

deepsia

Dynamic on-line intErnet Purchasing System Based on Intelligent Agents

Agentes Móveis

Aplicações na Web

Francisco Milagres

ICMC / USP

16 de Maio de 2002

- Introdução
- Visão geral de Agentes
- Agentes Móveis
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- Aplicações
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

- **Introdução**
- Visão geral de Agentes
- Agentes Móveis
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- Aplicações
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

Termo *agentes* em várias áreas

- Inteligência Artificial – Sistemas Especialistas
- Arquitetura de Objetos Distribuídos
- Sistemas de Aprendizado Adaptativo
- Algoritmos Genéticos

Agentes de Software: são programas que estão em um ambiente de execução e possuem algumas propriedades...

(Stan Franklin, 1996)



Propriedades de um agente

deepsia

Reativos	Percebe as mudanças no ambiente e atua de acordo com essas mudanças
Autônomos	Tem o controle sobre suas próprias ações
Orientados a objetivos	Não responde simplesmente em resposta ao ambiente
Contínuo	É um processo continuamente em execução
Comunicativo	Capaz de se comunicar com outros agentes
Adaptativo	Muda o seu comportamento baseado em suas experiências anteriores
Móvel	Habilidade para se transportar
Caráter	Personalidade e <i>estado emocional</i>

- Introdução
- **Visão geral de Agentes**
- Agentes Móveis
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- Aplicações
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

$$\text{Agentes} = \text{Código} + \text{Estado}$$

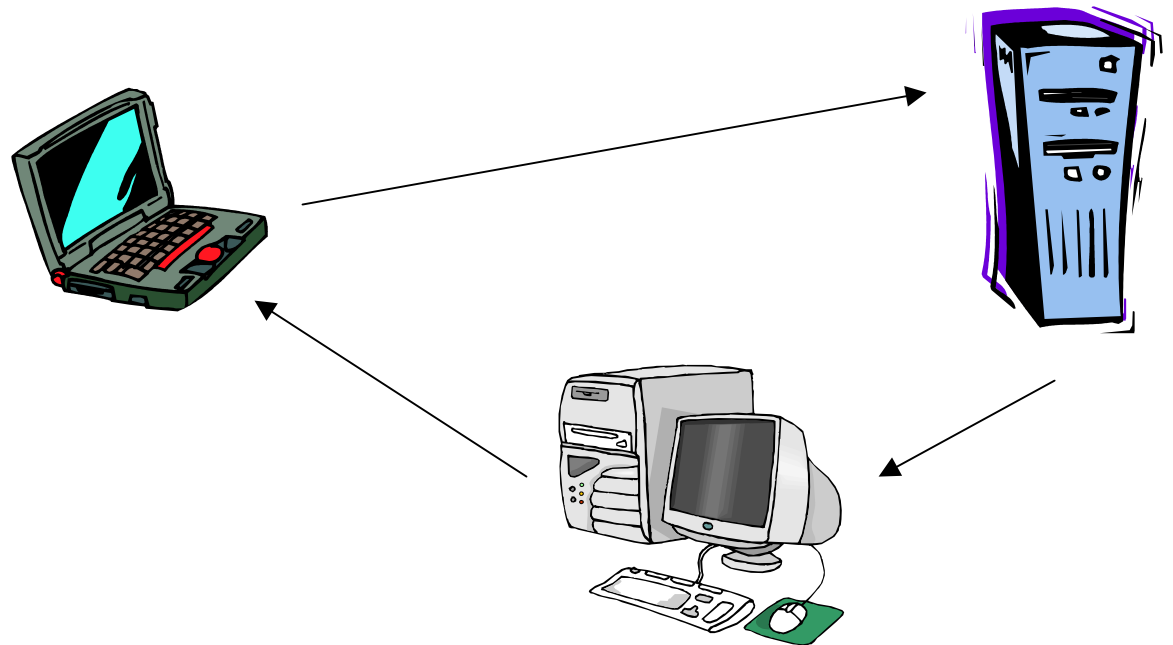
- Estado do Agente: pode ser seu estado de execução e valores de atributos do agente (*frame stacks, ciclo*)
- Código: classe necessária para o agente se executar
- Sistema de Agentes: plataforma que pode criar, interpretar, executar, transferir e receber agentes

Unidade de Execução = Agente
Ambiente Computacional = Sistema de Agentes

- Introdução
- Visão geral de Agentes
- **Agentes Móveis**
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- Aplicações
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

