Artificial Intelligence

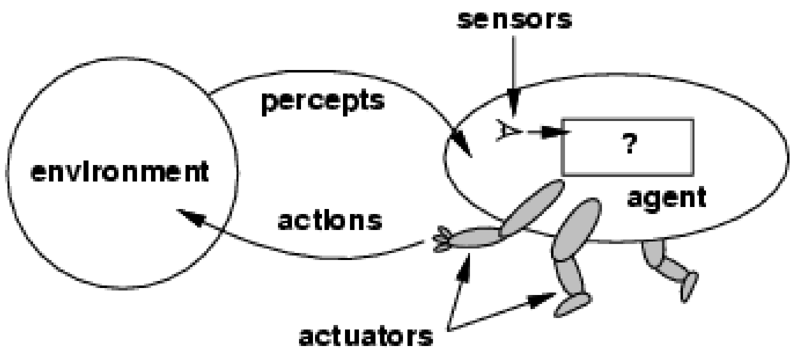
Engr. Madeha Mushtaq
Department of Computer Science
Iqra National University

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

Agents

- A software agent receives keystrokes, file contents, and network packets as sensory inputs and acts on the environment by displaying on the screen, writing files, and sending network packets.
- We use the term percept to refer to the agent's perceptual inputs at any given instant.
- An agent's percept sequence is the complete history of everything the agent has ever perceived.



Agents and environments

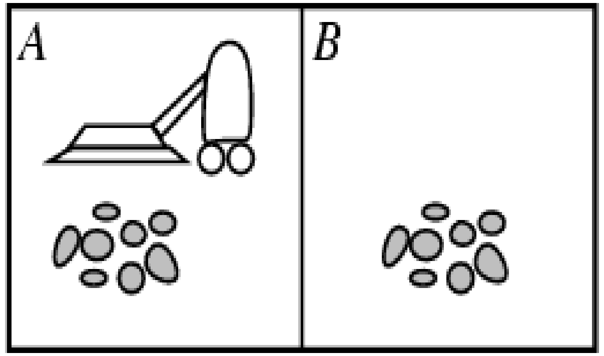
- Mathematically speaking, an agent's behavior is described by the agent function that maps any given percept sequence to an action.
- The agent function maps from percept histories to actions:

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

- Internally, the agent function for an artificial agent will be implemented by an agent program.
- The agent function is abstract mathematical description; the agent program is a concrete implementation, running within some physical system.
- The agent program runs on the physical architecture to produce *f*.

Vacuum-cleaner world

- This particular world has just two locations: squares A and B.
- The vacuum agent perceives which square it is in and whether there is dirt in the square.
- It can choose to move left, move right, suck up the dirt, or do nothing.
- One very simple agent function is the following: if the current square is dirty, then suck; otherwise, move to the other square.



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*
⋮	⋮
[*A*, *Clean*], [*A*, *Clean*], [*A*, *Clean*]	*Right*
[*A*, *Clean*], [*A*, *Clean*], [*A*, *Dirty*]	*Suck*
⋮	⋮

- Percepts: location and state of the environment, e.g., [A,Dirty], [B,Clean]
- Actions: *Left*, *Right*, *Suck*, *NoOp*

Rational agents

- What is 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, based on the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

Rational agents

- 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, amount of noise generated, etc.
- Rationality is distinct from omniscience (all-knowing with infinite knowledge).

Rational agents

- Agents can perform actions in order to modify future percepts so as to obtain useful information (information gathering, exploration)
- An agent is autonomous if its behavior is determined by its own percepts & experience (with ability to learn and adapt) without depending solely on build-in knowledge.
- For example, a vacuum-cleaning agent that learns to foresee where and when additional dirt will appear will do better than one that does not.

