe uma grande quantidade de conhecimento específico do domínio sobre como resolvê-lo.
- Uma indicação para quando devem ser utilizados:
 - Quando existe um especialista humano capaz de solucionar o problema.

Sistemas Baseados em Conhecimento

- Possuem como principais características:
 - Uma Base de Conhecimento:
 - Onde são construídas sentenças em uma linguagem de representação, modelando o problema que se deseja resolver.
 - Um mecanismo de raciocínio capaz de:
 - Realizar inferências sobre esta base e
 - Obter conclusões a partir deste conhecimento
- Construção de um SBC:
 - Deve-se capturar o conhecimento dos membros da organização e organizá-lo, disponibilizado-o na Base de conhecimento.
 - Com a Base construída, esse conhecimento deve:
 - Ser permanentemente acessível,
 - Ser facilmente recuperável,
 - Poder ser amplamente utilizado por todos, independentemente de sua capacitação.

Conceitos e Definições

- Sistemas Baseados em Conhecimento:
 - Usam o conhecimento representado explicitamente para resolver problemas.
 - Manipulam conhecimento e informação de forma inteligente.
 - São desenvolvidos para serem usados em problemas que requerem uma quantidade considerável de conhecimento humano e de especialização.
- Pontos centrais no desenvolvimento de um SBC:
 - Conhecimento;
 - Processo de resolução de problemas.

Conceitos e Definições

- Duas operações utilizadas no processo:
 - Capacidade de raciocínio:
 - Como chegar a certas conclusões (ou gerar um novo conhecimento) interpretando o conhecimento adquirido até o momento.
 - Mas não é suficiente para a resolução adequada de problemas.
 - Deve-se guiar o processo de raciocínio.
 - Método de resolução de problemas:
 - Guiar o processo de raciocínio de maneira que apenas conclusões relevantes ao problema em questão sejam consideradas.
 - Determinar uma sequência de operações de raciocínio que encontre boas soluções.
 - Existem heurísticas aqui, isto é, o conhecimento de regras práticas que auxiliam a reduzir o esforço de busca por soluções.

Conceitos e Definições

- Propriedades:
 - Distinção entre SBCs e sistemas convencionais:
 - Tudo que se sabe sobre o problema deve estar explicitamente representado na Base de Conhecimento do sistema;
 - A Base de Conhecimento deve ser usada por um agente capaz de interpretá-la.
 - A representação necessita ser interpretada para possuir significado.
 - Distinção entre SBCs e sistemas determinísticos:
 - Os problemas resolvidos são aqueles sobre os quais não é conhecido um procedimento determinístico que garanta uma resolução efetiva.

SBCs e Sistemas Convencionais

Sistemas Convencionais	Sistemas Baseados em Conhecimento
Estrutura de Dados	Representação do Conhecimento
Dados e Relações entre Dados	Conceitos, Relações entre Conceitos e Regras
Tipicamente usa algoritmos determinísticos	Busca Heurística
Conhecimento embutido no código do programa	Conhecimento representado explicitamente e separado do programa que o manipula e interpreta
Explicação do raciocínio é difícil	Podem e devem explicar seu raciocínio

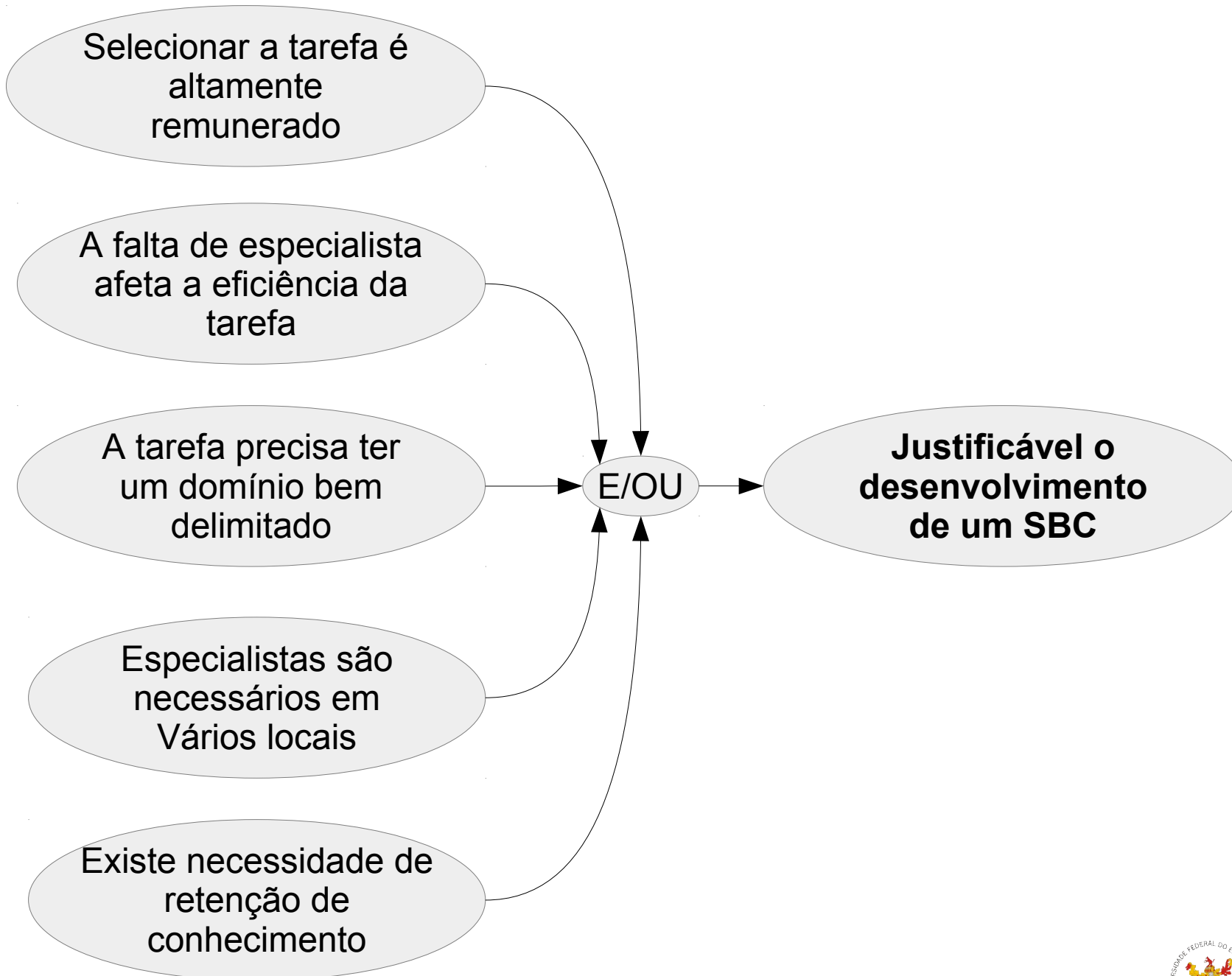
SI, SBC e SE



SBC e SE

- Diferenças entre os SBCs e os Ses:
 - SBCs são sistemas capazes de resolver problemas usando conhecimento específico sobre o domínio da aplicação,
 - Os SE são SBCs que resolvem problemas ordinariamente resolvidos por um especialista humano.
 - Eles requerem conhecimento sobre a habilidade, a experiência e as heurísticas usadas pelo especialista.
 - Além disso, seu processo de desenvolvimento envolve uma profunda interação com o especialista.
- O SBC pode ser classificado como SE quando:
 - Desenvolvimento voltado para aplicações com conhecimento manipulado de um domínio específico.
 - O conhecimento manipulado conta com um alto grau de especialização.
- Na maioria dos trabalhos na área de IA, os termos SBC e SE são usados indistintamente.