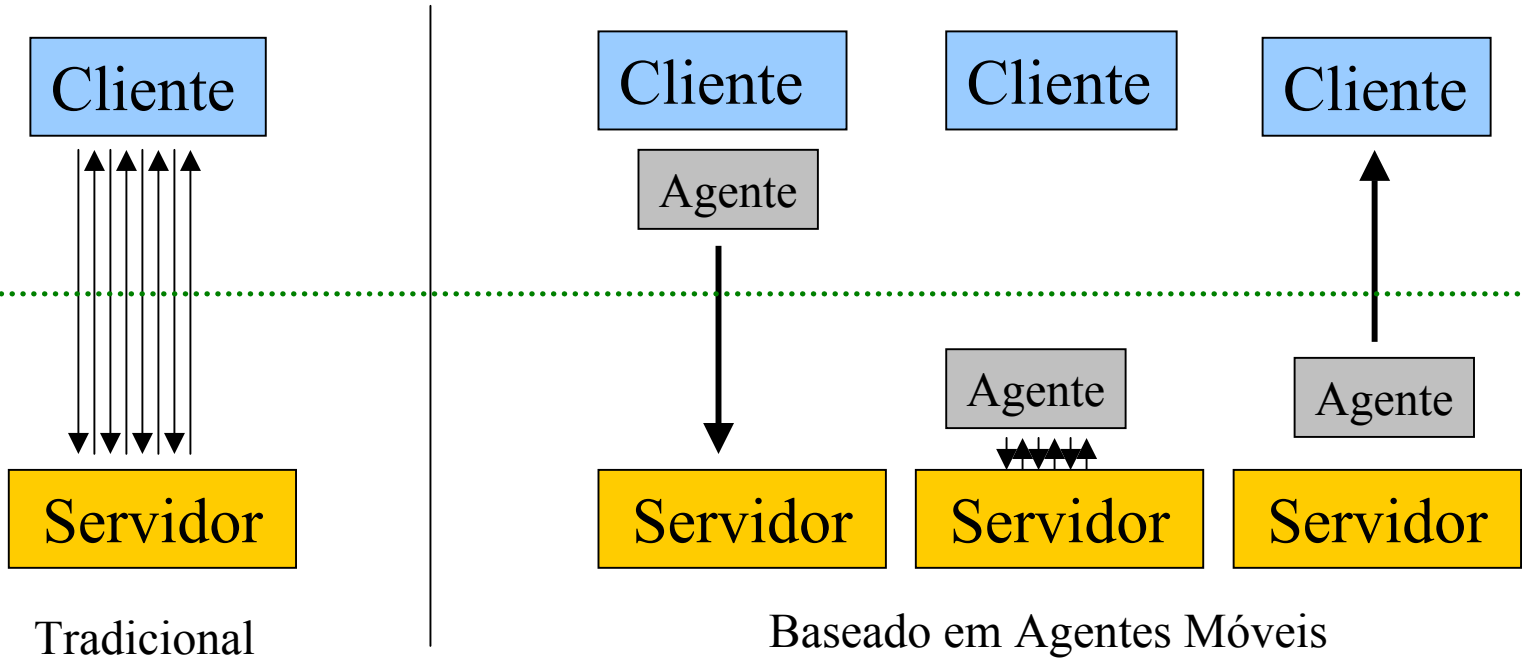
Definição

- Agentes de software
- Se move de um computador para outro



Cliente-Servidor vs. Agentes Móveis

deepsia

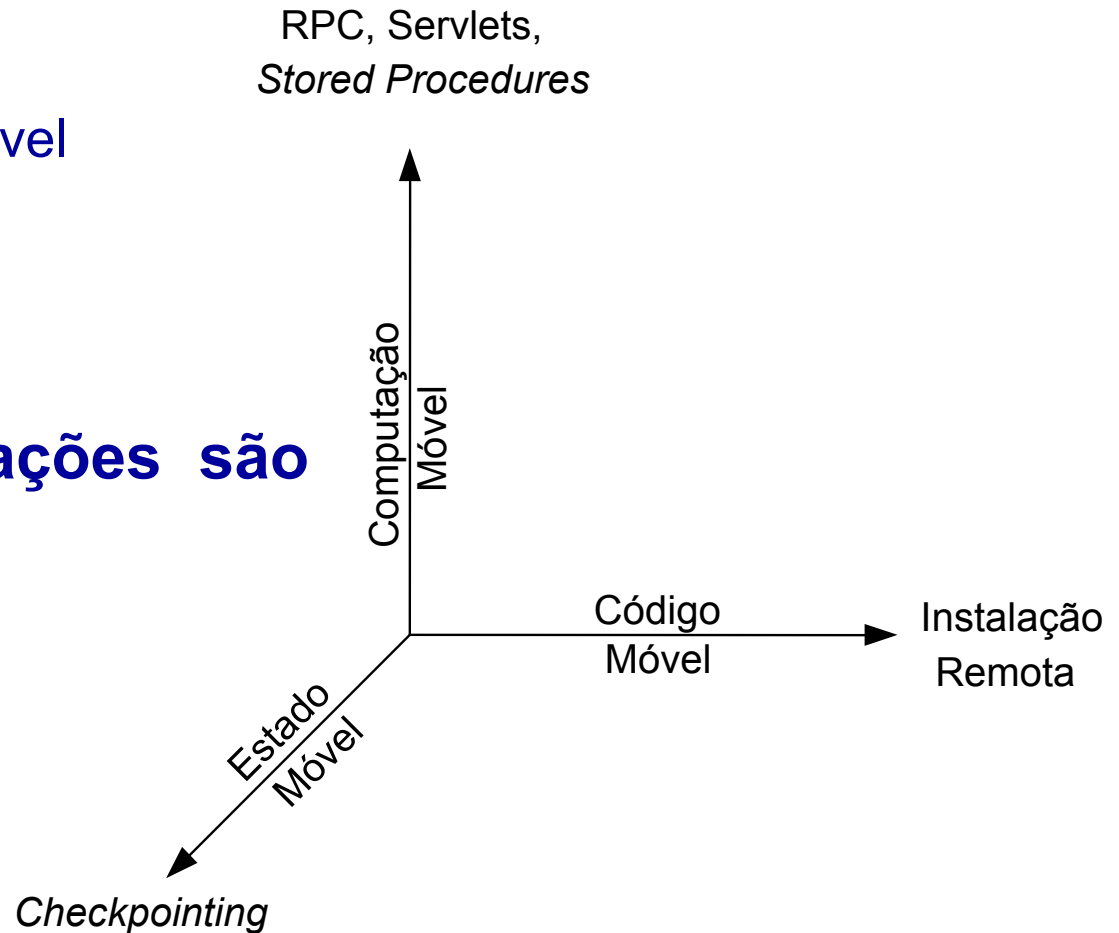


- Introdução
- Visão geral de Agentes
- Agentes Móveis
- **Caracterização de Mobilidade**
- Vantagens dos Agentes Móveis
- Aplicações
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

