

# Agentes Inteligentes

Prof. Marcelo de Souza

85ECS – Engenharia de Software Orientada a Agentes  
Universidade do Estado de Santa Catarina



# Agentes

## Revisitando o conceito de agente



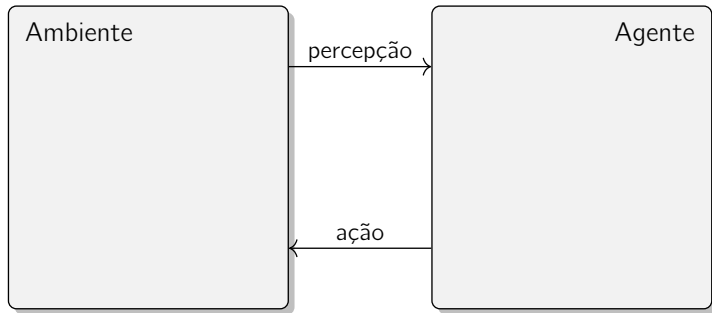
Um agente é um sistema de computador que está **situado** em algum ambiente, e que é capaz de agir de forma **autônoma** nesse ambiente para atender aos seus objetivos de projeto.

# Agentes

## Revisitando o conceito de agente



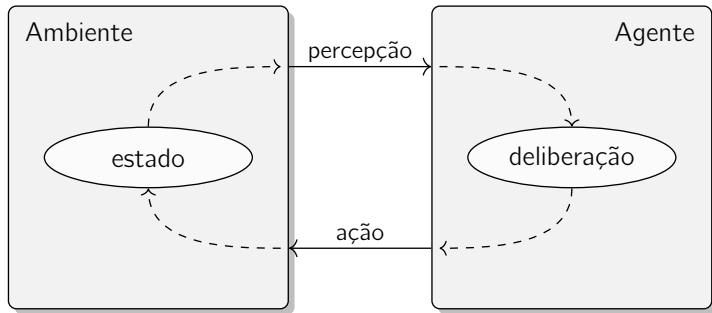
Um agente é um sistema de computador que está **situado** em algum ambiente, e que é capaz de agir de forma **autônoma** nesse ambiente para atender aos seus objetivos de projeto.



# Agentes

## Revisitando o conceito de agente

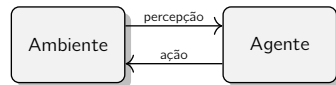
Um agente é um sistema de computador que está **situado** em algum ambiente, e que é capaz de agir de forma **autônoma** nesse ambiente para atender aos seus objetivos de projeto.



# Um exemplo para discussão



**Agente** buscador de vulnerabilidades e ameaças em uma rede interna – agente  $\mathcal{A}$

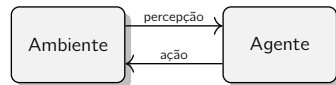


# Um exemplo para discussão



**Agente** buscador de vulnerabilidades e ameaças em uma rede interna – agente  $\mathcal{A}$

- ▶ **Ambiente:** computadores conectados à rede (sistema de arquivos);
- ▶ **Objetivo:** analisar os arquivos da rede e identificar potenciais perigos (e.g. *malwares*);

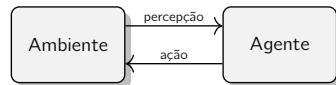




# Um exemplo para discussão

**Agente** buscador de vulnerabilidades e ameaças em uma rede interna – agente  $\mathcal{A}$

- ▶ **Ambiente:** computadores conectados à rede (sistema de arquivos);
- ▶ **Objetivo:** analisar os arquivos da rede e identificar potenciais perigos (e.g. *malwares*);
- ▶ **Percepção:** estrutura de diretórios, arquivos, propriedade e conteúdo dos arquivos, etc.;
- ▶ **Ação:** análise detalhada de suspeitas, reportar achados, colocar em quarentena, etc.



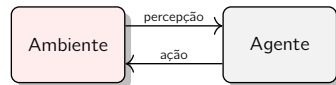


# Propriedades dos ambientes

Com exemplos baseados no agente  $\mathcal{A}$

**Acessível**  $\times$  **inaccessível** (ou completo  $\times$  incompleto)

- ▶ É acessível se o agente tem a visão completa do ambiente.
- ▶ Acessível:  $\mathcal{A}$  é limitado a um conjunto de diretórios predeterminado;
- ▶ Inaccessível:  $\mathcal{A}$  percorre a rede e observa somente os arquivos do computador atual.