Construção de um SBC



Características

- Em sua forma geral, os sistemas especialistas:
 - Possuem suporte para inspecionar seu processo de raciocínio,
 - Tanto para representar os passos intermediários para alcançar a ação,
 - Quanto para responder questões sobre o processo de solução do problema.
 - Permitem modificar de forma fácil suas habilidades,
 - Adicionando e/ou removendo regras de sua base de conhecimento.
 - Possuir inferências heurísticas,
 - Permitindo utilizar seu conhecimento (geralmente imperfeito) para obter soluções úteis.

Explicações do conhecimento

- O conhecimento de um sistema especialista deve permitir sua inspeção, provendo:
 - Informações sobre a solução do problema;
 - Explicações sobre as decisões e escolhas que o programa fez ou está fazendo.
- Explicações são importantes para os humanos,
 - Tanto um médico quando um engenheiro vão necessitar de motivos para acreditar nas respostas do Sistema Especialista.
 - Um humano não vai aceitar a opinião de outro humano ou de uma máquina sem entender as justificativas para isso.

Características

- Demais atributos que um SE pode fornecer:
 - Permitir que os usuários interajam com o sistema (desejável):
 - Interação com sua linguagem natural: português ou inglês, por exemplo.
 - Funcionar com informações incompletas ou incertas:
 - Utilização de técnicas estatísticas, ou
 - Lógica Nebulosa.
 - Apresentar velocidade;
 - Apresentar confiabilidade;
 - Apresentar precisão de suas recomendações (respostas).

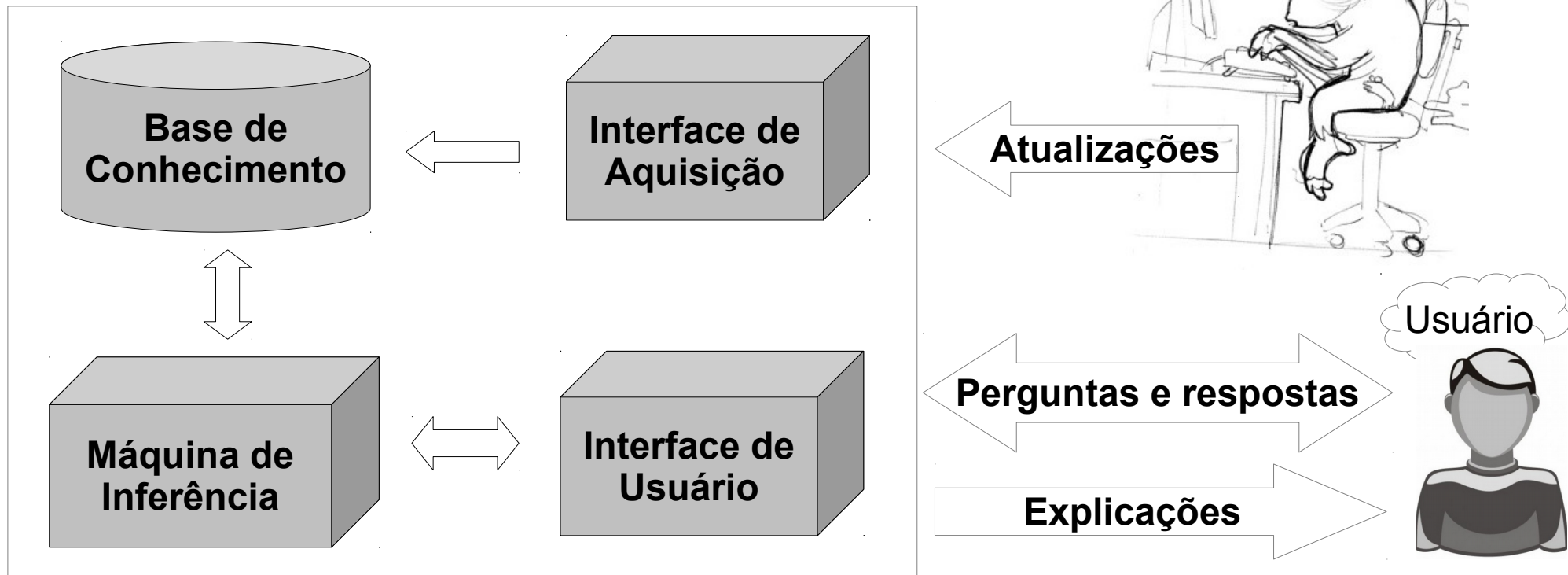
Sistemas Especialistas (SE)

- Categorias comuns dos SE:
 - Interpretação:
 - Obtendo informações de alto nível, extraídas de coletas de dados em seu estado original ou não processados.
 - Prognóstico (predição):
 - Projetando prováveis consequências de determinadas situações.
 - Diagnose:
 - Utilizar os sintomas observáveis para determinar a causa de defeitos/problemas em casos complexos.
 - Esboço (*design*):
 - Encontrar uma configuração de componentes do sistema que atenda os objetivos, satisfazendo um conjunto de restrições de projeto.

Sistemas Especialistas (SE)

- Categorias comuns dos SE:
 - Planejamento:
 - Elaborar uma sequência de ações para atingir uma meta, porém seguindo as condições iniciais e as restrições de tempo.
 - Monitoramento:
 - Comparar o comportamento real de um sistema com o comportamento esperado.
 - Ensino:
 - Auxiliar no processo de educação de domínios técnicos e teóricos.
 - Controle (administração):
 - Dirigir/Administrar o comportamento de um ambiente complexo.

Arquitetura de um SE



Arquitetura de um SE: Base de Conhecimento

- Componente responsável pelo armazenamento do conhecimento.
- Deve usar algum dos modelos de representação:
 - Lógica, Regas de Produção, Redes semânticas, Quadros, etc.
- Os sistemas de Produção costumam ser os mais utilizados, pois possuem:
 - Modularidade:
 - Cada regra define um pequeno e independente pedaço do conhecimento.
 - Permitindo grande facilidade para adicionar novas regras.
 - Regras incorretas podem ser facilmente alteradas ou excluídas.
 - Uniformidade:
 - Todas as regras utilizam um mesmo modelo de representação.
 - Permitindo entender o conteúdo do conhecimento armazenado nas regras.
 - Facilidade de explicar as decisões e soluções obtidas.

Representação do Conhecimento

- Definição:
 - Algo que substitui o objeto ou fenômeno real, de modo a permitir a uma entidade determinar as consequências de um ato pelo pensamento ao invés de sua realização (Davis, Shrobe, & Szolovits, 1983).
- Deve apresentar as seguintes características:
 - ser compreensível ao ser humano, permitindo interpretar o estado de conhecimento do sistema;
 - abstrair-se dos detalhes de como funciona internamente o processador de conhecimento que a interpretará;
 - ser robusta, permitindo sua utilização mesmo que não aborde todas as situações possíveis;
 - ser generalizável, ao contrário do conhecimento em si que é individual.
- Existem várias técnicas de Representação de Conhecimento.
 - Para avaliar essas técnicas existem os critérios:
 - adequação lógica: observa se o formalismo usado é capaz de expressar o conhecimento que se deseja representar;
 - conveniência notacional: verifica as convenções da linguagem de representação. Se essas forem muito complicadas, a tarefa de codificação torna-se extremamente complexa.

