CSE 604

Artificial Intelligence

Chapter 2: Intelligent Agents

Adapted from slides available in Russell & Norvig's textbook webpage

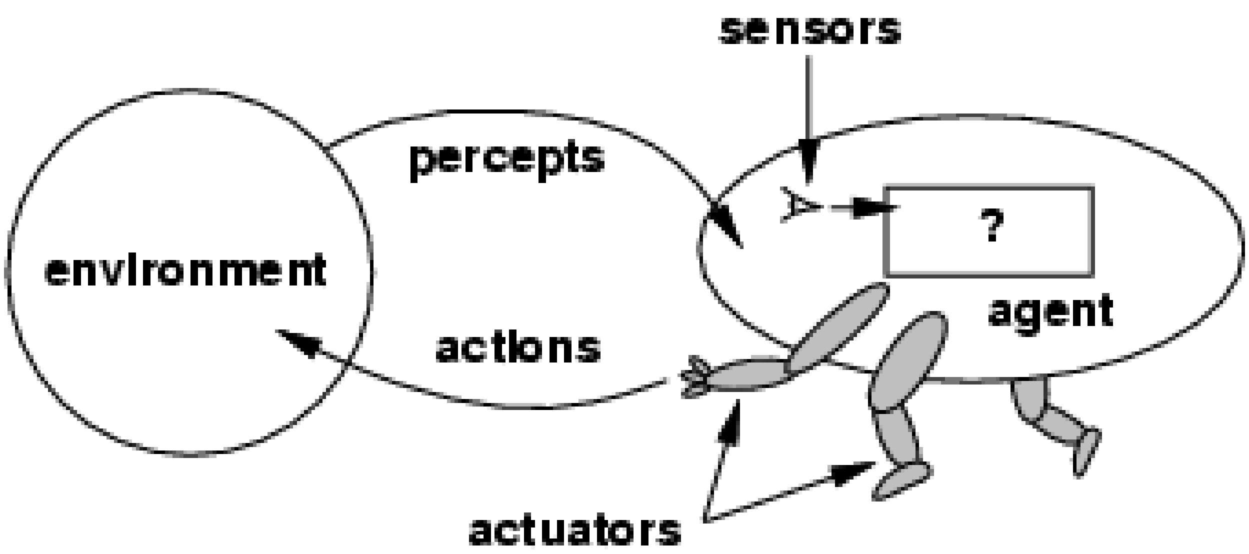
IIT
University of Dhaka

Outline

- Agents and environments
- Rationality
- PEAS (Performance measure, Environment, Actuators, Sensors)
- Environment types
- Agent types

Agents

- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators



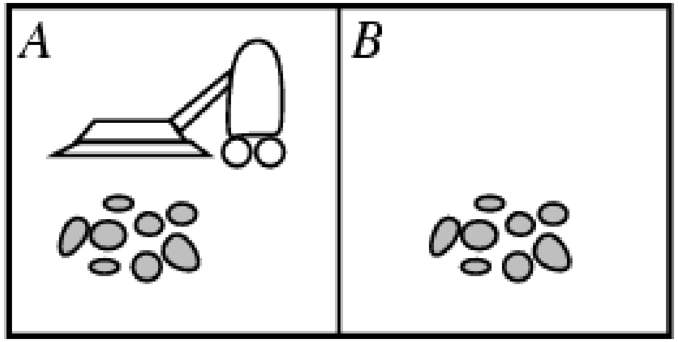
Examples of Agents

- **Human agent**: eyes, ears, and other organs for sensors; hands, legs, mouth, and other body parts for actuators

- **Robotic agent**: cameras and infrared range finders for sensors; various motors for actuators

- **Software agent**: receives keystrokes, file contents, network packets as sensory inputs; acts by displaying on screen, writing files etc.

Vacuum-cleaner world



- **Percepts**: location and contents, e.g., [A, Dirty]
- **Actions**: *Left*, *Right*, *Suck*, *NoOp*

Vacuum-cleaner world..

