Chapter 2

Intelligent Agents

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;

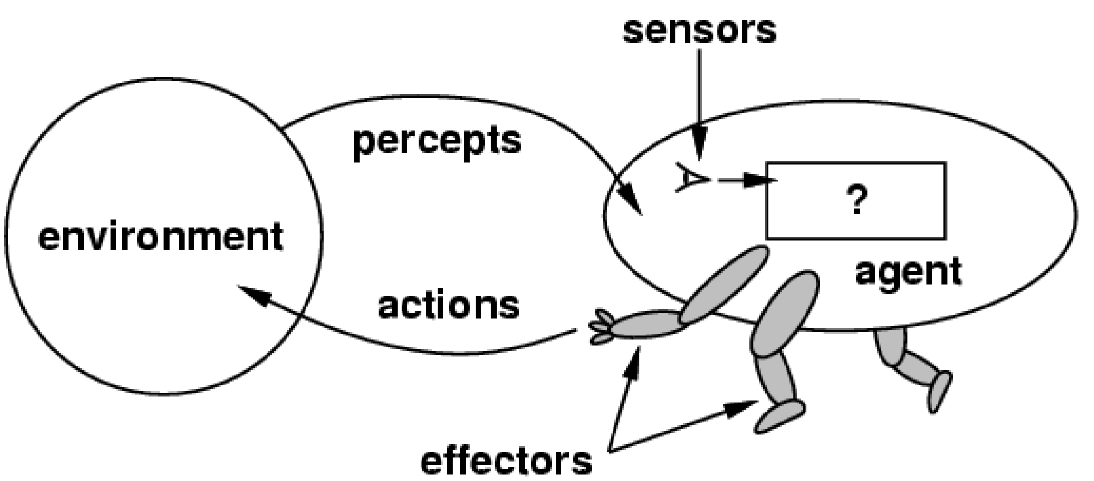
hands, legs, mouth, and other body parts for actuators

Robotic agent:

cameras and infrared range finders for sensors; various motors for actuators

How to design an intelligent agent?

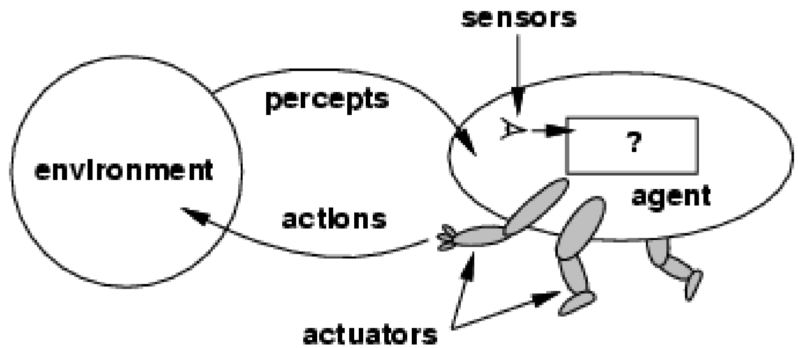
- An **intelligent agent** perceives its environment via **sensors** and acts rationally upon that environment with its **effectors**.
- A discrete agent receives **percepts** one at a time, and maps this percept sequence to a sequence of discrete **actions**.
- Properties
  - –Autonomous
  - –Reactive to the environment
  - –Pro-active (goal-directed)
  - –Interacts with other agents via the environment



Sensors/percepts and effectors/actions?