Paradigmas de representação de conhecimento

- Técnicas frequentemente usadas
 - Lógica;
 - Sistemas de Produção;
 - Redes Semânticas;
 - Quadros (Frames);
 - Árvores de Decisão.

Representação por Lógica

- Criada por Aristóteles há mais de 2300 anos:
 - Uma linguagem para representar os processos envolvidos no pensamento.
- Trabalha sobre proposições:
 - Simples:
 - p : está chovendo;
 - q : a raiz quadrada de 4 é 2;
 - Compostas:
 - P : está chovendo e ventando;
 - Q : se está chovendo, então a pista está molhada.

Representação por Lógica

- E pode-se aplicar operações lógicas:
 - Negação;
 - Conjunção;
 - Disjunção Inclusiva;
 - Disjunção exclusiva;
 - Condicional *ou* Implicação;
 - Bicondicional *ou* Dupla negação;

Representação por Lógica

- Inferência com proposições:
 - Premissas: p_1, p_2, \dots, p_n, c ;
 - Onde: “c” é a consequência final, obtida com as premissas.
 - Tem-se:

“Regra de inferência” *ou* “Argumento válido”

$$p_1, p_2, \dots, p_n \rightarrow c$$

ou

$$\frac{p_1, p_2, \dots, p_n}{c}$$

Regras de Inferências

- Adição $\frac{a}{a \vee b}$
- Simplificação $\frac{a \wedge b}{a}$
- União $\frac{a, b}{a \wedge b}$
- Absorção $\frac{a \rightarrow b}{a \rightarrow a \wedge b}$
- Modus Ponens $\frac{a \rightarrow b, a}{b}$
- Modus Tollens $\frac{a \rightarrow b, \neg b}{\neg a}$
- Silogismo Disjuntivo $\frac{a \vee b, \neg b}{a}$
- Silogismo Hipotético $\frac{a \rightarrow b, b \rightarrow c}{a \rightarrow c}$

Representação por Lógica

- Lógica de Predicados:
 - Objetos: substantivos (casa, lápis, maria, etc.);
 - Relações: verbos para descrever relações entre objetos;
 - Funções: relações em que existe somente um valor para uma dada entrada;
 - Domínio: um conjunto de objetos;
 - Elementos do domínio: elementos do domínio;
 - Sentenças atômicas. Ex: Pai(Luis, Pedro);
 - Sentenças complexas. Ex: irmão(Luis, Pedro) \wedge casado(Lucas, Maria);
 - Quantificadores: $\forall(x)$ traidor(x) \rightarrow enforcado(x).

Representação por Sistemas de Produção

- Proposto pelo matemático Emil Post em 1943:
 - Para tratar procedimentos computáveis.
- As regras estão no formato:
 - condição \rightarrow ação
 - SE <condição> ENTÃO <ação>
 - Utilizando a regra de inferência “Modus Ponens”

Sistemas de Produção

- Proposto pelo matemático Emil Post em 1943:
 - Demonstrou que um procedimento computável poderia ser modelado como um sistema de produção.
- Pode ser utilizado para:
 - implementar métodos de busca; e
 - para modelar a solução humana de problemas.
- Fornece um controle guiado por padrão de um processo de solução de problemas.
- Consiste em:
 - Um conjunto de *regras de produção*,
 - E uma *memória de trabalho*; e
 - Um ciclo de controle do tipo *reconhece-atua*.

Conjunto de regras de produção

- São chamadas simplesmente de *produções*;
- Cada *produção* define uma porção de conhecimento para a solução de um problema.
- A *produção* é um par condição-ação, onde:
 - A *condição* da regra é um padrão que determina quando a regra pode ser aplicada para um caso do problema;
 - A *ação* define o passo da solução do problema associado.

Memória de trabalho

- Contém uma descrição do *estado atual do mundo* num processo de raciocínio.
- A descrição é um padrão comparado com a condição para selecionar ações apropriadas para resolver o problema.
- Quando o elemento da condição de uma regra casa com o conteúdo da memória de trabalho, a ação associada com esta condição pode ser realizada.
- As ações das regras de produção são projetadas especificamente para alterar o conteúdo da memória de trabalho.

O ciclo reconhece-atua.

Estrutura de controle do sistema

- A *memória de trabalho* é inicializada com a descrição inicial do problema;
- O estado atual da solução do problema é mantido como um *conjunto de padrões* na memória de trabalho;
- Estes padrões são comparados com as condições das regras de produção,
 - É produzido o *conjunto de conflito*, que é um subconjunto de regras de produção.
 - As produções do *conjunto de conflito* estão *habilitadas* neste momento.
- Uma destas soluções no conjunto de conflito é selecionada, ou seja, a solução é *disparada*. Esta ação modifica a memória de trabalho.
- O ciclo de controle então se repete utilizando a memória de trabalho modificada.
- O processo termina quando o conteúdo da memória de trabalho não casa mais com nenhuma condição.

Conflitos

- A *resolução de conflito* escolhe uma regra para ser disparada no *conjunto de conflitos*.
 - Pode ser a escolha da primeira regra;
 - Ou a utilização de heurísticas para a seleção de regras.
 - Aqui é o local do sistema de produção que permite-se adicionar um controle heurístico a um algoritmo de busca.
- O modelo *puro* de sistema de produção finaliza no momento que nenhuma produção é mais encontrada.
 - Porém, muitas implementações práticas permitem regredir até um estado anterior da memória de trabalho.

Exemplo

Conjunto de produção:

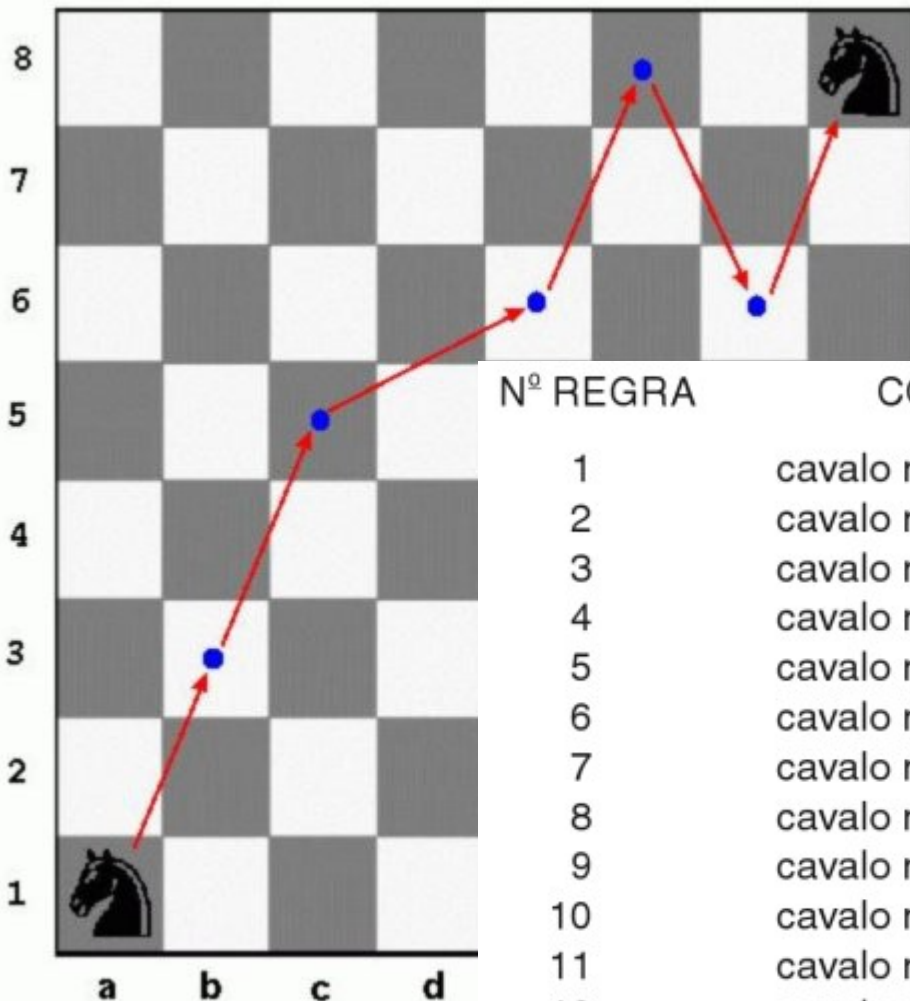
1 $ba \rightarrow ab$

2 $ca \rightarrow ac$

3 $cb \rightarrow bc$

N ^o Iteração	Memória de trabalho	Conjunto de conflito	Regra disparada
0	cbaca	1, 2, 3	1
1	cabca	2	2
2	acbca	2, 3	2
3	acbac	1, 3	1
4	acabc	2	2
5	aacbc	3	3
6	aabcc	∅	Parar

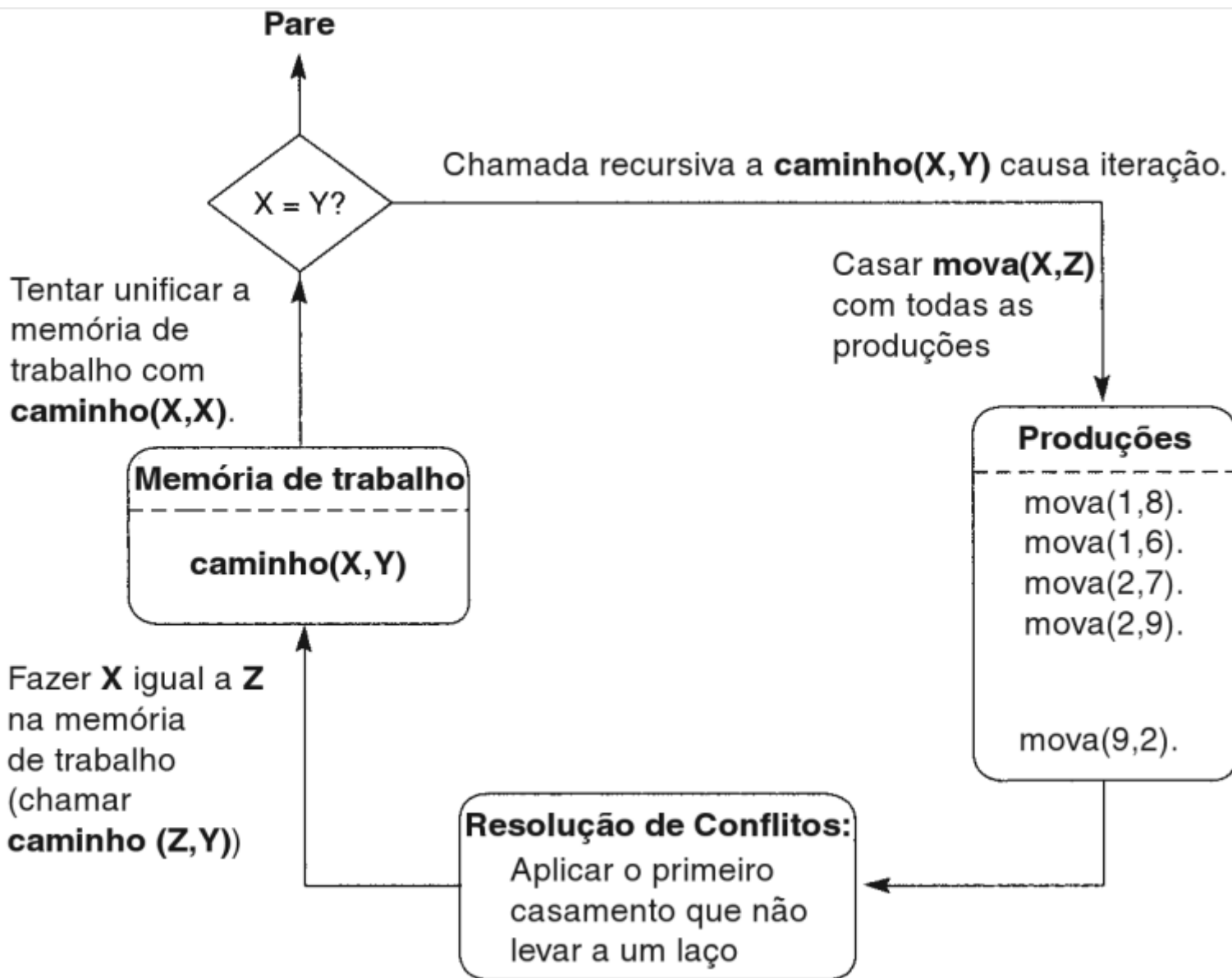
Problema do percurso do cavalo



Mova o cavalo da posição 1 até a posição 2 em um tabuleiro 3x3

Nº REGRA	CONDIÇÃO	AÇÃO
1	cavalo no quadrado 1	→ mova cavalo para quadrado 8
2	cavalo no quadrado 1	→ mova cavalo para quadrado 6
3	cavalo no quadrado 2	→ mova cavalo para quadrado 9
4	cavalo no quadrado 2	→ mova cavalo para quadrado 7
5	cavalo no quadrado 3	→ mova cavalo para quadrado 4
6	cavalo no quadrado 3	→ mova cavalo para quadrado 8
7	cavalo no quadrado 4	→ mova cavalo para quadrado 9
8	cavalo no quadrado 4	→ mova cavalo para quadrado 3
9	cavalo no quadrado 6	→ mova cavalo para quadrado 1
10	cavalo no quadrado 6	→ mova cavalo para quadrado 7
11	cavalo no quadrado 7	→ mova cavalo para quadrado 2
12	cavalo no quadrado 7	→ mova cavalo para quadrado 6
13	cavalo no quadrado 8	→ mova cavalo para quadrado 3
14	cavalo no quadrado 8	→ mova cavalo para quadrado 1
15	cavalo no quadrado 9	→ mova cavalo para quadrado 2
16	cavalo no quadrado 9	→ mova cavalo para quadrado 4

