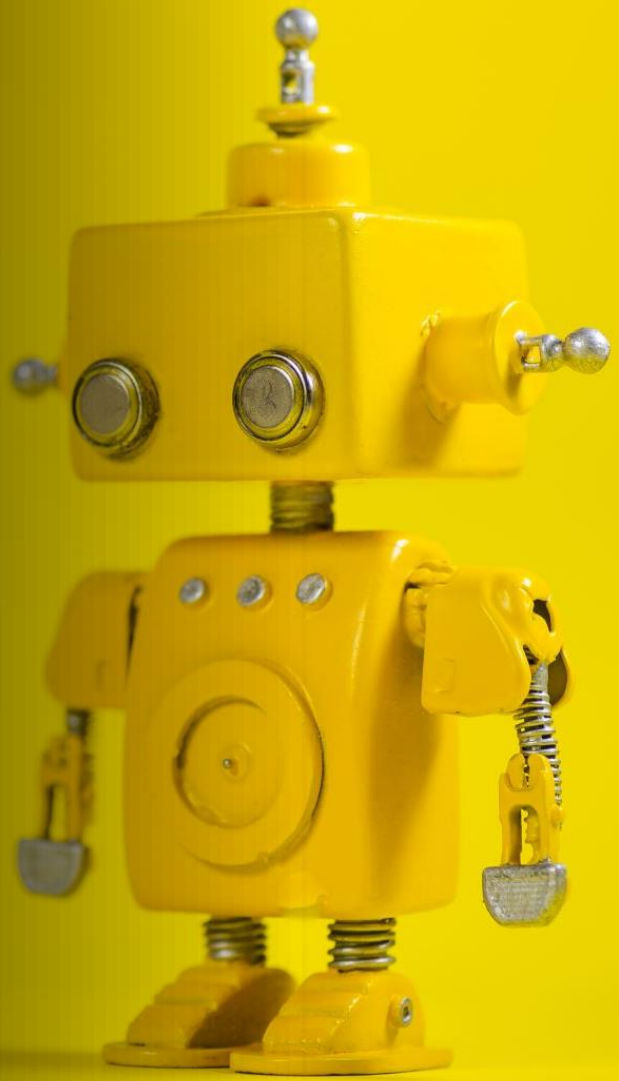
Intelligent Agents

Ahmed Ibrahim



Agenda

- Agent and environments
- Nature of environments influences agent design
- Basic “skeleton” agent designs

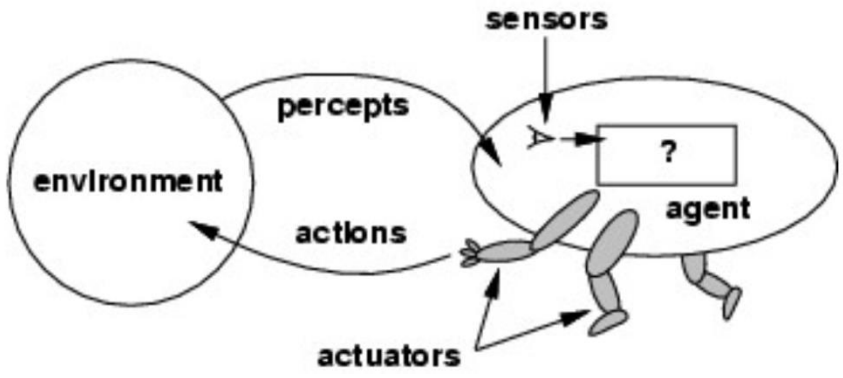
Agent and Environments

What do we mean by an agent?

