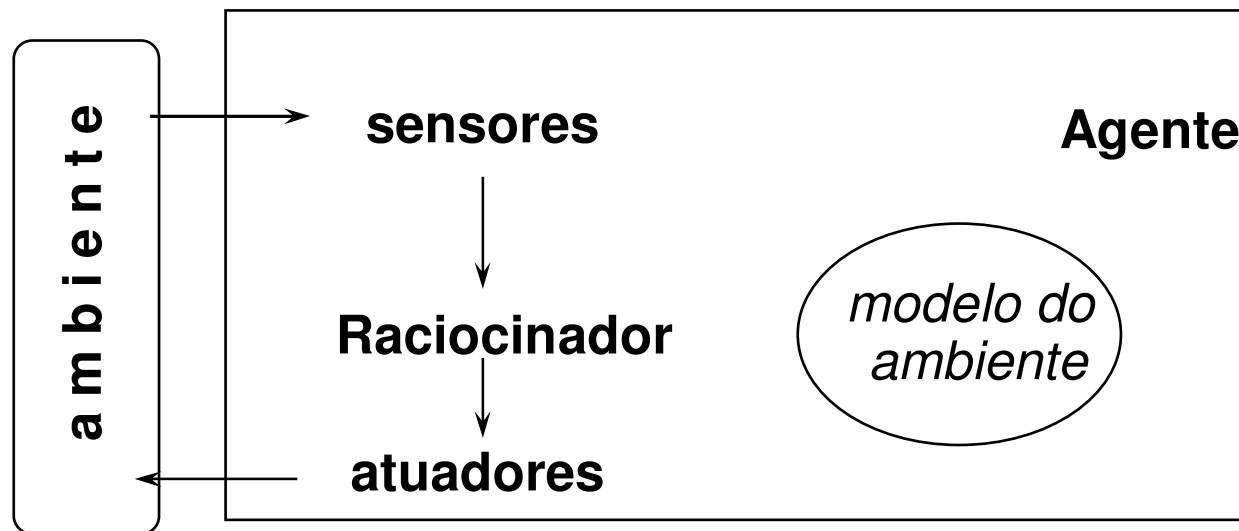


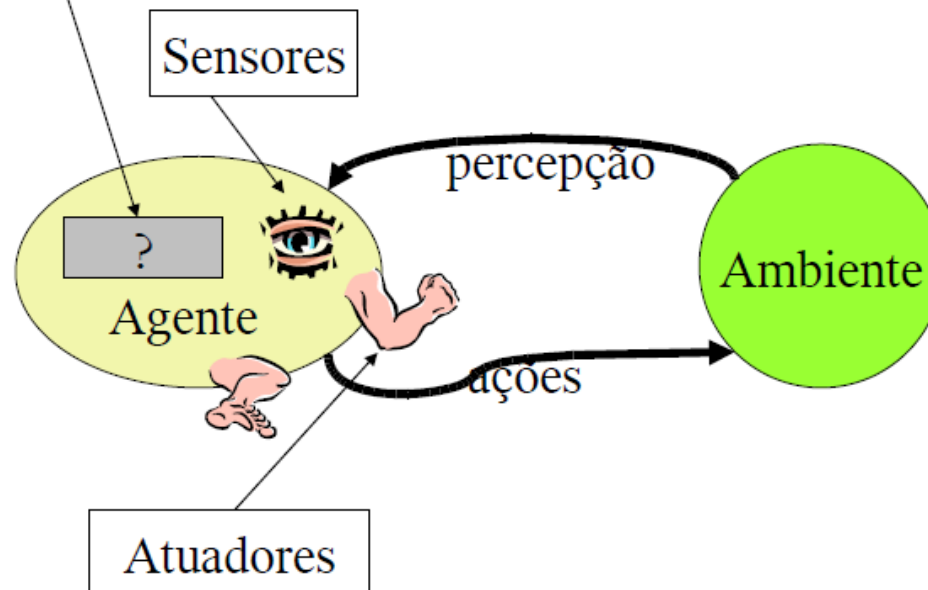
# Agentes Inteligentes

# O que é um agente

- Agente é qualquer entidade que:
  - percebe seu ambiente através de sensores (ex. câmeras, microfone, teclado, finger...)
  - age sobre ele através de atuadores (ex. vídeo, auto-falante, impressora, braços, ftp, ...)
- Mapeamento: seqüência de percepções => ação



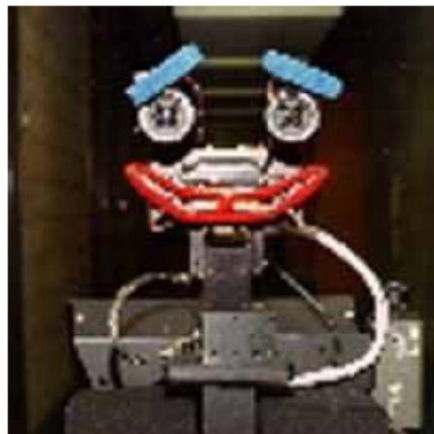
Como projetar isto?