Percept sequence	Action
$[A, Clean]$	$Right$
$[A, Dirty]$	$Suck$
$[B, Clean]$	$Left$
$[B, Dirty]$	$Suck$
$[A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Dirty]$	$Suck$
$\vdots$	$\vdots$
$[A, Clean], [A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Clean], [A, Dirty]$	$Suck$
$\vdots$	$\vdots$

Figure 2.3 Partial tabulation of a simple agent function for the vacuum-cleaner world shown in Figure 2.2.

Rational Agent

- A rational agent chooses whichever action maximizes the expected value of the performance measure given the percept sequence to date
- Rational ≠ omniscient
  - percepts may not supply all relevant information
- Rational ≠ clairvoyant
  - action outcomes may not be as expected
- Hence, rational ≠ successful
- Rational ⇒ exploration, learning, autonomy

Rational agents

- An agent should strive to "do the right thing", based on what it can perceive and the actions it can perform.

- Performance measure: An objective criterion for success of an agent's behavior

- E.g., performance measure of a vacuum-cleaner agent could be :
  - amount of dirt cleaned up
  - amount of time taken
  - amount of electricity consumed

PEAS

- Specifying the task environment:
  - **P**erformance measure
  - **E**nvironment
  - **A**ctuators
  - **S**ensors

PEAS

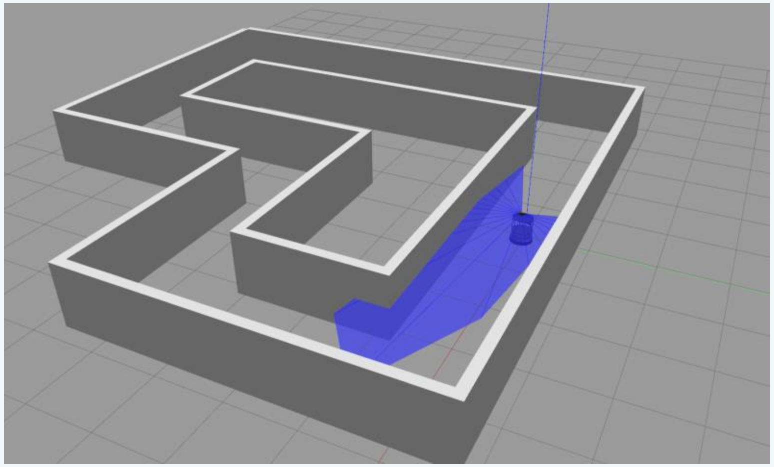
- Agent: **Part-picking robot**
  - **Performance measure**: % of parts in correct bins
  - **Environment**: Conveyor belt, parts, bins
  - **Actuators**: Jointed arm and hand
  - **Sensors**: Camera, joint angle sensors

PEAS

- Agent: **Automated car**
  - **Performance measure**: Safe, fast, legal, comfortable trip
  - **Environment**: Roads, other traffic, pedestrians
  - **Actuators**: Steering wheel, accelerator, brake
  - **Sensors**: Camera, GPS, Speedometer, engine sensor

Environment types

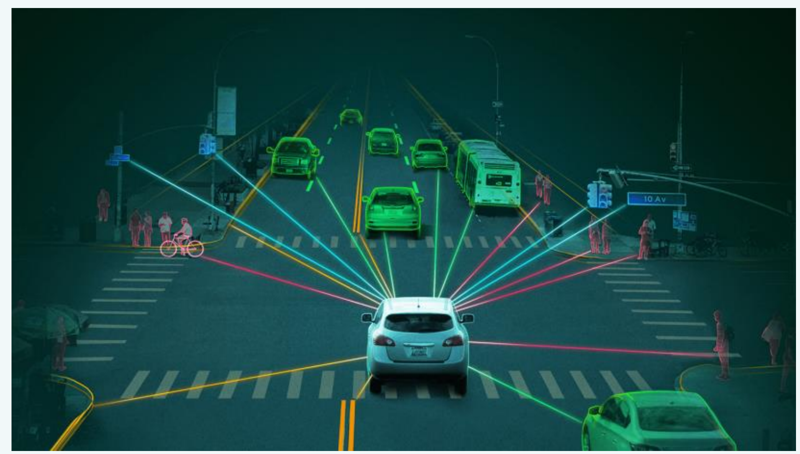
- Fully observable vs. partially observable
- Single agent vs. multiagent
- Deterministic vs. stochastic
- Episodic vs. sequential
- Static vs dynamic
- Discrete vs continuous