Agents and Environments

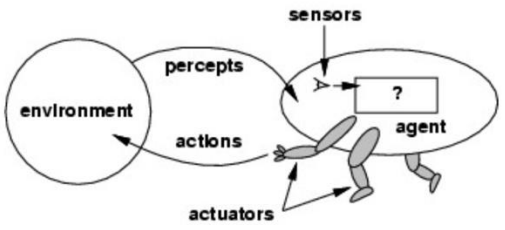
- Agents include humans, robots, softbots, etc.
- The **agent function** maps from **percept** histories to actions:

$$f : \mathcal{P}^* \rightarrow \mathcal{A}$$

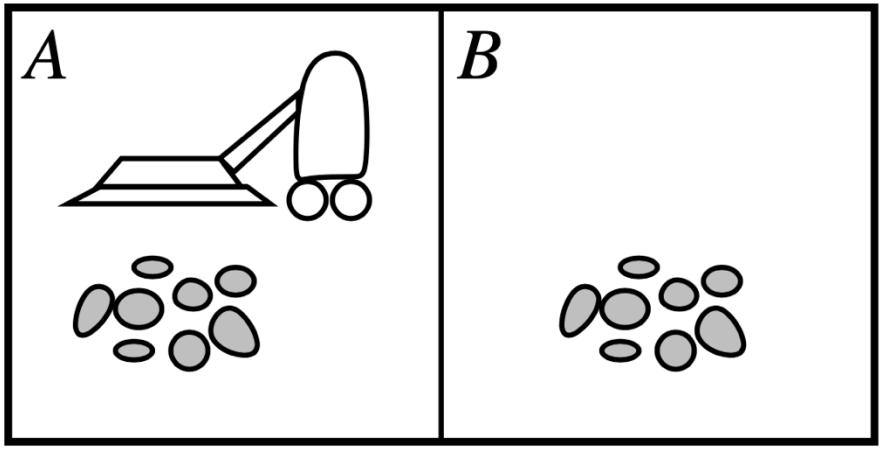


Terminologies

- Percept – the agent's perceptual inputs
- Percept sequence – the complete history of everything the agent has perceived
- The agent program runs on the physical architecture to produce f
- Agent = architecture + program

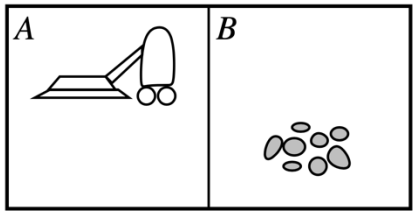


Vacuum-cleaner world



- Percepts $\mathcal{P}^*$: location and contents, e.g., [A, Dirty]
- Actions $\mathcal{A}$: *Left*, *Right*, *Suck*, *NoOp*

A vacuum-cleaner agent



Percept sequence
$[A, Clean]$
$[A, Dirty]$
$[B, Clean]$
$[B, Dirty]$
$[A, Clean], [A, Clean]$
$[A, Clean], [A, Dirty]$
⋮

Rationality

- An agent should "do the right thing".
- Performance measure – An objective criterion for success of an agent's behavior.
- Back to the vacuum-cleaner example
  - Amount of dirt cleaned within certain time
  - +1 credit for each clean square per unit time

Rationality, cont.

- Keep in mind that, rational at any given time depends on four things:
  - The performance measure that defines the criterion of success.
  - The agent's prior knowledge of the environment.
  - The actions that the agent can perform.
  - The agent's percept sequence to date.

Rational Agent

For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever **built-in knowledge** the agent has.

Rational Agent

- A simple agent that cleans a square if it is dirty and moves to the other square if not
- Assumption
  - performance measure: 1 point for each clean square at each time step
  - environment is known a priori
  - actions = {left, right, suck, no-op}
  - agent can recognise the location and dirt in that location

Omniscience, Learning and Autonomy

- Distinction between rationality and omniscience
- An agent can also learn from what it perceives
- An agent is autonomous if its behavior is determined by its own experience (with ability to learn and adapt)

Nature of environments influences agent design

Recall: Rational Agent

For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever **built-in knowledge** the agent has.

