

Chapter 2

Intelligent Agents

Outline

- ❖ Agents and environments
- ❖ Rationality
- ❖ Nature of Environments
- ❖ The Structure of Agents

Intelligent Agents

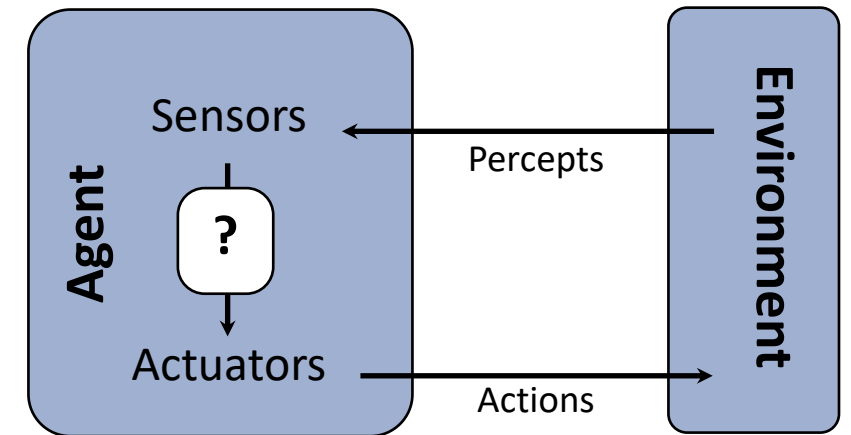
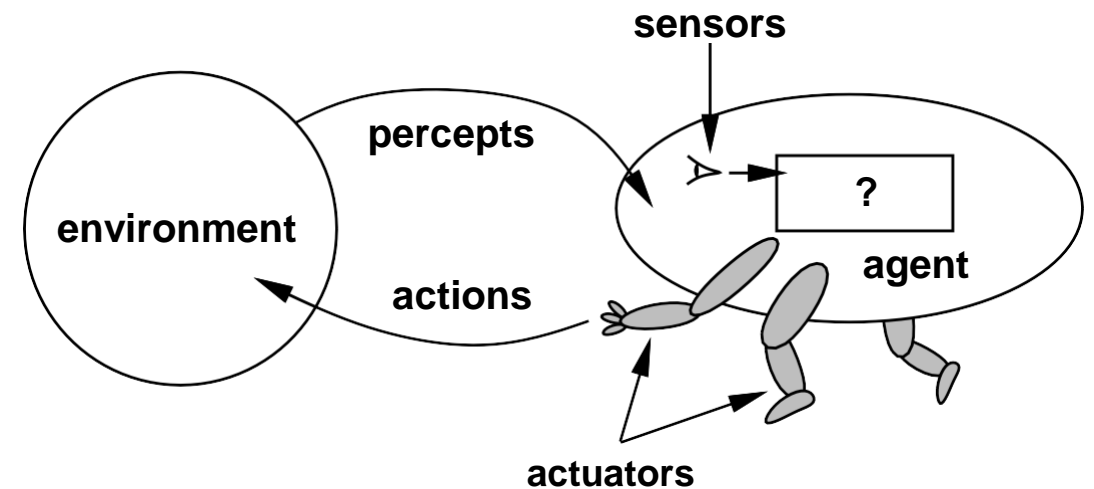
❖ An **agent** is anything that can be viewed as **perceiving** its **environment** through **sensors** and **acting** upon that environment through **actuators**

❖ 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



Percept and Percept Sequence

- ❖ Percepts: agent's perceptual input at any given instant
- ❖ Percept Sequence: is the complete history of everything the agent has ever perceived.

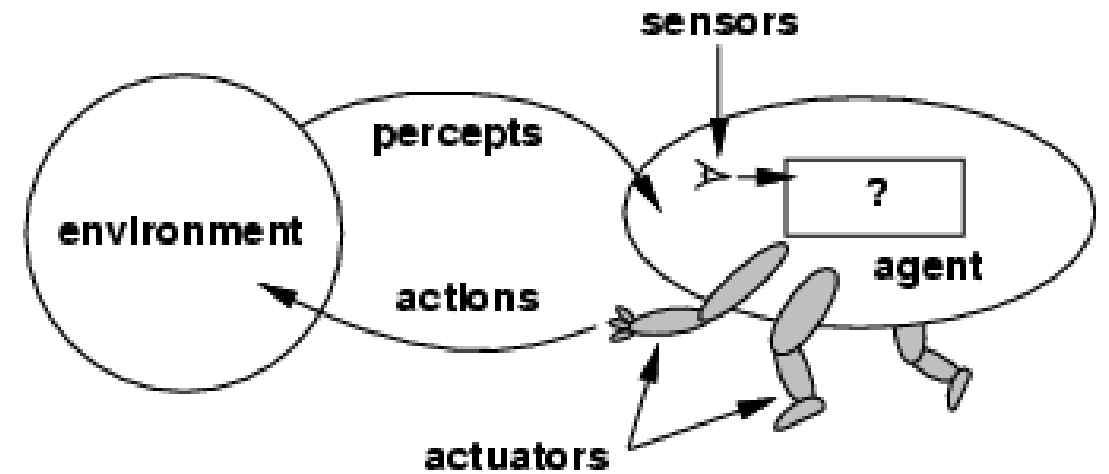
Agent Behavior

- ❖ The **agent function** maps from percept histories to actions:

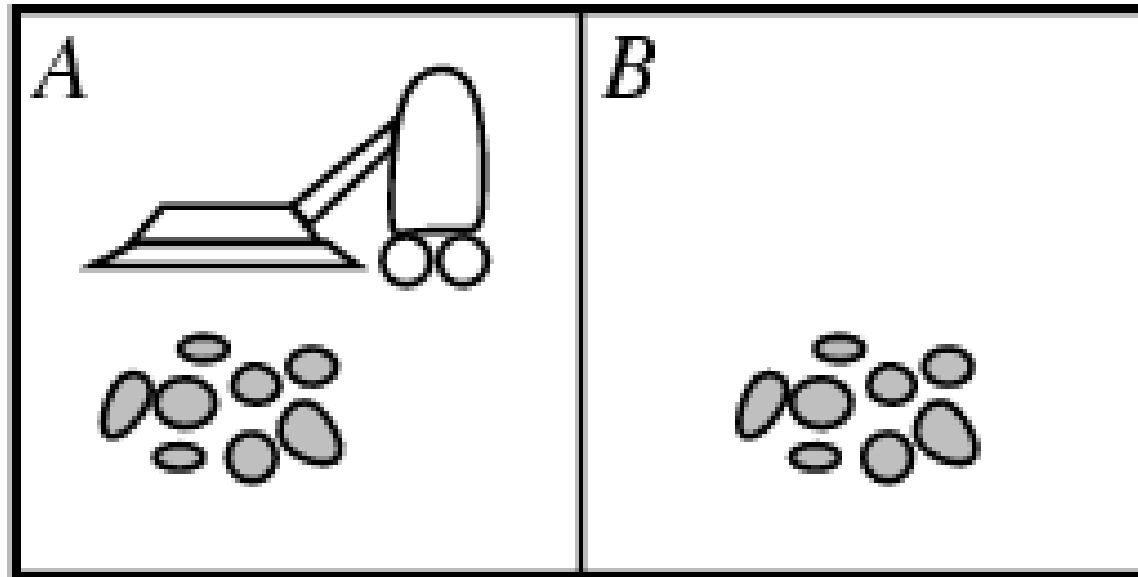
$$[f: \mathcal{P}^* \rightarrow \mathcal{A}]$$

