Intelligent Agents

Chapter 2

Reminders

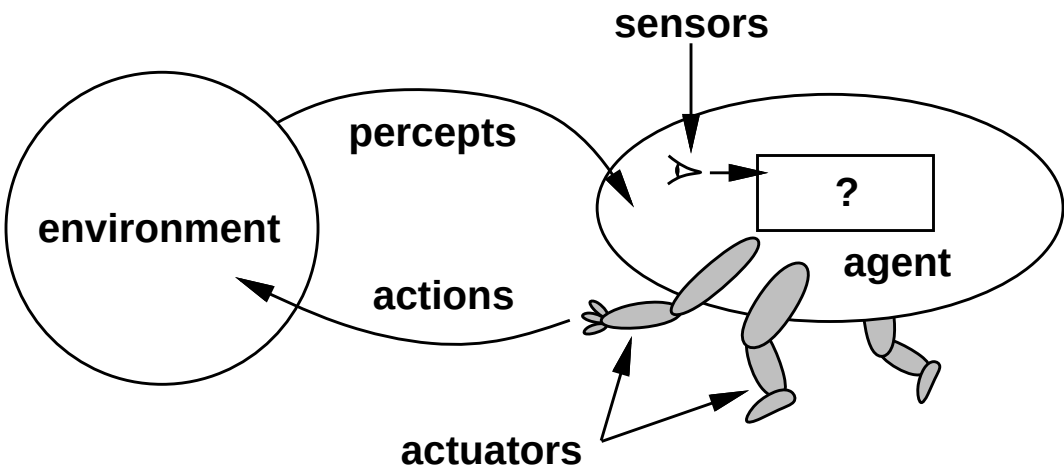
Assignment 0 (lisp refresher) due 1/28

Lisp/emacs/AIMA tutorial: 11-1 today and Monday, 2

Outline

- ♢ Agents and environments
- ♢ Rationality
- ♢ PEAS (Performance measure, Environment, Actuators, Ser
- ♢ Environment types
- ♢ Agent types

Agents and environments



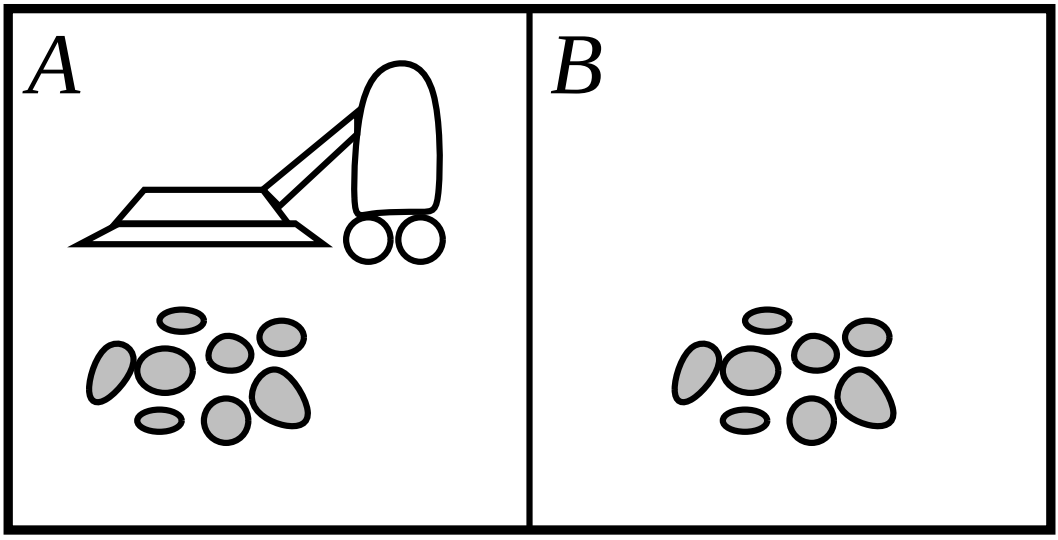
Agents include humans, robots, softbots, thermostats, etc.

