

# Agenti

- Conoscono l'**ambiente**
- Prendono decisioni **reagendo** a cambiamenti nell'**ambiente**
- Possono **imparare** da situazioni precedenti
- **Attuatori**: Effettuano **Azioni** → sull'**ambiente**
- **Sensori**: Ottengono **Percetti** ← dall'**ambiente**

## PAGE: Percepts, Actions, Goals and the Environment

Classificazione che si basa sull'ambiente:

- Fully (Accessibile) VS partially Observable
  - Se l'agente può ottenere lo stato completo [tutto quello che mi serve per il goal] dell'ambiente con i suoi sensori ad ogni momento nel tempo
  - il Fully accessible environment è facile per l'agente perché non c'è bisogno di mantenere uno stato interno per tracciare la storia del mondo
- Deterministic VS Stochastic [non-deterministico]
  - Se lo stato dell'agente, inclusa l'azione selezionata, determinano completamente il prossimo stato dell'ambiente allora l'ambiente è deterministico ⇒ L'agente non si deve preoccupare di cambiamenti incerti
  - In un ambiente stocastico ci sono delle variabili random che entrano in gioco con probabilità per decidere quali azioni compiere
  - Casuale ≠ stocastico: il primo assume una distribuzione uniforme di casualità
- Static VS Dynamic
  - Se i cambiamenti nell'ambiente avvengono solo se causati dall'agente stesso e non da altri fattori allora l'ambiente è statico e quindi l'agente può effettuare pianificazione per ottenere il risultato, e non ha bisogno di conoscere bene l'ambiente per scegliere la prossima azione.
- Discrete VS Continuous

- Un ambiente discreto è formato da un numero discreto di percetti e permette di affrontare il problema tramite una matrice che associa un certo stato ad un'azione [sia per gli ambienti finiti che quelli discreti in generale]
- Per alcuni casi specifici gli ambienti continui sono comunque interessanti [e.g.: Parcheggio automatico]
- Episodic VS Sequential
  - Se per scegliere la prossima azione è solo necessario il percetto corrente allora l'ambiente è episodico
  - In un ambiente sequenziale l'agente necessita della memoria delle azioni passate per determinare la prossima azione.
- Single-agent VS Multi-agent
  - Se un solo agente è nell'ambiente → Allora l'ambiente è single-agent
  - Gli ambienti multi-agente sono normalmente più challenging che quelli single-agent, si dividono in:
    - **Cooperativo:** Agenti cooperano per ottenere un "joint goal"
    - **Competitivo:** Gli agenti cercano di ottenere i loro obiettivi con relazioni di conflitto con altri agenti
    - **Strategico:** Se un ambiente è deterministico a meno delle azioni di altri agenti. [Ci possono essere agenti che competono e che cooperano]
  - Agente + Ambiente  $\Rightarrow$  Mondo

## Classificazione di Weiss

- **Reactive agents:** La decisione di quale azione eseguire è descritta da una mappatura diretta da condizione ad azione.
- **Layered(-architecture) agents:** La decisione di quale azione eseguire utilizza layer, per ognuno dei quali ragiona in modo reattivo su diversi livelli di astrazione
- **Logic-based agents:** Le decisioni di quali azioni da eseguire sono ottenute tramite ragionamento logico [Agente ha obiettivo e prende decisioni per raggiungerlo]
- **Belief-Desire-Intention (BDI) agents:** Le decisioni dipendono dal processamento dei dati che rappresentano i beliefs [affermazione o proposizione]

che viene assunta come vera], le intenzioni dell'agente piani di lavoro, I desideri sono gli obbiettivi che gli vengono imposti ma che non sono ancora diventate intenzioni

## **Altra classificazione:**

- **Simple (Reflex) Agents**

- Agente semplice che agisce sulla base del percetto corrente e basta

- **Model-Based (Reflex) Agents**

- model-based agent:
  - Mantiene un modello interno che dipende dalla storia dei percetti
  - Utilizza il modello per predirre l'impatto delle azioni future
- Sono utilizzabili anche in ambienti parzialmente osservabili

- **Goal-Based Agents**

- Sono agenti basati su modello con obbiettivi espliciti
  - Scelgono cosa fare per raggiungere gli obbiettivi correnti che descrivono situazioni che sono correntemente considerate desiderabili.
  - Sono permessi multipli goal non in conflitto

- **Utility-Based Agents**

- Goal-Based agents che:
  - Hanno una funzione di utility per associare utilità con goals
  - Le azioni che sono scelte per effettuare i goal con la massima utilità