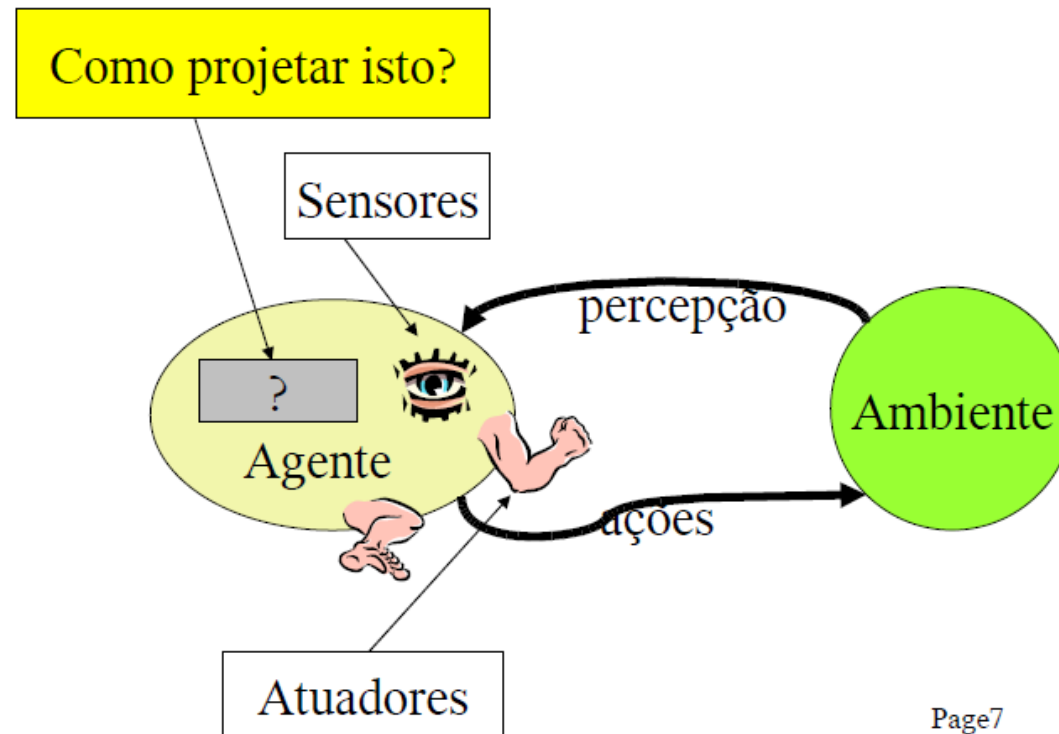
- Um **agente humano** tem olhos, ouvidos, e outros órgãos como sensores, e mãos, pernas, boca, e outras partes como atuadores



- Um **agente robótico** utiliza câmeras e dispositivos de infra-vermelho como sensores e vários motores como atuadores



- Um objetivo da Inteligência Artificial é projetar agentes que façam um bom trabalho de atuar em um ambiente



---

# Medida de Desempenho (MD)

- Critério que define o grau de sucesso de um agente na realização de uma dada tarefa
    - Esta medida deve ser imposta do exterior
    - Má escolha da MD pode acarretar comportamento indesejado
    - Compromissos entre objetivos múltiplos conflitantes
    - Resta o problema de saber quando avaliar o desempenho
    - Exs. aspirador de pó, provador de teoremas, filtragem de *e-mails*, policial de trânsito, avaliador de clima...
-

---

## Um exemplo

- Como exemplo de medida de desempenho, considere um agente aspirador-de-pó que deve limpar uma dada área



---

## Um exemplo

- **Como exemplo de medida de desempenho, considere um agente aspirador-de-pó encarregado de limpar uma certa área**
    - uma medida de desempenho plausível é medir a quantidade de poeira coletada em um período de oito horas
    - Uma medida mais sofisticada levaria em conta a quantidade de eletricidade consumida e a quantidade de ruído produzido
-



---

# Agente Racional (McCarthy & Hayes 69, Newell 81)

- Agente Racional: **fazer a melhor coisa possível**
    - segue o **princípio da racionalidade**: dada uma **seqüência de percepções**, o agente escolhe, segundo seus **conhecimentos**, as **ações** que satisfazem melhor seu **objetivo**.
  - Problema
    - estado inicial + ações => estado final (objetivo)
  - Racionalidade  $\neq$  Onisciência, limitações de:
    - sensores
    - atuadores
    - raciocinador (conhecimento, tempo, etc.)
    - Agir para obter mais dados perceptivos é **racional**
-

---

# Autonomia e Utilidade

## ■ Autonomia

- ❑ Capacidade de adaptação a situações novas, para as quais não foi fornecido todo o conhecimento necessário com antecedência
- ❑ Duas implementações: aprendizagem e/ou programação declarativa

## ■ Para construir um sistema inteligente, utilizamos

- ❑ linguagem
  - ❑ inferência
  - ❑ conhecimento
-

---

# A metáfora de agente decompõe

## 1) Problema em:

- ❑ percepções, ações, objetivos, e ambiente (e outros agentes)

## 2) Tipo de conhecimento em:

- ❑ Quais são as propriedades relevantes do mundo
- ❑ Como o mundo evolui
- ❑ Como identificar os estados desejáveis do mundo
- ❑ Como interpretar suas percepções
- ❑ Quais as conseqüências de suas ações no mundo
- ❑ Como medir o sucesso de suas ações
- ❑ Como avaliar seus próprios conhecimentos

## 3) Arquitetura e método de resolução de problema

---

# Agente de Policia

**Agente**



**raciocínio**

**Conhecimento:**

- leis
- comportamento dos indivíduos,...

**Objetivo:**

- fazer com que as leis sejam respeitadas

**Ações:**

- multar
- apitar
- parar, ...