Caracterização de Mobilidade

- Três possibilidades
 - Código Móvel
 - Computação Móvel
 - Estado Móvel

- **Várias combinações são possíveis...**



- Permite que código executável *migre* para um outro computador
- Pode utilizar o modelo *push* ou *pull*
 - Pull: *Applets*
 - Push: *Instalação remota*

Agentes móveis usam *push*

- Às vezes, um agente pode gerar um código *pull*
- O código pode ser binário ou fonte

Vantagens:

- Dinamicamente pode mudar as capacidades
 - Download de novo código para adicionar / substituir / se atualizar
 - Remover código não utilizado

Problemas:

- Considerações de segurança com relação à código não confiável
 - O código pode ser malicioso ou com *bugs*

- **Evolução da Computação Remota**

- RPC, RSH, Servlets, Stored Procedures, CORBA

- **Permite que um sistema execute seu processamento em outro sistema**

- **Utiliza recursos em um sistema remoto**

- CPU, memória, etc

- **Acessa recursos de um sistema remoto**

- Arquivos, bancos de dados, etc

Evolução da *captura de estado*

- *Checkpointing*

Permite que o estado de execução de um processo seja capturado e movido

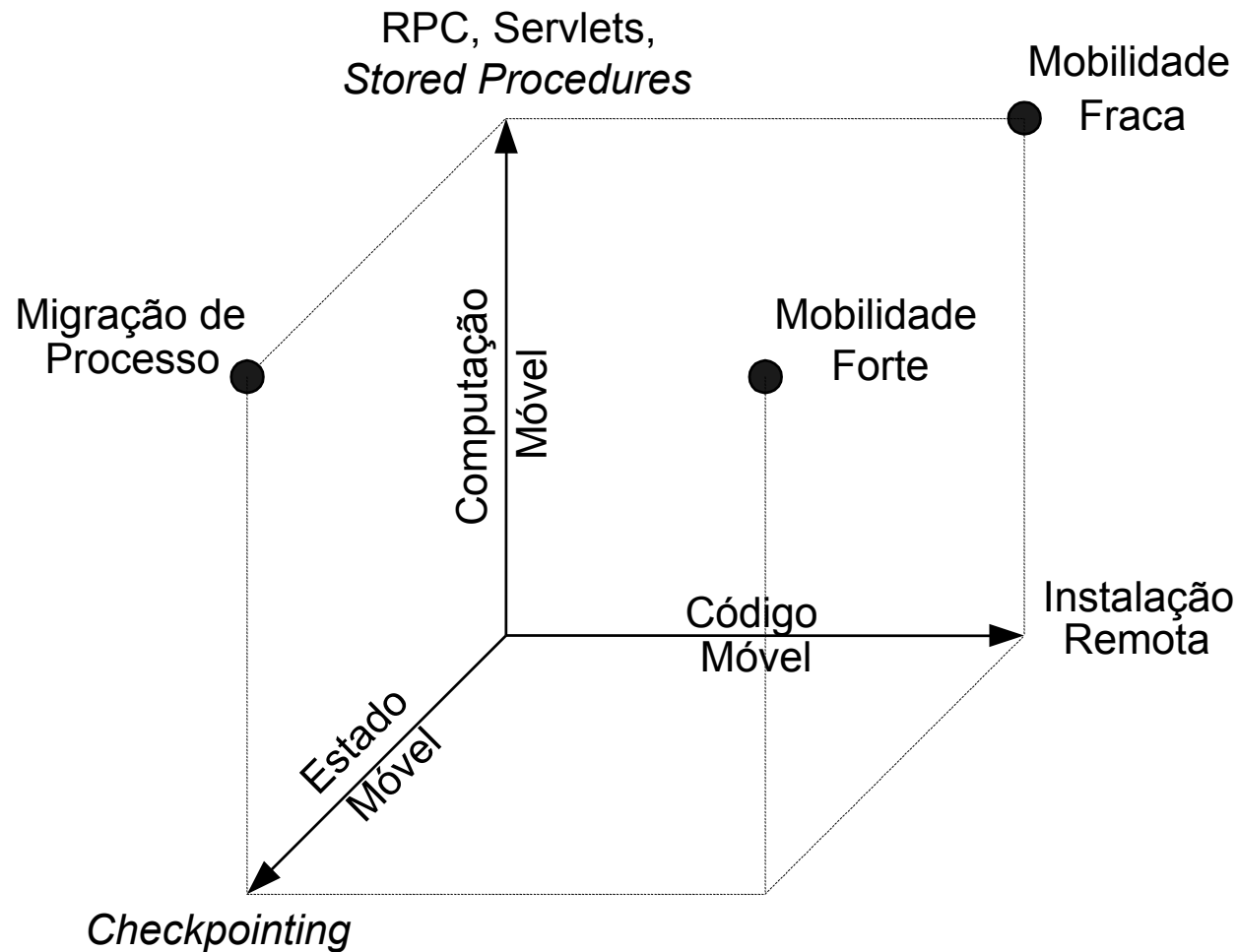
O estado de um processo pode ser dependente ou independente de arquitetura

Pode conter:

- Estado de um único ou de múltiplos *threads*
- Código

Combinações das Possibilidades

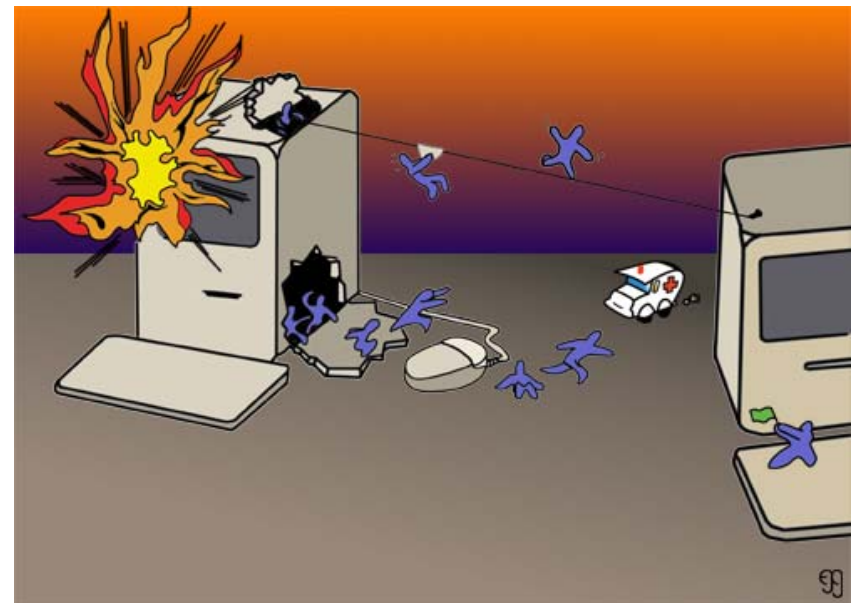
deepsia



Move o estado de execução com o agente

Porque é importante?

- Computacionalmente equivalente à mobilidade fraca
- No entanto, mais simples e de abstração mais natural
- Além do mais, mais fácil de se programar agentes móveis
- Mais importante:
- Estado móvel permite a *mobilidade obrigatória*

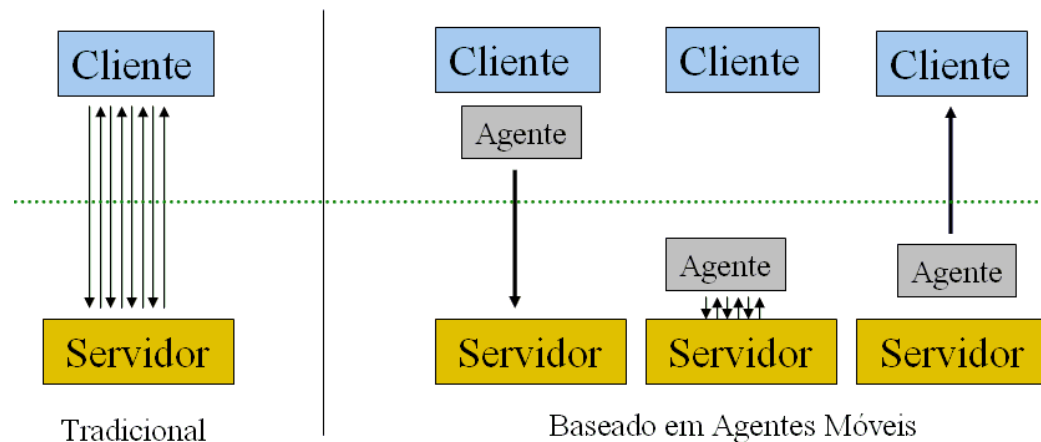


- Introdução
- Visão geral de Agentes
- Agentes Móveis
- Caracterização de Mobilidade
- **Vantagens dos Agentes Móveis**
- Aplicações
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

Vantagens dos Agentes Móveis

deepsia

- Menor uso da largura de banda da rede
- Operação *não conectada*
 - Curtos períodos *on-line*
 - Necessita de baixas prioridades de execução
 - Suporta unidades móveis
- Interações com baixas latências