Task Environment

- Before we design an intelligent agent, we must specify its “task environment”:
- PEAS:
  - Performance measure
  - Environment
  - Actuators
  - Sensors

PEAS

- Example: Agent = taxi driver
  - Performance measure: Safe, fast, legal, comfortable trip, maximize profits
  - Environment: Roads, other traffic, pedestrians, customers
  - Actuators: Steering wheel, accelerator, brake, signal, horn
  - Sensors: Cameras, sonar, speedometer, GPS, odometer, engine sensors, keyboard

PEAS

- First, what is the performance measure?
- Desirable qualities include getting to the correct destination;
- minimizing fuel consumption and wear and tear; minimizing the trip time or cost; minimizing violations of traffic laws and disturbances to other drivers;
- Maximizing safety and passenger comfort.
- Next, what is the driving environment that the taxi will face?
- Any taxi driver must deal with a variety of roads, ranging from rural lanes and urban alleys to 12-lane freeways.
- The roads contain other traffic, pedestrians, stray animals, road works, police cars etc.

PEAS

- The actuators for an automated taxi include those available to a human driver: control over the engine through the accelerator and control over steering and braking.
- The basic sensors for the taxi will include one or more controllable video cameras so that it can see the road; it might augment these with infrared or sonar sensors to detect distances to other cars and obstacles.

PEAS

- Example: Agent = Medical diagnosis system

  Performance measure: Healthy patient, minimize costs, lawsuits

  Environment: Patient, hospital, staff

  Actuators: Screen display (questions, tests, diagnoses, treatments, referrals)

  Sensors: Keyboard (entry of symptoms, findings, patient's answers)

PEAS

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

Environment types

- Fully observable (vs. partially observable):
- If an agent's sensors give it access to the complete state of the environment at each point in time, then we say that the task environment is fully observable.
- An environment might be partially observable because of noisy and inaccurate sensors or because parts of the state are simply missing from the sensor data.
- If the agent has no sensors at all then the environment is unobservable.

Environment types

- Deterministic vs. stochastic: If the next state of the environment is completely determined by the current state and the action executed by the agent, then we say the environment is deterministic; otherwise, it is stochastic.
- In principle, an agent need not worry about uncertainty in a fully observable, deterministic environment.
- If the environment is partially observable, however, then it could appear to be stochastic.

Environment types

- Episodic vs. sequential:
- An agent’s action is divided into atomic episodes.
- In each episode the agent receives a percept and then performs a single action.
- The next episode does not depend on the actions taken in previous episodes.
- In sequential environments, on the other hand, the current decision could affect all future decisions.

Environment types

- Static (vs. dynamic):
- If the environment can change while an agent is deliberating, then we say the environment is dynamic for that agent; otherwise, it is static.
- Static environments are easy to deal with because the agent need not keep looking at the world while it is deciding on an action, nor need it worry about the passage of time.
- Taxi driving is clearly dynamic: the other cars and the taxi itself keep moving while the driving algorithm dithers about what to do next.
- Chess, when played with a clock, is semi dynamic.
- Crossword puzzles are static.

Environment types

- The discrete/continuous distinction applies to the state of the environment, to the way time is handled, and to the percepts and actions of the agent.
- For example, the chess environment has a finite number of distinct states (excluding the clock).
- Chess also has a discrete set of percepts and actions.
- Taxi driving is a continuous-state and continuous-time problem: the speed and location of the taxi and of the other vehicles sweep through a range of continuous values and do so smoothly over time.

Environment types

- Single agent vs. multi-agent:
- The distinction between single-agent and multiagent environments may seem simple enough.
- For example, an agent solving a crossword puzzle by itself is clearly in a single-agent environment, whereas an agent playing chess is in a two agent environment.

task environm.	observable	determ./ stochastic	episodic/ sequential	static/ dynamic	discrete/ continuous	agents
crossword puzzle	fully	determ.	sequential	static	discrete	single
chess with clock	fully	strategic	sequential	semi	discrete	multi
poker	partial	stochastic	sequential	static	discrete	multi
back gammon	fully	stochastic	sequential	static	discrete	multi
taxi driving	partial	stochastic	sequential	dynamic	continuous	multi
medical diagnosis	partial	stochastic	sequential	dynamic	continuous	single
image analysis	fully	determ.	episodic	semi	continuous	single
partpicking robot	partial	stochastic	episodic	dynamic	continuous	single
refinery controller	partial	stochastic	sequential	dynamic	continuous	single
interact. Eng. tutor	partial	stochastic	sequential	dynamic	discrete	multi

Agent Programs

- The job of AI is to design an agent program that implements the agent function—the mapping from percepts to actions.
- We assume this program will run on some sort of computing device with physical sensors and actuators—we call this the architecture:
- agent = architecture + program .