PEAS

- Specifying the task environment is always the first step in designing agent
- PEAS:
  - **P**erformance, **E**nvironment, **A**ctuators, **S**ensors

Self-driving Car

Performance Measure	Environment	Actuators	Sensors
safe, fast, legal, comfortable trip, maximize profits	roads, other traffic, pedestrians, customers	steering, accelerator, brake, signal, horn, display	camera, sonar, speedometer, GPS, odometer, engine sensors, keyboard, accelerator

DARPA urban challenge 07

The Environment Properties

- Fully observable (vs. partially observable)
- Deterministic (vs. stochastic)
- Episodic (vs. sequential)
- Static (vs. dynamic)
- Single agent (vs. multiagent)

Practical Exercise

- Develop PEAS description for the following task environment:
  - Robot soccer player
- Analyze the properties of the task environments

PEAS for Robot Soccer Player

Performance Measure	Environment	Actuators	Sensors
To play, make goals and win the game	Soccer field, teammates and opponents, referee, audience	Navigator, legs of robot, view detector for robot	Camera, communicators, sensors

The properties of the task environment of Robot soccer player are

- Observable: It is partially observable.
- Deterministic: No.
- Static: No.
- Discrete: No.
- Single agent: No

Basic “skeleton” agent designs

An Agents

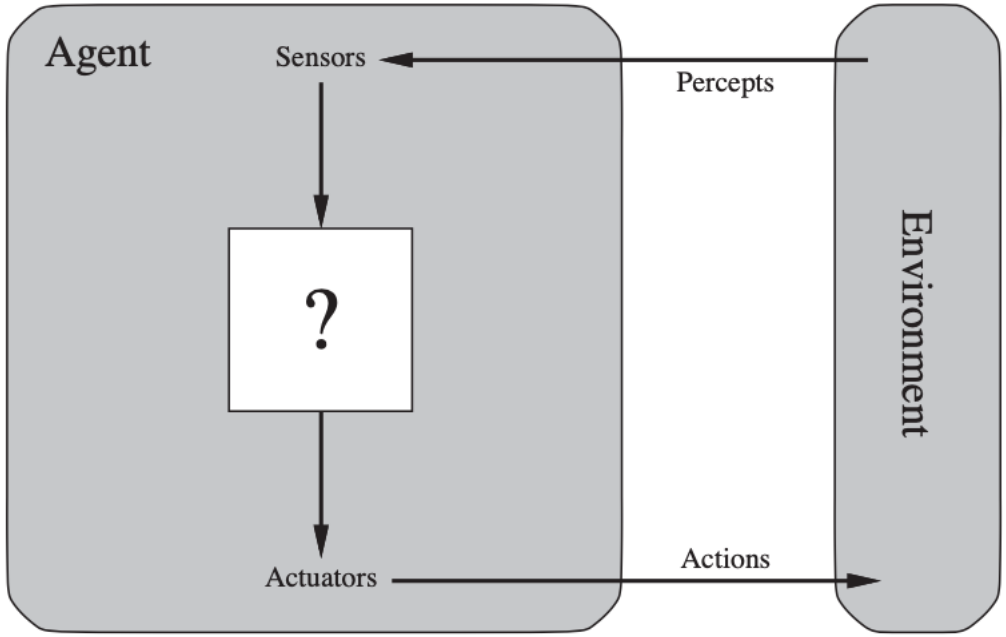


Figure 2.1 Agents interact with environments through sensors and actuators.