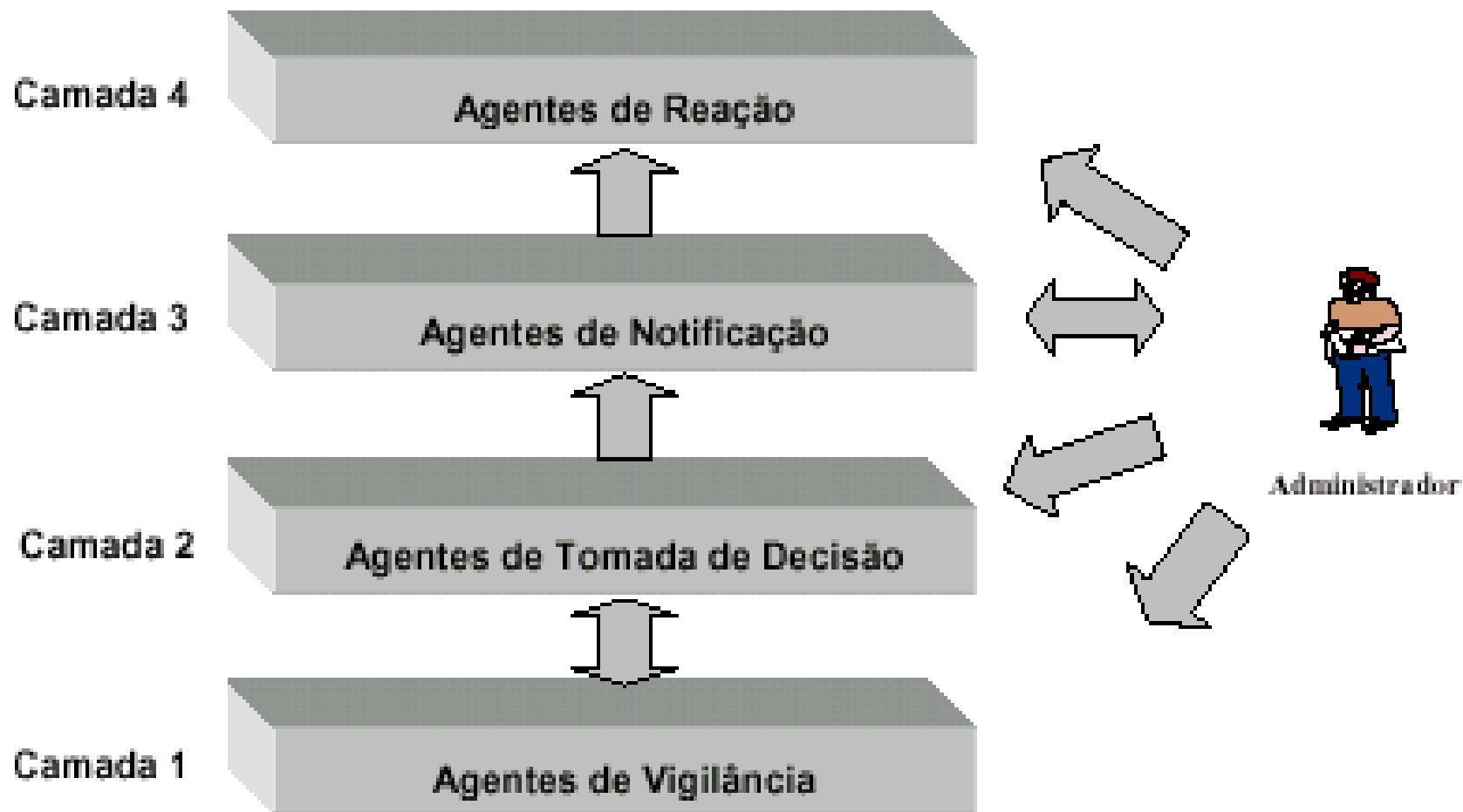


- Introdução
- Visão geral de Agentes
- Agentes Móveis
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- **Aplicações**
 - Case 1: Sistema de Detecção de Intrusões
 - Case 2: Sistema Multi-Agentes DEEPSIA

- Introdução
- Visão geral de Agentes
- Agentes Móveis
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- **Aplicações**
 - **Case 1: Sistema de Detecção de Intrusões**
 - Case 2: Sistema Multi-Agentes DEEPSIA

Case 1: Sistema de Detecção de Intrusão

deepsia

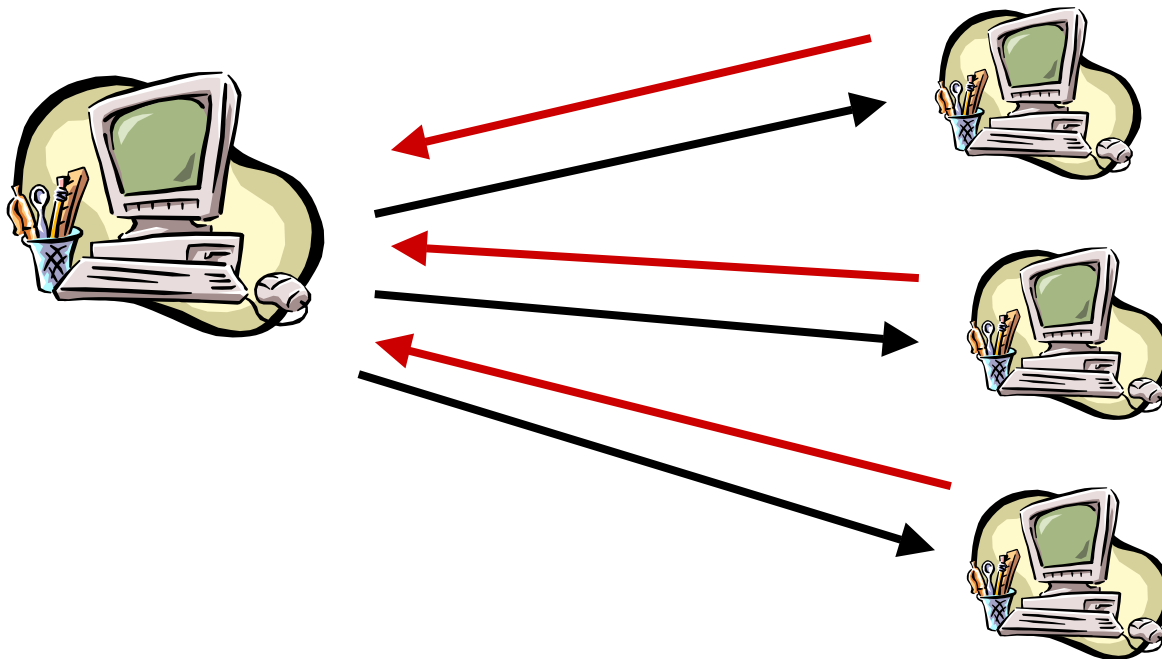


(Bernardes, 1999)