Fully Observable	Partially Observable
Agent can observe (see/hear/perceive) all relevant information from the environment	Agent can observe only partial information from the environment
	

Single Agent	Multiagent
Our agent is the only intelligent agent in the environment	There are multiple intelligent agents which can be either cooperative or competitive
	

Deterministic	Stochastic
Agent can fully determine the outcome of it’s action (next step, not necessarily the full task)	Agent is uncertain of the outcome of it’s action
	

Episodic	Sequential
Agent's actions are completely independent of each other, not linked to past or future actions	Agent's actions are dependent on it's past/future actions. The actions form a sequence.
	

Static	Dynamic
While the agent is in the process of taking it’s action, the environment doesn’t change	The environment is constantly changing even when the agent is taking an action
	

Discrete	Continuous
Agent's task can be broken down into discrete set of actions (you can make a list of agent's actions A_1, A_2, ..., A_n)	Actions are happening continuously and can not be listed, i.e., you cannot say where one action ends and the other begins
	

Environment types

	Chess with a clock	Chess without a clock	Taxi driving
Fully observable	Yes	Yes	No
Deterministic	Strategic	Strategic	No
Episodic	No	No	No
Static	Semi	Yes	No
Discrete	Yes	Yes	No
Single agent	No	No	No

- The environment type largely determines the agent design
- The real world is (of course) partially observable, stochastic, sequential, dynamic, continuous, multi-agent