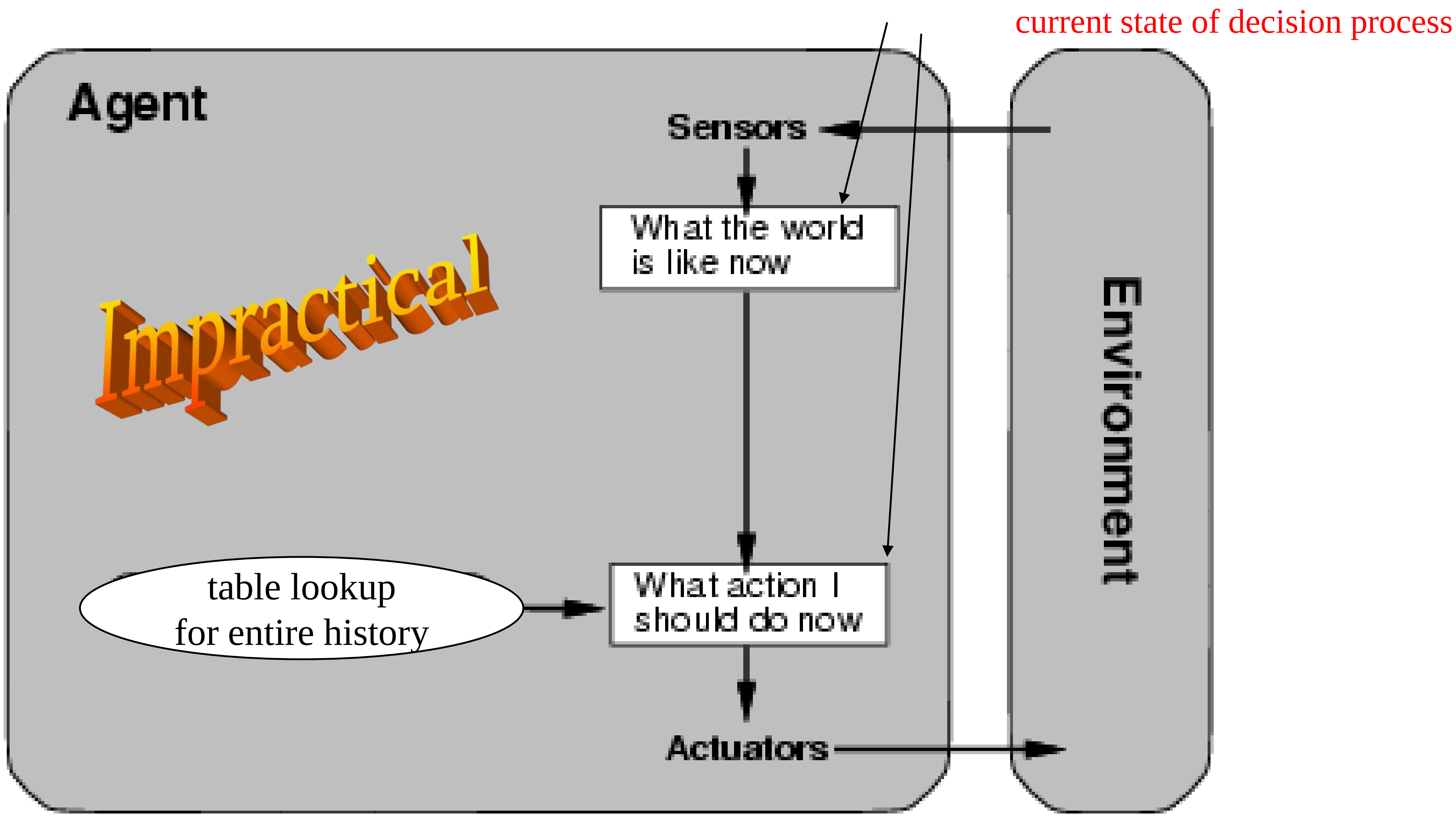
Agent types

- There are five basic kinds of agent programs that embody the principles underlying almost all intelligent systems:
- Table Driven agent
- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents

Table Driven Agent

- For this approach, designers must construct a table that contains the appropriate action for every possible percept sequence.
- The daunting size of these tables means that:
  - no physical agent in this universe will have the space to store the table,
  - the designer would not have time to create the table,
  - no agent could ever learn all the right table entries from its experience,
  - even if the environment is simple enough to yield a feasible table size, the designer still has no guidance about how to fill in the table entries.

