

Sistemas Baseados em Conhecimento e SHELLS

Huei Diana Lee

Inteligência Artificial
CECE/UNIOESTE-FOZ


Sistemas Baseados em Conhecimento (SBC)

- Programas de computador que usam **conhecimento** representado explicitamente para resolver problemas
- SBCs são desenvolvidos para serem usados em problemas que requerem, para serem resolvidos, uma quantidade considerável de:
 - conhecimento humano
 - perícia



Pontos Importantes no Desenvolvimento de um SBC

- Aquisição do Conhecimento: grande gargalo na construção de um SBC
- Ferramentas de suporte a construção de SBC
 - Linguagens de programação
 - Shells



Sistemas Especialistas (SE)

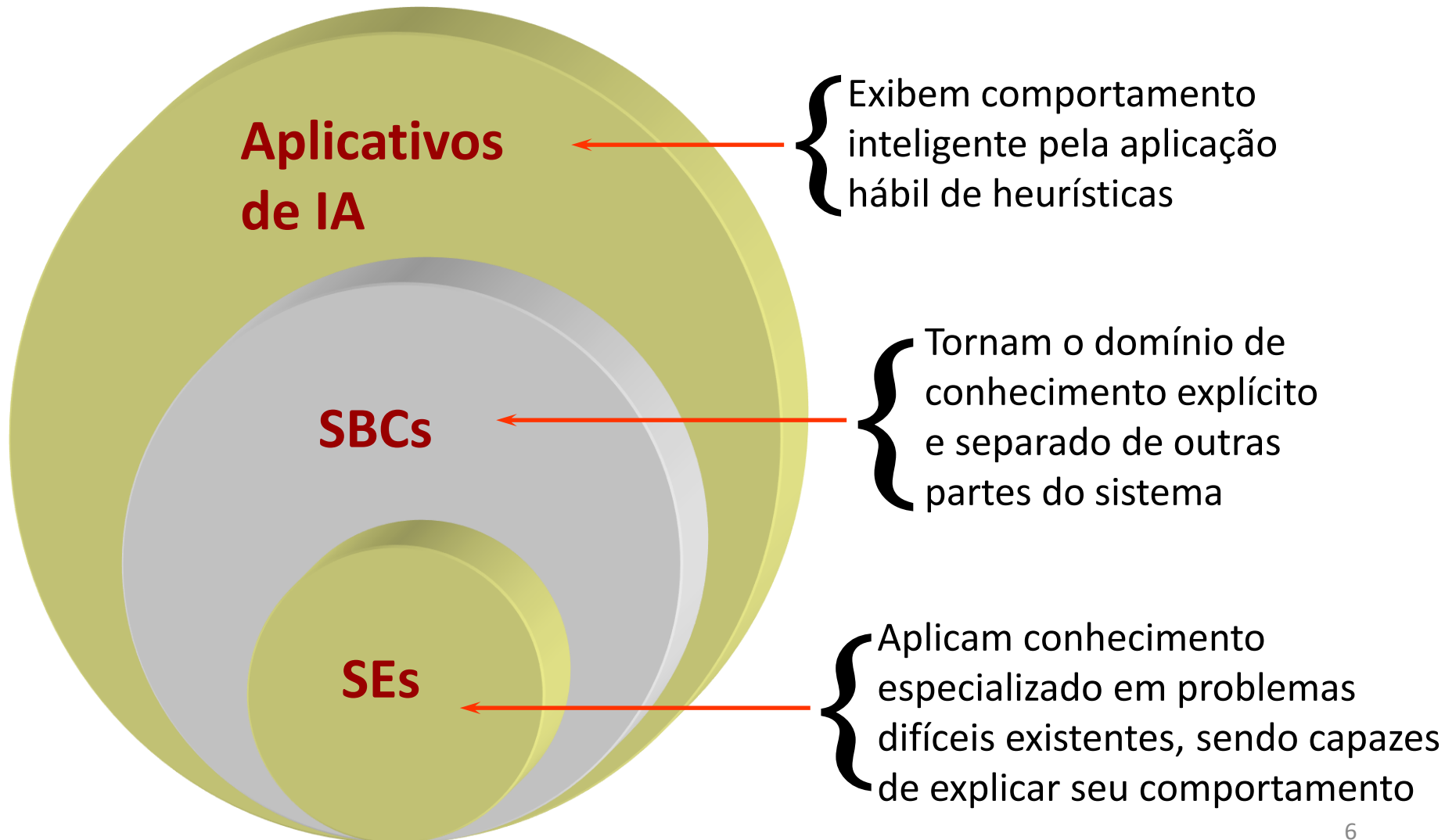
“SE são sistemas que são capazes de oferecer soluções para problemas específicos num dado domínio ou que consigam aconselhar, de uma maneira ou num nível comparável ao de especialistas naquela área.”

*Lucas and van der
Gaag &
Linda van der Gaag
Principles of Expert Systems
(1991)*

Sistemas Convencionais x SBC

Sistemas Convencionais	SBC
Estrutura de dados	Representação de conhecimento
Dados e relações entre os dados	Conceitos, relações entre conceitos e regras
Tipicamente usam algoritmos determinísticos	Podem incluir algoritmos com 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 deve explicar seu raciocínio