Sistema de Detecção de Intrusão

deepsia

Camada 1: Agentes de Vigilância



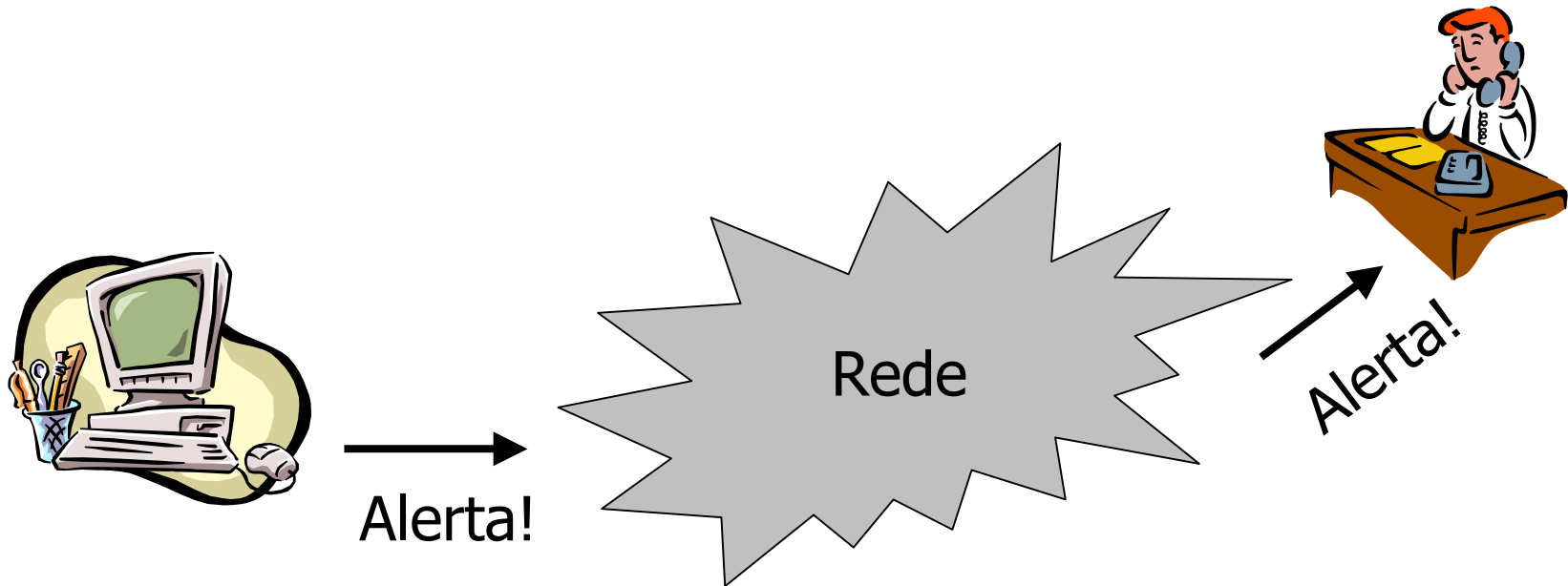
Sistema de Detecção de Intrusão

deepsia

Camada 2: Agentes de Tomada de Decisão



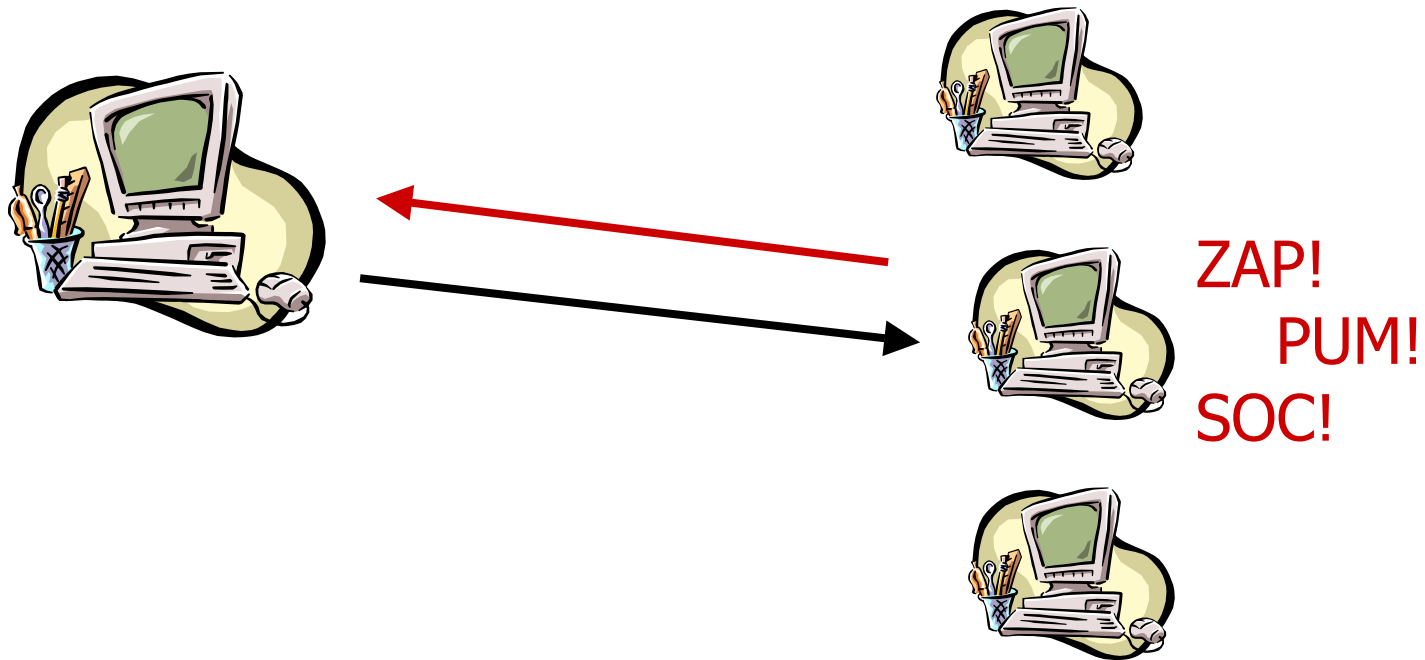
Camada 3: Agentes de Notificação



Sistema de Detecção de Intrusão