- ❖ The **agent program** runs on the physical **architecture** to produce f

agent = architecture + program



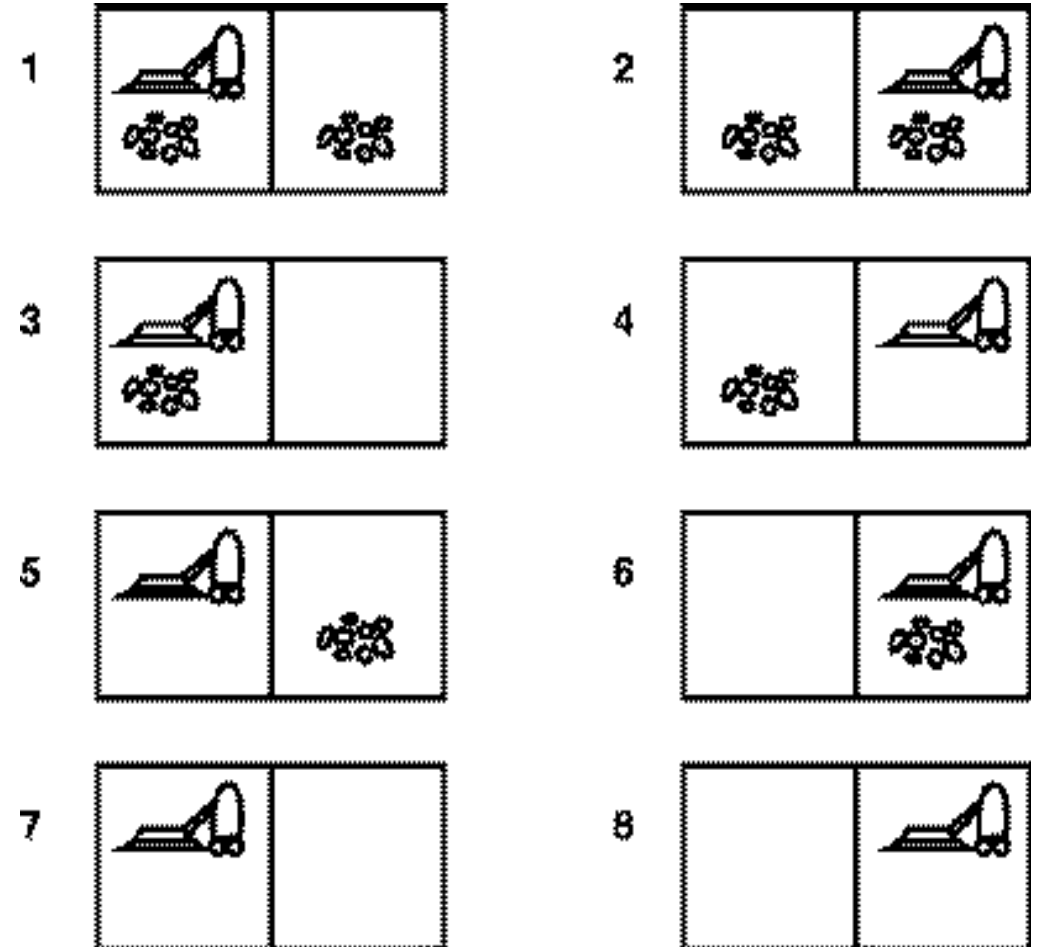
VACUUM-CLEANER World



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

A vacuum-cleaner agent

Percept sequence	Action
$[A, \textit{Clean}]$	<i>Right</i>
$[A, \textit{Dirty}]$	<i>Suck</i>
$[B, \textit{Clean}]$	<i>Left</i>
$[B, \textit{Dirty}]$	<i>Suck</i>
$[A, \textit{Clean}], [A, \textit{Clean}]$	<i>Right</i>
$[A, \textit{Clean}], [A, \textit{Dirty}]$	<i>Suck</i>
\vdots	\vdots



Outline

- ❖ Agents and environments
- ❖ **Rationality**
- ❖ Nature of Environments
- ❖ The Structure of Agents

Rational agents

- ❖ An agent should strive to "**do the right thing**", based on what it can perceive and the actions it can perform. The right action is the one that will cause the agent to be most successful.
- ❖ **Performance measure:** An objective criterion for success of an agent's behavior
- ❖ For example: vacuum-cleaner agent
 - could be amount of dirt cleaned up
 - amount of time taken
 - amount of electricity consumed
 - amount of noise generated

PEAS

- ❖ PEAS: Performance measure, Environment, Actuators, Sensors
- ❖ To design a rational agent, we must specify the **task environment**
- ❖ Consider, e.g., the task of designing an automated taxi driver
 - Performance measure
 - Environment
 - Actuators
 - Sensors

Internet shopping agent

❖ Performance measure??

❖ Environment??

❖ Actuators??

❖ Sensors??

Interactive English Tutor

❖ Performance measure??

❖ Environment??

❖ Actuators??

❖ Sensors??

Outline

- ❖ Agents and environments
- ❖ Rationality
- ❖ **Nature of Environments**
- ❖ The Structure of Agents

Environmental Types

- ❖ Observability
- ❖ Time Constraint
- ❖ Environmental Representation
- ❖ Consequences
- ❖ Environmental predictability
- ❖ Environment knowledge
- ❖ Number of agents

Environment types

	Solitaire	Backgammon	Internet shopping	Taxi
<u>Observable??</u> <u>Deterministic??</u> <u>Episodic??</u> <u>Static??</u> <u>Discrete??</u> <u>Single-agent??</u>				

Outline

- ❖ Agents and environments
- ❖ Rationality
- ❖ Nature of Environments
- ❖ **The Structure of Agents**

Agent types

Four basic types in order of increasing generality:

1. Simple reflex agents
2. Model-based reflex agents
3. Goal-based agents
4. Utility-based agents

Simple reflex agents

It acts according to a rule whose condition matches the current state, as defined by the percept.