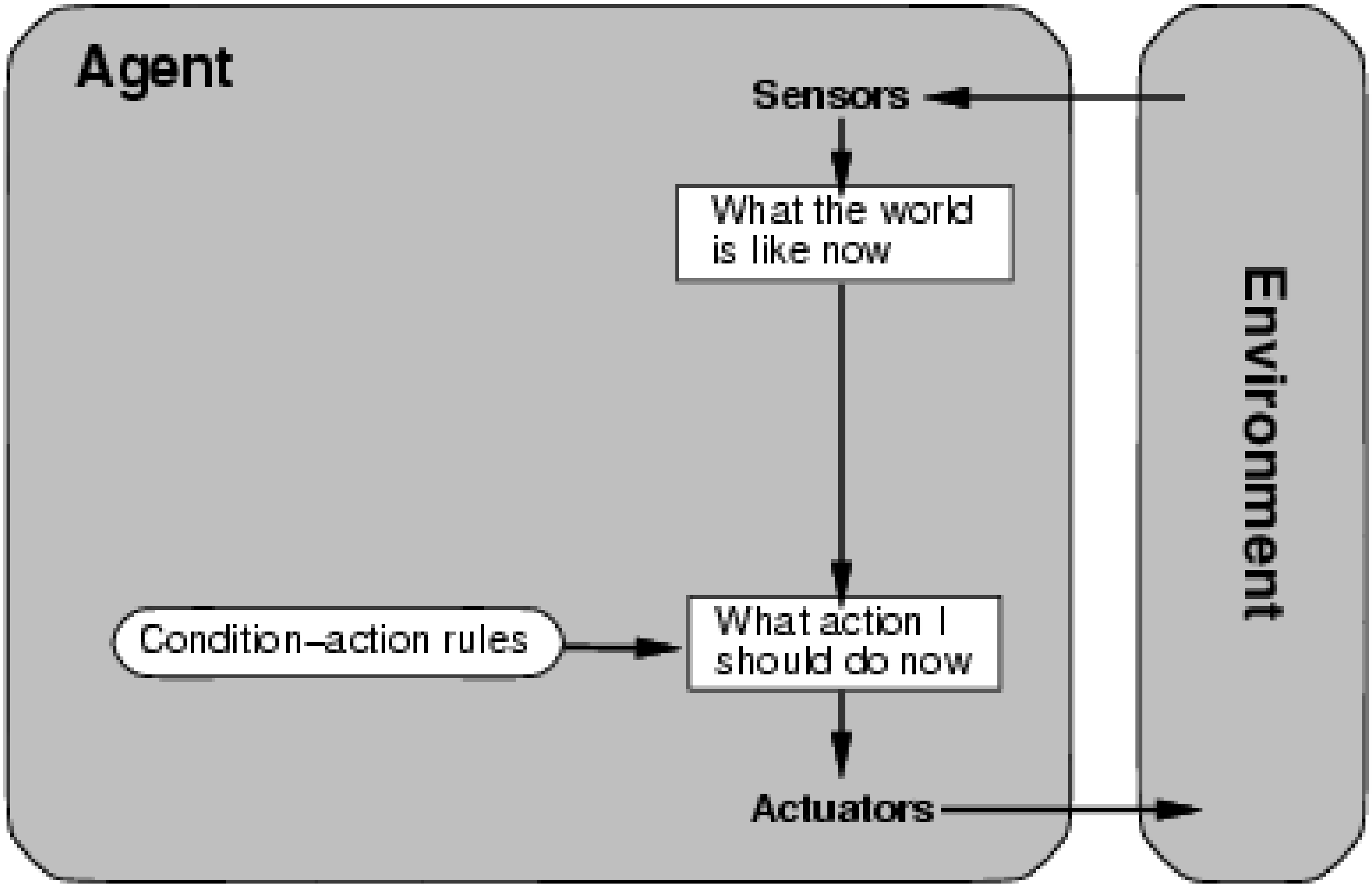
Check this

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle						
Chess with a clock						
Poker						
Backgammon						
Taxi driving						
Medical diagnosis						
Image analysis						
Part-picking robot						
Refinery controller						
Interactive English tutor						

Agent types

- Four basic types in order of increasing generality:
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
  - Utility-based agents

Simple reflex agents



Simple reflex agents

Thermostats

It’s 6pm in the winter? Crank that heat up. It’s noon in the summer? This simple reflex agent, with its limited intelligence, will turn on the AC.

Automatic doors

While its perceived intelligence is low, automatic doors are often examples of simple reflex agents. This AI agent senses a human in front of a door, and it opens.