Algoritmo recursivo de caminho



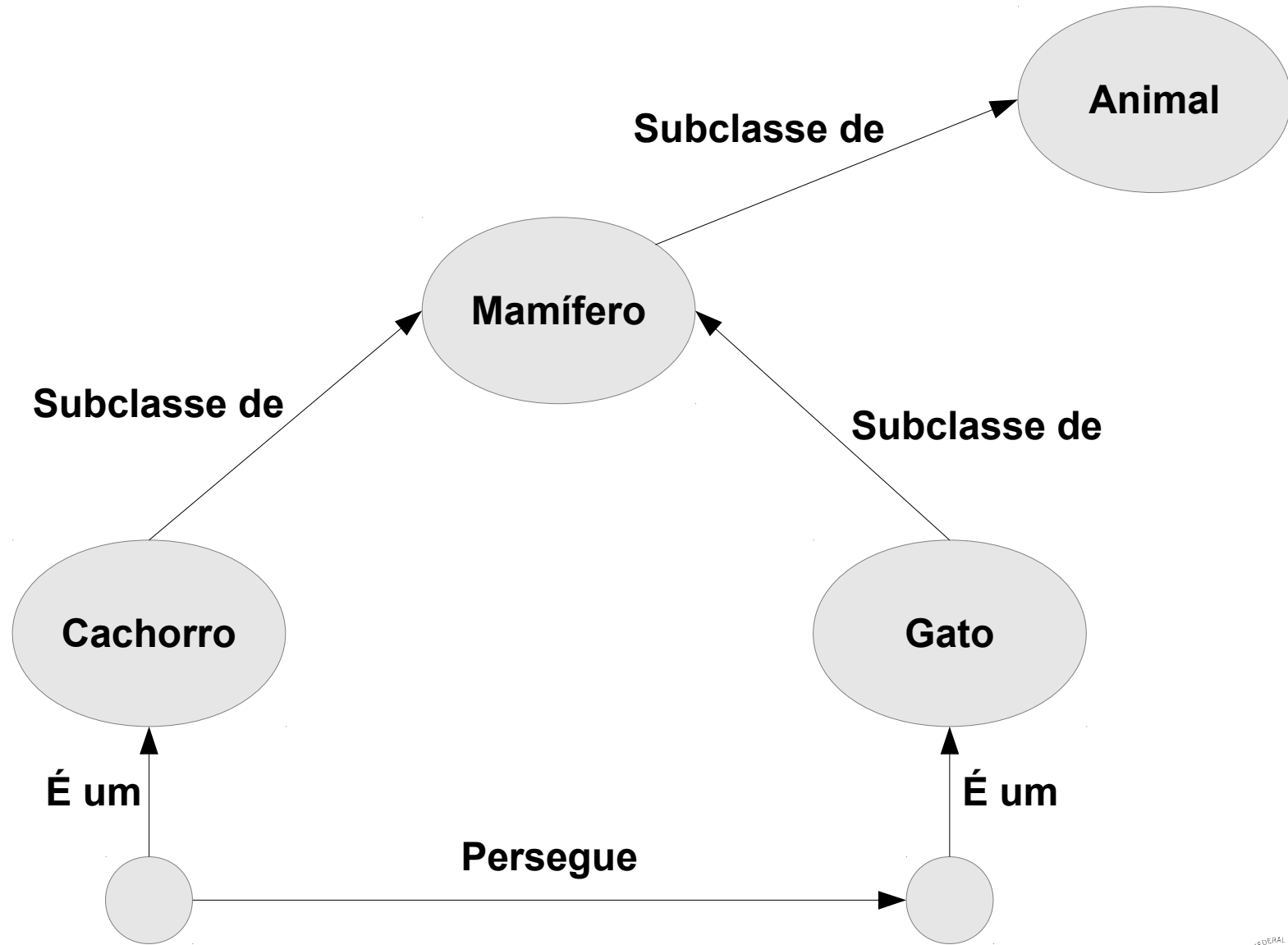
Redes Semânticas

- Consiste em um conjunto de nodos conectados por um conjunto de arcos.
 - Os nodos, em geral, representam objetos.
 - Os arcos representam relações binárias entre os objetos.
- Originalmente foram usadas para suporte a linguagem natural. Em 1968 Ross Quillian as usou para representar modelos psicológicos de memória humana chamado memórias semânticas.
- Quillian desenvolveu um programa que define palavras em inglês de forma similar a dicionários.
 - Em vez de definir palavras formalmente, cada definição simplesmente conduz a outras definições em uma forma desestruturada e, possivelmente circular.
 - Ao procurar uma palavra, percorremos a “rede” até que estejamos satisfeitos com o que compreendemos da palavra original.

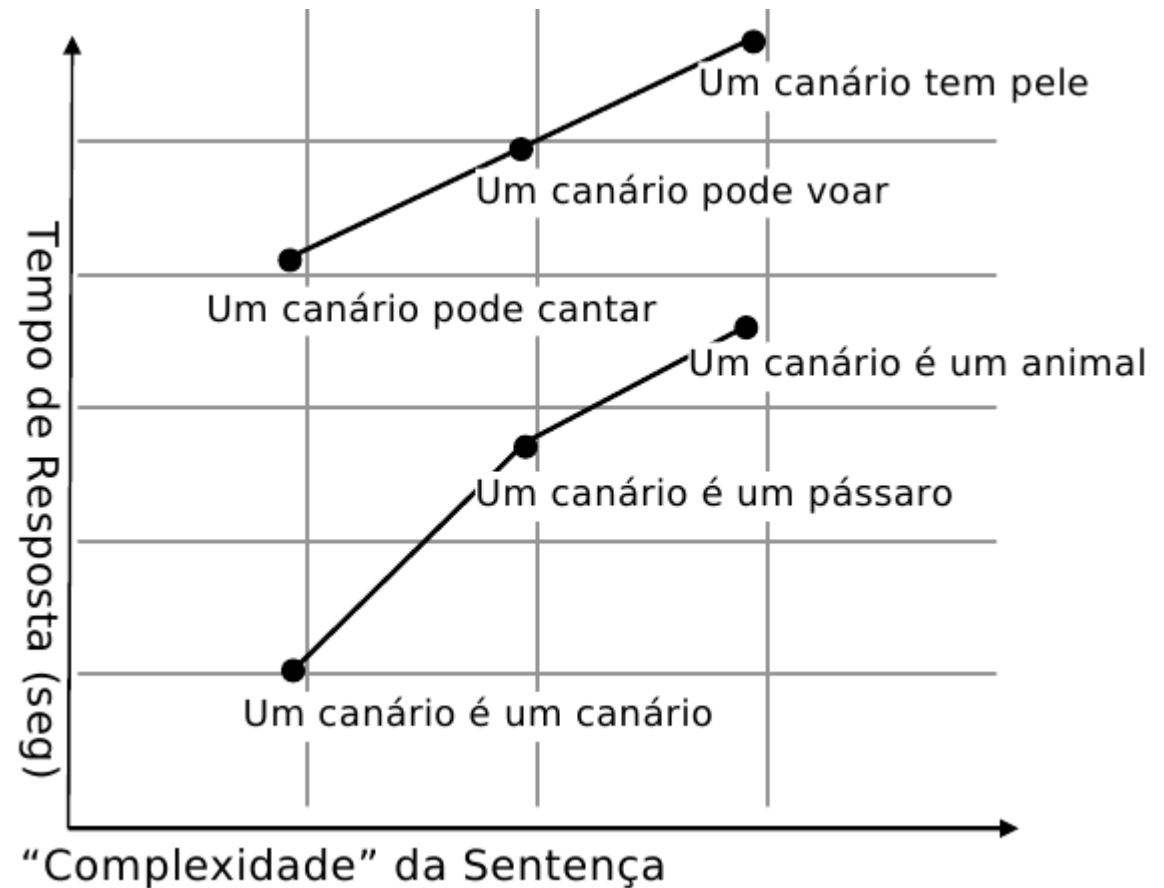
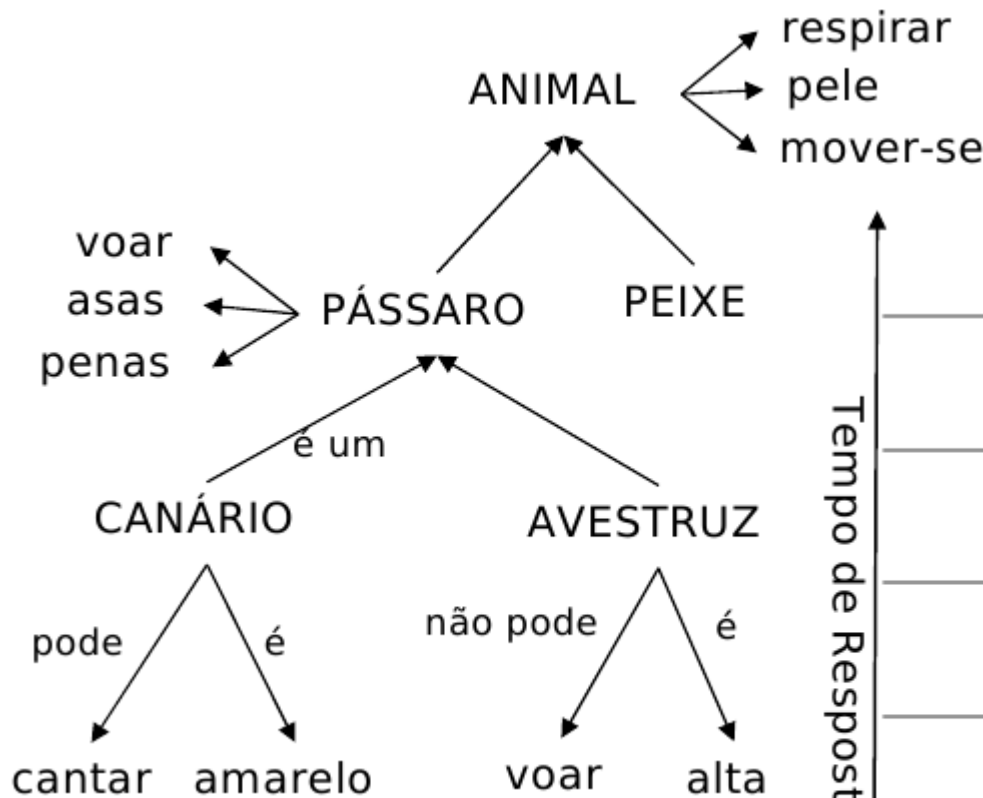
Redes Semânticas