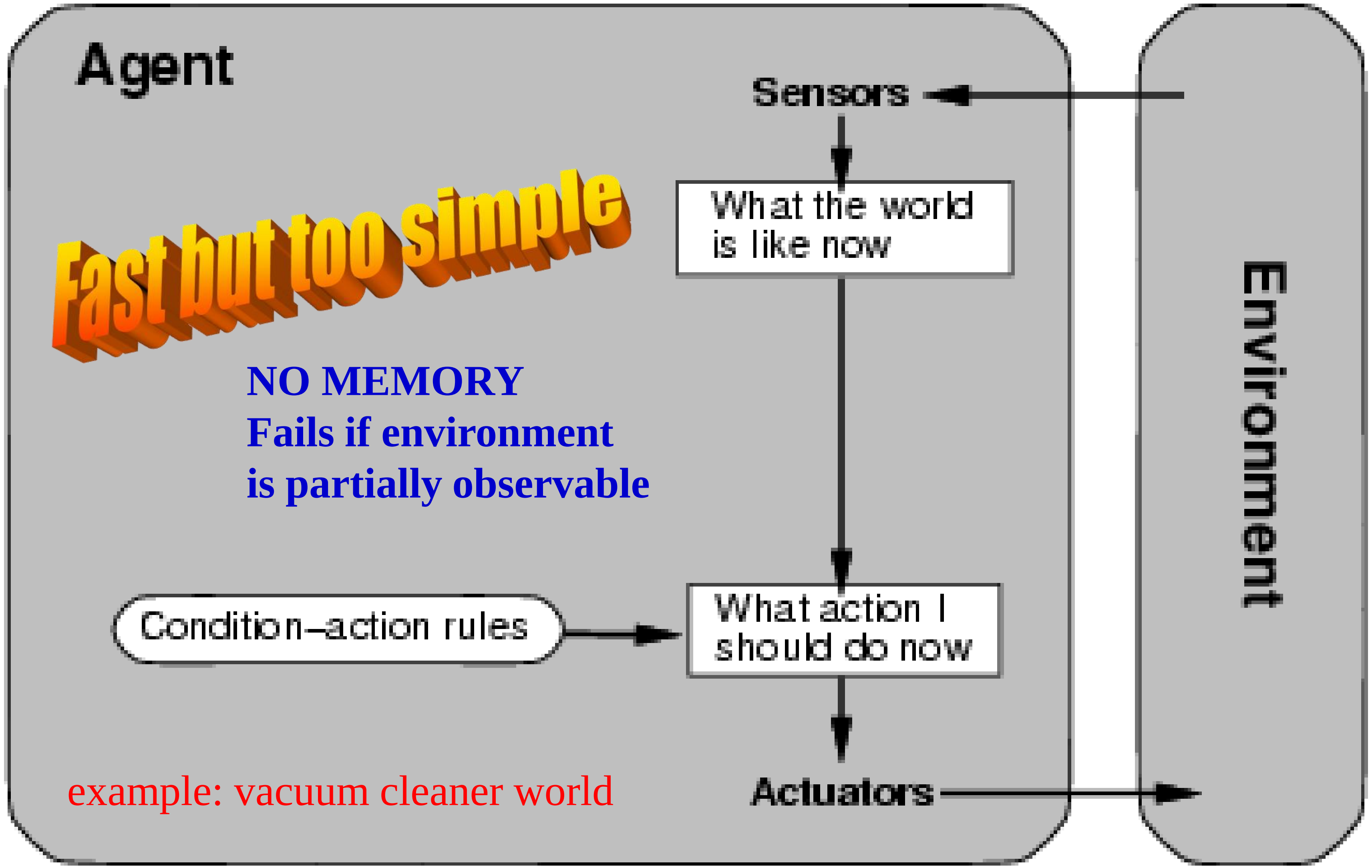
Table Driven Agent



Simple reflex agents

- The simplest kind of agent is the simple reflex agent.
- These agents select actions on the basis of the current percept, ignoring the rest of the percept history.
- For example, the vacuum agent is a simple reflex agent, because its decision is based only on the current location and on whether that location contains dirt.
- The agent program for vacuum cleaner world:
  - *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*