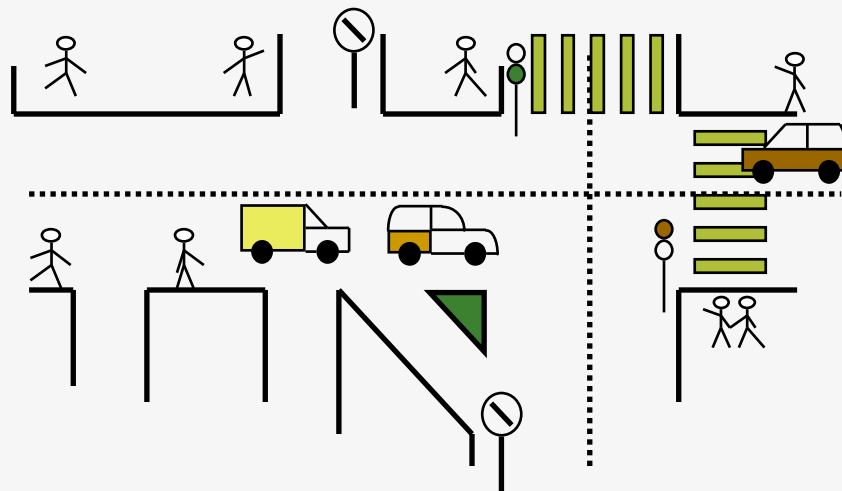
**percepção**



**execução**



**Ambiente**



# Exemplos de Agentes

<b>Agente</b>	<b>Dados perceptivos</b>	<b>Ações</b>	<b>Objetivos</b>	<b>Ambiente</b>
Diagnóstico médico	Sintomas, paciente, exames respostas, ...	Perguntar, prescrever exames, testar	Saúde do paciente, minimizar custos	Paciente, gabinete, ...
Análise de imagens de satélite	Pixels	imprimir uma categorização	categorizar corretamente	Imagens de satélite
Tutorial de português	Palavras digitadas	Imprimir exercícios, sugestões, correções, ...	Melhorar o desempenho do estudante	Conjunto de estudantes
Filtrador de mails	mensagens	Aceitar ou rejeitar mensagens	Aliviar a carga de leitura do usuário	Mensagens, usuários
Motorista de taxi	Imagens, velocímetro, sons	brecar, acelerar, dobrar, falar com passageiro, ...	Segurança, rapidez, economia, conforto,...	Ruas, pedestres, carros, ...
Músico de jazz	Sons seus e de outros músicos, grades de acordes	Escolher e tocar notas no andamento	Tocar bem, se divertir, agradar	Musicos, publico, grades de acordes

# Ambiente

## ■ **Classes de ambientes**

- ❑ Físico: robôs
- ❑ Software: softbots
- ❑ Realidade virtual (simulação do ambiente físico): softbots e avatares

## ■ **Propriedades de um ambiente**

- ❑ acessível (completamente observável) x inacessível (parcialmente observável)
- ❑ estático x dinâmico
- ❑ determinista x não-determinista
- ❑ discreto x contínuo
- ❑ episódico x não-episódico (seqüencial)
- ❑ tamanho: número de percepções, ações, objetivos,...

---

# Ambientes: propriedades (1/2)

- Acessível: quando os *sensores* do agente conseguem perceber o estado completo do ambiente.
  - Determinístico: o próximo estado do ambiente pode ser completamente determinado pelo estado atual e as ações selecionadas pelo agente.
  - Episódico: a experiência do agente é dividida em *episódios*. Cada episódio consiste em o agente perceber e então agir. Cada episódio não depende das ações que ocorreram em episódios prévios.
-

---

# Ambientes: propriedades (2/2)

- **Estático: o ambiente não muda enquanto o agente está escolhendo a ação a realizar.**
    - Semi-dinâmico: o ambiente não muda enquanto o agente delibera, mas o "score" do agente muda.
  - **Discreto: quando existe um número distinto e claramente definido de percepções e ações em cada turno.**
  - **Contínuo: percepções e ações mudam em um espectro contínuo de valores.**
-



# Exemplos de Ambientes

<b>Agente</b>	<b>acessível</b>	<b>determinista</b>	<b>episódico</b>	<b>estático</b>	<b>discreto</b>
<b>xadrez sem relógio</b>	Sim	Sim	Não	Sim	Sim
<b>xadrez com relógio</b>	Sim	Sim	Não	Semi	sim
<b>gamão</b>	sim	não	não	sim	sim
<b>motorista de taxi</b>	Não	Não	Não	Não	Não
<b>médico</b>	Não	Não	Não	Não	Não
<b>tutor</b>	Não	Não	Não	Não	Sim
<b>Analisador de imagem</b>	Sim	Sim	Sim	Semi	Não
<b>Busca na web</b>	Não	Não	Sim	Não	Sim
<b>Filtrador de mail</b>	Sim	Não	Sim	Não	Sim
<b>Músico</b>	Sim	Não	Não	Não	Não

+ O Tamanho do ambiente é dado por:  
número de percepções, ações e objetivos possíveis

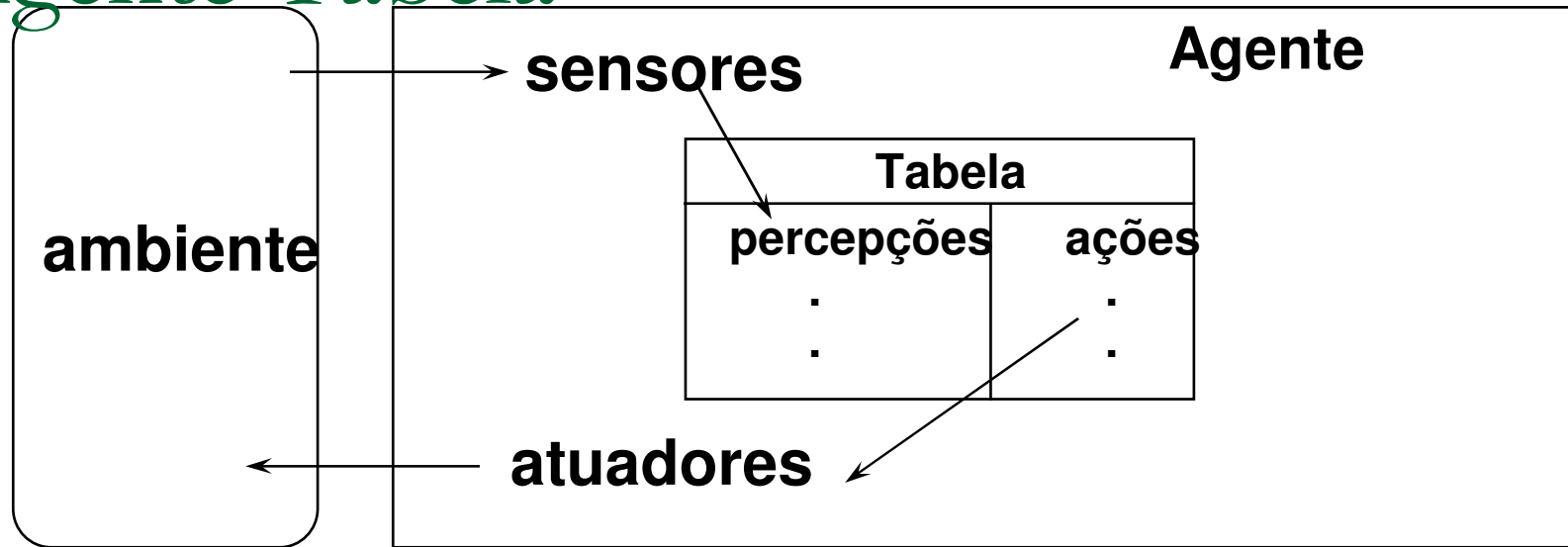
# Algoritmo Básico

- função agenteSimples (percept) retorna ação  
memória := atualizaMemória (memória, percept)  
ação := escolheMelhorAção(memória)  
memória := atualizaMemória (memória, ação)  
retorna ação
- Arquiteturas
  - Agente tabela
  - Agente reativo simples
  - Agente reativo baseado em modelos
  - Agente baseado em objetivos
  - Agente baseado em utilidade
  - Agente com aprendizagem