IA, SBCs e SEs



- Diagnóstico
- Monitoramento
- Predição
- Planejamento

Áreas de Aplicação

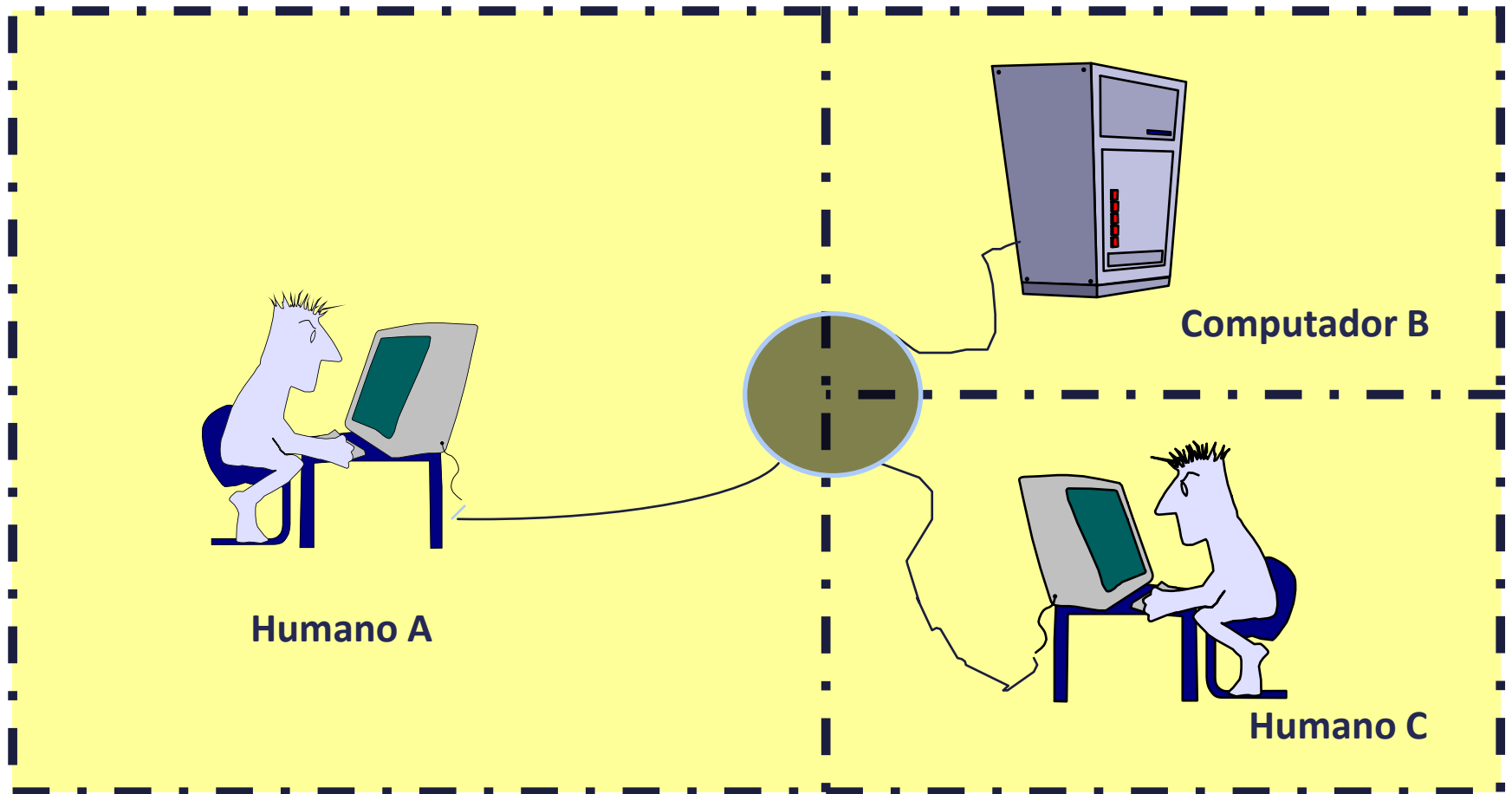
- Projeto
- Depuração
- Reparo
- Instrução
- Controle

Desempenho de SBC

Para fazer com que um Sistema Baseado em Conhecimento chegue perto do desempenho de um especialista humano, o sistema deve:

- ter **grande quantidade de conhecimento** disponível
- conseguir **ter acesso a este conhecimento** rapidamente e
- ser capaz de **raciocinar adequadamente** com este conhecimento

Teste de Turing





SBC

Difere de sistemas
convencionais no modo
como **incorpora o**
conhecimento ao sistema
e na **quantidade**
considerável de
conhecimento humano e
de **perícia** necessários

Componentes Básicos de SBC

- Sistemas Baseados em Conhecimento:
 - Conhecimento + Inferência
- SBC têm dois componentes essenciais:
 - Uma **base de conhecimento** que captura (representa) o conhecimento específico do domínio
 - Um **motor de inferência** consistindo de algoritmos para manipular o conhecimento representado na base de conhecimento

Arquitetura de um SBC

Base de Conhecimento (BC)

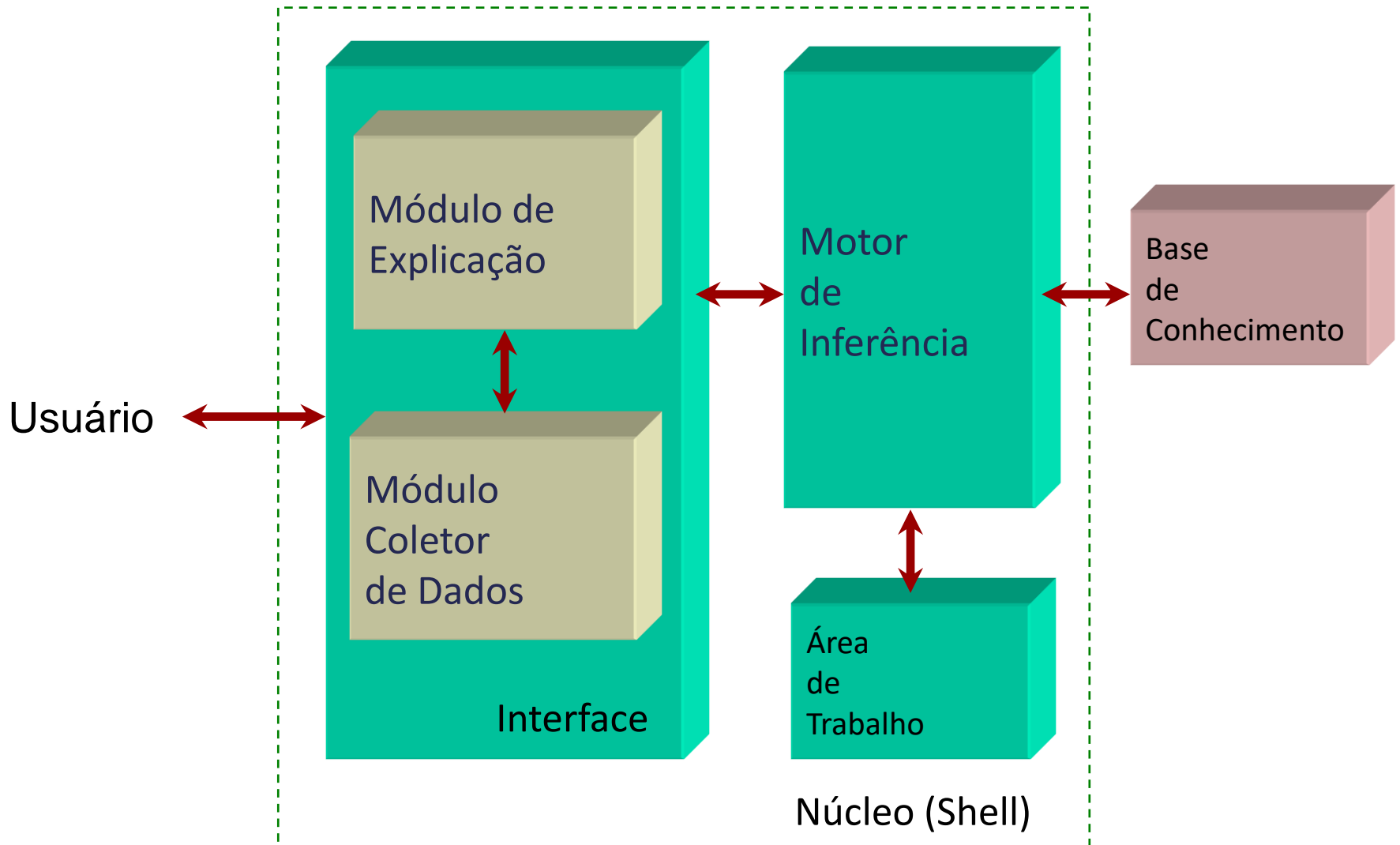
Motor de Inferência (MI)