Agent Program vs. Agent Function

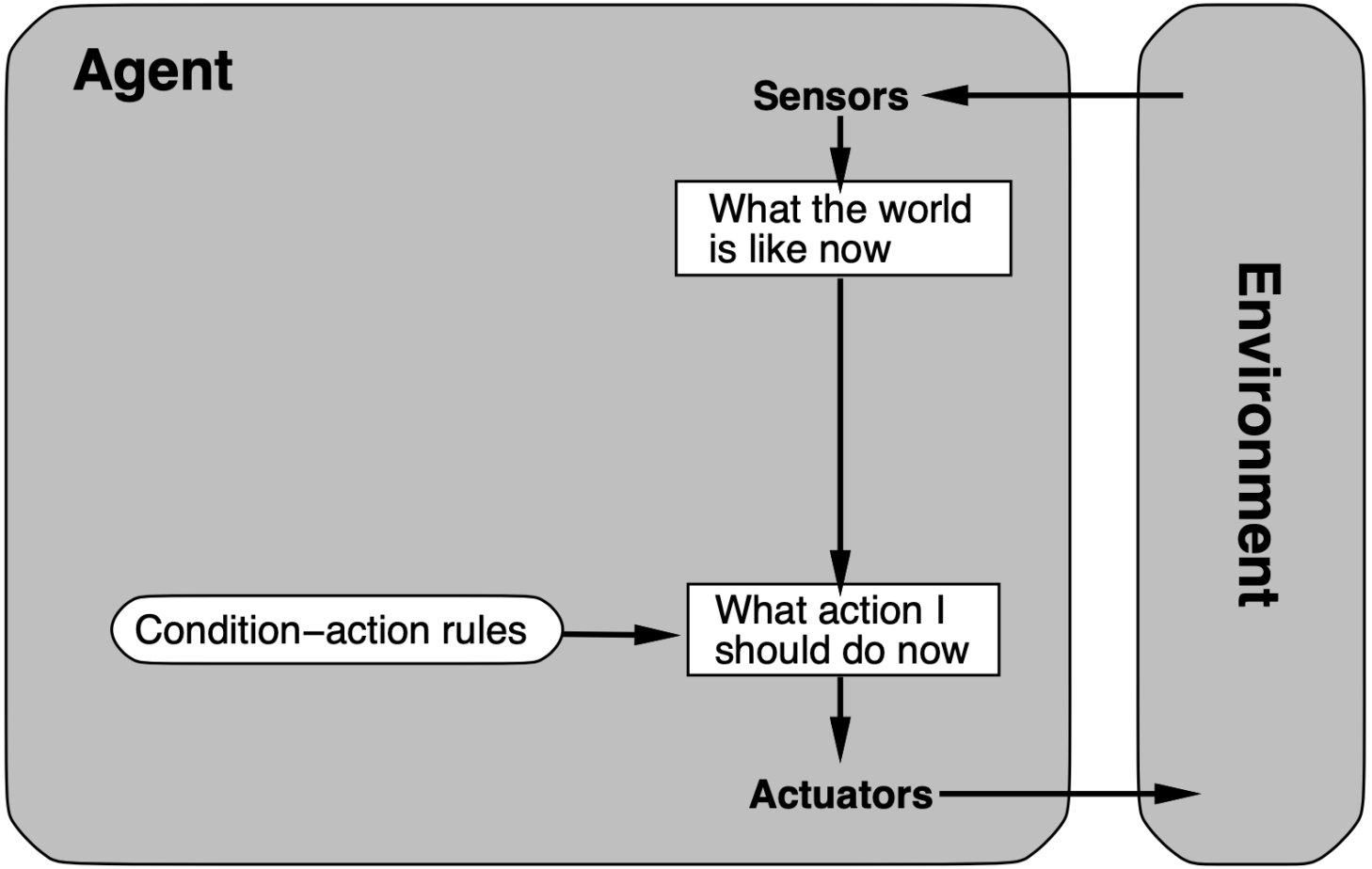
The agent function is an abstract mathematical description; the agent program is a concrete implementation, running within some physical system.

Agent function – a function that specifies the agent's action in response to **every possible** percept sequence.