Simple reflex agents



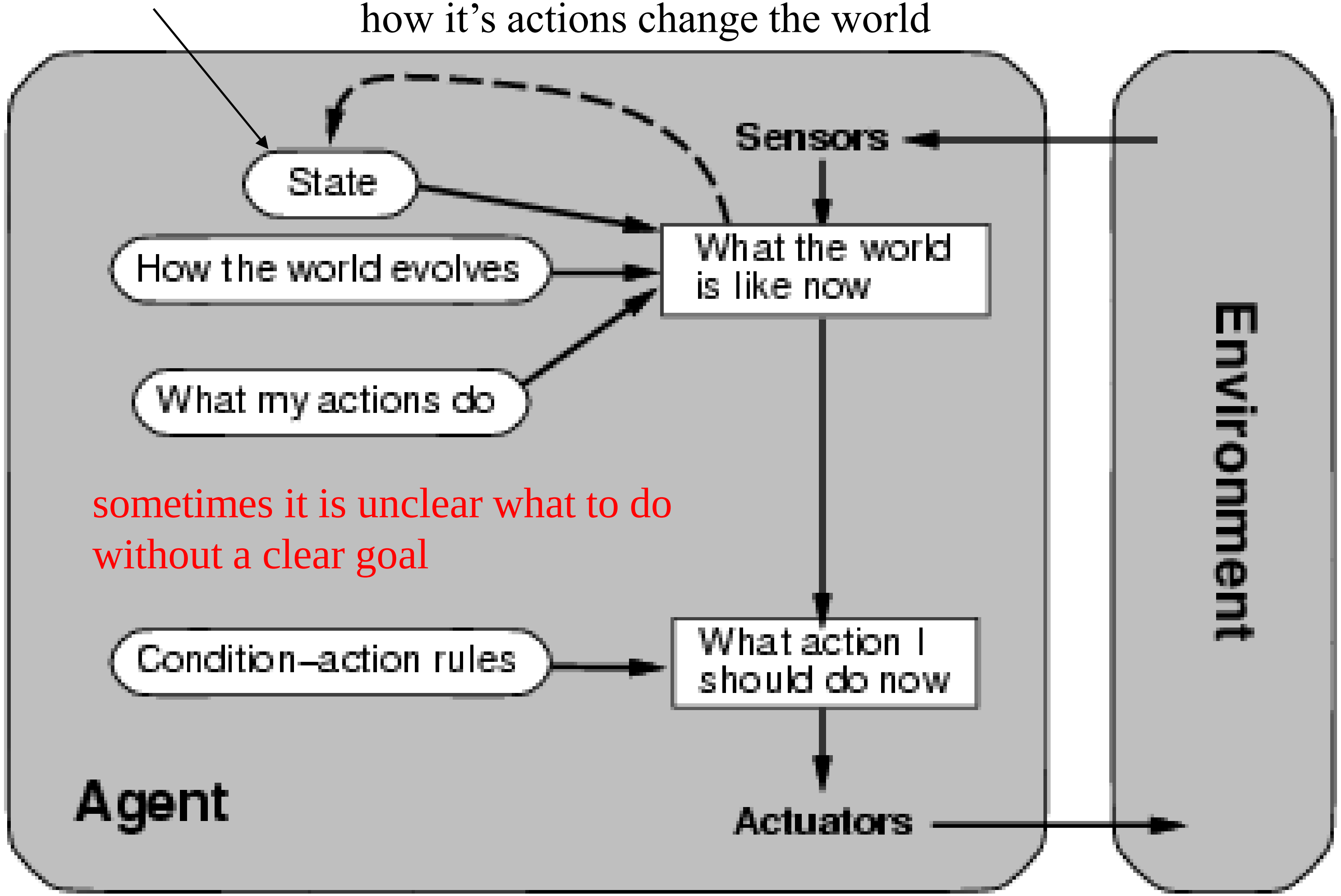
Model-based reflex agents

- The knowledge about “how the world works”—whether implemented in simple Boolean circuits or in complete scientific theories—is called a model of the world.
- An agent that uses such a model is called a model-based agent.