Interface com usuário

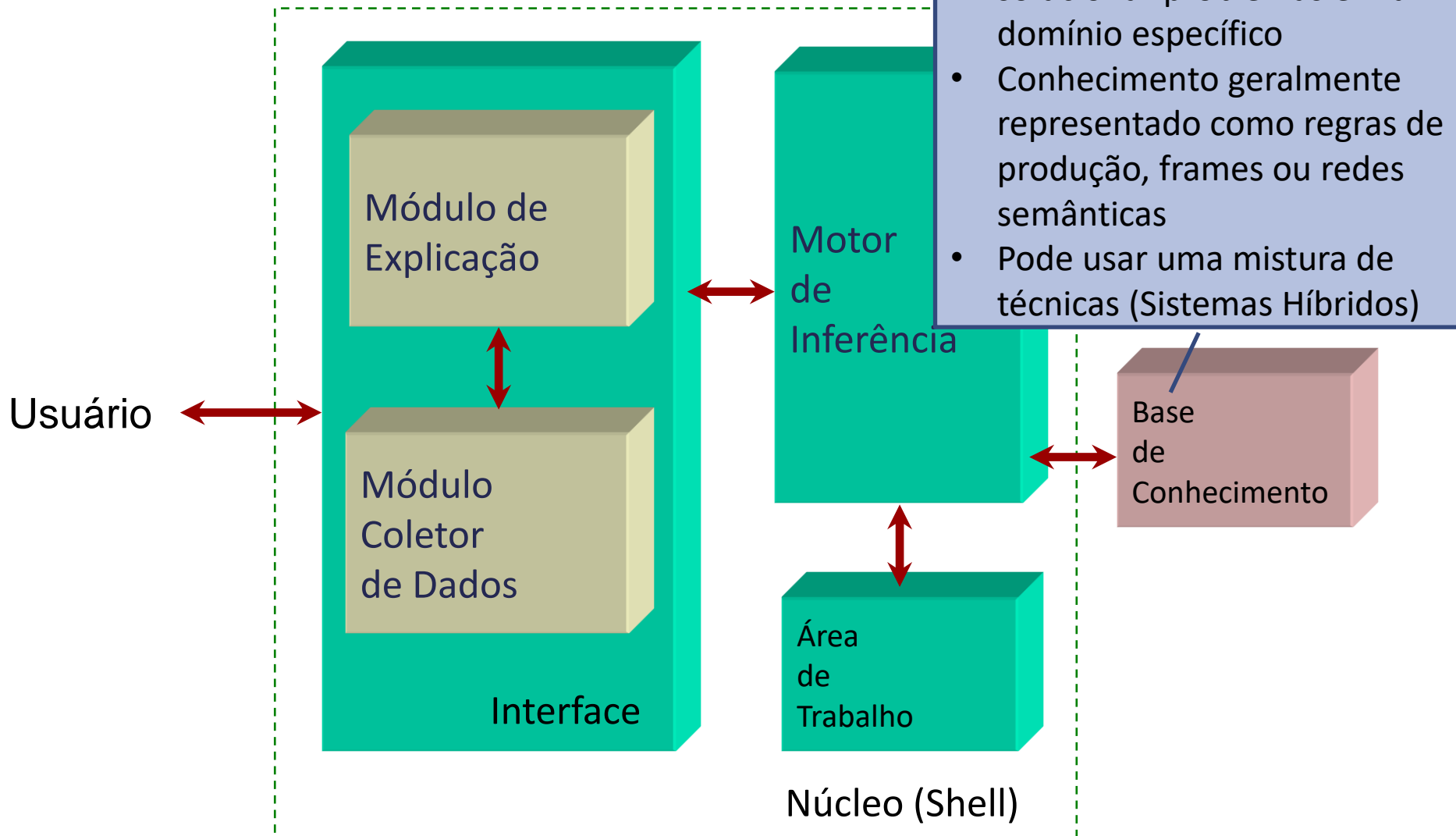
- Módulo Coletor de Dados (MCD)
- Módulo de Explicação (ME)