autonomia  
complexidade

# Agente Tabela



## ■ Limitações

- ❑ Mesmo Problemas simples -> tabelas muito grandes
  - ex. xadrez  $30^{100}$
- ❑ Nem sempre é possível, por ignorância ou questão de tempo, construir a tabela
- ❑ Não há autonomia nem flexibilidade
- ❑ Este agente só tem fins didáticos!!! Não vale nem a pena pensar nele

## ■ Ambientes

- ❑ acessível, determinístico, episódico, estático, discreto e minúsculo!

---

## Programa de Agente: Tabela de Consulta

**função** Agente-Tabela(*percepção*) **retorna** *ação*  
**estática:** *sequência*, inicialmente uma sequência vazia  
          *tabela*, uma tabela indexada por sequência de  
          *percepção*, inicialmente vazia  
  
insira *percepção* final *sequência*  
      *ação* <- procura\_tabela(*sequência*, *tabela*)  
**retorna** *ação*

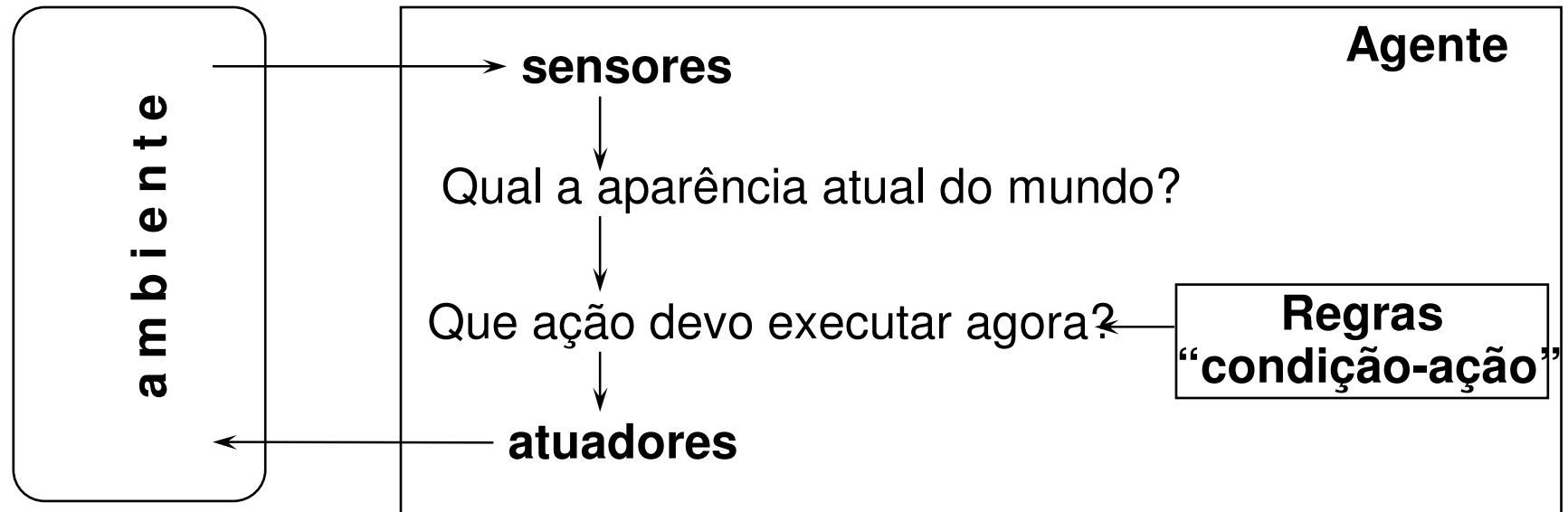
---

---

## Por que a abordagem por tabela falha?

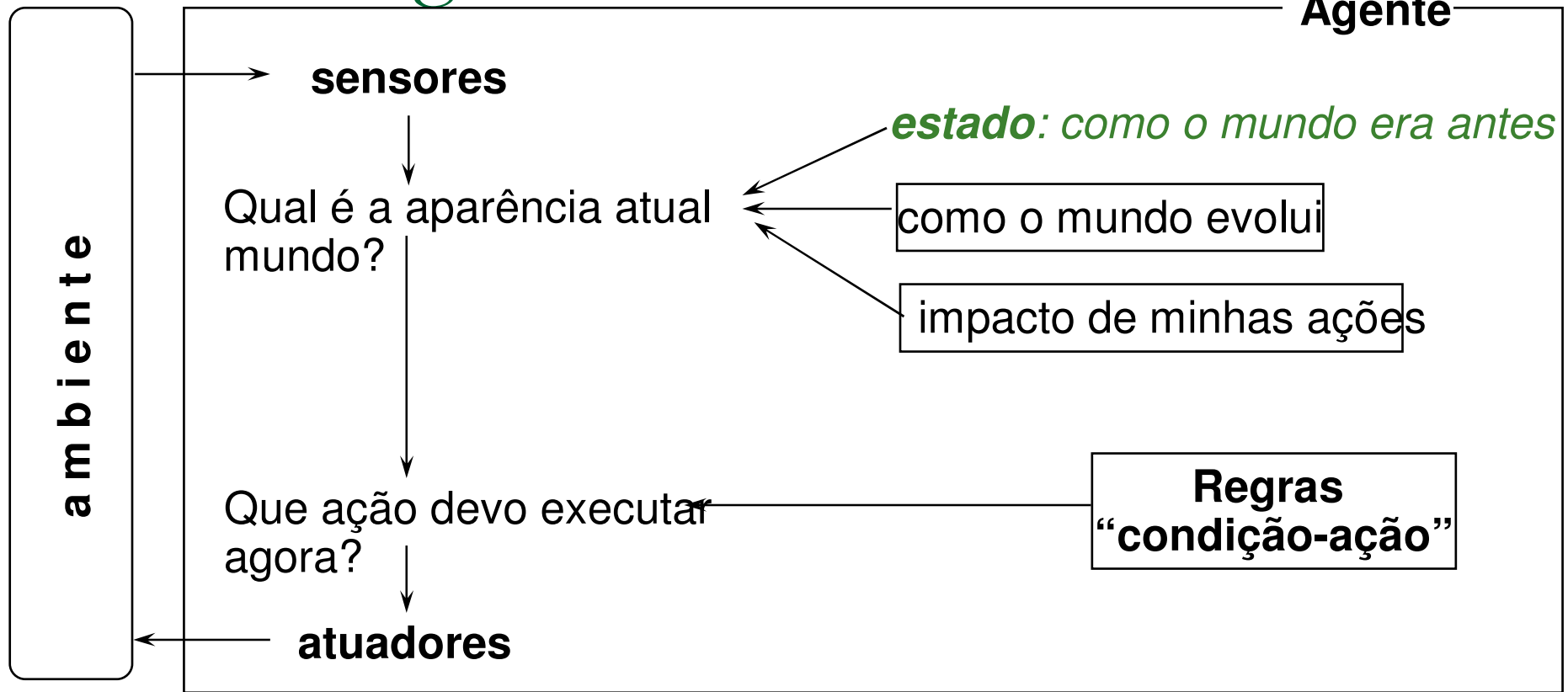