The agent function maps from percept histories to actions:

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

The agent program runs on the physical architecture to produ

Vacuum-cleaner world



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

Actions: $Left$, $Right$, $Suck$, $NoOp$

A vacuum-cleaner agent

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

```
function REFLEX-VACUUM-AGENT( [location,status]) returns an
    if status = Dirty then return Suck
    else if location = A then return Right
    else if location = B then return Left
```

What is the **right** function?

Can it be implemented in a small agent program?

Rationality

Fixed performance measure evaluates the environment sequenc
- one point per square cleaned up in time T?
- one point per clean square per time step, minus one per
- penalize for $> k$ dirty squares?

A rational agent chooses whichever action maximizes the expe
the performance measure given the percept sequence to date

Rational $\neq$ omniscient
- percepts may not supply all relevant information

Rational $\neq$ clairvoyant
- action outcomes may not be as expected

Hence, rational $\neq$ successful

Rational $\Rightarrow$ exploration, learning, autonomy

PEAS

To design a rational agent, we must specify the task environm

Consider, e.g., the task of designing an automated taxi:

Performance measure??

Environment??

Actuators??

Sensors??

PEAS

To design a rational agent, we must specify the task environm

Consider, e.g., the task of designing an automated taxi:

Performance measure?? safety, destination, profits, legality, co

Environment?? US streets/freeways, traffic, pedestrians, weath

Actuators?? steering, accelerator, brake, horn, speaker/display

Sensors?? video, accelerometers, gauges, engine sensors, keybo

Internet shopping agent

Performance measure??

Environment??

Actuators??

Sensors??

Internet shopping agent

Performance measure?? price, quality, appropriateness, efficien

Environment?? current and future WWW sites, vendors, shipp

Actuators?? display to user, follow URL, fill in form

Sensors?? HTML pages (text, graphics, scripts)

Environment types

	Solitaire	Backgammon	Internet shop
Observable?? Deterministic?? Episodic?? Static?? Discrete?? Single-agent??			

Environment types

	Solitaire	Backgammon	Internet shop
Observable??	Yes	Yes	No
Deterministic??			
Episodic??			
Static??			
Discrete??			
Single-agent??			

Environment types

	Solitaire	Backgammon	Internet shop
Observable??	Yes	Yes	No
Deterministic??	Yes	No	Partly
Episodic??			
Static??			
Discrete??			
Single-agent??			

Environment types

	Solitaire	Backgammon	Internet shop
Observable??	Yes	Yes	No
Deterministic??	Yes	No	Partly
Episodic??	No	No	No
Static??			
Discrete??			
Single-agent??			

Environment types

	Solitaire	Backgammon	Internet shop
Observable??	Yes	Yes	No
Deterministic??	Yes	No	Partly
Episodic??	No	No	No
Static??	Yes	Semi	Semi
Discrete??			
Single-agent??			

Environment types

	Solitaire	Backgammon	Internet shop
Observable??	Yes	Yes	No
Deterministic??	Yes	No	Partly
Episodic??	No	No	No
Static??	Yes	Semi	Semi
Discrete??	Yes	Yes	Yes
Single-agent??			

Environment types

	Solitaire	Backgammon	Internet shopp
Observable??	Yes	Yes	No
Deterministic??	Yes	No	Partly
Episodic??	No	No	No
Static??	Yes	Semi	Semi
Discrete??	Yes	Yes	Yes
Single-agent??	Yes	No	Yes (except auc

The environment type largely determines the age

The real world is (of course) partially observable, stochastic
dynamic, continuous, multi-agent