- Redes Semânticas Elementares
 - Usa-se nodos para representar substantivos, adjetivos, pronomes e nomes próprios.
 - Os arcos são reservados basicamente para representar verbos transitivos e preposições.
 - Relações de inclusão entre classes são representadas por relações “subclasse-de”.
 - Os nodos rotulados representam classes genéricas, enquanto que os nodos anônimos representam indivíduos específicos.
 - Para saber se um nodo representa uma instância, é só observar se ele está na origem de algum arco do tipo “é-um”.

Exemplo de rede semântica



Redes Semânticas



Além da habilidade de associar conceitos, os humanos também organizam hierarquicamente o seu conhecimento. A informação é armazenada em níveis apropriados mais altos da taxonomia.

Quadros (*Frames*)

- Os Quadros ou Cenários (*Frames*), e sua variação, os roteiros (*Scripts*)
 - introduzidos para permitir a expressão das estruturas internas dos objetos,
 - mantendo a possibilidade de representar herança de propriedades.
- As pessoas, ao enfrentarem uma nova situação, guardam o repertório do comportamento para situações similares.
 - Ex.: alguém que já assistiu alguma vez a um júri popular sabe que tipo de “quadro” irá encontrar se for a outro.
- As ideias fundamentais foram introduzidas por Marvin Minsky em 1975, no artigo “A framework to represent knowledge”.
- Origina-se nas mesmas ideias das linguagens de programação orientadas a objetos.

Quadros (*Frames*)

- Segundo Minsky (1975):
 - “Quando alguém encontra uma nova situação (ou modifica substancialmente o seu entendimento sobre um problema), recupera da memória uma estrutura chamada ‘frame’. Esta estrutura é um arcabouço memorizado que deve ser adaptado para se adequar à realidade, alterando detalhes, conforme a necessidade”.
- Um quadro consiste em um conjunto de atributos (*slots*) que através de seus valores, descrevem as características do objeto representado pelo quadro.
- Os valores atribuídos aos atributos podem ser
 - valores do objeto em particular,
 - valores *default*,
 - ponteiros para outros quadros (que criam redes de dependências) e
 - conjuntos de regras de procedimento que podem ser implementados.
- Os conjuntos de procedimentos indicam que procedimento deve ser executado quando certas condições forem satisfeitas.

Quadros (*Frames*)

- Os quadros também são organizados em uma hierarquia de especialização, criando outra dimensão de dependência entre eles (herança).
 - Permite assim especificar propriedades de uma classe de objetos através da declaração de herança desta classe à outra.
- O processo de herança e instanciação favorece a reutilização de código.
- São úteis para domínio de problemas onde a forma e o conteúdo do dado desempenham um papel importante na solução do problema.

Exemplo

- Quadro: Cadeira
 - Slot: número de pernas - inteiro (default: 4);
 - Slot: tipo-de-encosto - curvo, reto, não-tem (default: curvo);
 - Slot: tipo-de-assento - redondo, anatômico, reto (default: anatômico);
 - Slot: número-de-braços - 2,1,0 (default: 0);
 - Slot: cor - preta, branca, incolor, azul (default: incolor);

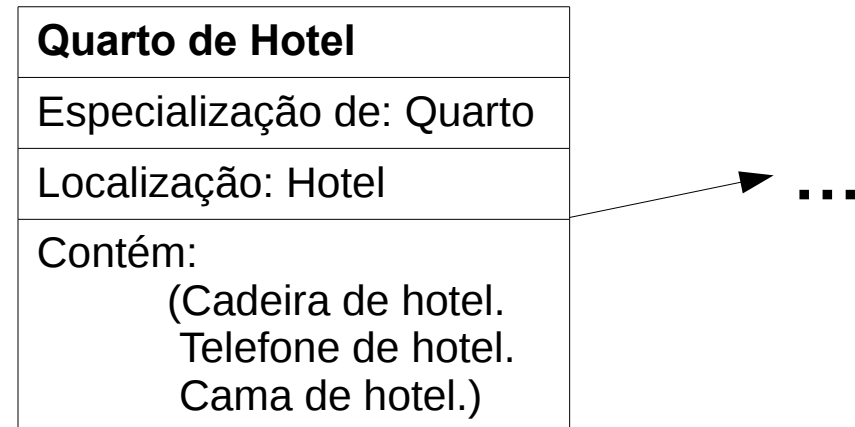
- Quadro: Cadeira-do-Renato

É-UM Cadeira

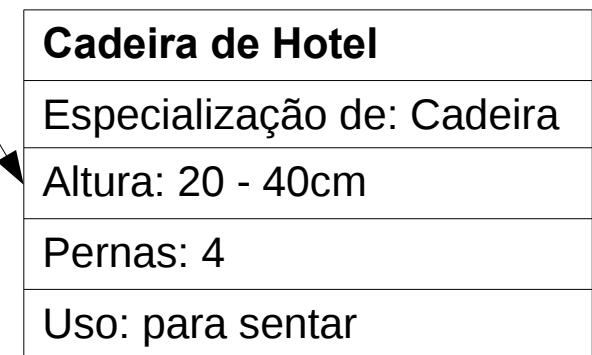
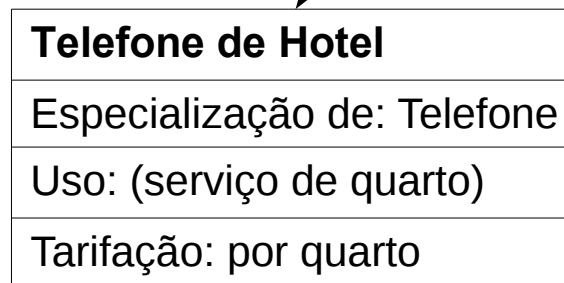
- Slot: número de pernas – 4;
- Slot: tipo-de-encosto – (default: curvo);
- Slot: tipo-de-assento – redondo;
- Slot: número-de-braços – 0;
- Slot: cor – (default: incolor);

Exemplo

- Parte de uma descrição por *frame* de um quarto de hotel.
- Cada *frame* individual pode ser visto como uma estrutura de dados.