- ***Número excessivo de entradas***
    - $35^{100}$  entradas necessárias para um jogo de xadrez
  - ***Leva muito tempo para construir a tabela***
  - ***Sem autonomia***
    - Mudanças no ambiente podem tornar o agente totalmente inútil
  - ***Aprendizagem leva muito tempo***
    - Há tantas possibilidades que aprender o valor correto de cada sequência vai consumir tempo “infinito”
-

# Agente Reativo Simples



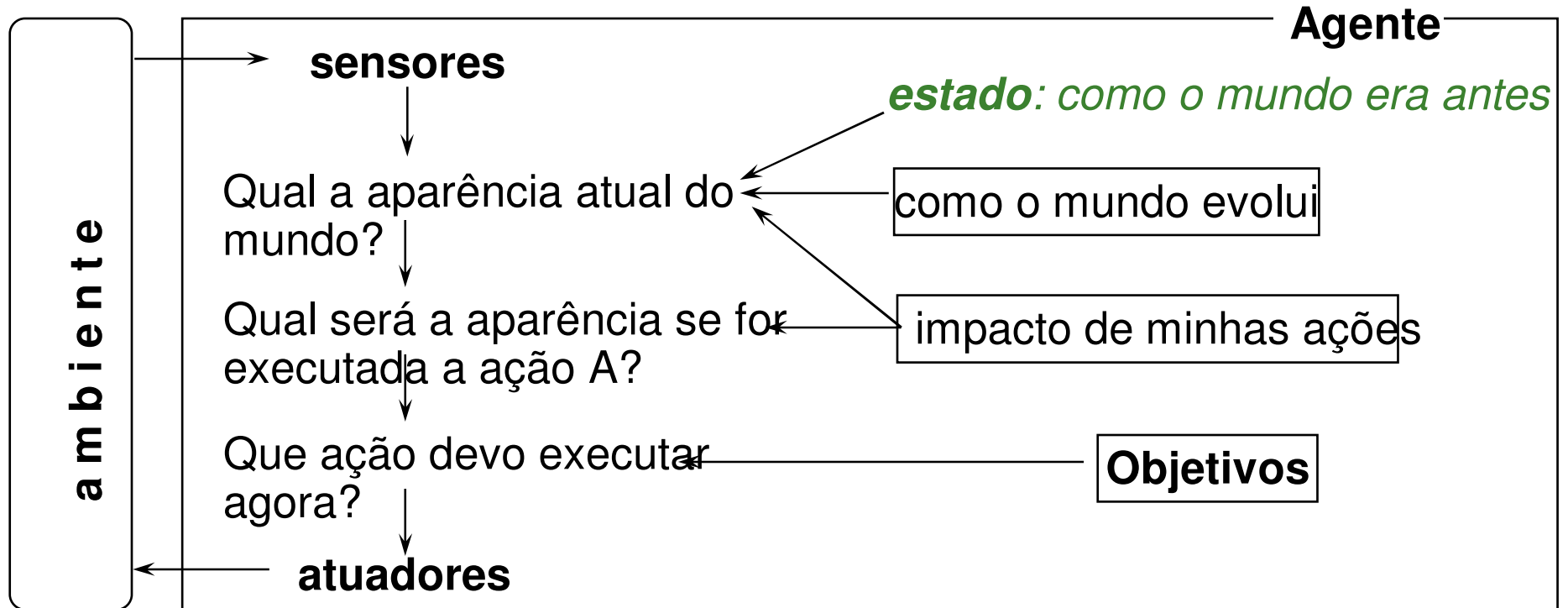
- Vantagens e desvantagens
  - Regras condição-ação: representação inteligível, modular e eficiente
    - ex. **Se** velocidade > 60 **então** multar
  - Não pode armazenar uma seqüência de percepções, pouca autonomia
- Ambientes:
  - Reflexo imprescindível em ambientes dinâmicos
  - Acessível, episódico, pequeno

# Agente Reativo Baseado em Modo



- Desvantagem: **pouca autonomia**
  - não tem objetivo, não encadeia regras
- Ambientes: **determinístico e pequeno**
  - Ex. Tamagotchi