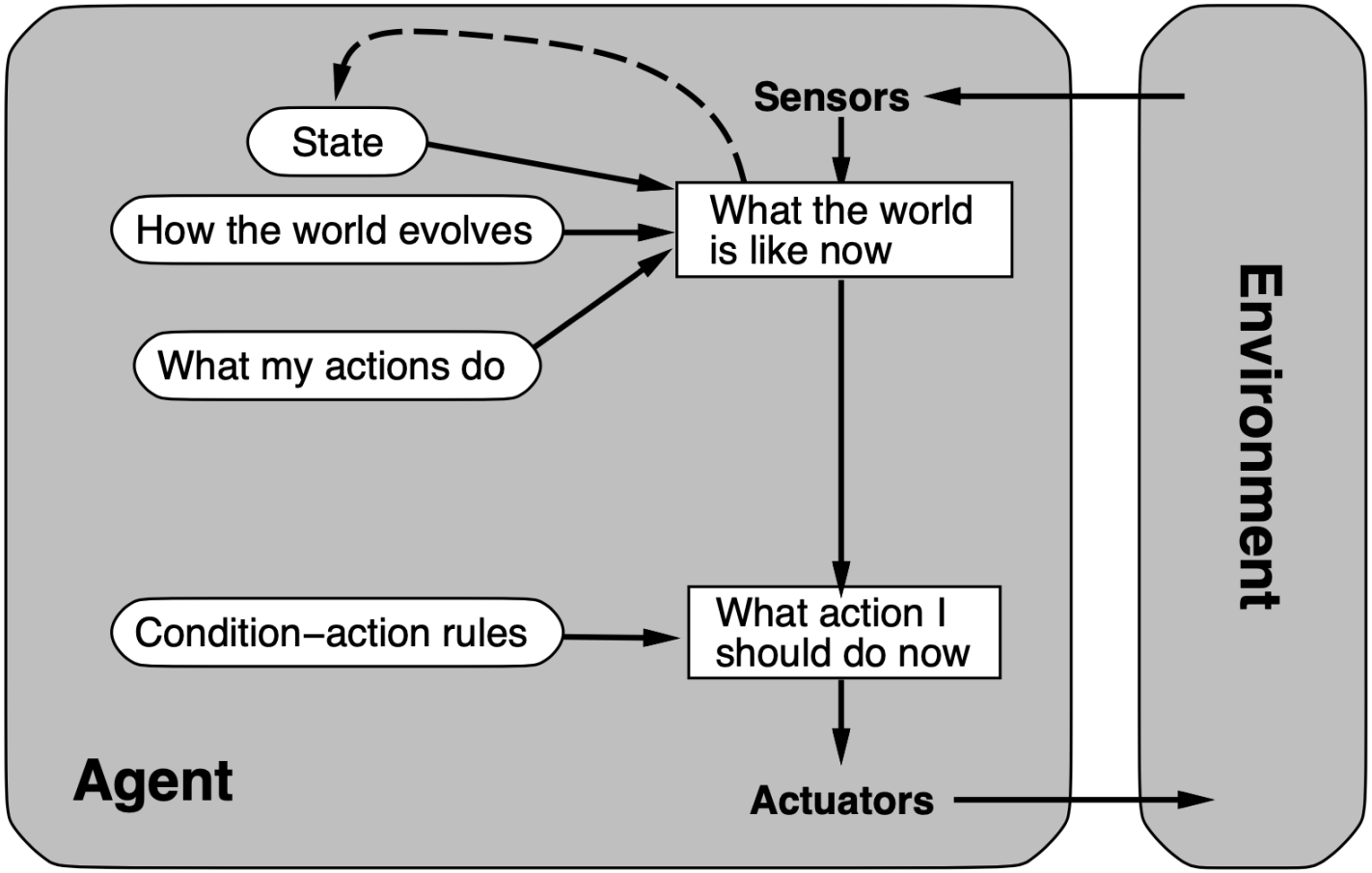
Agent Types

- Four basic types in order of increasing generality:
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
  - Utility-based agents
- All these can be turned into learning agents