- *Slots do frame contém:*
 - Identificação *frame*
 - Relação com outros *frames*
 - Descritores de requisitos (altura do acento)
 - Informação sobre uso
 - Informação *default* (cadeira tem 4 pernas)



Quadros (*Frames*)

- Quadros superam o poder das redes semânticas pois permitem que objetos complexos sejam representados como um único *frame*, em vez de uma grande estrutura de rede.
- Os *frames* tornam mais fácil organizar o conhecimento hierarquicamente.

Arquitetura de um SE: Interfaces de Aquisição e do Usuário

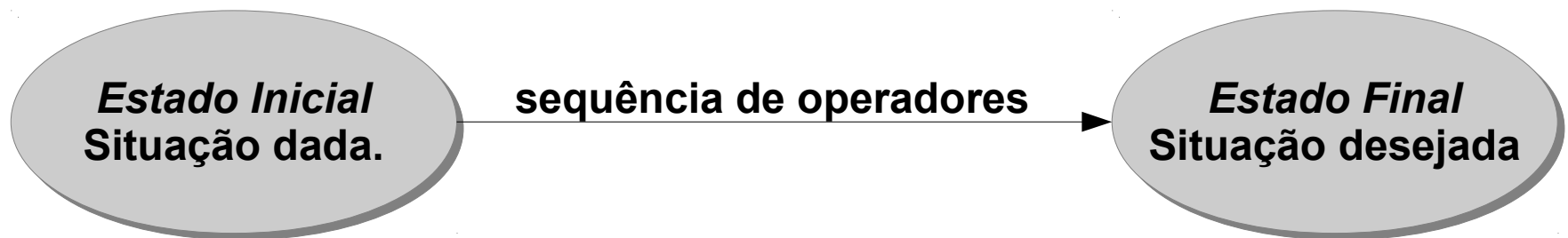
- de Aquisição:
 - Permitir ao especialista definir e manipular regras;
 - Apoiar a construção inicial da Base de Conhecimento;
 - Permitir atualizações:
 - Correções de regras;
 - Adição de novas regras.
- de Usuário:
 - Responsável da interação do usuário com o sistema:
 - Para obter respostas às perguntas; e
 - Explicações referentes de como chegou na conclusão.
 - Responsável com o grau de satisfação do usuário, pois é a interface de comunicação do SE com o usuário.

Arquitetura de um SE: Máquina de Inferência

- Responsável pelo processamento:
 - Das perguntas do usuário;
 - Dos fatos armazenados na base de conhecimento; e
 - Pela obtenção das conclusões e explicações que serão fornecidas ao usuário.
- É capaz de gerar novos conhecimentos, baseando-se:
 - Nos Fatos;
 - Nas Suposições; e
 - Nos Conhecimentos já existentes em sua base.

Arquitetura de um SE: Máquina de Inferência

Através de uma Sequência de Operadores, a Máquina de Inferência torna-se capaz de sair de um *Estado Inicial* e alcançar um *Estado Final*.



**Encontrar uma sequência de operadores que levem
de um *Estado Inicial* para um *Estado Final*.**

Processo de resolução de problemas

Máquina de Inferência

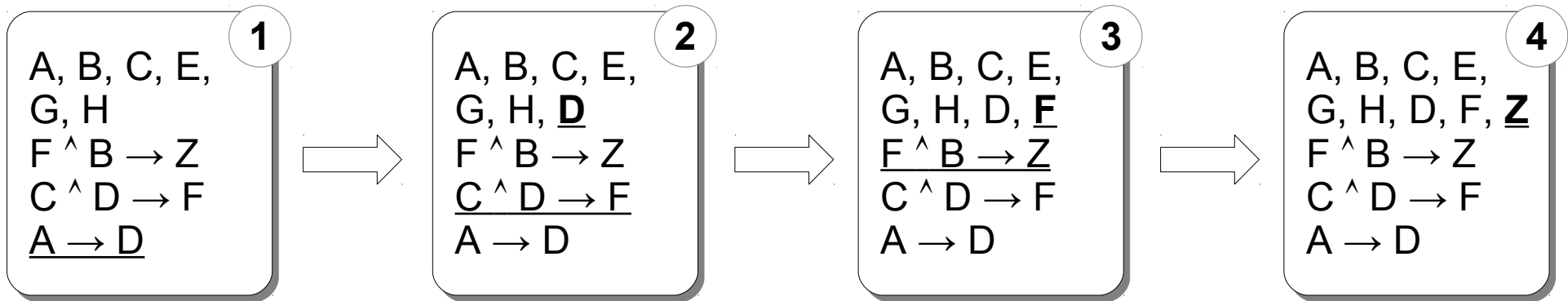
- Seu conhecimento pode ser representado de diferentes formas.
- Exemplos:
 - Com Sistemas de Produção (mais comum):
 - Encadeamento direto;
 - Encadeamento reverso;
 - Redes Semânticas.

Máquina de Inferência

- Sistema de Produção, Encadeamento Direto (Prova Direta):
 1. A memória de trabalho recebe dados sobre o problema;
 2. O processo de inferência deduz outros dados utilizando regras.
Para isso, compara os dados da memória de trabalho com as premissas das regras referentes.
 3. Então, adiciona à memória de trabalho os dados inferidos (conclusão de regras válidas);
 4. Efetua perguntas ao usuário para confirmar as premissas adicionais.
 5. O processo termina quando ocorre a confirmação da regra de conclusão, para que possa ser aceita como resposta final.

Exemplo

Deseja-se provar que Z é verdadeira.



A Memória de Trabalho é representada pela caixa;

O Conjunto de Produção é formado pelas sentenças da caixa;

O Conjunto de Conflito são as sentenças que podem ser inferidas;

A Regra Disparada é a sentença sublinhada;

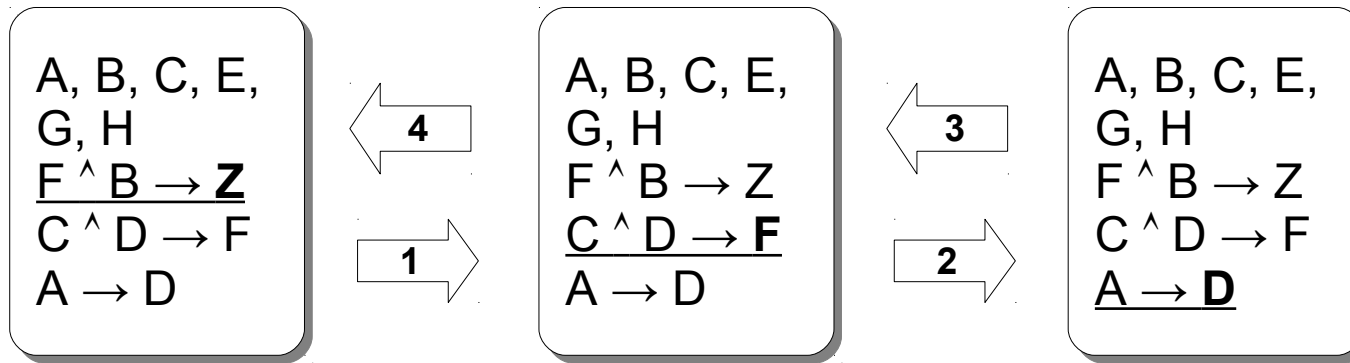
Cada caixa equivale a uma Iteração do sistema.