function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action

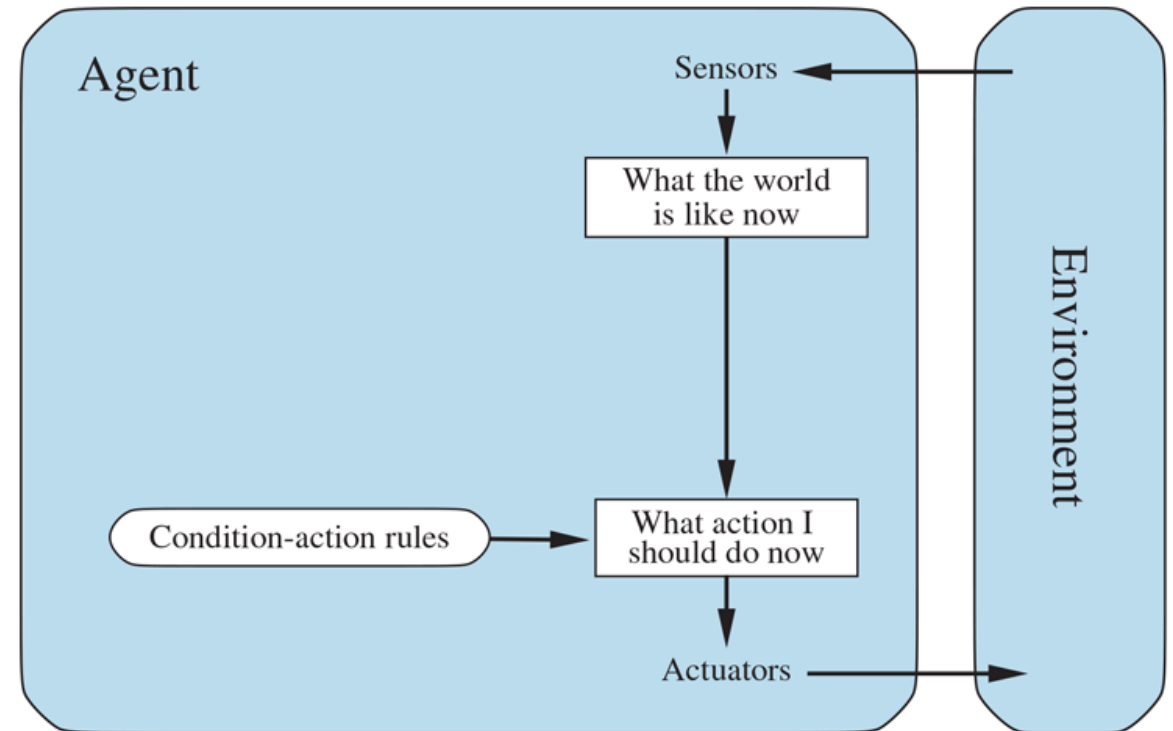
persistent: *rules*, a set of condition–action rules

state \leftarrow INTERPRET-INPUT(*percept*)

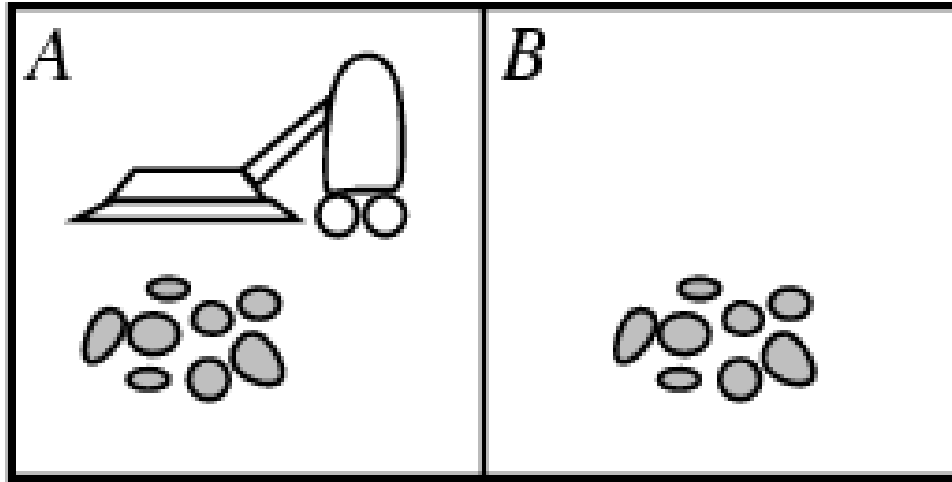
rule \leftarrow RULE-MATCH(*state*, *rules*)

action \leftarrow *rule*.ACTION

return *action*



VACUUM-CLEANER World



❖ Percepts: location and contents e.g., [A,Dirty]