Model-based reflex agents

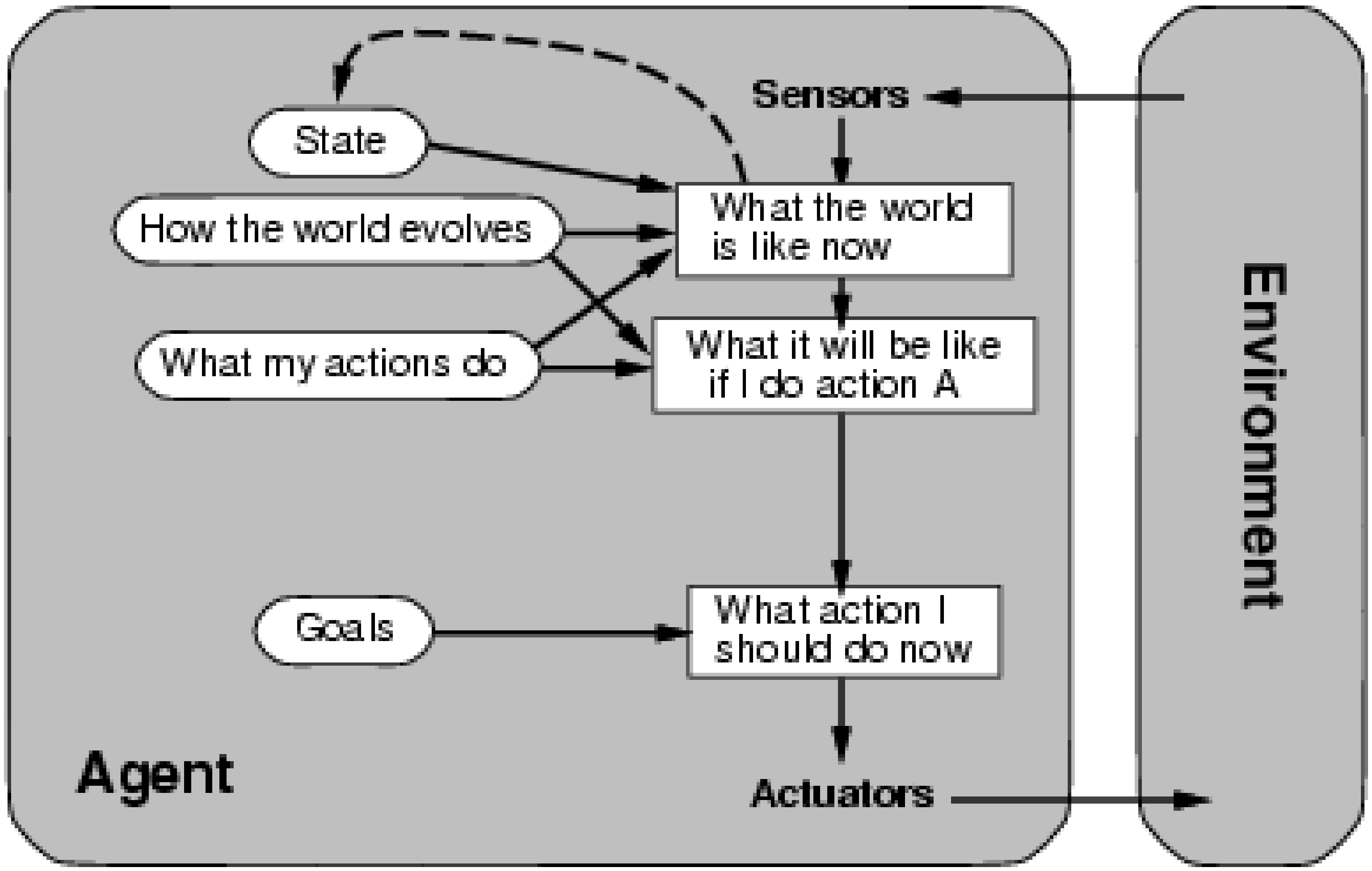
description of
current world state

Models the world by:
modeling how the world chances
how it’s actions change the world