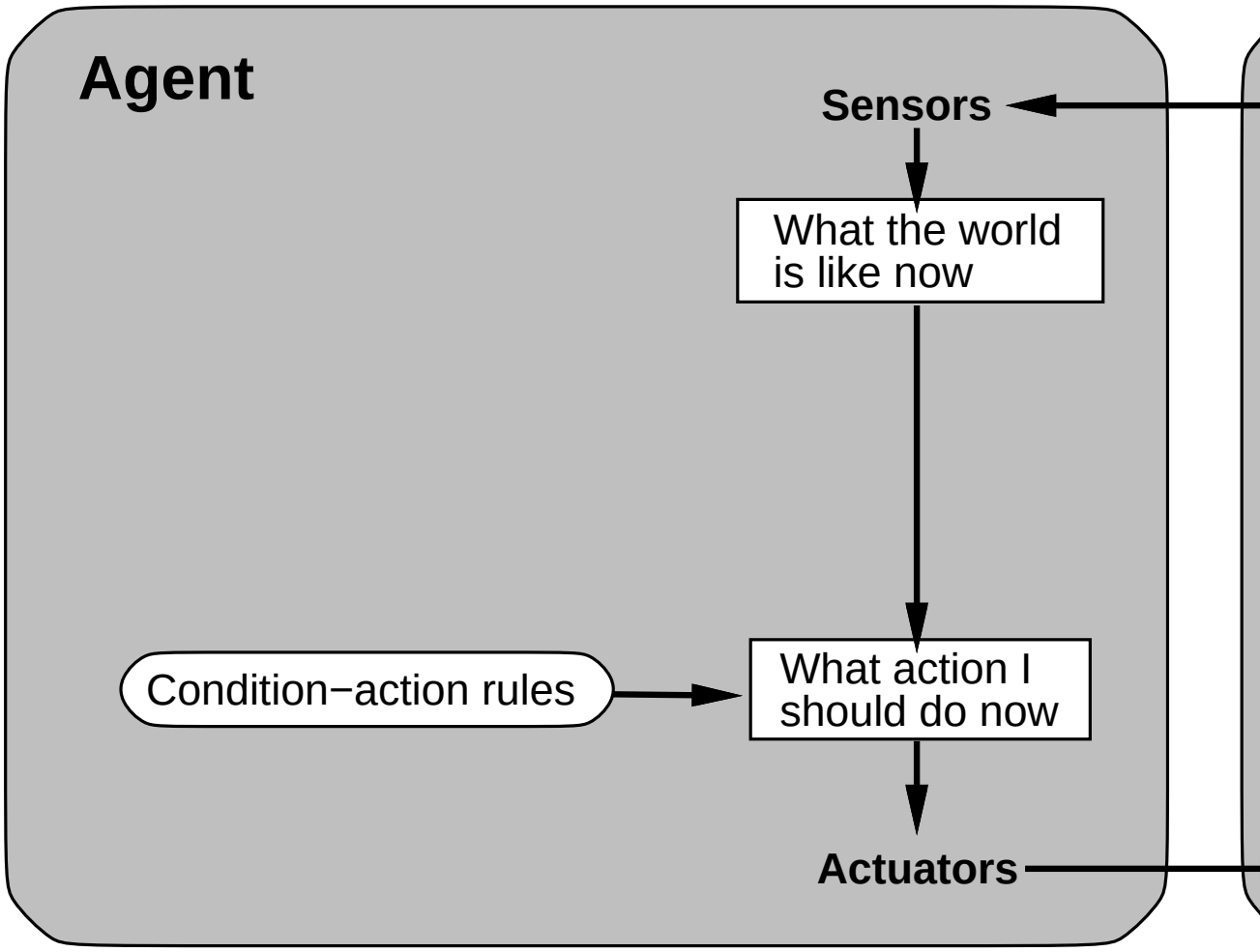
Agent types

Four basic types in order of increasing generality:

- simple reflex agents
- reflex agents with state
- goal-based agents
- utility-based agents