❖ Actions: *Left, Right, Suck, NoOp*

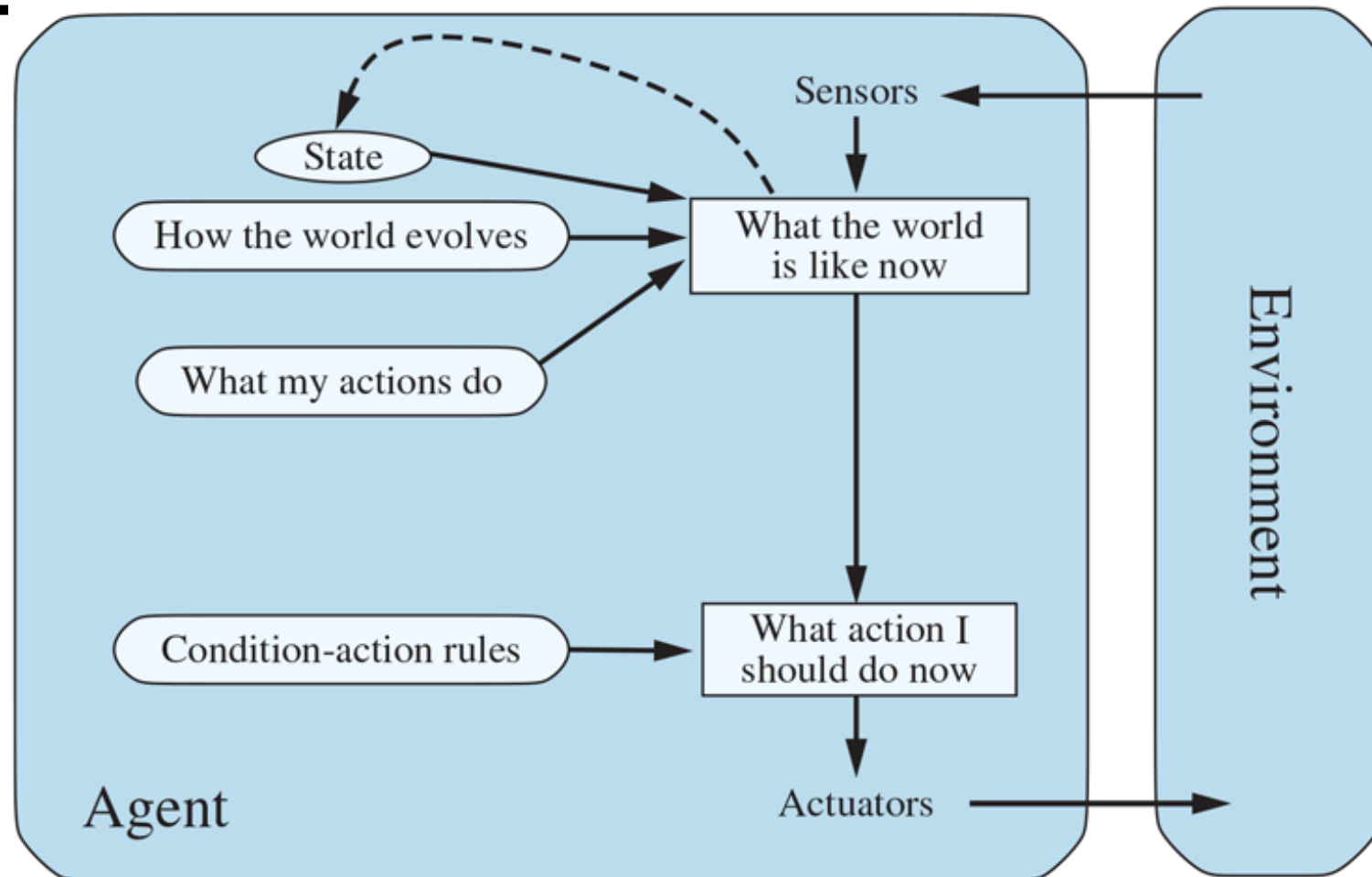
function REFLEX-VACUUM-AGENT(*[location,status]*) **returns** an action

if *status = Dirty* **then return** *Suck*

else if *location = A* **then return** *Right*

else if *location = B* **then return** *Left*

A model-based reflex agent.



Model-based agents

function MODEL-BASED-REFLEX-AGENT(*percept*) **returns** an action

persistent: *state*, the agent's current conception of the world state

transition_model, a description of how the next state depends on
the current state and action

sensor_model, a description of how the current world state is reflected
in the agent's percepts