# Propriedades dos ambientes

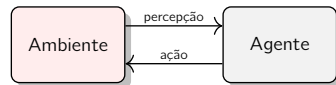
Com exemplos baseados no agente  $\mathcal{A}$

**Acessível**  $\times$  **inacessível** (ou completo  $\times$  incompleto)

- ▶ É acessível se o agente tem a visão completa do ambiente.
- ▶ Acessível:  $\mathcal{A}$  é limitado a um conjunto de diretórios predeterminado;
- ▶ Inacessível:  $\mathcal{A}$  percorre a rede e observa somente os arquivos do computador atual.

**Determinístico**  $\times$  **não determinístico**

- ▶ É determinístico quando qualquer ação do agente tem garantidamente um único efeito.
- ▶ Determinístico:  $\mathcal{A}$  é o único processo operando nos arquivos e tem máximos privilégios;
- ▶ Não determinístico:  $\mathcal{A}$  concorre com outros processos (agentes) e/ou está limitado.



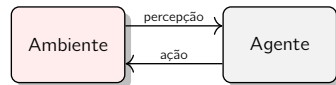


# Propriedades dos ambientes

Com exemplos baseados no agente  $\mathcal{A}$

## Estático × dinâmico

- ▶ É estático quando a ação do agente é a única fonte de mudança no ambiente.
- ▶ Estático: os arquivos são (garantidamente) manipulados somente por  $\mathcal{A}$ ;
- ▶ Dinâmico: outros processos manipulam os arquivos de forma concorrente.





# Propriedades dos ambientes

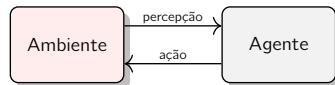
Com exemplos baseados no agente  $\mathcal{A}$

## Estático $\times$ dinâmico

- ▶ É estático quando a ação do agente é a única fonte de mudança no ambiente.
- ▶ Estático: os arquivos são (garantidamente) manipulados somente por  $\mathcal{A}$ ;
- ▶ Dinâmico: outros processos manipulam os arquivos de forma concorrente.

## Discreto $\times$ contínuo

- ▶ É discreto quando há um número fixo e finito de percepções e ações.
- ▶ Discreto:  $\mathcal{A}$  está limitado ao ambiente virtual que, por definição, é discreto;
- ▶ Não determinístico:  $\mathcal{A}$  também analisa aspectos físicos (e.g. temperatura).

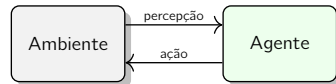




# Agentes simples (e desinteressantes)

## **Termostato** (ar condicionado)

- ▶ Percepção: temperatura  $T$ ;
- ▶ Ação: ligar, desligar.
  - ▶ Se  $T$  está acima do desejado: liga (ou mantém);
  - ▶ Senão, desliga (ou mantém).



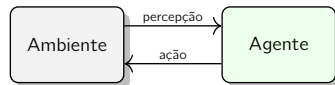
# Agentes simples (e desinteressantes)

## **Termostato** (ar condicionado)

- ▶ Percepção: temperatura  $T$ ;
- ▶ Ação: ligar, desligar.
  - ▶ Se  $T$  está acima do desejado: liga (ou mantém);
  - ▶ Senão, desliga (ou mantém).

## **Monitor de e-mail**

- ▶ Percepção: caixa de entrada;
- ▶ Ação: notificar.
  - ▶ Se há novo e-mail, notifica;
  - ▶ Senção, não faz nada.





# Agentes simples (e desinteressantes)

## **Termostato** (ar condicionado)

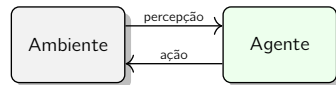
- ▶ Percepção: temperatura  $T$ ;
- ▶ Ação: ligar, desligar.
  - ▶ Se  $T$  está acima do desejado: liga (ou mantém);
  - ▶ Senão, desliga (ou mantém).

## **Monitor de e-mail**

- ▶ Percepção: caixa de entrada;
- ▶ Ação: notificar.
  - ▶ Se há novo e-mail, notifica;
  - ▶ Senção, não faz nada.