deepsia

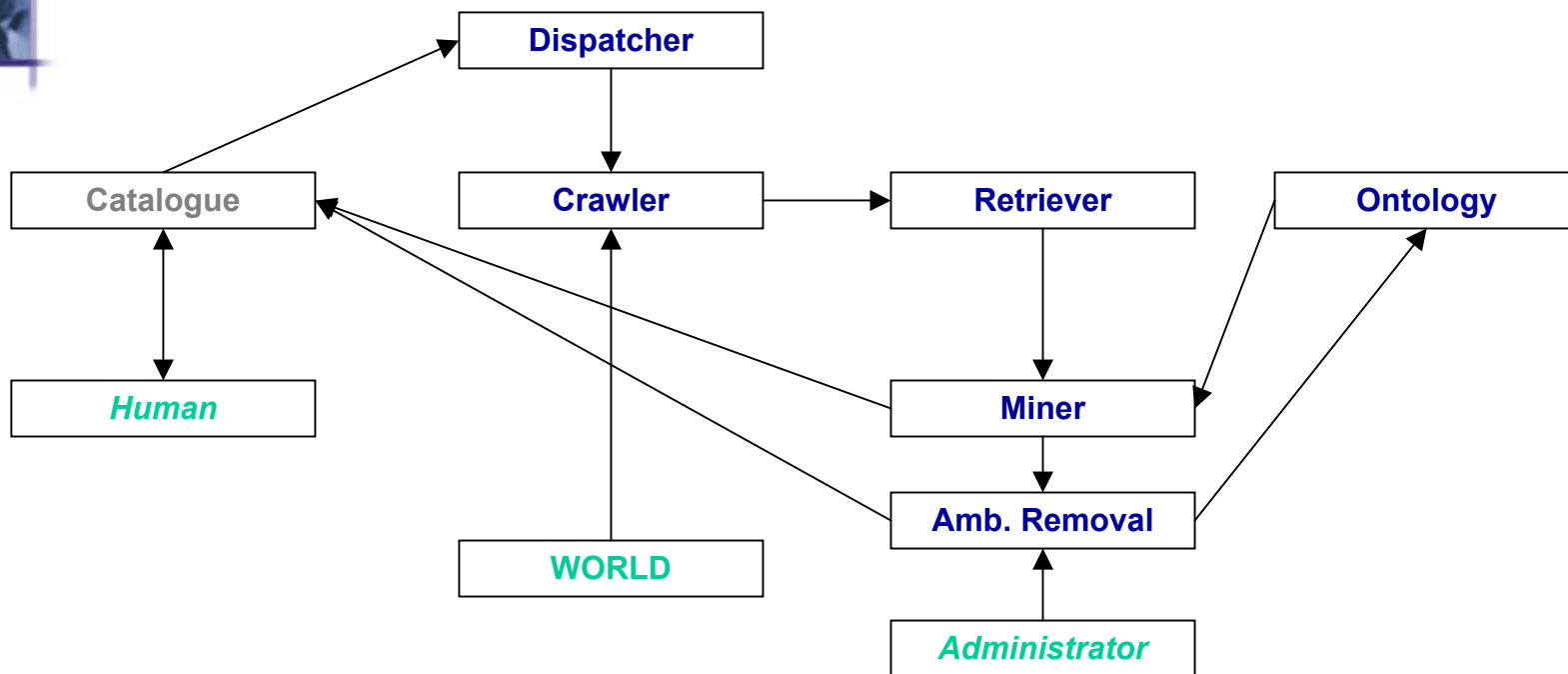
Camada 4: Agentes de Reação



- Introdução
- Visão geral de Agentes
- Agentes Móveis
- Caracterização de Mobilidade
- Vantagens dos Agentes Móveis
- **Aplicações**
 - Case 1: Sistema de Detecção de Intrusões
 - **Case 2: Sistema Multi-Agentes DEEPSIA**