- Humans
  - Sensors: Eyes (vision), ears (hearing), skin (touch), tongue (gustation), nose (olfaction), neuromuscular system (proprioception)
  - Percepts:
    - At the lowest level – electrical signals from these sensors
    - After preprocessing – objects in the visual field (location, textures, colors, …), auditory streams (pitch, loudness, direction), …
  - Effectors: limbs, digits, eyes, tongue, …
  - Actions: lift a finger, turn left, walk, run, carry an object, …

Agents and environments



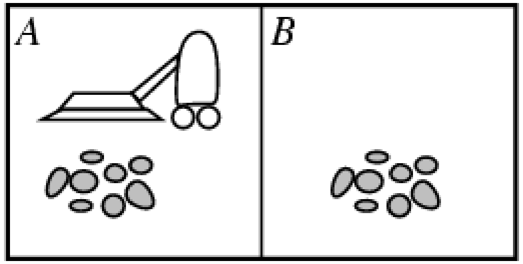
- The agent function maps from percept histories to actions:

  [*f*: P* → A]

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

- agent = architecture + program

Vacuum-cleaner world



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

Rational agents

- Performance measure: An objective criterion for success of an agent's behavior, e.g.,
  - Robot driver?
  - Chess-playing program?
  - Spam email classifier?

- Rational Agent: selects actions that is *expected* to maximize its performance measure,
  - given percept sequence
  - given agent’s built-in knowledge

  - sidepoint: how to maximize expected future performance, given only historical data

Rational agents

- Rational Agent → Always try to maximize performance.
- No Agent is Omniscience. Rationality is distinct from omniscience (all-knowing with infinite knowledge)

- 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 built-in knowledge

- To survive, agents must have:
  - Enough built-in knowledge to survive.
  - The ability to learn

Task Environment

- Before we design an intelligent agent, we must specify its "task environment":

PEAS:

Performance measure

Environment

Actuators

Sensors

DARPA Robotics Challenge

- A prize competition funded by the US Defense Advanced Research Projects Agency, held from 2012 to 2015

- Aimed to develop semi-autonomous ground robots that could do complex tasks in dangerous, degraded, human-engineered environments

DARPA Robotics Challenge

The initial task requirements for robot entries are

- Drive a utility vehicle at the site
- Travel dismounted across rubble
- Remove debris blocking an entryway
- Open a door and enter a building
- Climb an industrial ladder and traverse an industrial walkway
- Use a tool to break through a concrete panel
- Locate and close a valve near a leaking pipe
- Connect a fire hose to a standpipe and turn on a valve

PEAS

- Example: Agent = robot driver in DARPA Challenge
  - Performance measure:
    - Time to complete course
  - Environment:
    - Roads, other traffic, obstacles
  - Actuators:
    - Steering wheel, accelerator, brake, signal, horn
  - Sensors:
    - Optical cameras, lasers, sonar, accelerometer, speedometer, GPS, odometer, engine sensors,

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)

Environment types

- Fully observable (vs. partially observable):
  - An agent's sensors give it access to the complete state of the environment at each point in time.

- Deterministic (vs. stochastic):
  - The next state of the environment is completely determined by the current state and the action executed by the agent.
  - If the environment is deterministic except for the actions of other agents, then the environment is strategic
  - Deterministic environments can appear stochastic to an agent (e.g., when only partially observable)

- Episodic (vs. sequential):
  - An agent's action is divided into atomic episodes. Decisions do not depend on previous decisions/actions.

Environment types

- Static (vs. dynamic):
  - The environment is unchanged while an agent is deliberating.
  - The environment is semidynamic if the environment itself does not change with the passage of time but the agent's performance score does

- Discrete (vs. continuous):
  - A discrete set of distinct, clearly defined percepts and actions.
  - How we represent or abstract or model the world

- Single agent (vs. multi-agent):
  - An agent operating by itself in an environment. Does the other agent interfere with my performance measure?

Characteristics of environments

	Fully observable?	Deterministic	Episodic	Static	Discrete?	Single agent?
Solitaire						
Driving						
Internet shopping						
Medical diagnosis						

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Driving	**No**	**No**	**No**	**No**	**No**	**No**
Internet shopping						
Medical diagnosis						

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Driving	No	No	No	No	No	No
Internet shopping	**No**	**No**	**No**	**No**	**Yes**	**No**
Medical diagnosis						

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Driving	No	No	No	No	No	No
Internet shopping	No	No	No	No	Yes	No
Medical diagnosis	**No**	**No**	**No**	**No**	**No**	**Yes**

→ Lots of real-world domains fall into the hardest case!

task environm.	observable	deterministic/ 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						
taxi driving	partial	stochastic	sequential	dynamic	continuous	multi
medical diagnosis						
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. tutor	partial	stochastic	sequential	dynamic	discrete	multi

What is the environment for the DARPA Challenge?