rules, a set of condition–action rules

action, the most recent action, initially none

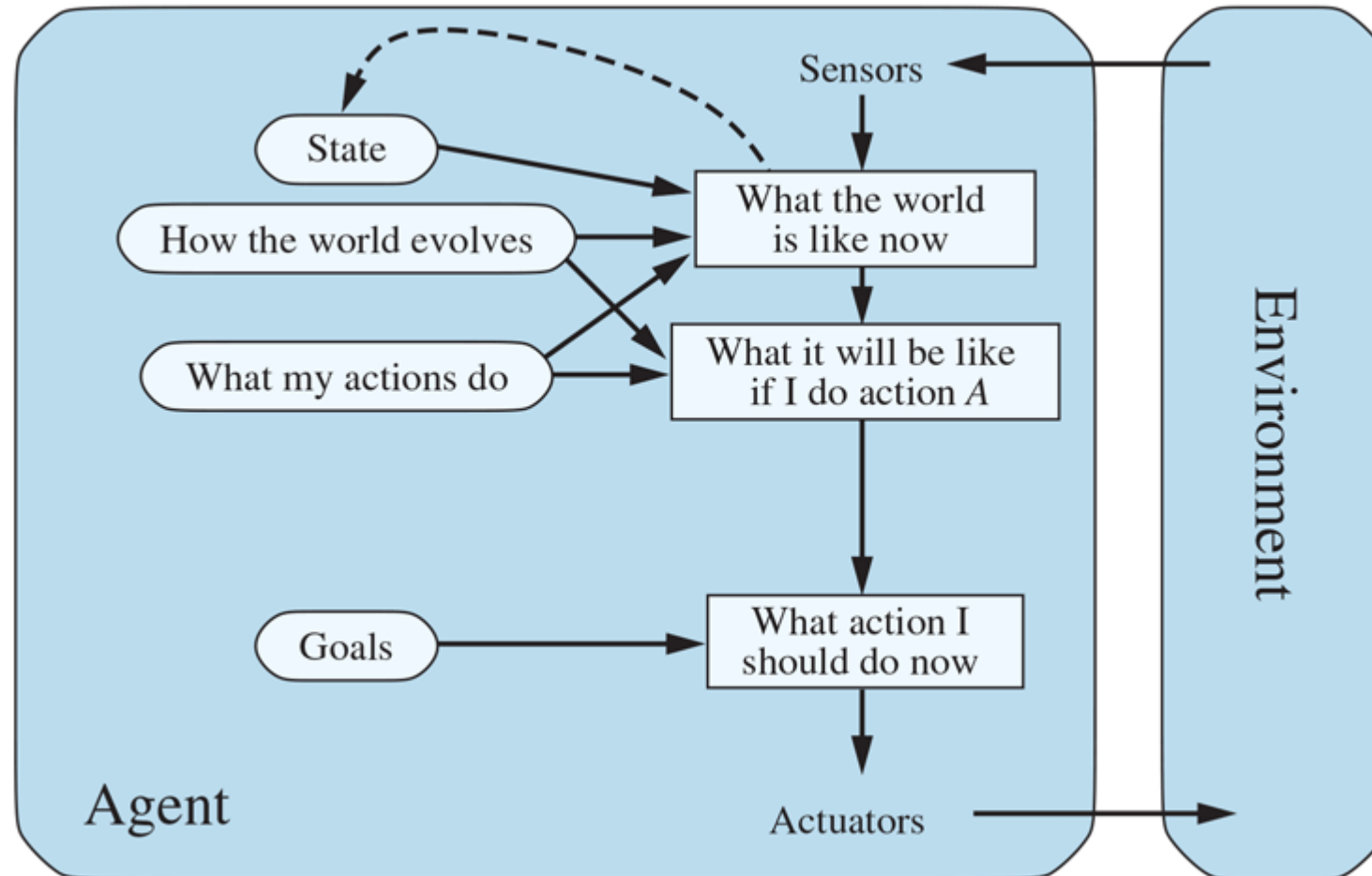
state \leftarrow UPDATE-STATE(*state*, *action*, *percept*, *transition_model*, *sensor_model*)

rule \leftarrow RULE-MATCH(*state*, *rules*)