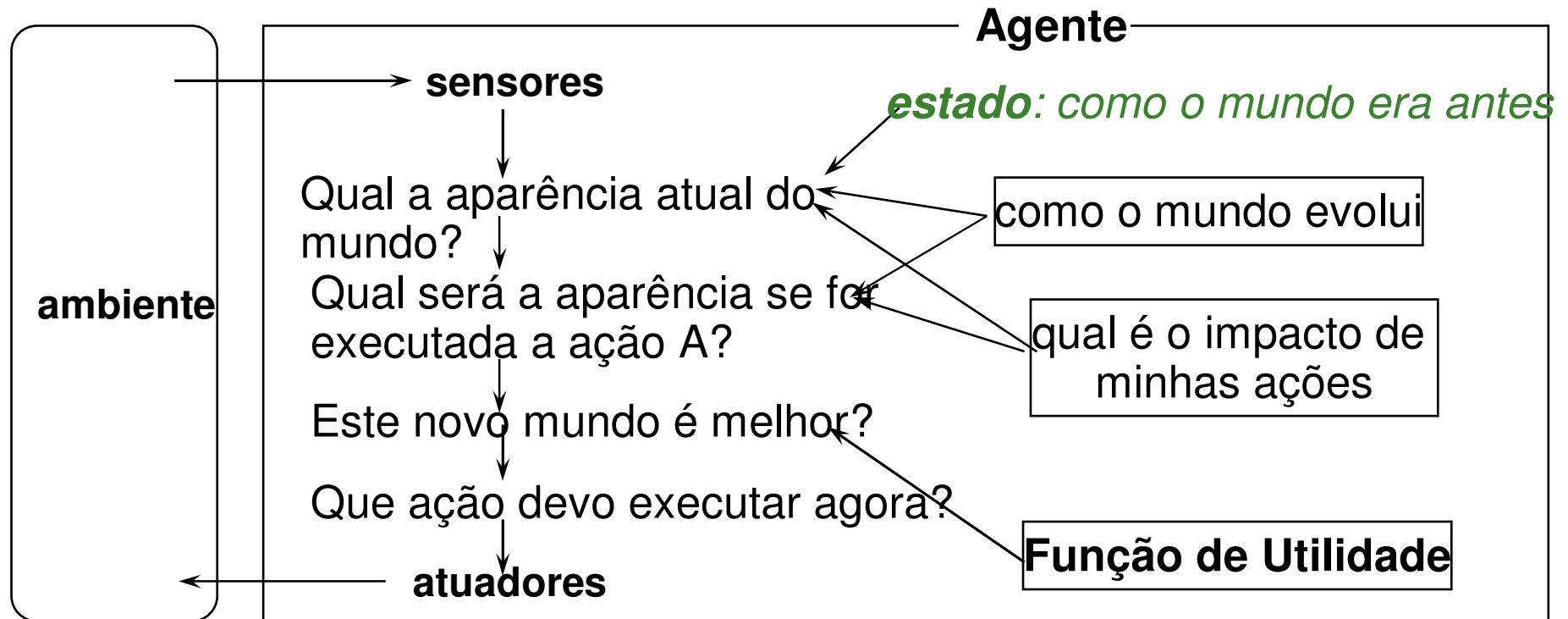
# Agente Baseado em Objetivo



- Vantagens e desvantagens:
  - Mais *complicado e ineficiente*, porém mais *flexível, autônomo*
  - Não trata objetivos conflitantes
- Ambientes: determinístico
  - ex.: xaque-mate no xadrez