Área de Trabalho

Arquitetura



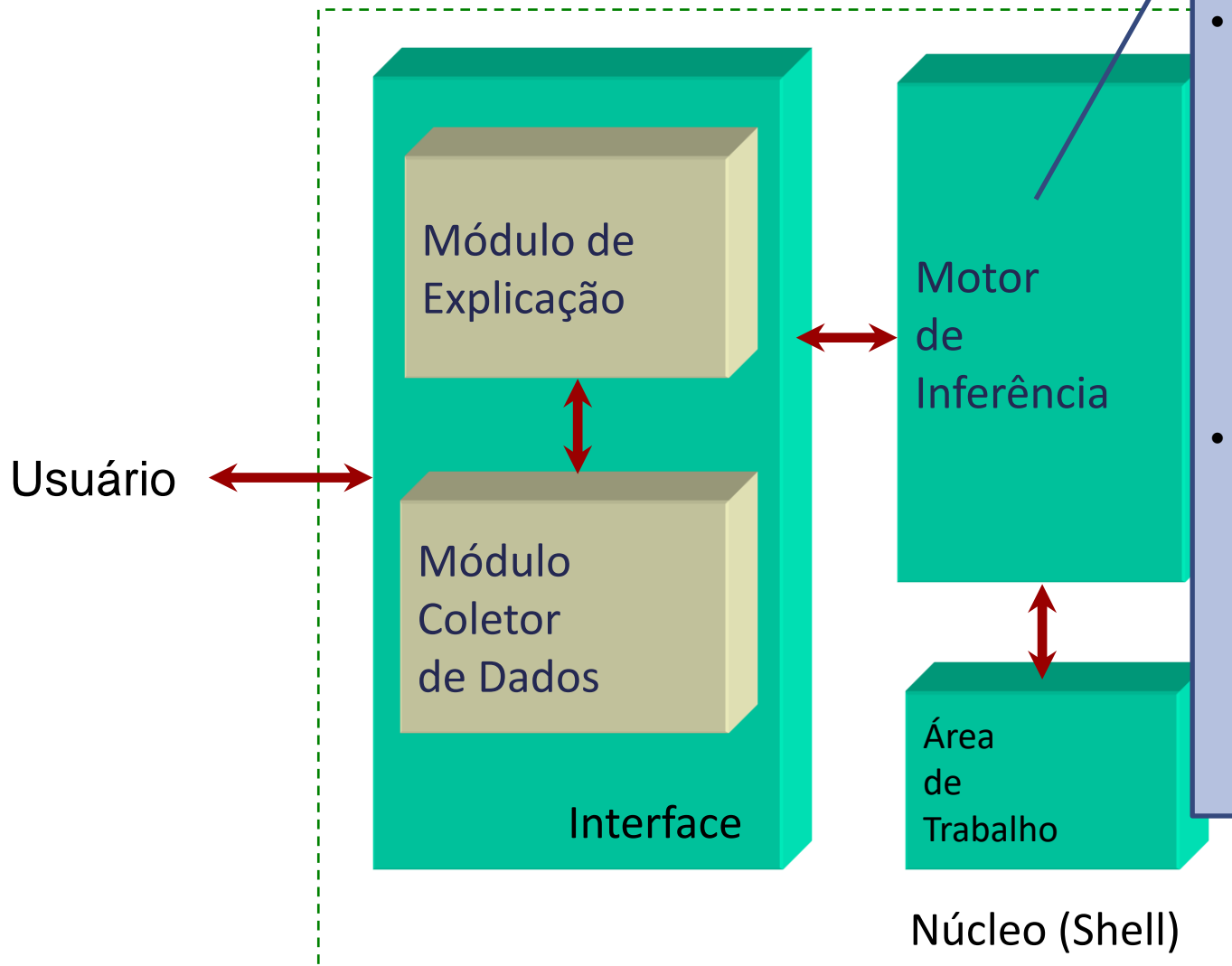
Arquitetura



Certeza do Conhecimento

- Conhecimento também pode incluir probabilidades ou fatores de incerteza para:
 - melhorar o grau de certeza na tomada de decisões
 - ajudar a resolver conflitos e
 - melhorar recursos de explicação
- Técnicas para lidar com incerteza incluem:
 - Método Bayesiano
 - Lógica Fuzzy

Arquitetura



- Responsável por aplicar as estratégias de inferência e controle
- Usa uma linha de raciocínio:
 - Backward chaining ou encadeamento regressivo
 - Forward chaining ou encadeamento progressivo
- Processa informações contidas na BC e Área de Trabalho, tentando encontrar uma solução para o problema no qual está trabalhando

Encadeamento Regressivo (Backward Chaining)

MI utiliza linhas de raciocínio que partem das metas a serem atingidas em direção aos dados

“É possível provar as metas especificadas a partir das informações disponíveis?”

Encadeamento Regressivo (Backward Chaining)

- A direção da busca é das metas para os dados
- Se cond1 e cond2 e ... condn Então ação1



direção da busca

- É possível provar as hipóteses a partir dos dados?

Encadeamento Progressivo (Forward Chaining)

Oposto de backward chaining

“O que é possível concluir, ou seja, quais as metas que podem ser provadas a partir das informações disponíveis?”

Encadeamento Progressivo (Forward Chaining)

- A direção da busca é dos dados para as metas ou hipóteses
- Se cond1 e cond2 e .. condn Então ação1



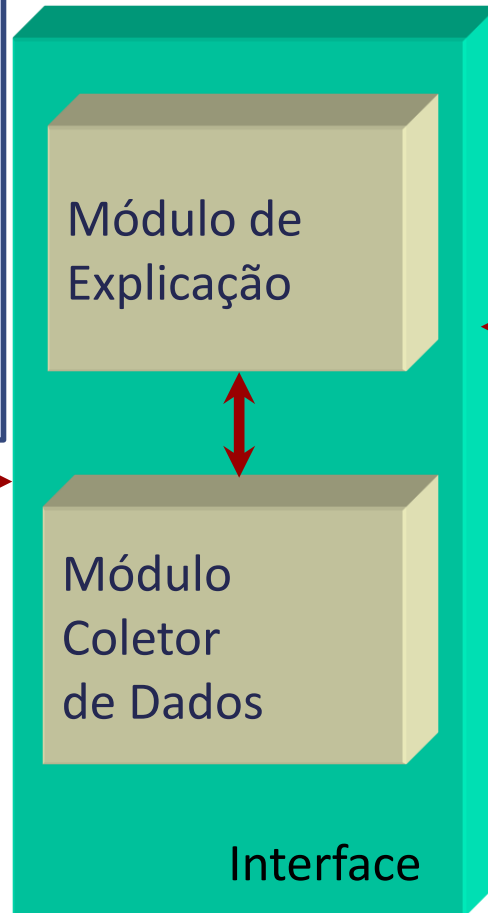
direção da busca

- O que podemos concluir a partir dos dados?

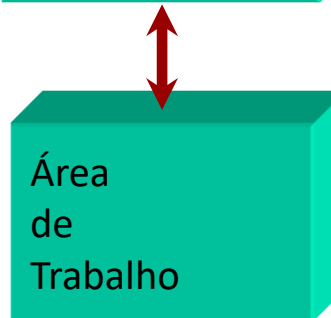
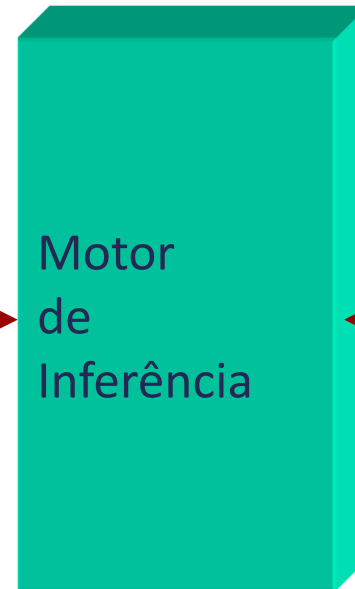
É um processador de linguagem projetado para processar e produzir comunicação orientada a problemas entre o usuário e o sistema

Usuário

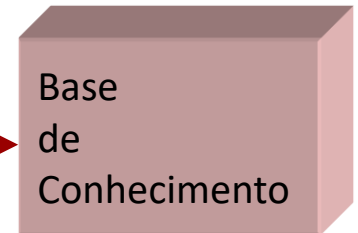
Arquitetura



Interface



Núcleo (Shell)