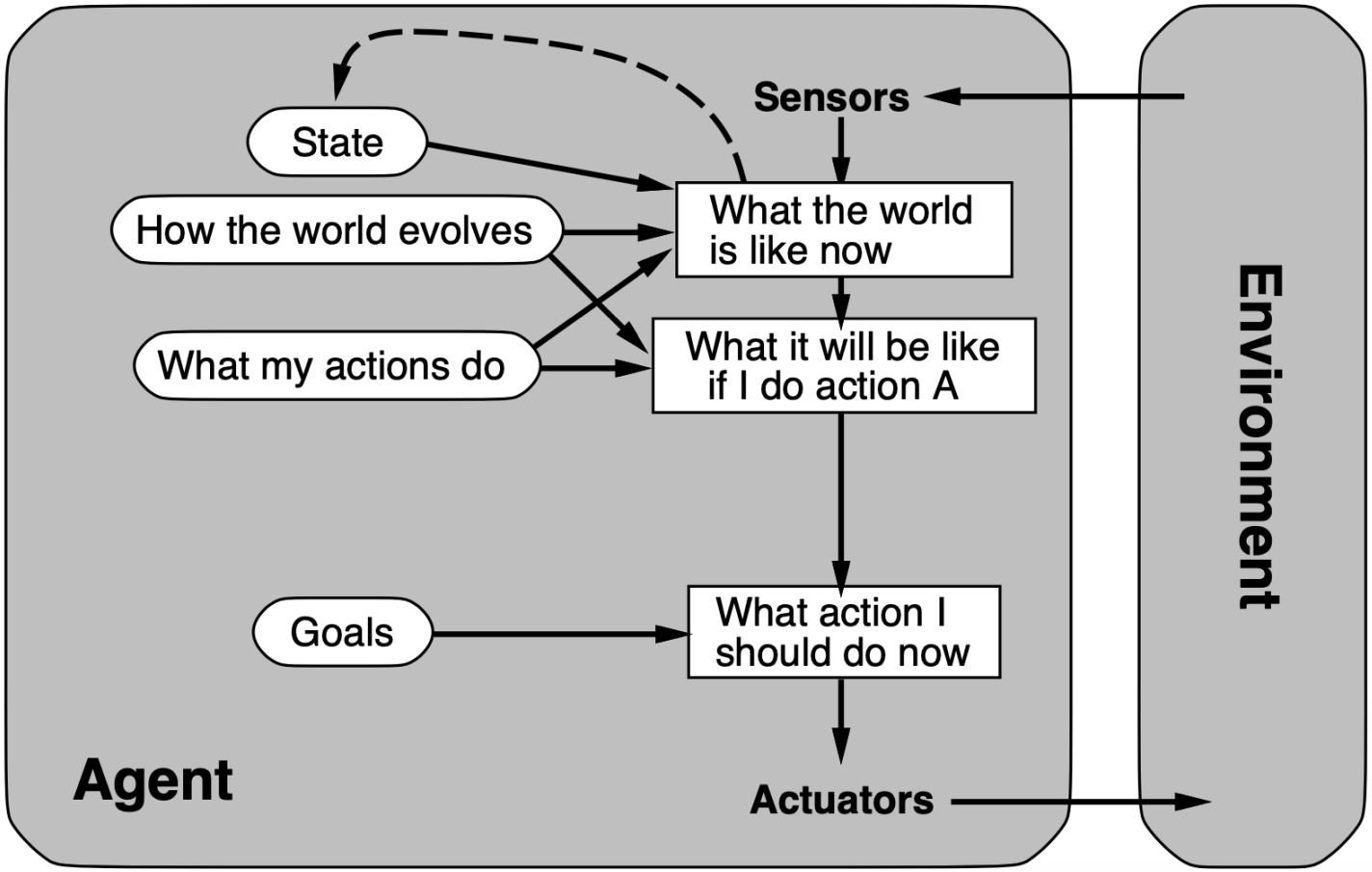
Simple Reflex Agents



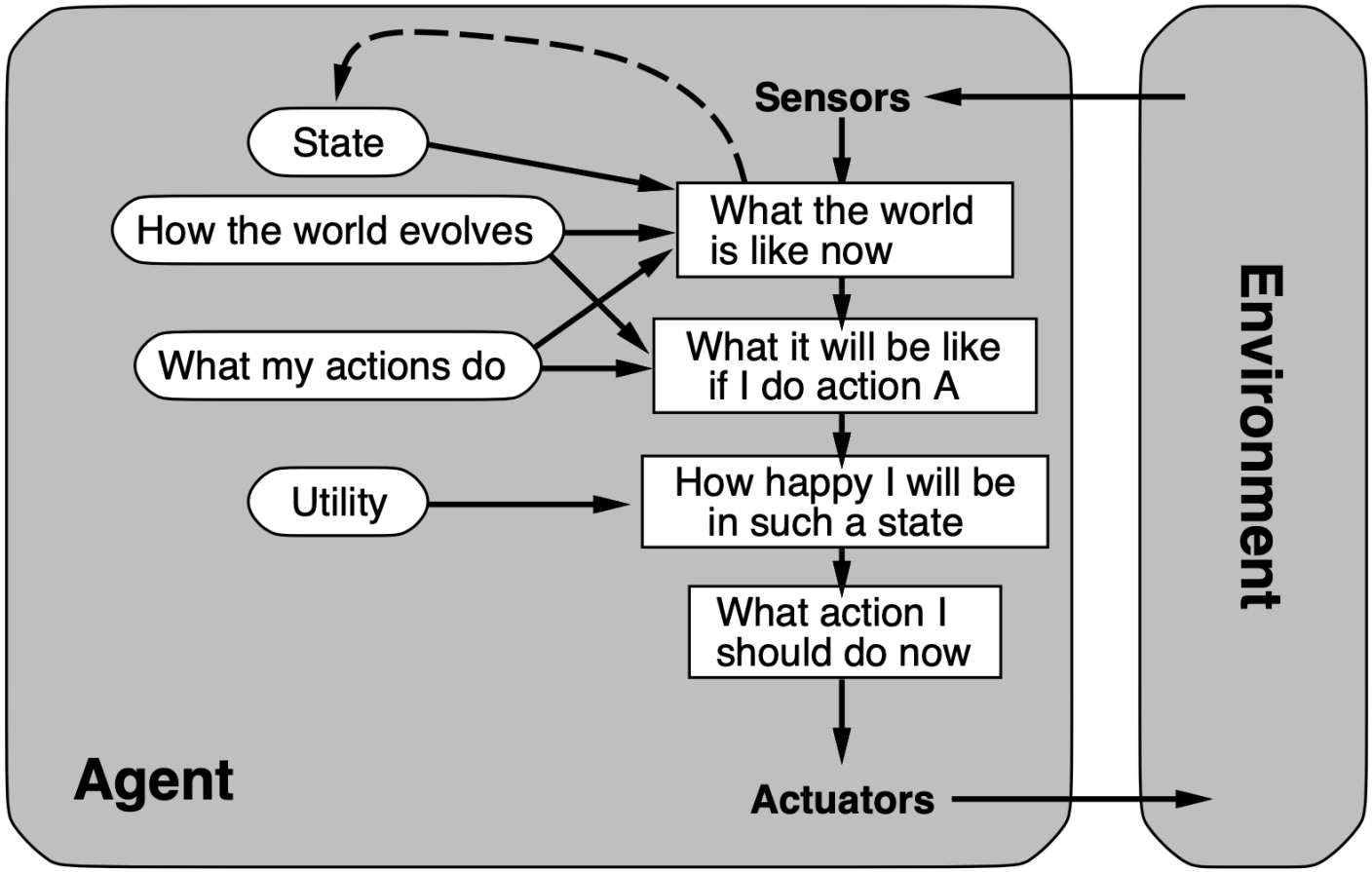
Model-based Reflex Agents



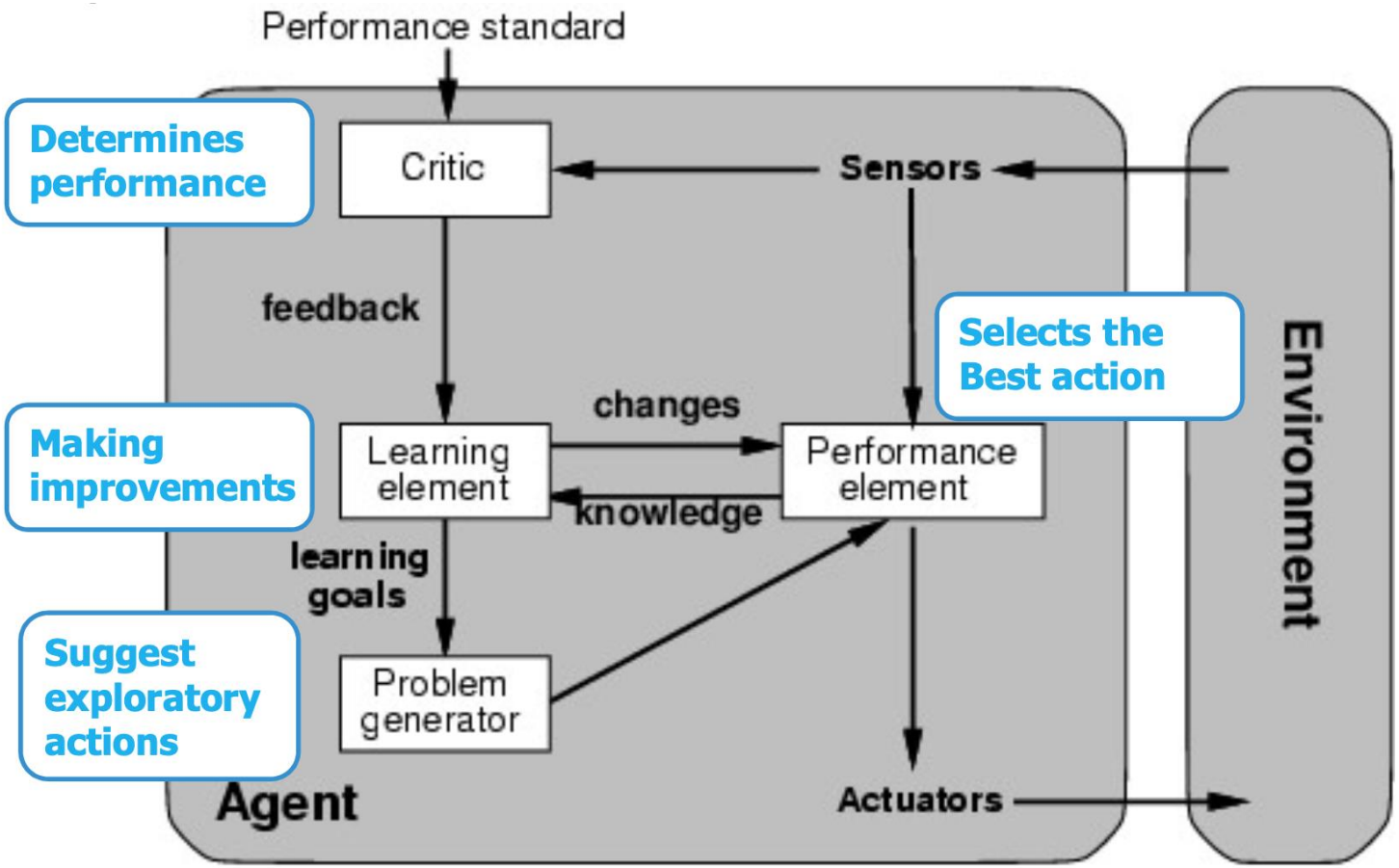
Goal-based Agents



Utility-based Agents



Learning Agents



Performance measure vs. utility function

- A performance measure (typically imposed by the designer) is used to evaluate the behavior of the agent in environment.
- A utility function is used by an agent itself to evaluate how desirable states are.

Remember that, an agent program runs in cycles of: (1) perceive, (2) think then (3) act.

Conclusion

- Agents interact with environments through actuators and sensors.
- A perfectly rational agent maximizes expected performance
- Environments are categorized along several dimensions: observable? deterministic? episodic? static? discrete? single-agent?
- Several basic agent architectures exist

To DO

- This week you need to do the following:
  - Read Chapter 2 from the textbook
  - Make yourself familiar with the pseudocode of the following:
    - simple reflex agents
    - reflex agents with state
    - goal-based agents
    - utility-based agents