- Agent = robotic vehicle

- Environment = 130-mile route through desert
  - Observable?
  - Deterministic?
  - Episodic?
  - Static?
  - Discrete?
  - Agents?

Agent types

- Five basic types in order of increasing generality:
  - Table Driven agent
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
    - Problem-solving agents
  - Utility-based agents
    - Can distinguish between different goals
  - Learning agents

Some agent types

- **Table-driven agents**
  - use a percept sequence/action table in memory to find the next action. They are implemented by a (large) **lookup table**. It is not autonomous.
- **Simple reflex agents**
  - are based on **condition-action rules**, implemented with an appropriate production system. They are stateless devices which do not have memory of past world states. It can not save history.
- **Agents with memory**
  - have **internal state**, which is used to keep track of past states of the world.
- **Agents with goals**
  - are agents that, in addition to state information, have **goal information** that describes desirable situations. Agents of this kind take future events into consideration. **Never thinks about cost.**
- **Utility-based agents**
  - base their decisions on **classic axiomatic utility theory** in order to act rationally. **Always thinks about cost.**

Table-driven/reflex agent Architecture

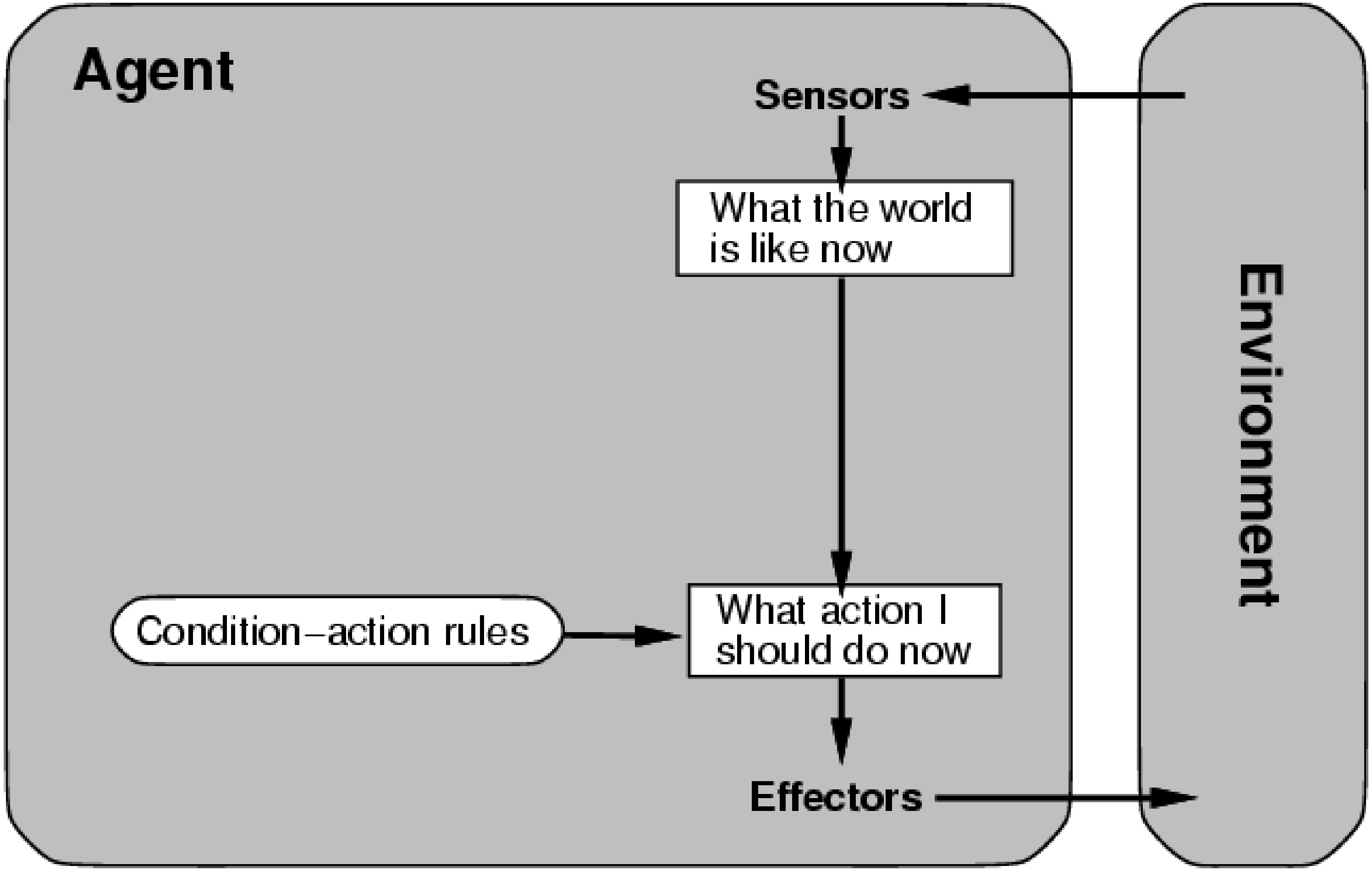


Table-driven agents

- **Table lookup** of percept-action pairs mapping from every possible perceived state to the optimal action for that state

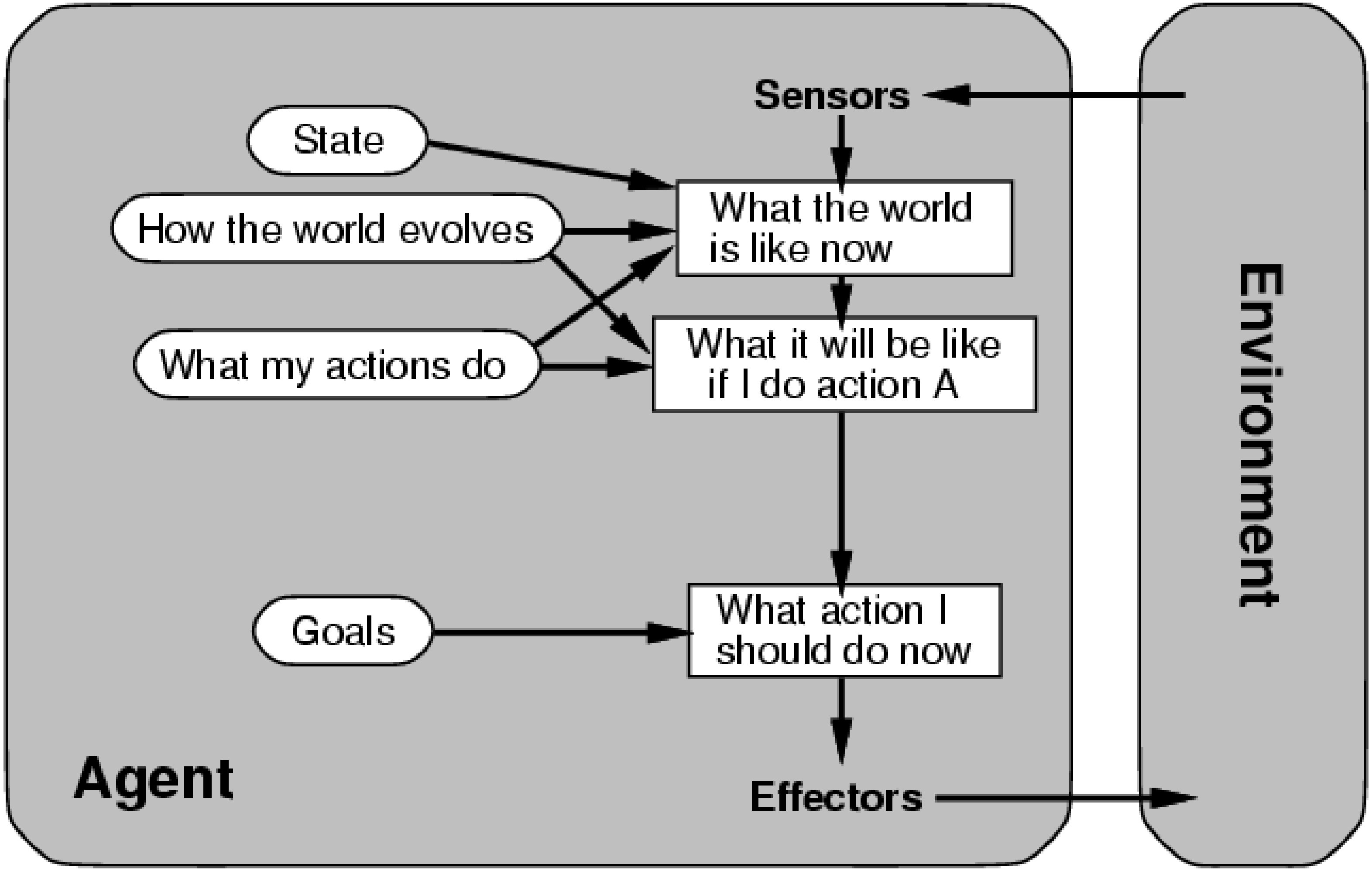
- **Problems**
  - Too big to generate and to store (Chess has about 10^{120} states, for example)
  - No knowledge of non-perceptual parts of the current state
  - Not adaptive to changes in the environment; requires entire table to be updated if changes occur
  - Looping: Can't make actions conditional on previous actions/states