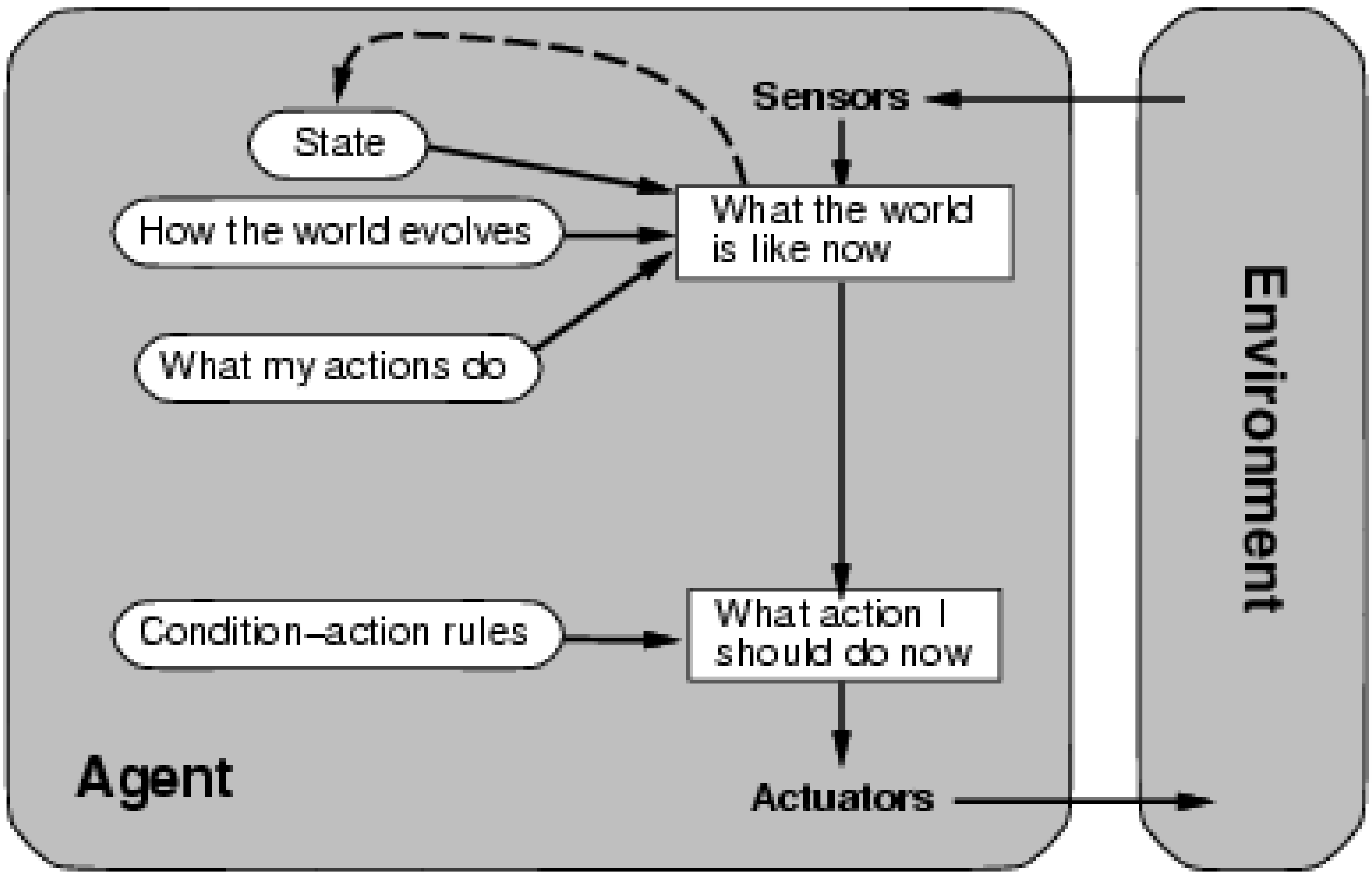
Smoke detectors

This AI agent operates from your kitchen ceiling. Yep, it’s a simple reflex agent, too.

Basic spam filters

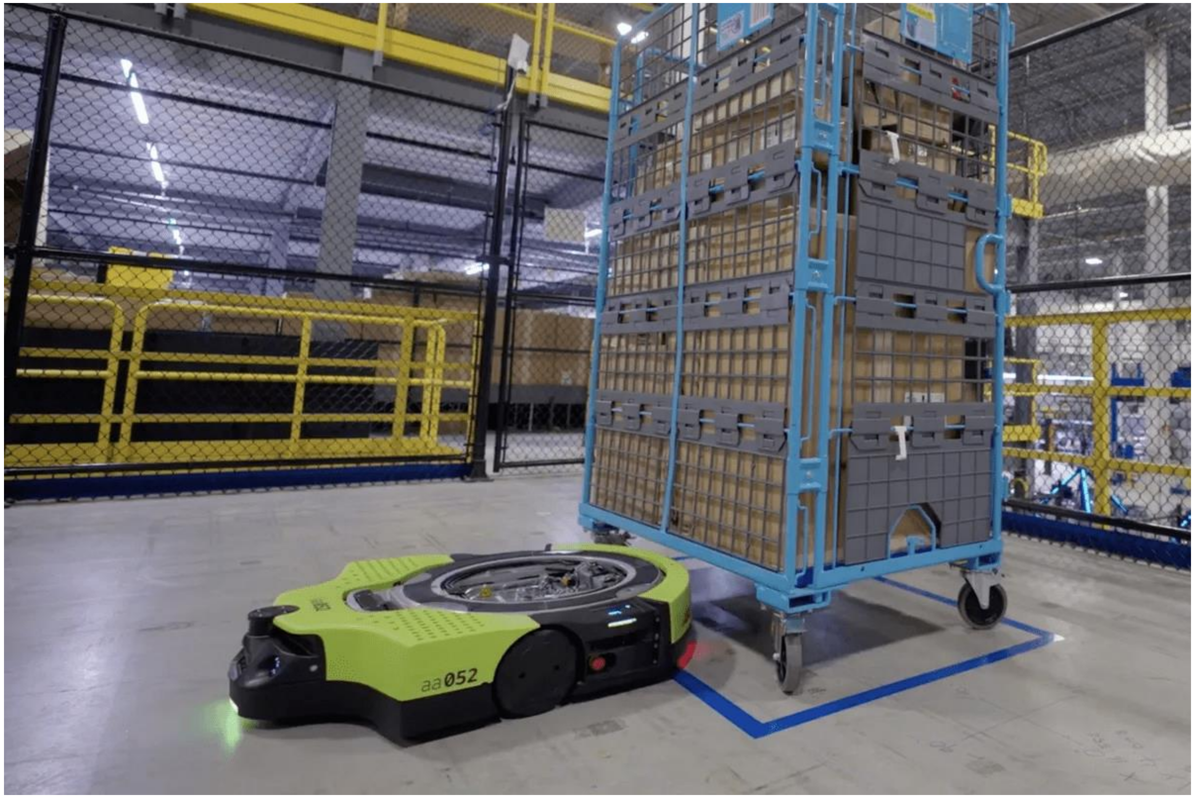
Some agents in artificial intelligence have been helping us daily for years. The email spam filter is one of these. Basic versions don’t use natural language processing, but rather keywords or the sender’s reputation.