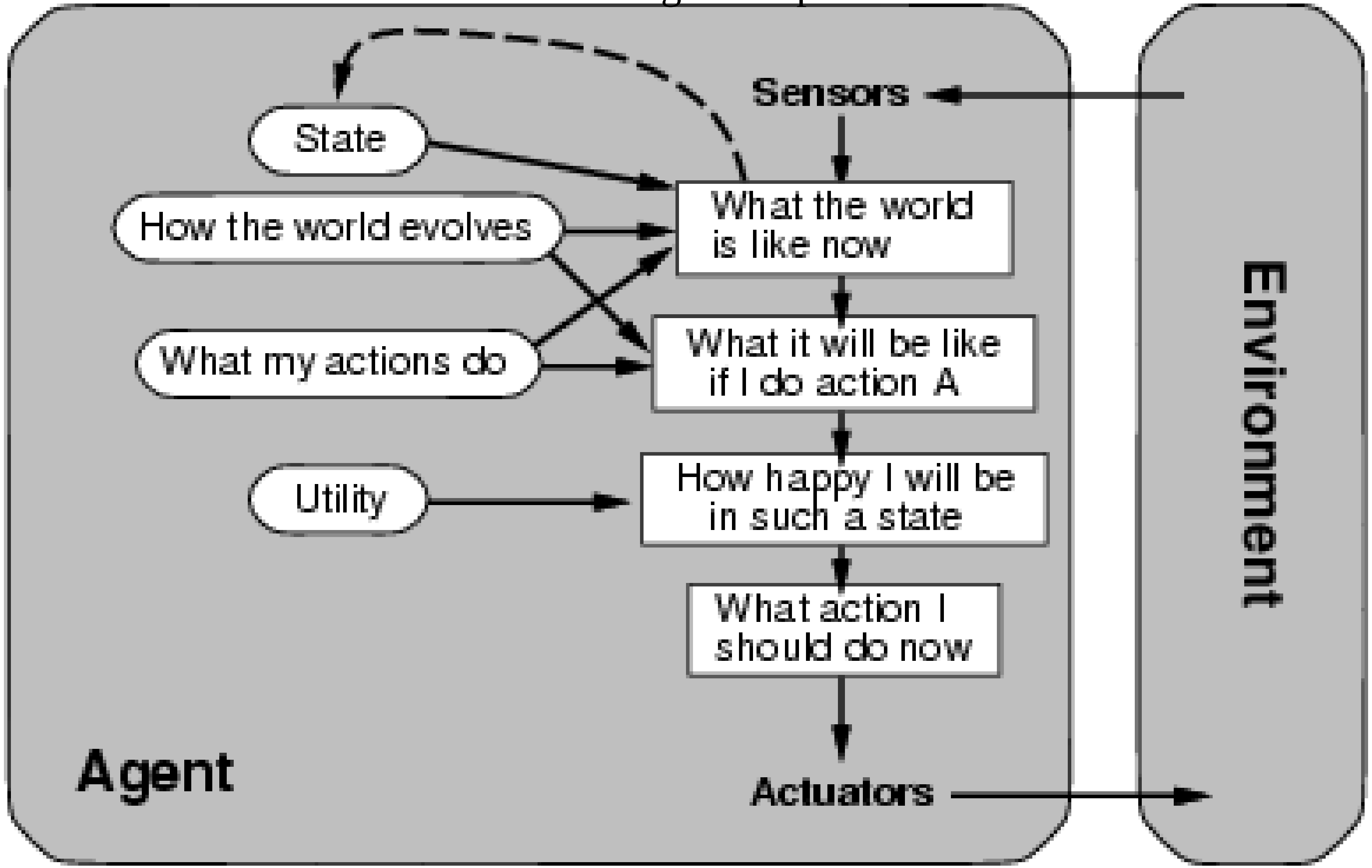
Goal-based agents

Goals provide reason to prefer one action over the other.
We need to predict the future: we need to plan & search