All these can be turned into learning agents

Simple reflex agents



Example

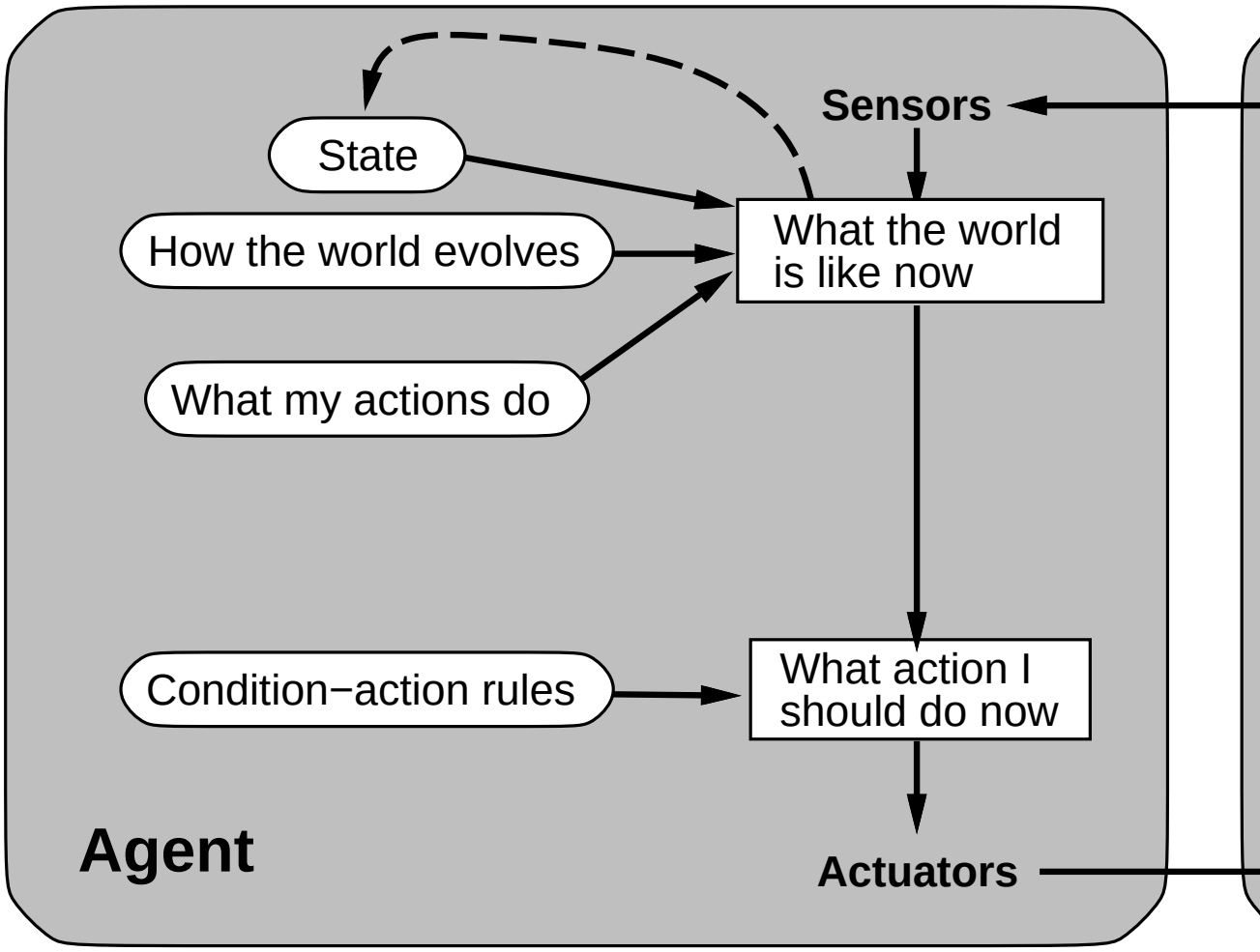
function REFLEX-VACUUM-AGENT([*location*,*status*]) **returns** an

if *status* = *Dirty* **then return** *Suck*
else if *location* = *A* **then return** *Right*
else if *location* = *B* **then return** *Left*

```
(setq joe (make-agent :name 'joe :body (make-age
                      :program (make-reflex-vacu

(defun make-reflex-vacuum-agent-program ()
  #'(lambda (percept)
      (let ((location (first percept)) (status (
        (cond ((eq status 'dirty) 'Suck)
              ((eq location 'A) 'Right)
              ((eq location 'B) 'Left)))))
```