Model-based reflex agents



Model Based Reflex Agent

A robot vacuum updating its map to avoid obstacles



Robots navigating warehouses or delivering packages use internal maps of their operations management. They update their model when new obstacles appear, ensuring efficient pathfinding and avoiding collisions.

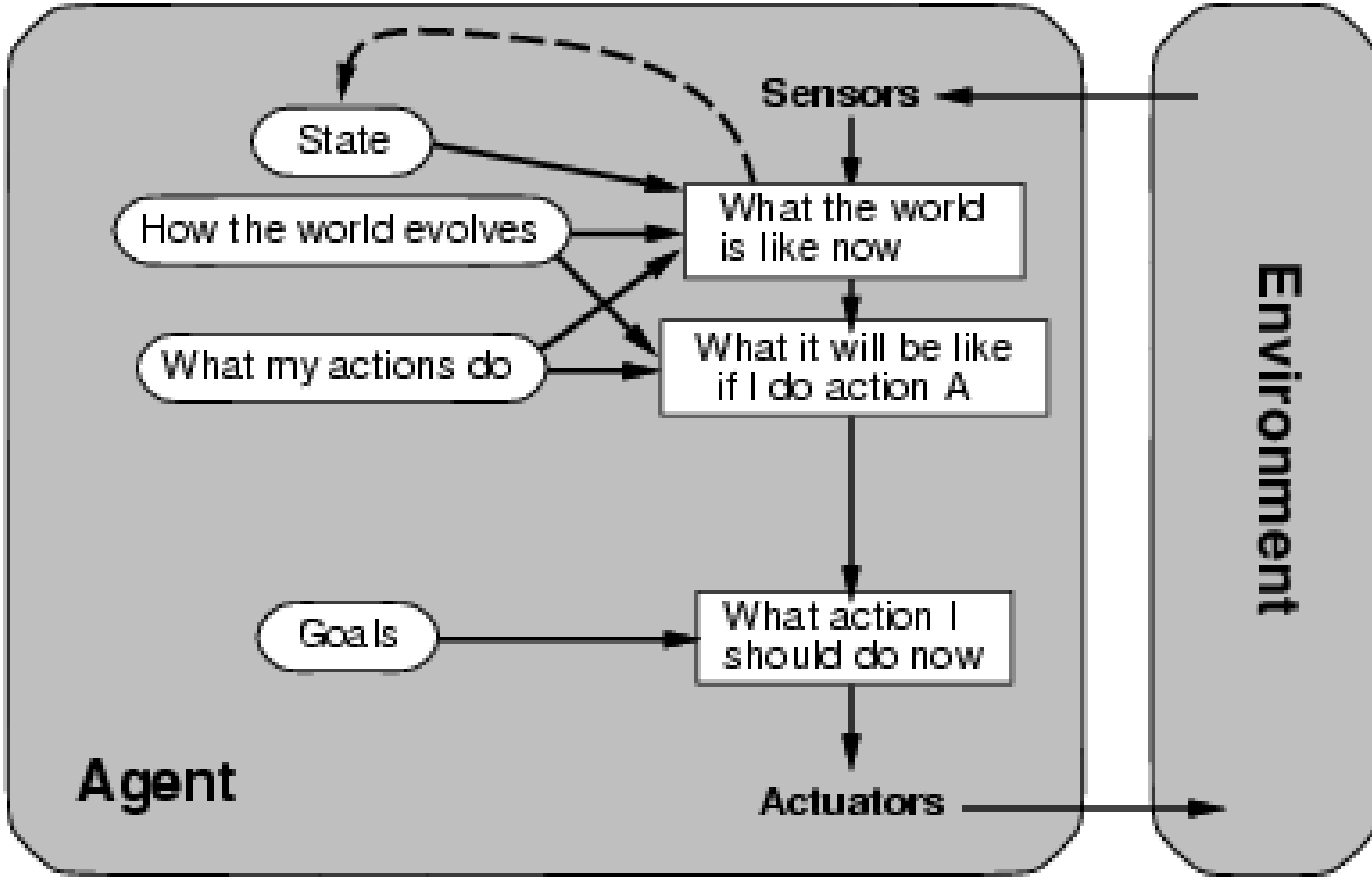
Model Based Agent..