Simple reflex agents

- **Rule-based reasoning** to map from percepts to optimal action; each rule handles a collection of perceived states

- **Problems**
  - Still usually too big to generate and to store
  - Still not adaptive to changes in the environment; requires collection of rules to be updated if changes occur
  - Still can't make actions conditional on previous state

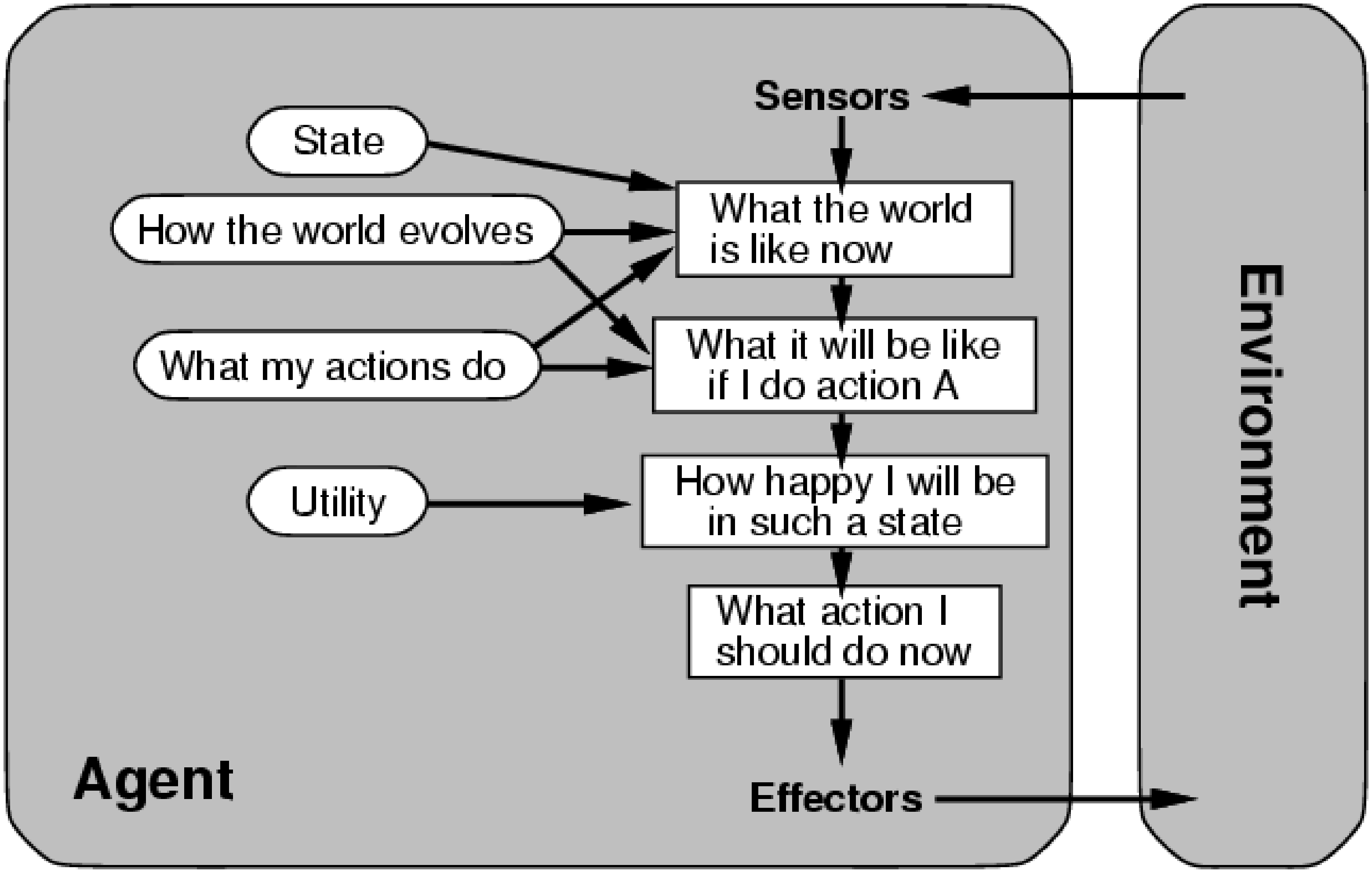
Goal-based agent: Architecture



Goal-based agents

- Choose actions so as to achieve a (given or computed) goal.
- A goal is a description of a desirable situation.
- Keeping track of the current state is often not enough - need to add goals to decide which situations are good
- **Deliberative** instead of **reactive**.
- May have to consider long sequences of possible actions before deciding if goal is achieved – involves consideration of the future, *“what will happen if I do...?”*

Complete utility-based agent



Utility-based agents

- When there are multiple possible alternatives, how to decide which one is best?
- A goal specifies a crude distinction between a happy and unhappy state, but often need a more general performance measure that describes “degree of happiness.”
- Utility function **U: State $\rightarrow$ Reals** indicating a measure of success or happiness when at a given state.

Summary

- An **agent** perceives and acts in an environment, has an architecture, and is implemented by an agent program.
- An **ideal agent** always chooses the action which maximizes its expected performance, given its percept sequence so far.
- An **autonomous agent** uses its own experience rather than built-in knowledge of the environment by the designer.

Summary (Contd.)

- An **agent program** maps from percept to action and updates its internal state.
  - **Reflex agents** respond immediately to percepts.
  - **Goal-based agents** act in order to achieve their goal(s).
  - **Utility-based agents** maximize their own utility function.
- **Representing knowledge** is important for successful agent design.
- The most challenging environments are
  - partially observable
  - stochastic
  - sequential
  - Dynamic
  - continuous
  - contain multiple intelligent agents