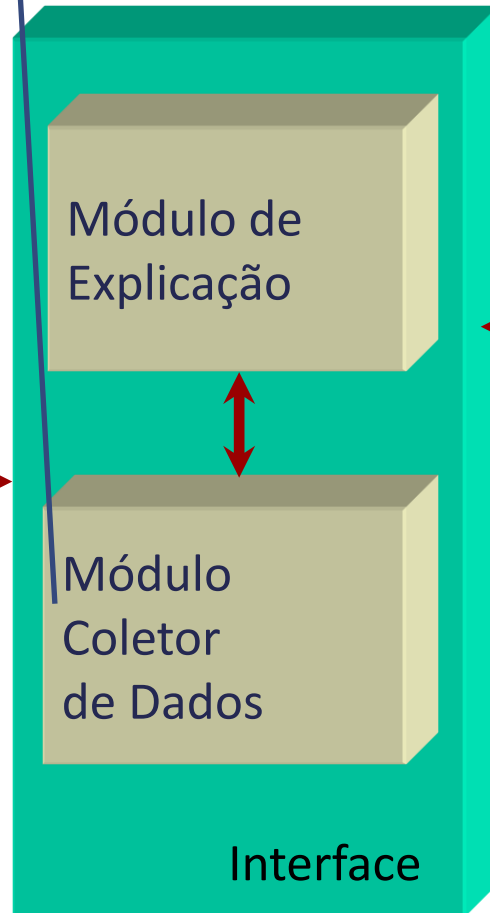


Base de Conhecimento

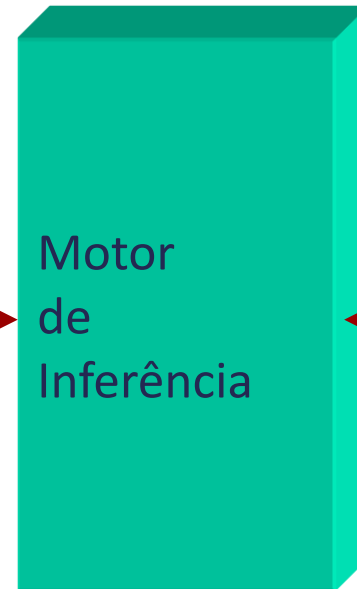
Arquitetura

- Acionado pelo MI quando este necessita de dados específicos
- Pergunta ao usuário e obtém as respostas, enviando-as ao MI

Usuário

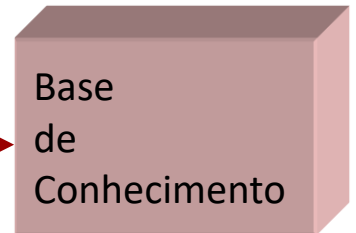


Interface



Área de Trabalho

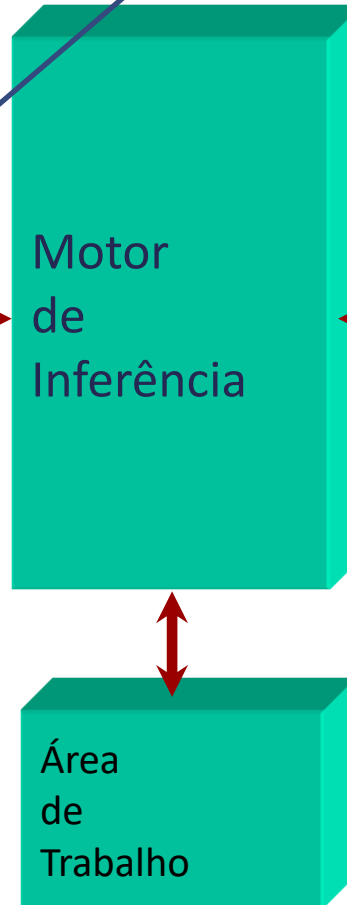
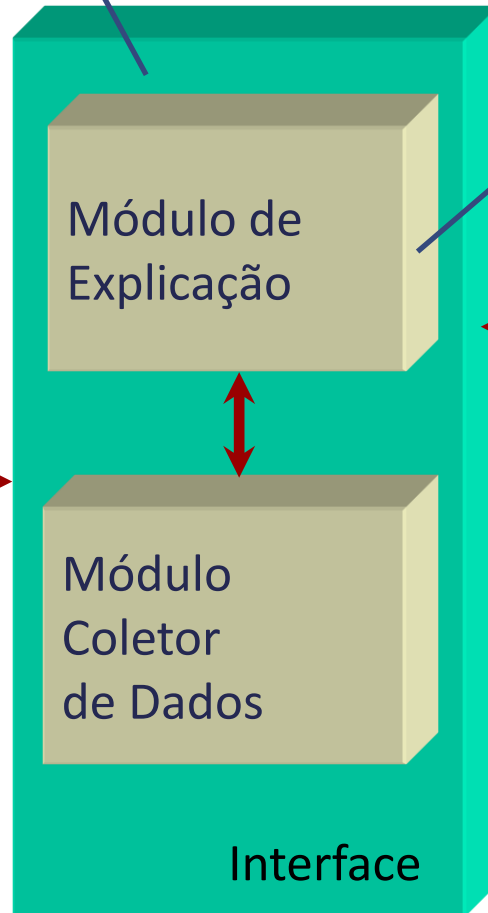
Núcleo (Shell)



Arquitetura

A pedido do usuário, indica como o MI chegou a uma determinada conclusão ou porque realizou uma pergunta