Satisfazem a definição de agentes (situados e autônomos), mas **não são inteligentes**.

- ▶ Tarefa e funcionamento interno muito simples!





# Agentes inteligentes

## Algumas características

### Autônomos

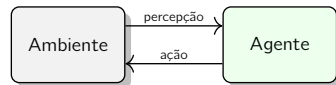
- ▶ Reativos;
- ▶ Pró-ativos;
- ▶ Híbridos.

### Adaptativos

- ▶ Melhoram seu desempenho com o tempo e a experiência → aprendizagem!

### Sociais

- ▶ Coordenação;
- ▶ Cooperação ou competição;
- ▶ Negociação

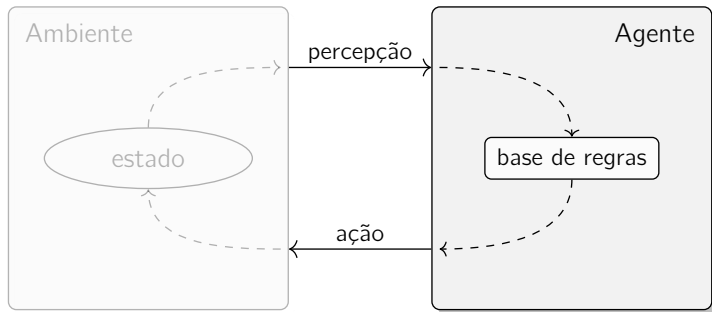


# Estrutura/arquitetura de agentes

## Agentes (puramente) reativos

Respondem a mudanças no ambiente via uma base de regras.

- ▶ Exemplo: aspirador de pó – detecta sujeira no chão, aspira e se move no ambiente;
- ▶ Base de regras:  $\{sujo \rightarrow aspira; limpo \rightarrow desloca\}$ .



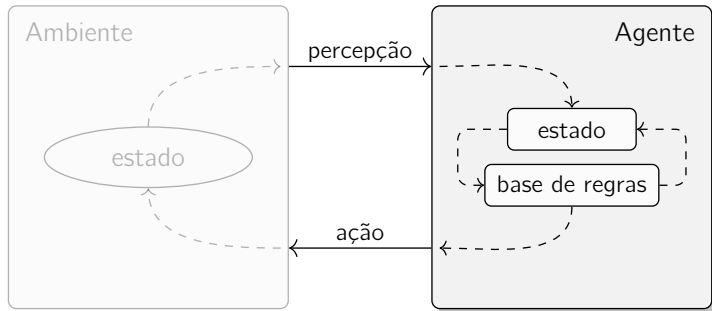


# Estrutura/arquitetura de agentes

## Agentes reativos com estado (interno)

Armazenam um modelo do estado do ambiente e/ou informações internas.

- ▶ Exemplo: aspirador de pó com mapa do ambiente e memória;
- ▶ Regras que definem o deslocamento com base em locais já visitados (i.e. regiões limpas).

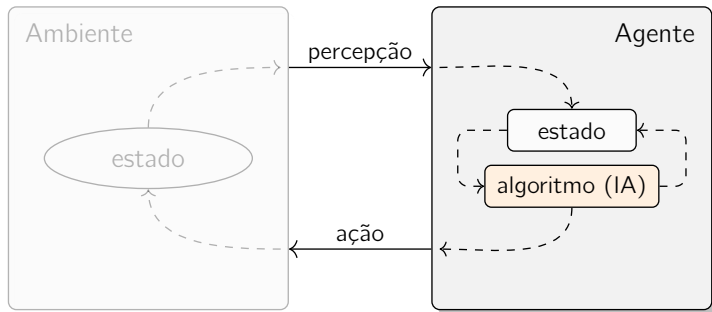


# Estrutura/arquitetura de agentes

## Agentes pró-ativos

Implementam algoritmos (mais sofisticados) para a deliberação (tomada de decisão).

- ▶ Geralmente, tentam maximizar uma função de utilidade;
- ▶ Exemplo: aspirador de pó com mapa do ambiente, memória e planejador de ações;
- ▶ Planeja sua rota maximizando a área limpa e minimizando o tempo (ou energia) gasto.



85ECS – Engenharia de Software Orientada a Agentes  
Prof. Marcelo de Souza