Reflex agents with state



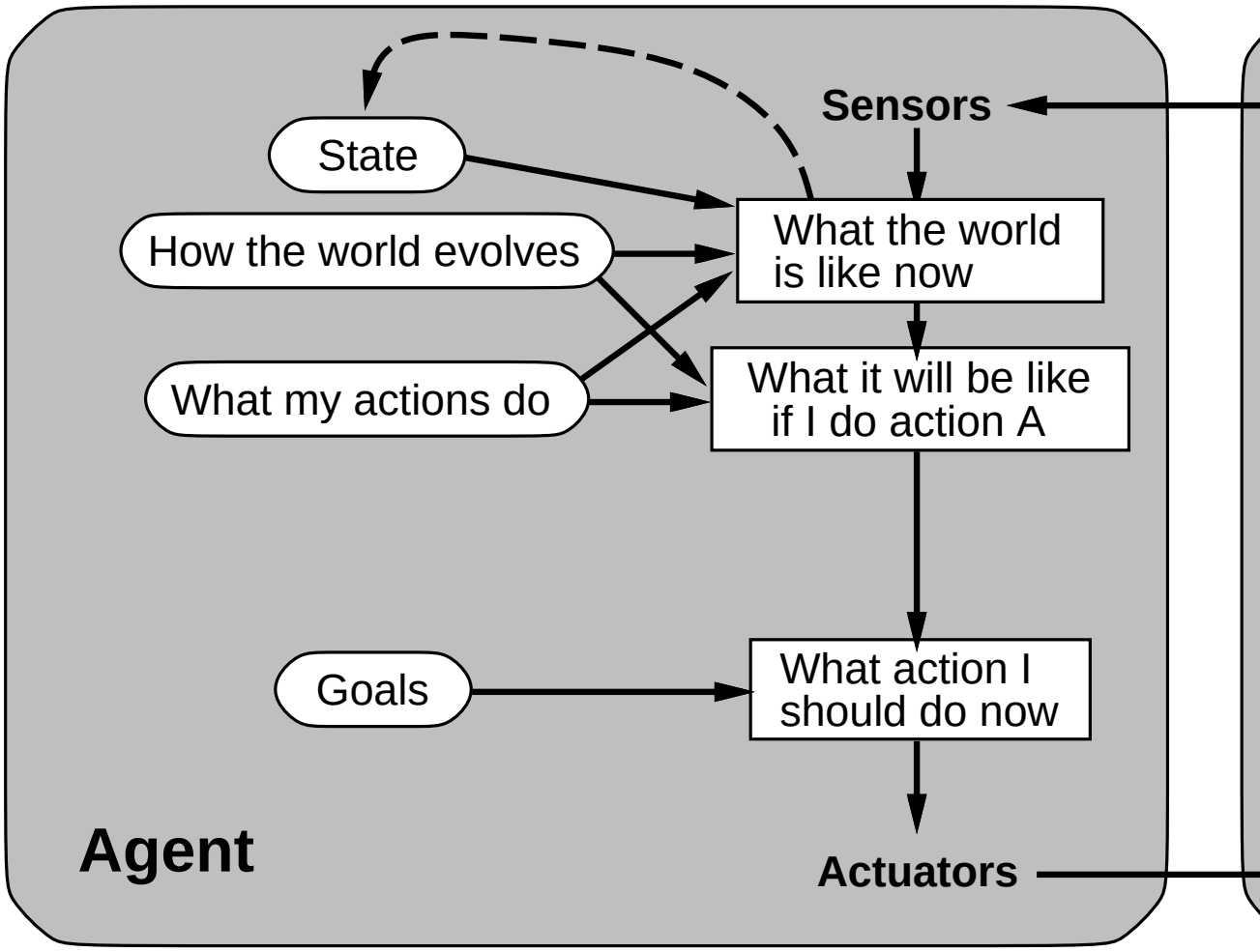
Example

function REFLEX-VACUUM-AGENT([*location*,*status*]) **returns** an
static: *last_A*, *last_B*, numbers, initially ∞