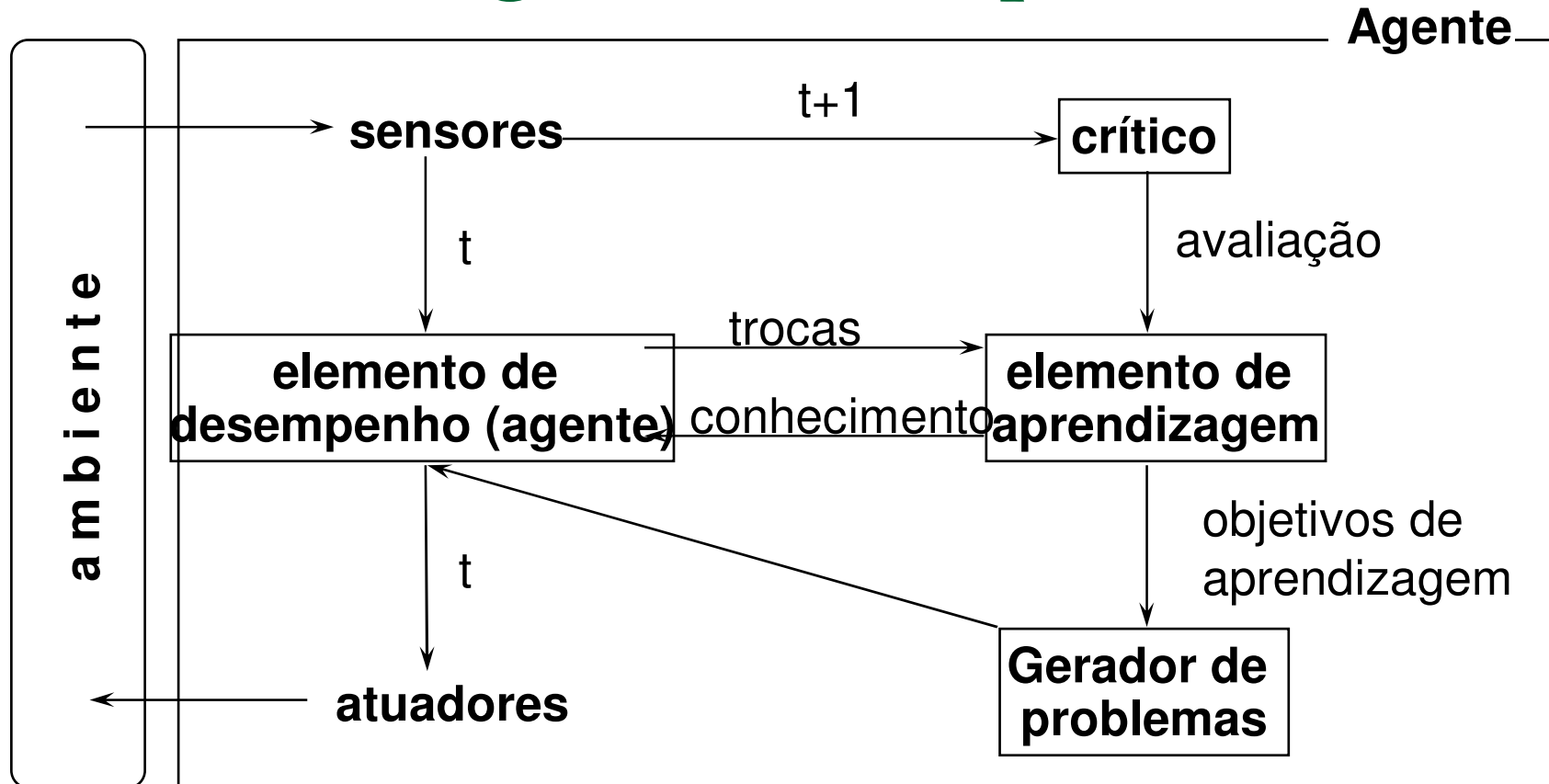


# Agente Baseado em Utilidade



- Ambiente: **sem restrição**
- Desvantagem: **não tem adaptabilidade**
- Ex. **motorista**

# Agente com Aprendizado



- Ambiente: **sem restrição**
- Vantagem: **tem adaptabilidade (aprende)**
- Ex. **motorista sem o mapa da cidade**

---

# Simulação do Ambiente

- Às vezes é mais conveniente simular o ambiente
    - mais simples
    - permite testes prévios
    - evita riscos, etc...
  - O ambiente (programa)
    - recebe os agentes como entrada
    - fornece repetidamente a cada um deles as percepções corretas e recebe as ações
    - atualiza os dados do ambiente em função dessas ações e de outros processos (ex. dia-noite)
    - é definido por um estado inicial e uma função de atualização
    - deve refletir a realidade
-

# Simulação de ambiente

- função **simulaAmbiente** (**estado**,  
**funçãoAtualização**,**agentes**,**final**)

**repita**

**para** cada *agente* em *agentes* **faça**

Percept[*agente*] := pegaPercepção(*agente*,*estado*)

**para** cada *agente* em *agentes* **faça**

Action[*agente*] := Programa[*agente*] (Percept[*agente*])

*estado* := funçãoAtualização(*ações*, *agentes*, *estado*)

*scores* := avaliaDesempenho(*scores*,*agente*,*estado*) //opcional

**até** *final*

- Observação:
  - não cair em tentação “roubando” do ambiente a descrição do que aconteceu. Usar a memória do agente!

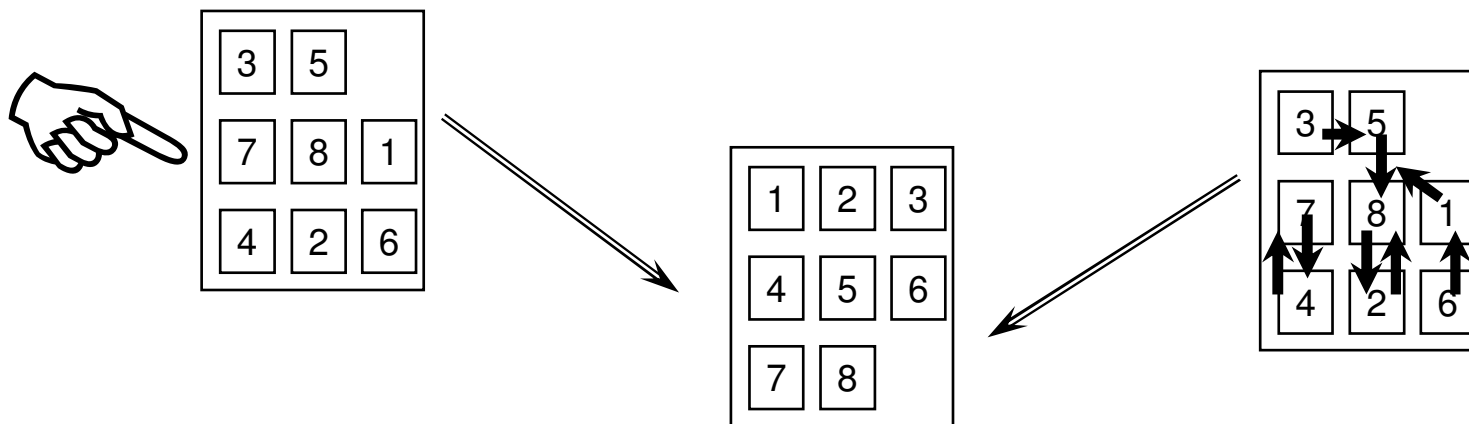
---

# Inteligência Coletiva

- Porque pensar a inteligência/racionalidade como propriedade de um único indivíduo?
  - Não existe inteligência ...
    - ❑ Em um time de futebol?
    - ❑ Em um formigueiro?
    - ❑ Em uma empresa (ex. correios)?
    - ❑ Na sociedade?
  - Solução: IA Distribuída
    - ❑ Agentes simples que juntos resolvem problemas complexos tendo ou não consciência do objetivo global
    - ❑ Proposta por Marvin Minsky e em franca expansão...
    - ❑ o próprio ambiente pode ser modelado como um agente
-

# IA Distribuída: dois tipos de sistemas

- Resolução distribuída de problemas
  - consciência do objetivo global e divisão clara de tarefas
  - Exemplos: Robótica clássica, Busca na Web, Gerência de sistemas distribuídos, ...
- Sistemas Multi-agentes
  - não consciência do objetivo global e nem divisão clara de tarefas
  - Exemplos: n-puzzle, futebol de robôs, balanceamento de carga, robótica, ...