- This is a smarter version of the reflex agent. Instead of just reacting, it keeps an internal model of the world.
- Think of a robotic vacuum cleaner:
- It doesn’t just react to dirt it sees right now.
- It remembers where it’s been, where obstacles are, and which areas are already clean.

Simple vs Model based agent

Aspect	Simple Reflex Agents	Model-Based Reflex Agents
Decision Basis	Immediate input only	Current input + internal model
Memory	None	Retains past states to inform decisions
Environmental Suitability	Effective in fully observable, static environments	Better for dynamic or partially observable environments
Example	A basic vending machine dispensing snacks based on button press	A robot vacuum updating its map to avoid obstacles

Goal-based agents



"how the world works"—whether implemented in simple Boolean circuits or in complete scientific theories—is called a model of the world

Goal Based Agent..

Imagine a self-driving car:

- Its goal is to reach a destination.
- It simulates possible actions (turn left, go straight).
- It predicts which action will best achieve the goal.
- The shift here is from "what should I do now?" to "what will get me closer to my goal?" These agents are common in robotics and simulations where objectives are clear but the environment is dynamic.

Example

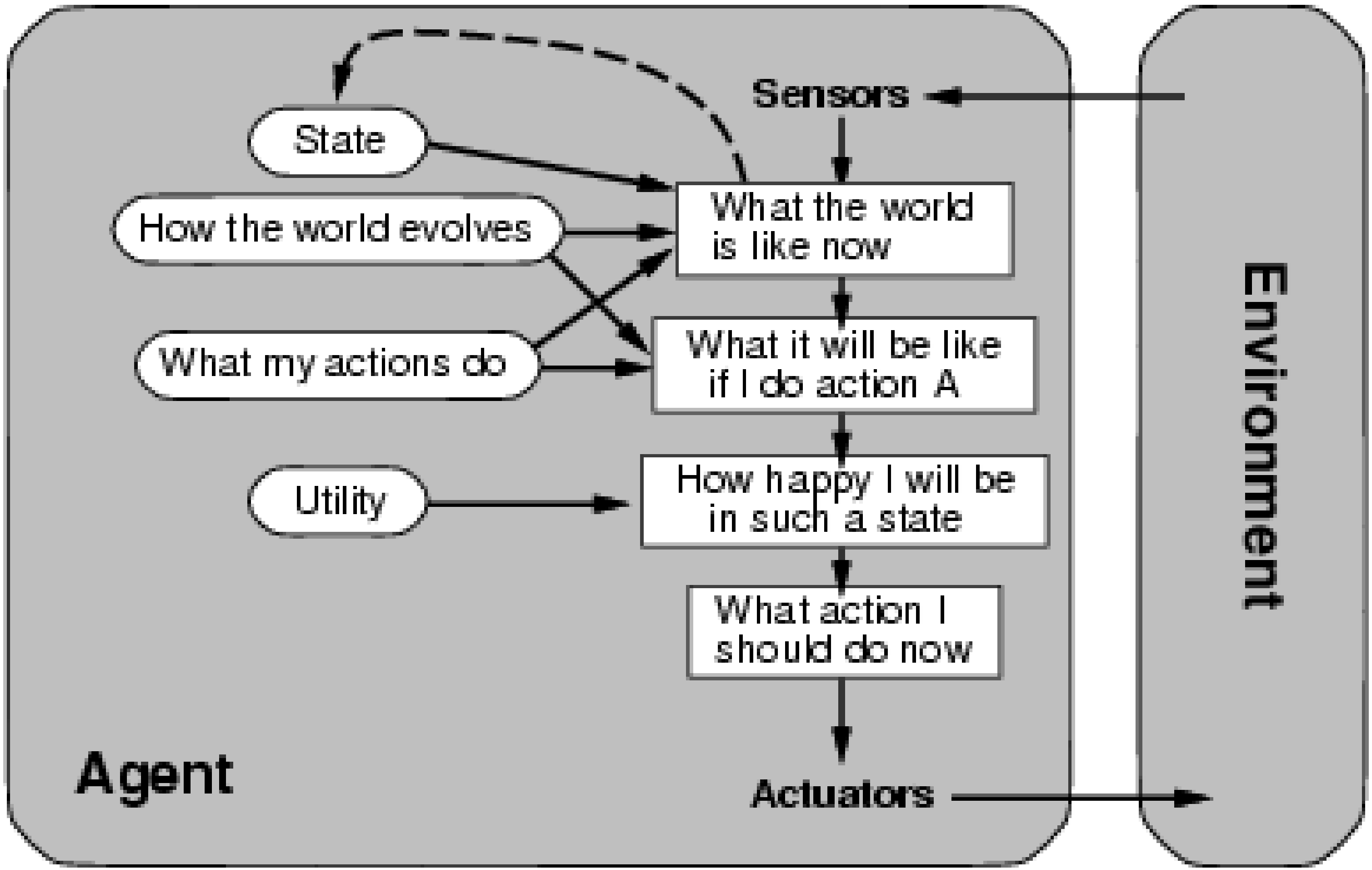
Game AI

In game AI, goal-based agents control non-player characters, enabling them to exhibit intelligent behavior and strategies. These agents enhance the gaming experience by creating challenging and realistic interactions for players.

Autonomous Vehicles

In autonomous vehicles, goal-based agents are used to navigate roads, avoid obstacles, and follow traffic rules. These agents ensure safe and efficient operation, contributing to the development of self-driving cars and other autonomous transport systems.

Utility-based agents



Utility

- The utility function of these intelligent agents is a mathematical representation of its preferences. The utility function maps to the world around it, deciding and ranking which option is the most preferable. Then a utility agent can choose the optimal action.

Financial Trading

Utility-based agents are well-suited for stock and cryptocurrency markets – they’re able to buy or sell based on algorithms that aim to maximize financial returns or minimize losses. This type of utility function can take into account both historical data and real-time market data.

Which One Should You Use?

- **Simple reflex agents** are fast but dumb. Good for predictable tasks.
- **Model-based agents** remember. Good for environments where state matters.
- **Goal-based agents** aim. Good when you have clear objectives.
- **Utility-based agents** optimize. Good when some outcomes are better than others.
- **Learning agents** improve. Good when you need adaptability.