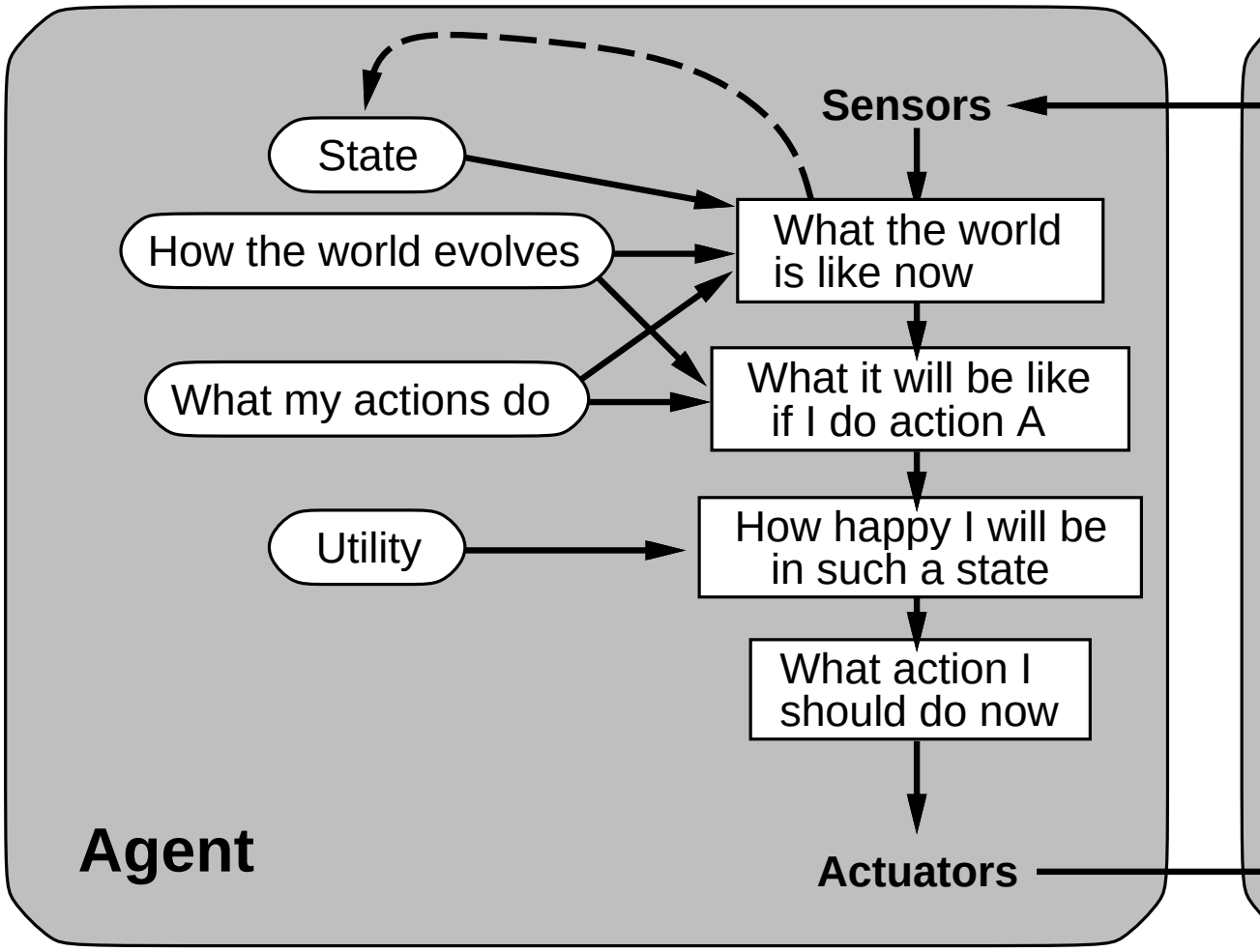
if *status* = *Dirty* **then** . . .

```
(defun make-reflex-vacuum-agent-with-state-progr
  (let ((last-A infinity) (last-B infinity))
  #'(lambda (percept)
      (let ((location (first percept)) (status (
        (incf last-A) (incf last-B)
        (cond
         ((eq status 'dirty)
          (if (eq location 'A) (setq last-A 0) (
          'Suck)
         ((eq location 'A) (if (> last-B 3) 'Rig
         ((eq location 'B) (if (> last-A 3) 'Lef
```