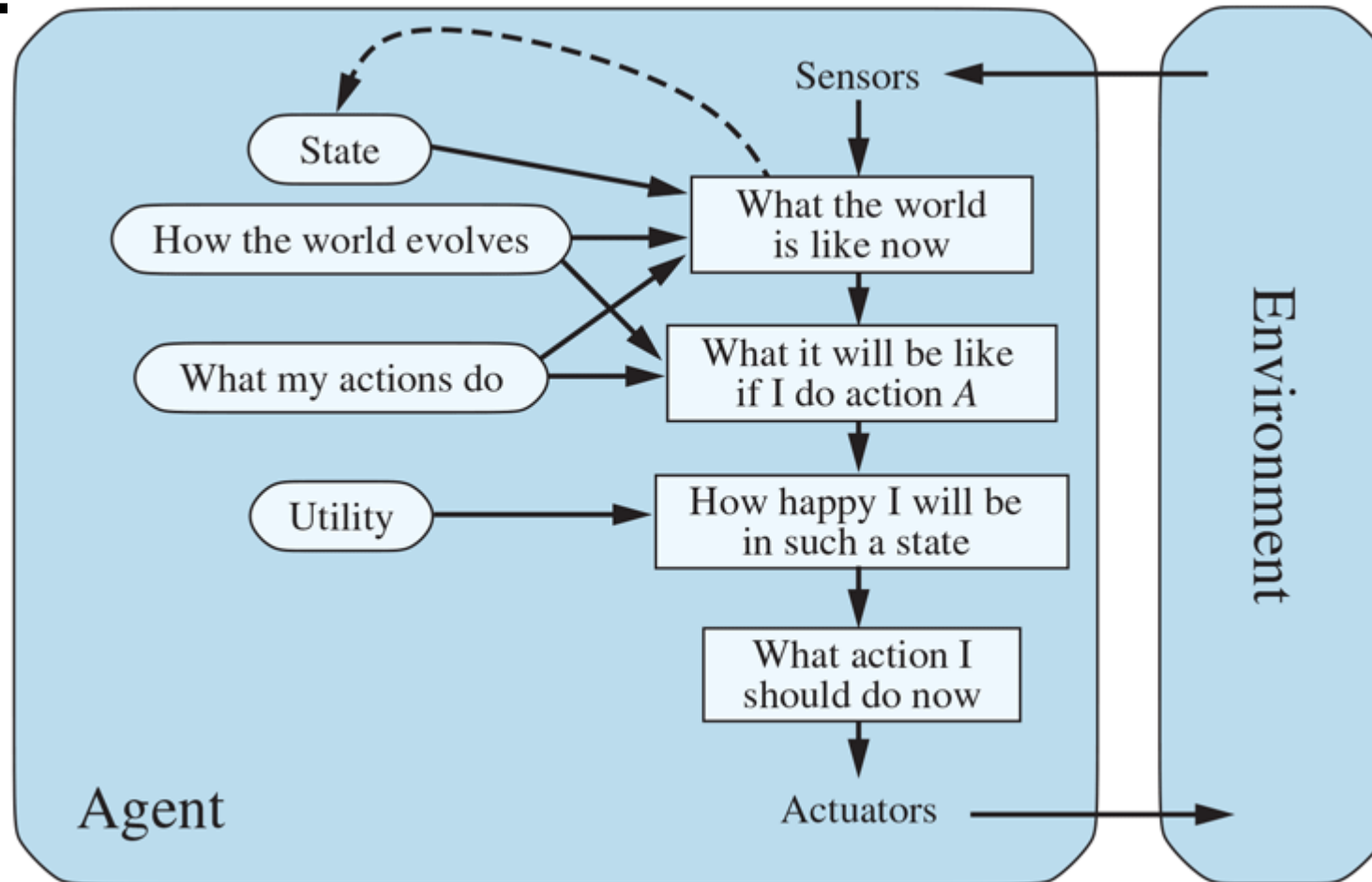
action \leftarrow *rule*.ACTION

return *action*

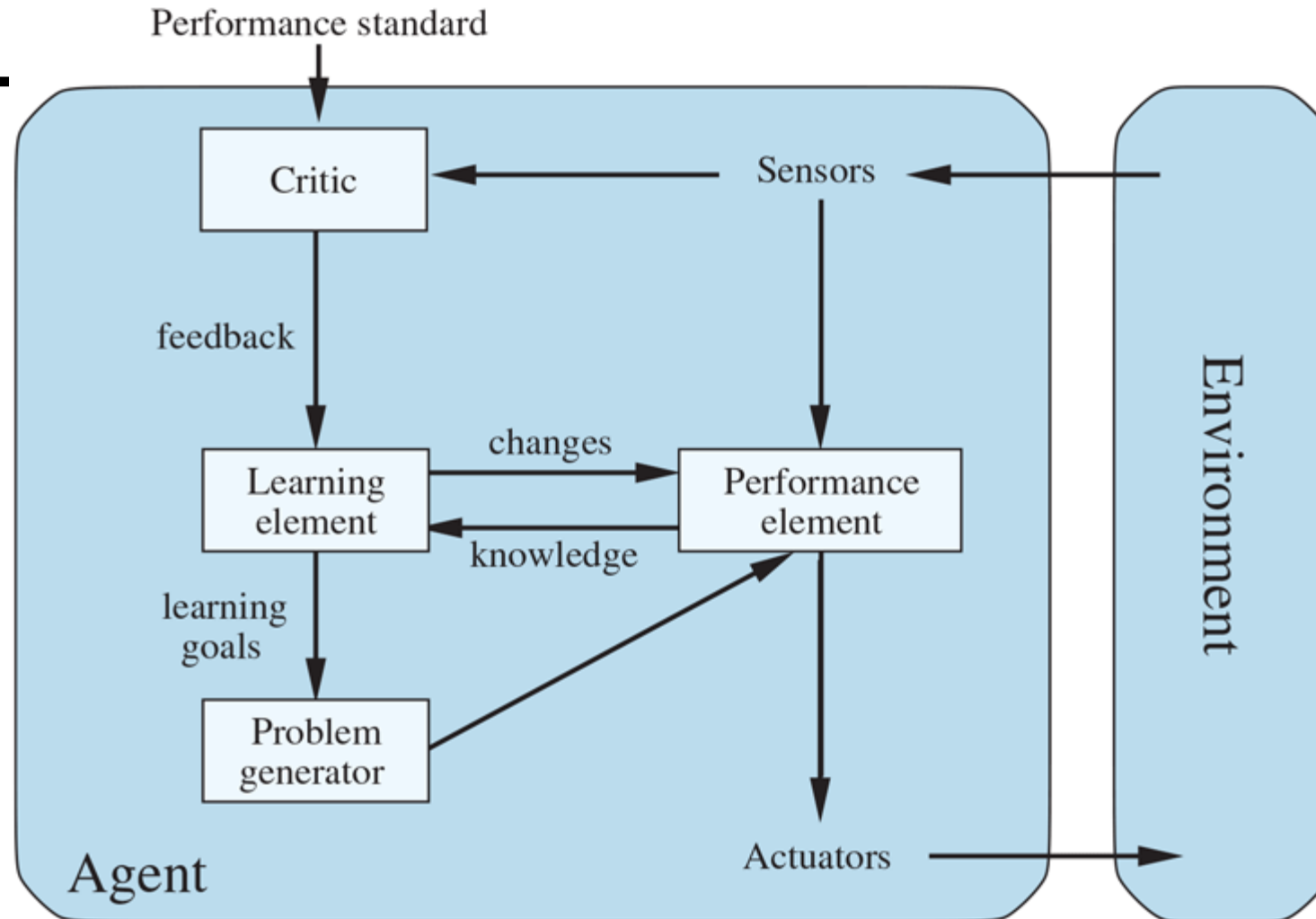
Goal-based agents



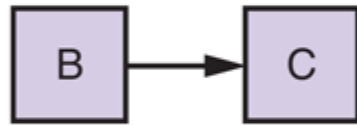
Utility-based agents



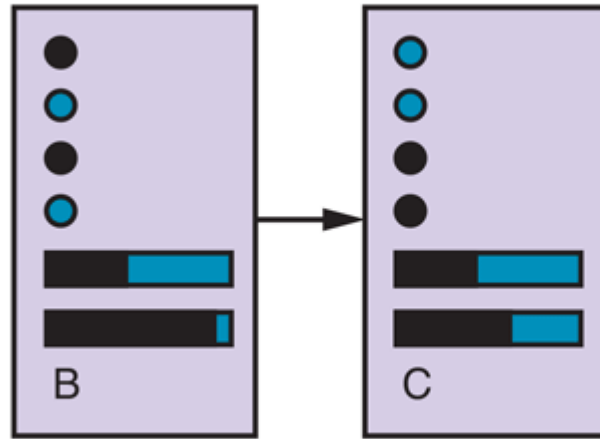
Learning agents



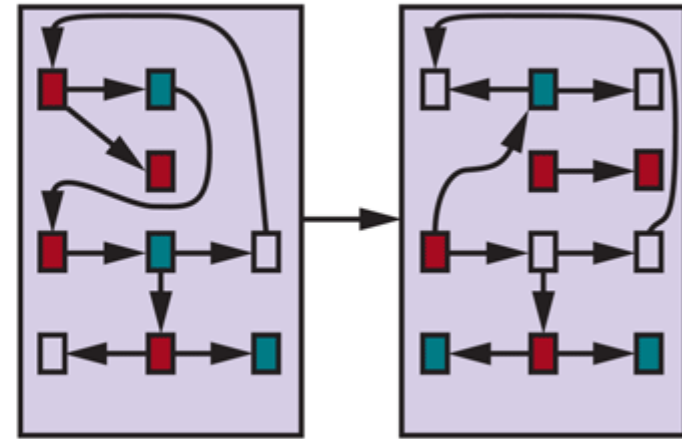
State and transition representation



(a) Atomic



(b) Factored



(c) Structured

Summary

- ❖ Agents interact with environments through actuators and sensors
- ❖ The agent function describes what the agent does in all circumstances. The performance measure evaluates the environment sequence
- ❖ A perfectly rational agent maximizes expected performance. Agent programs implement (some) agent functions
- ❖ PEAS descriptions define task environments
- ❖ Environments are categorized along several dimensions: observable? deterministic? episodic? static? discrete? single-agent?
- ❖ Several basic agent architectures exist: reflex, reflex with state, goal-based, utility-based

Consent

Data for this course is taken from several books specifically AIMA by Russel Norvig, slides and already similar course taught in other universities specifically MIT, UC-Berkley, and Stanford and is the sole property of the respective owner. The copyright infringement is not intended, and this material is solely used for the academic purpose or as a teaching material. This work is protected by United States copyright laws and is provided solely for the use of instructors in teaching their courses and assessing student learning. Dissemination or sale of any part of this work (including on the World Wide Web) will destroy the integrity of the work and is not permitted. The work and materials from it should never be made available to students except by instructors using the accompanying text in their classes. All recipients of this work are expected to abide by these restrictions and to honor the intended pedagogical purposes and the needs of other instructors who rely on these materials.