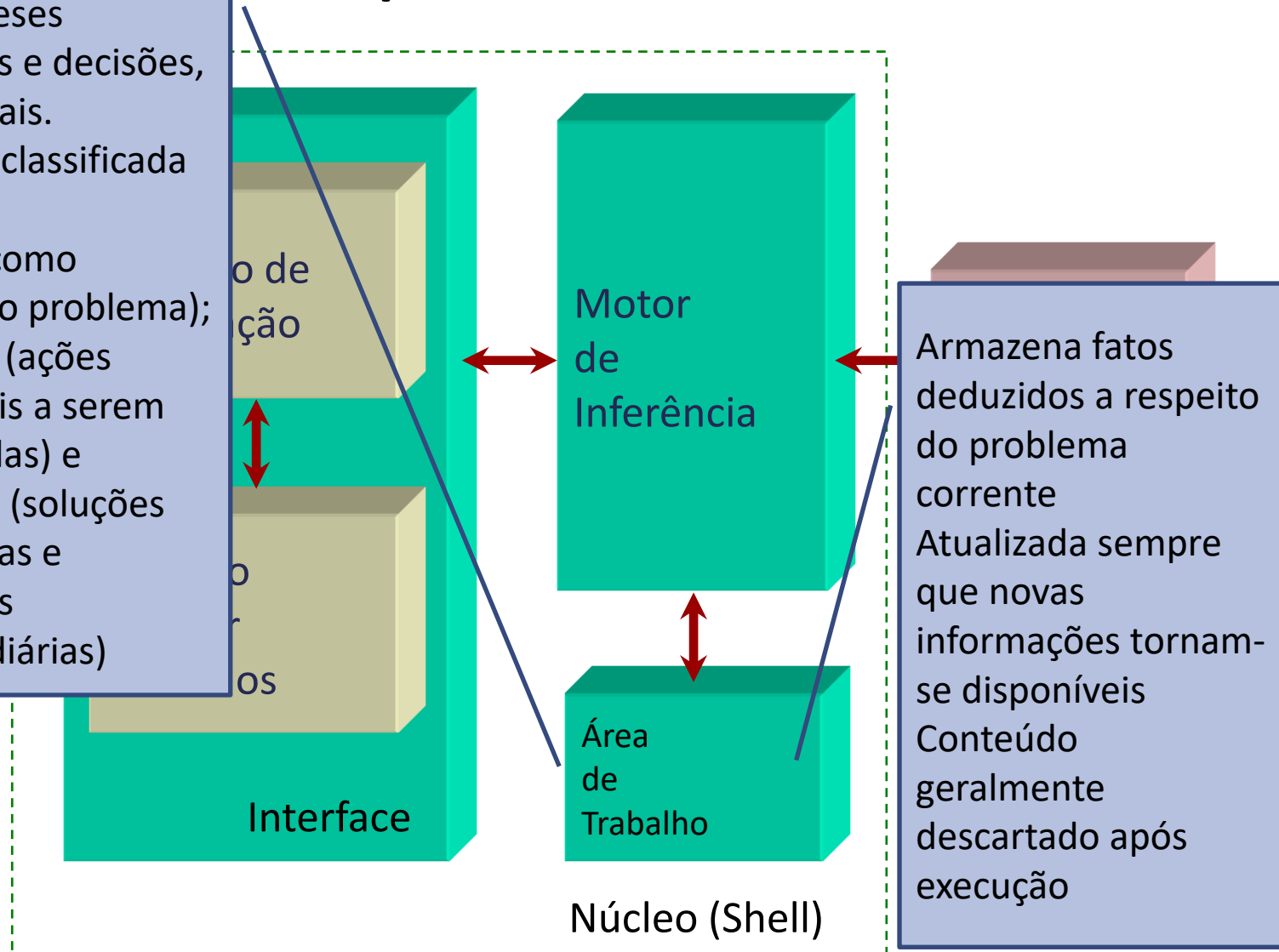
Usuário



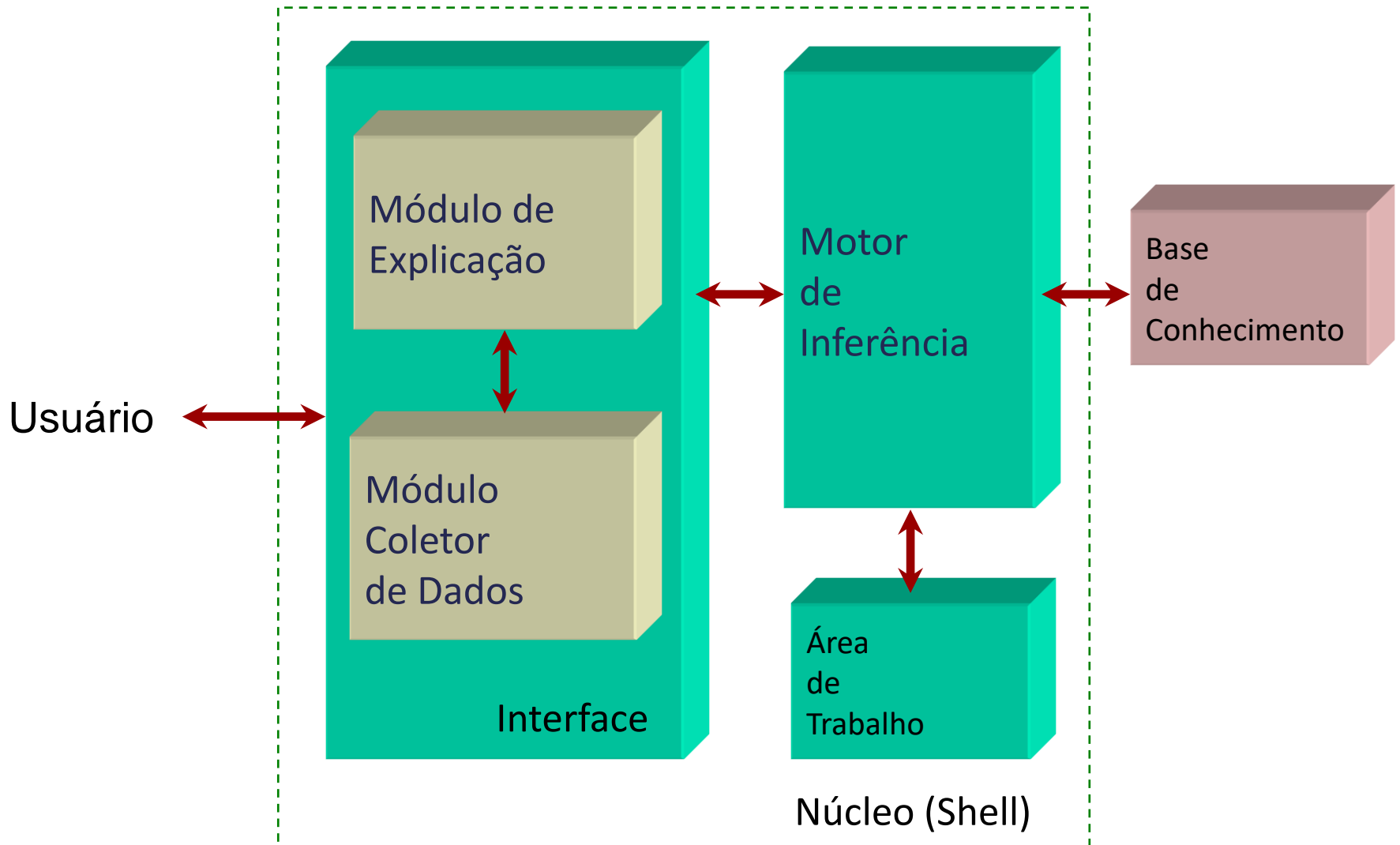
- Porque o sistema faz uma pergunta em particular?
- Como o sistema alcança a conclusão correta?
- Porque uma certa alternativa é rejeitada?
- Qual é a tática atual do sistema para alcançar a conclusão?

- A área (memória) de trabalho do sistema armazena condições iniciais, hipóteses intermediárias e decisões, e soluções finais.
- Informação é classificada em três tipos:
 - Planos (como resolver o problema);
 - Agendas (ações potenciais a serem executadas) e
 - Soluções (soluções candidatas e hipóteses intermediárias)

Arquitetura



Arquitetura



Usuários

SBCs são projetados para interagir com vários tipos de usuários, agindo de diferentes maneiras conforme as circunstâncias:

- Um leigo procurando ajuda direta: **modo consultor**
- Um estudante que quer aprender: **modo instrutor**
- Um construtor de SBC melhorando ou aumentando a base de conhecimentos: **modo acompanhante**
- Um especialista: **modo cooperativo**

Aquisição de Conhecimento

- Como o sistema adquire conhecimento?
 - Como o conhecimento é inicialmente trazido para dentro do sistema?
 - Como o sistema adquire novos conhecimentos?
 - Como este conhecimento pode ser testado?
- Quais componentes necessitamos acrescentar ao sistema para realizar estas funções?