Máquina de Inferência

- Sistema de Produção, Encadeamento Reverso (Prova Indireta):
 - Utiliza somente as regras que são relevantes a um problema em questão.
 - O processo inicia da conclusão a ser provada, tentando provar a validade de suas premissas.

Exemplo

Deseja-se provar que Z é verdadeira.

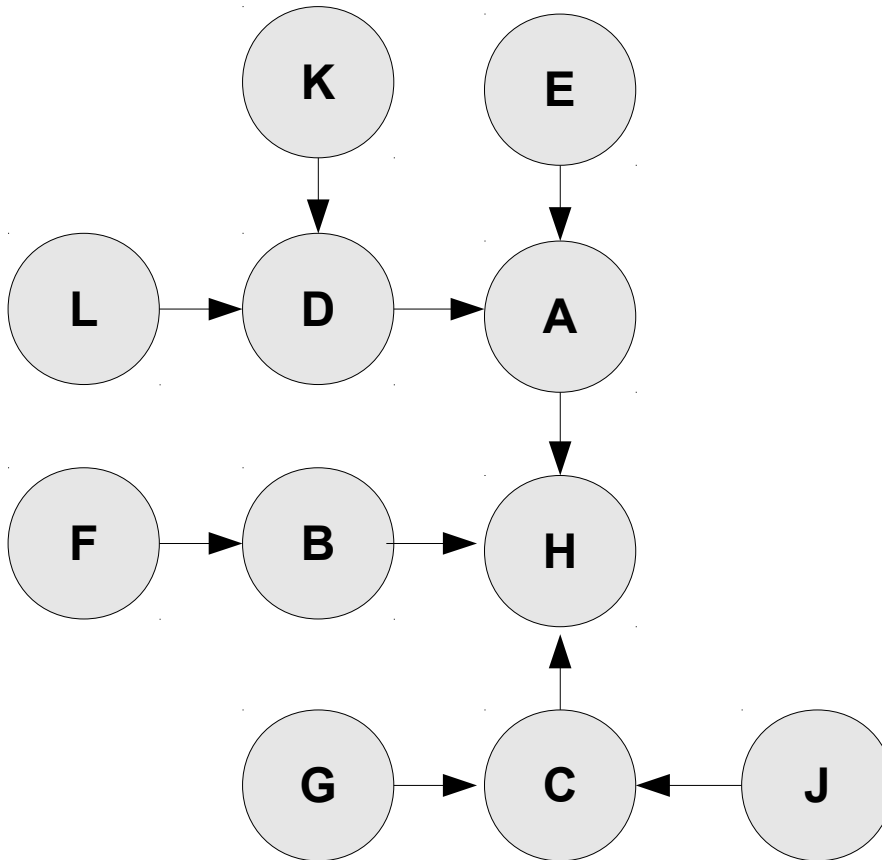


Iterações do Sistema de Produção:

1. Supõe Z , como B já é premissa, ainda precisa de F ;
 2. Supõe F , como C já é premissa, ainda precisa de D ;
 3. Por Modus Ponens, prova-se D ;
 4. Por conjunção e Modus Ponens, prova-se F ;
- Então, como Z necessitava somente de F , chega-se à sua prova.

Máquina de Inferência

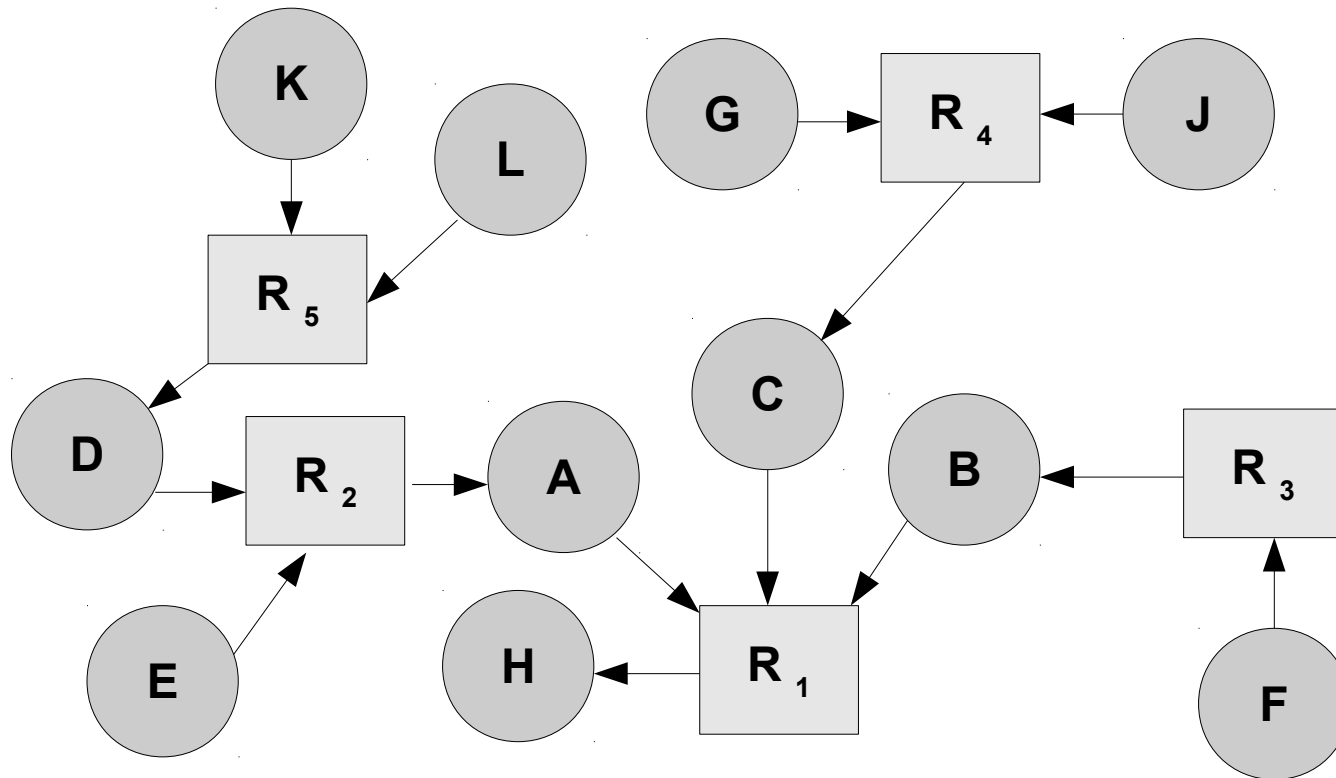
- Redes Semânticas também podem ser utilizadas.
- Exemplo:



Regras	
1	$A \wedge B \wedge C \rightarrow H$
2	$D \wedge E \rightarrow A$
3	$F \rightarrow B$
4	$G \wedge J \rightarrow C$
5	$K \wedge L \rightarrow D$

Exemplo

- Redes Semânticas com as regras incluídas:



Referências

Rezende, Solange Oliveira. *Sistemas inteligentes: fundamentos e aplicações*. Barueri, SP: Manole, 2003. 525 p. ISBN 8520416837.

ARTERO, Almir Olivette. *Inteligência Artificial: Teoria e Prática*. São Paulo: Livraria da Física, 2009.