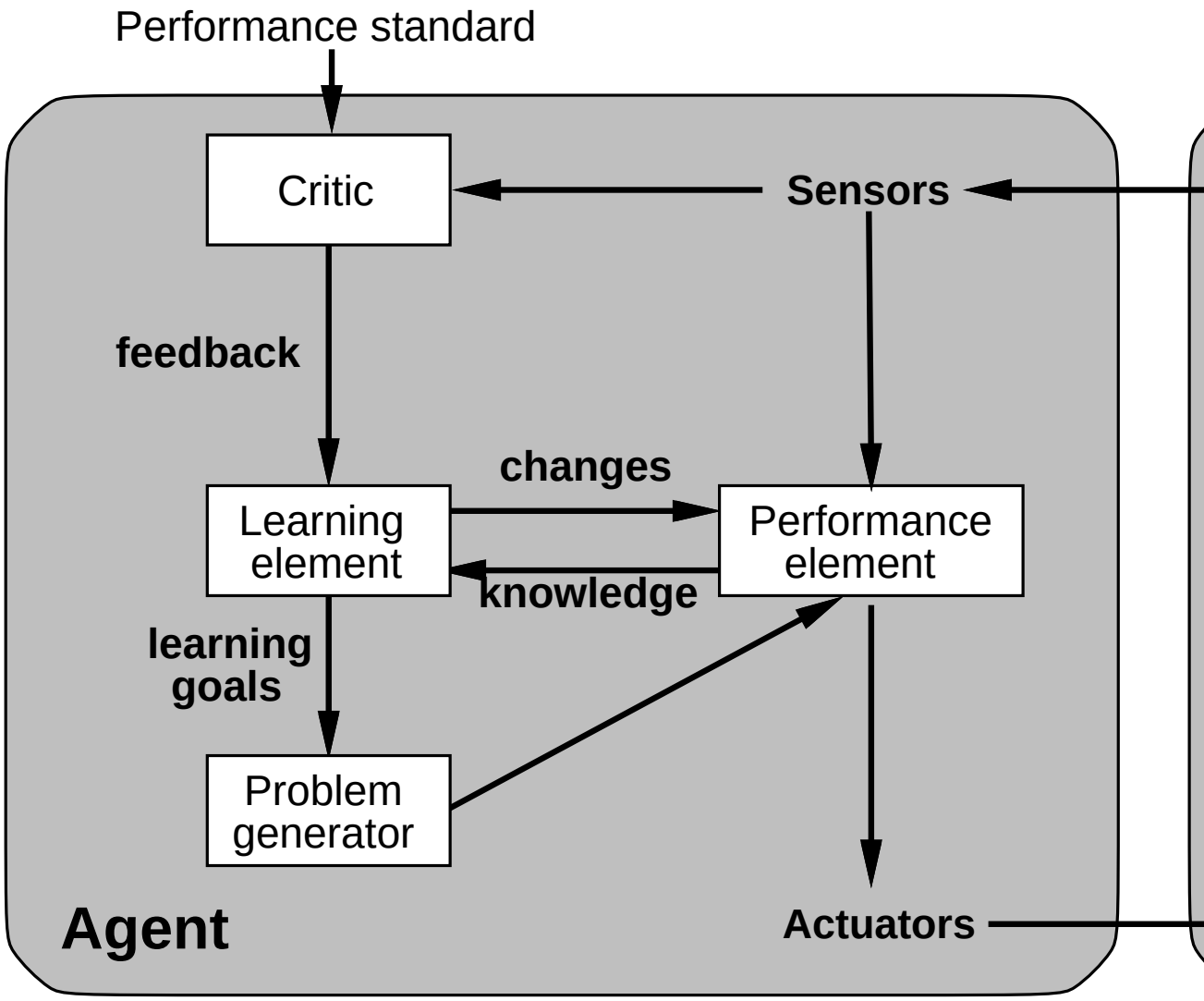
Goal-based agents



Utility-based agents



Learning agents



Summary

Agents interact with environments through actuators and sens

The agent function describes what the agent does in all circun

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? s

Several basic agent architectures exist:
reflex, reflex with state, goal-based, utility-based