Conhecimento Especialista

Adquirir conhecimento especialista para uma base de um SBC envolve:

- Obter informação dos especialistas e/ou de fontes documentadas;
- Classificação desta informação em declarativa (factual) e procedural ;
- Codificação desta informação num formato utilizado pelo SBC; e
- Verificação de consistência do conhecimento codificado com o conhecimento existente no sistema.

Engenheiro do Conhecimento

A equipe de Engenheiro do Conhecimento realiza as seguintes tarefas:

- Estrutura a área do problema;
- Interpreta, traduz e integra conhecimento especialista ao sistema;
- Traça analogias;
- Apresenta contra exemplos;
- Traz a luz conceitos difíceis; e
- Verifica a consistência do conhecimento.

Refinamento do Conhecimento

- Futuros SBCs poderão ser capazes de monitorar, analisar, aprender e melhorar suas próprias performances, resultando numa base de conhecimento mais aprimorada e num raciocínio mais efetivo
- Entretanto, nos SBCs atuais esta tarefa é realizada pelo Engenheiro de Conhecimento



SHELLS

Shell de um SBC

- Shell de um SBC é um SBC com uma base de conhecimento vazia
- “Shells são destinadas a permitir que não-programadores usufruam dos esforços de programadores que já resolveram um problema similar.”

Jackson

Introdução a Sistemas Especialistas

Processo de Desenvolvimento SBC

Especificação dos Requisitos



Aquisição de Conhecimento



Projeto



Implementação



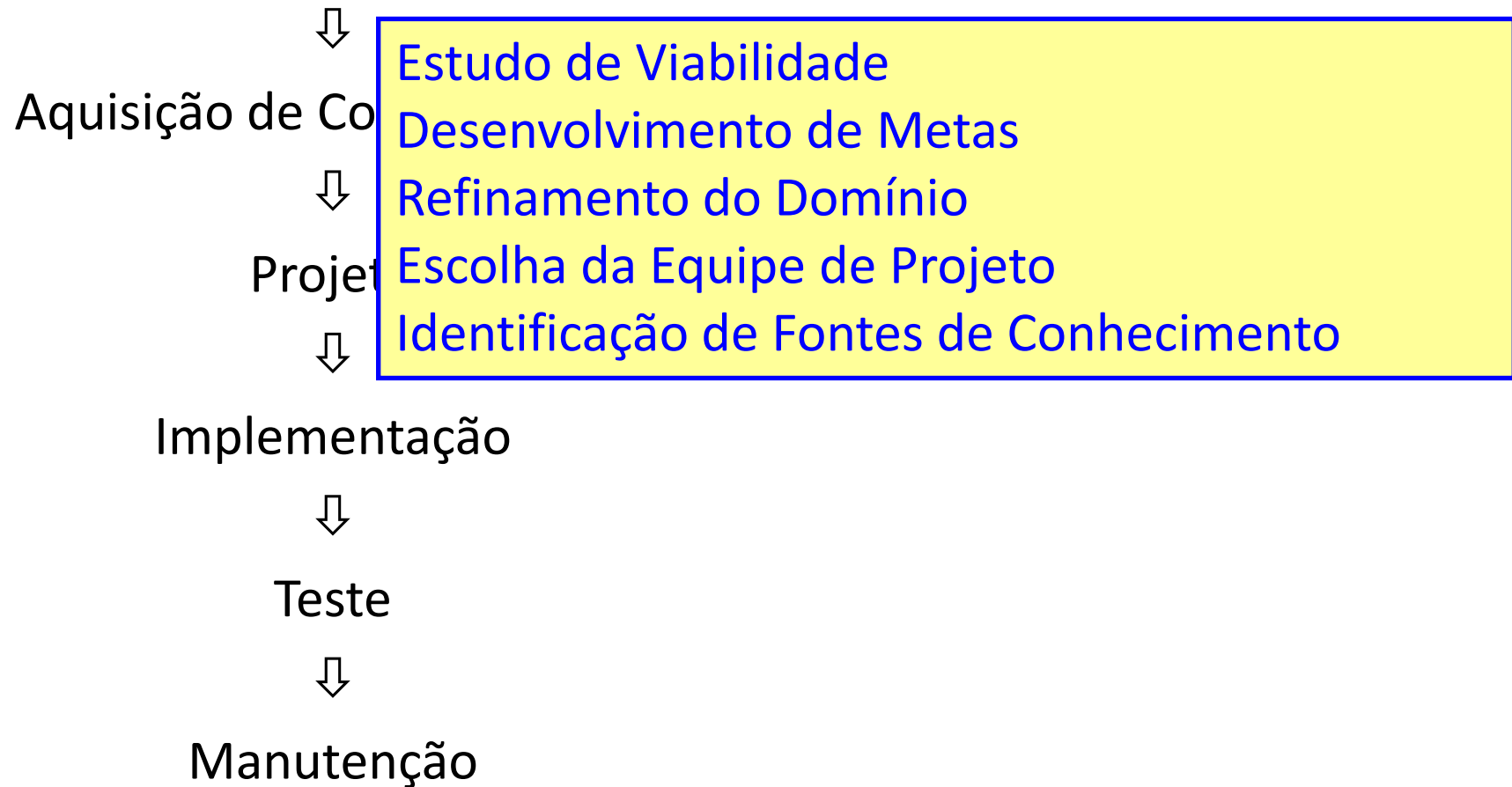
Teste



Manutenção

Processo de Desenvolvimento SBC

Especificação dos Requisitos



Processo de Desenvolvimento SBC

Especificação dos Requisitos



Aquisição de Conhecimento



Projeto



Implementação



Teste



Manutenção

Adquirir,
Organizar &
Estruturar o Conhecimento
Documentação
- Dicionário de Conhecimento