Agentes do Sistema Multi-Agentes

deepsia



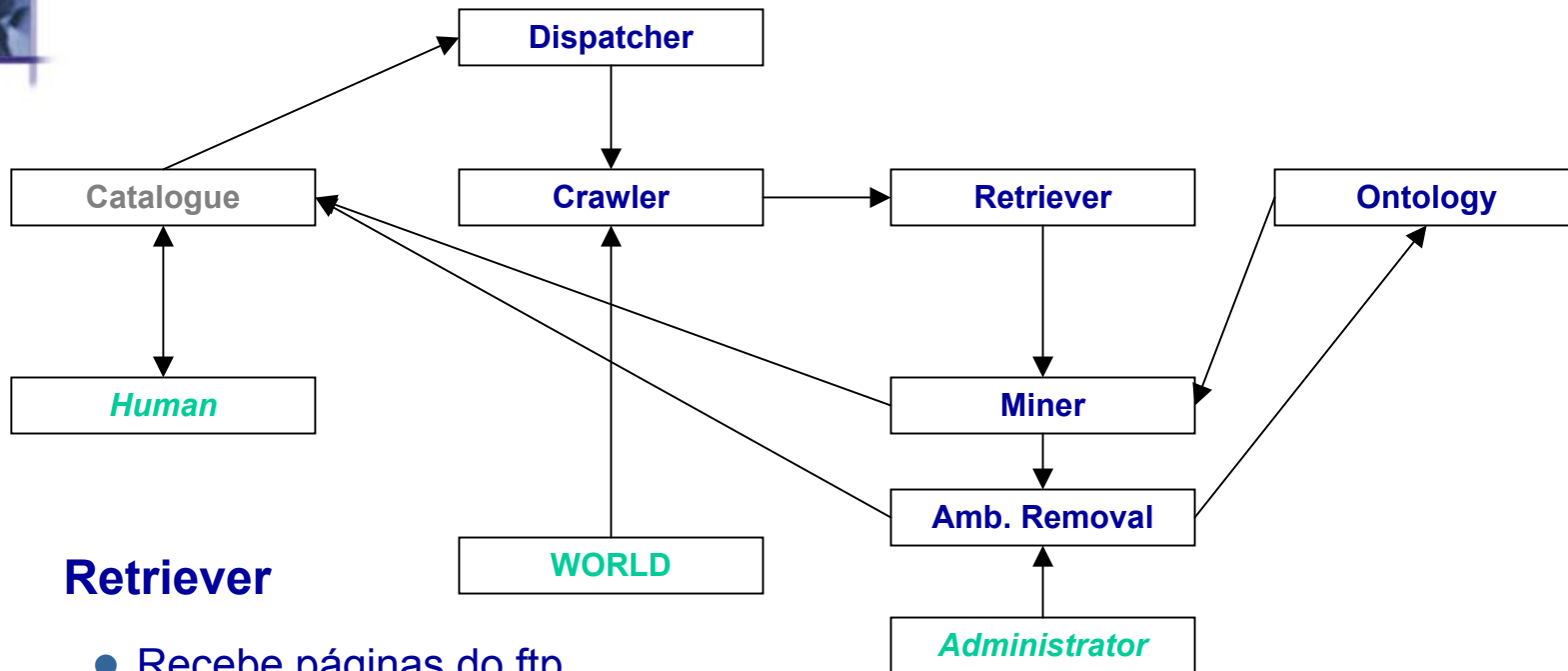
Dispatcher

- Responsável pela interação com o catálogo

Crawler

- Responsável pela navegação no site atual
- Filtragem de páginas não-interessantes através de classificação

Agentes utilizados (2)



Retriever

- Recebe páginas do ftp
- Notifica Miner sobre novas páginas no repositório do ftp

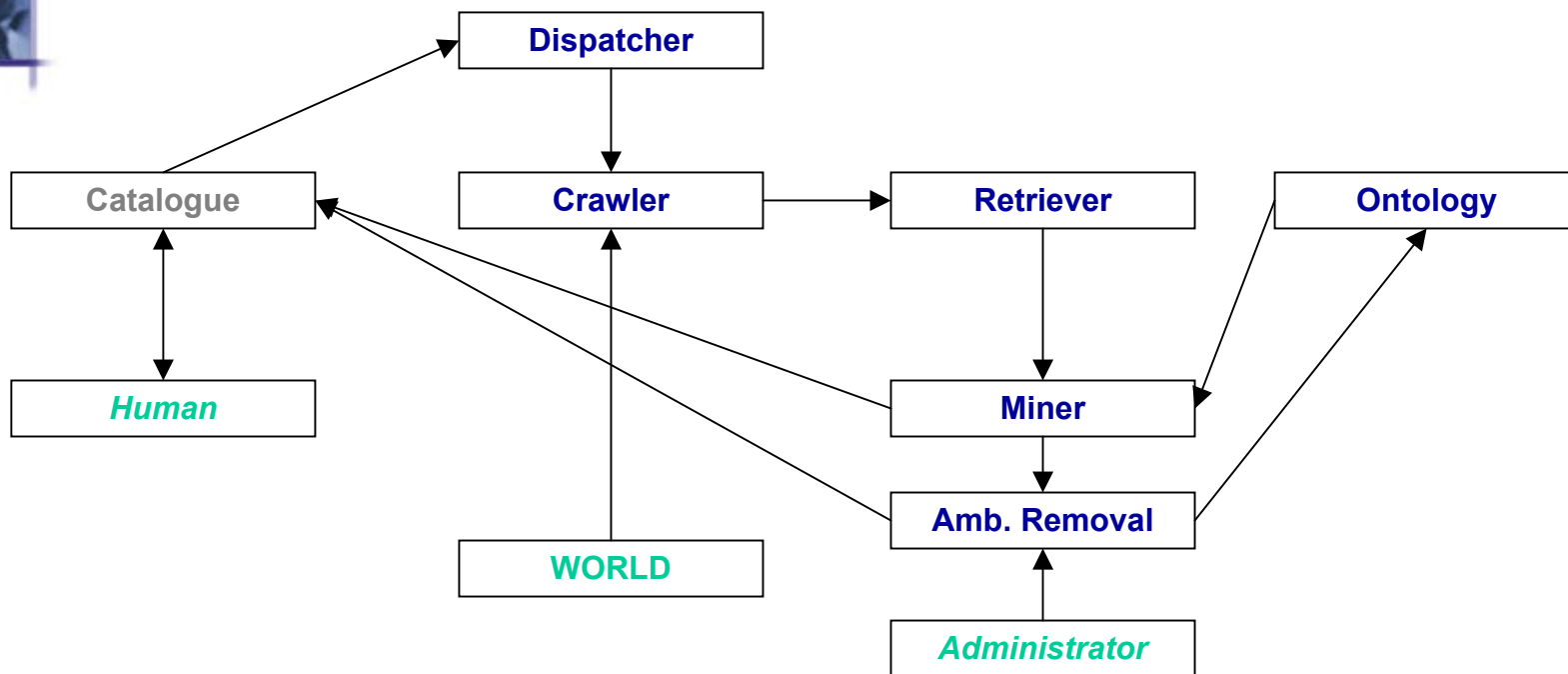
Ontology

- Responde a solicitações de classificação de produtos

Ambiguity Removal

- Interface para usuário que irá distinguir produtos

Agentes utilizados (3)



Miner

- Extrai produtos de uma página
- Calcula palavra chave para página que irá ser relacionada com os produtos
- Questiona agente *Ontology* sobre o conceito relacionado à palavra chave
- Envia produtos para o Catálogo ou para o agente *Ambiguity Removal*

Camadas existentes:

- **Abstract** – conjunto de métodos a serem implementados em níveis superiores
- **Base** – implementação dos métodos suportando TCP/IP
- **KQML** – implementação da sintaxe KQML baseada na camada Base
- **Router** – restringe a passagem de mensagens por um router de modo a poder haver agentes offline. Baseado no KQML Layer
- **Protocol** – baseado na camada Router implementa protocolos como SMTP ou FTP

Consórcio DEEPSIA

deepsia



Universidade de São Paulo
B R A S I L



ICMC-USP

Instituto de Ciências Matemáticas e de Computação



COMARCH



ZEUS CONSULTING S.A.



ULB



University of
Sunderland

Supported by



Homepage: <http://www.deepsia.com>