---

# Questões

- Questões centrais
    - comunicação
    - negociação (ex. compra-venda na Web)
    - estados mentais
    - crença, ...
  - Tensão (trade-off)
    - Quanto mais agentes, mais simples (sub-dividido) fica o problema
    - No entanto, mais complexa fica a comunicação e coordenação entre os agentes
-

---

*Evolução da noção de Agente  
além das fronteiras da IA....*

---



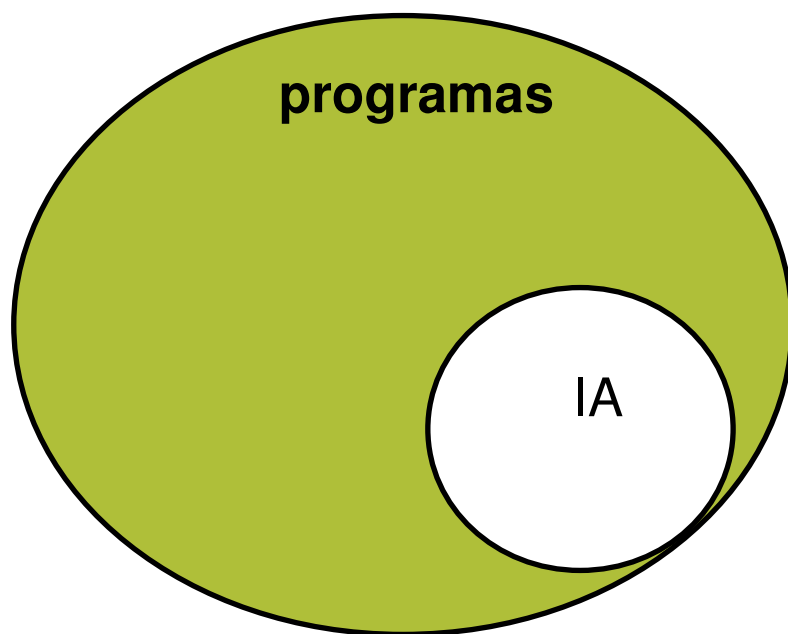
---

# Agente: ainda não há uma definição única

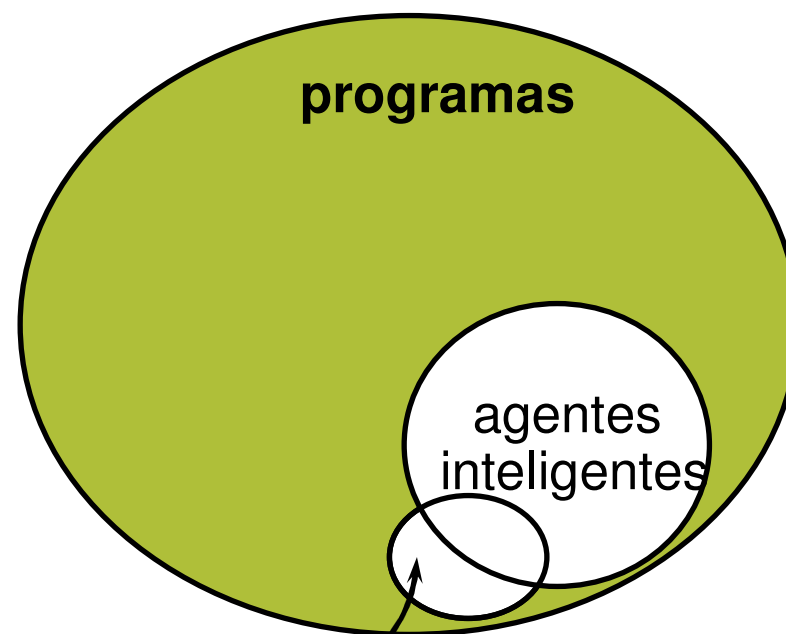
- **IBM:** Intelligent agents are software entities that carry out some set of operations *on behalf of an user*, and in doing so employ some *knowledge representation* of the *user's goals or desires*
  - **KidSim:** Agent is a *persistent software entity* (agents have their own ideas about how to accomplish tasks) dedicated to a *specific purpose* (smaller than multifunctions applications)
  - **SodaBot:** Software agents are programs that engage in *dialogs* to *negotiate* and *coordinate* transfer of information
-

# Técnicas & Problemas

**Antes....**



**Agora**



agentes móveis,  
agentes de software,  
robôs, ...

# Propriedades

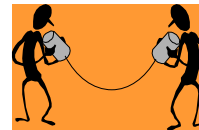
- Autonomia (IA)
  - raciocínio, comportamento guiado por objetivos
  - reatividade



- Adaptabilidade & aprendizagem (IA)



- Comunicação & Cooperação (IA)



- Personalidade (IA)



- Continuidade temporal



- Mobilidade



---

# Agentes na Internet

- Categorias por Tipos de Serviços:
    - ❑ Agentes de Busca e Recuperação (ex. Google)
    - ❑ Agentes que Filtram Informações (ex. KOM)
    - ❑ Agentes de Entrega Off-line (ex. PointCast)
    - ❑ Agentes Notificadores (ex. URL-Minder)
    - ❑ Agentes de Suporte ao Comércio (ex. BargainBot)
    - ❑ Agente corretor (interoperabilidade - ACL)
    - ❑ Outros...
-

---

# Conclusões

- Agentes em IA
    - Metodologia (metáfora) para projeto de sistemas
    - Sistemas multi-agentes e robótica
  - Agentes em computação
    - Adoção de uma nova metáfora (antropomórfica e sociológica). Extrapolação de OOP
      - IA: autômato -> mente
      - Agentes: objetos -> pessoas
    - Integração de técnicas de IA
    - Novas tecnologias próprias à Web (ex. mobilidade)
    - Marketing (moda)
  - Agentes: técnica ou metodologia ?
-

---

# Desenvolvimento de software inteligente

- Projeto:
    - Modelar tarefa em termos de ambiente, percepções, ações, objetivos e utilidade
    - Identificar o tipo de ambiente
    - Identificar a arquitetura de agente adequada ao ambiente e tarefa
  - Implementação
    - o gerador e o simulador de ambientes
    - componentes do agente (vários tipos de conhecimento)
    - Testar o desempenho com diferentes instâncias do ambiente
-

---

## Referências

- Stuart Russel and Peter Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, 1995.
    - Chapter 2: Intelligent Agents
-

---

# Fim

- Obrigado pela presença!