Processo de Desenvolvimento SBC

Especificação dos Requisitos



Aquisição de Conhecimento



Projeto



Implementação



Testes



Manutenção

Definição da Estrutura & Organização
do Conhecimento

Definir Métodos para Processamento

Selecionar Ferramenta de Software

Documentação

Processo de Desenvolvimento SBC

Especificação dos Requisitos



Aquisição de Conhecimento



Projeto



Implementação



Teste



Manutenção

Codificação

Documentação do Sistema

Manual do Usuário

Processo de Desenvolvimento SBC

Especificação dos Requisitos



Aquisição de Conhecimento



Projeto



Implementação



Teste



Manutenção

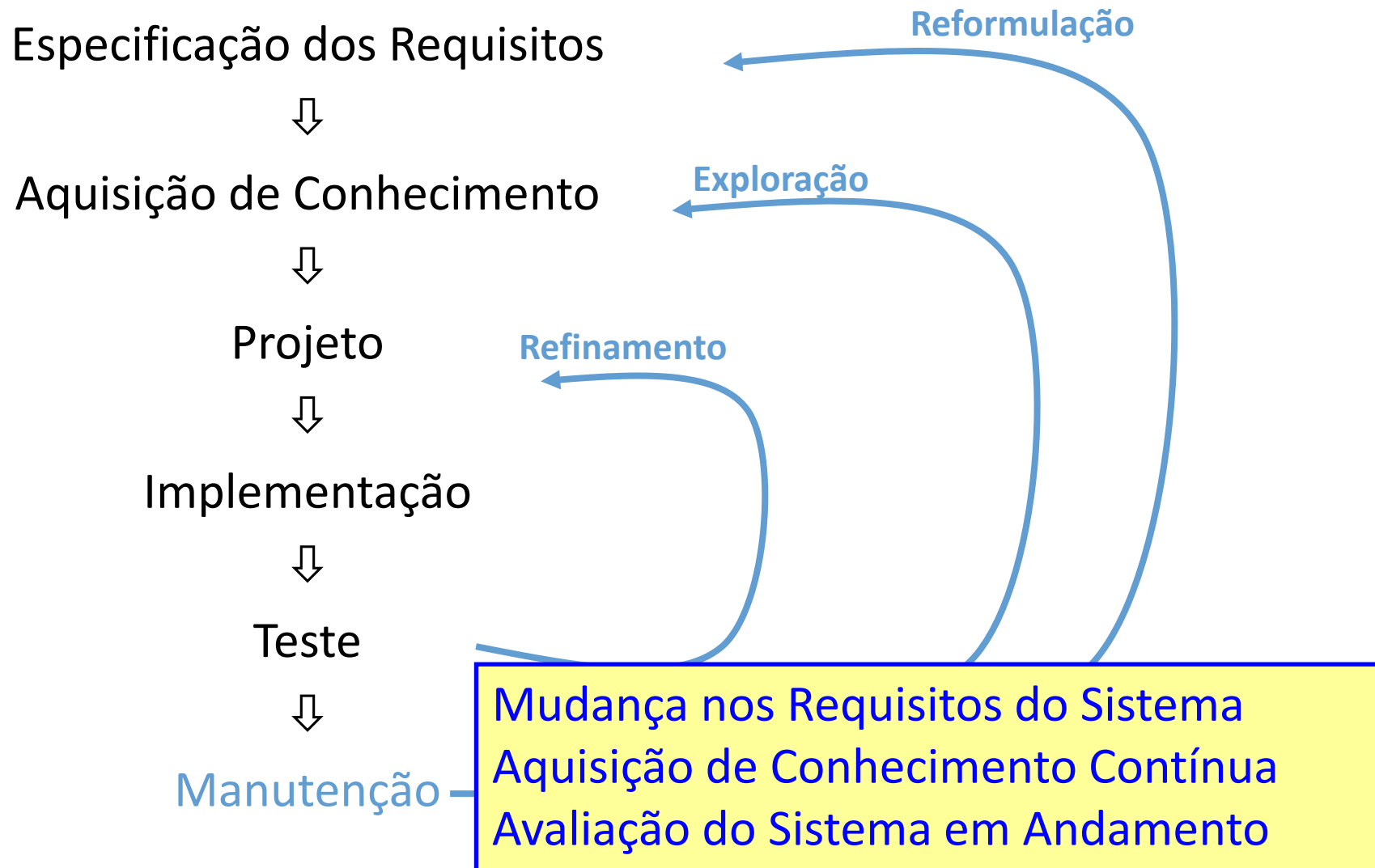
Exploração

Refinamento

Validação & Verificação

Processo Contínuo.. fases se misturam

Processo de Desenvolvimento SBC



Verificação e Validação (V V)

- Há a necessidade de assegurar que o sistema:
 - Funcione
 - Forneça resultados verdadeiros (corretos)
 - Satisfaça os requisitos do cliente

O IEEE define V V:

Verificação:

- Processo que determina se os produtos de uma dada fase de desenvolvimento do software satisfazem todos os requisitos estabelecidos durante a fase anterior.
- “O sistema foi construído corretamente?”

O IEEE define V V:

Validação:

- Processo de avaliar software no fim do processo de desenvolvimento para assegurar que está de acordo com os requisitos do software.
- “É o sistema certo?”

Verificação e Validação

As estratégias de teste mais eficazes são constituídas de vários testes pequenos e separados

O sistema pode ser testado com um conjunto estruturado de problemas teóricos, e com problemas reais

O desempenho pode ser avaliado pela comparação de informações históricas ou com recomendações paralelas de especialistas

Verificação e Validação

Técnicas de validação
incluem:

- Simulação
- Teste de Turing
- Revisão por especialista(s)
- Revisão pelos usuários finais

Verificação e Validação

Problema: Quais características devem ser avaliadas?

- Índice de acerto
- Velocidade
- Facilidade de Uso
- Flexibilidade
- Robustez
- Comportamento diante de Erro

Verificação e Validação

Problema: Qual o “benchmark” utilizado para desempenho?

- Tão bom quanto um “especialista razoável”?
- Tão bom quanto “os melhores”?
- Melhor que qualquer especialista humano?
- Como dizer se é ou não?

Quem pode julgar melhor o desempenho do melhor especialista?

Verificação e Validação

Problema: Escolha de Dados de Exemplo

- Os dados de teste são realistas?
- Como o sistema trata dados inválidos?
- E se dados reais forem raros?

Exemplos de *Shells* para SBC

- Expert SINTA: baseado em regras. Interface intuitiva.
- CLIPS: baseado em regras; interação mediante linguagem específica
- JESS : baseada em regras e objetos
- Mike : baseada em regras
- RT-Experts : baseada em regras (para sistemas de tempo real)
- Entre outros

Quando desenvolver um SBC?



Referências

Expert Sinta

<http://www.lia.ufc.br/~bezerra/exsinta/index.html>

CLIPS

<http://www.ghg.net/clips/CLIPS.html>

Baranauskas, J.A.; Material de aula.

Giarratano, J.; Riley, G. Expert Systems: Principles and Programming.
Boston, USA: PWS Publishing Company, 1998.

Bratko, I.; Prolog Programming for Artificial Intelligence, 3rd Edition,
Pearson Education, 2001.

Russell, Stuart J. & Norvig, Peter. Inteligência Artificial, 2ª edição,
Campus, 2004