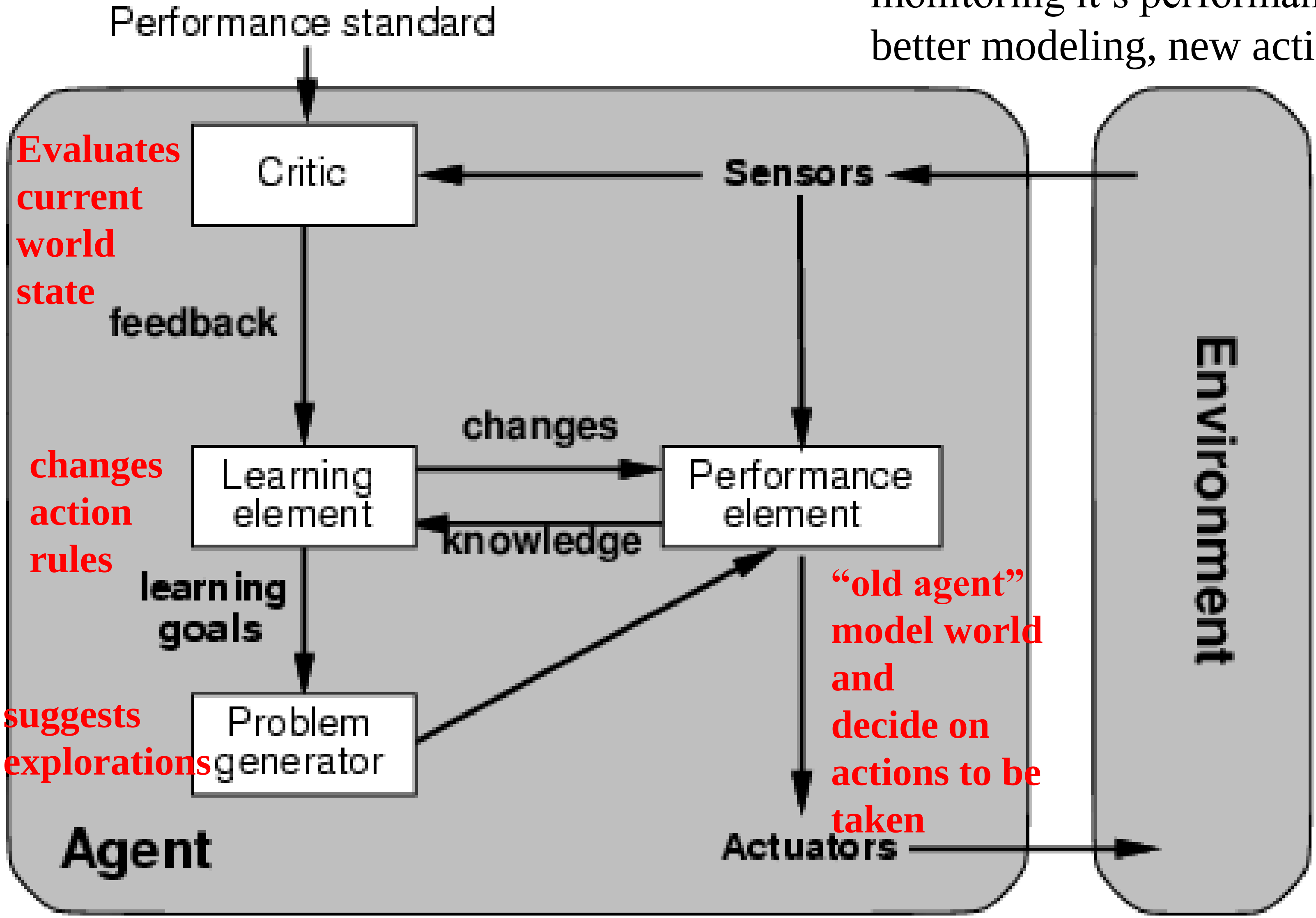
Utility-based agents

Some solutions to goal states are better than others.
Which one is best is given by a utility function.
Which combination of goals is preferred?



Learning agents

How does an agent improve over time? By monitoring it's performance and suggesting better modeling, new action rules, etc.



End of Slides