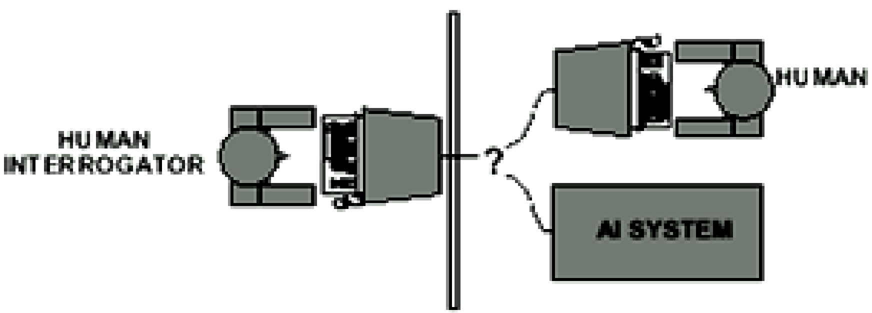
# Chapter Two
# Intelligent Agents

Faris A.
College of Computing
Debre Berhan University, Ethiopia

# Intelligent Agents (IA)

- The Turing test (The Imitation Game): Operational definition of intelligence.




**Computer needs to possess**: Natural language processing, Knowledge representation, Automated reasoning, and Machine learning

# Intelligent Agents (IA)…

- One time Turing test
  - AI System: hello My Name is ChatBot, what is your name?
  - Interrogator : Lawrent
  - AI System : Nice to meet you Lawrent
  - Interrogator: Can you test the Turing Test?
  - AI System : You be the Judge of that Lawrent
  - Interrogator: Tell me about yourself
  - AI System: I am an Artificial Linguistic Entity
  - Interrogator: Sorry, you just failed the Turing test

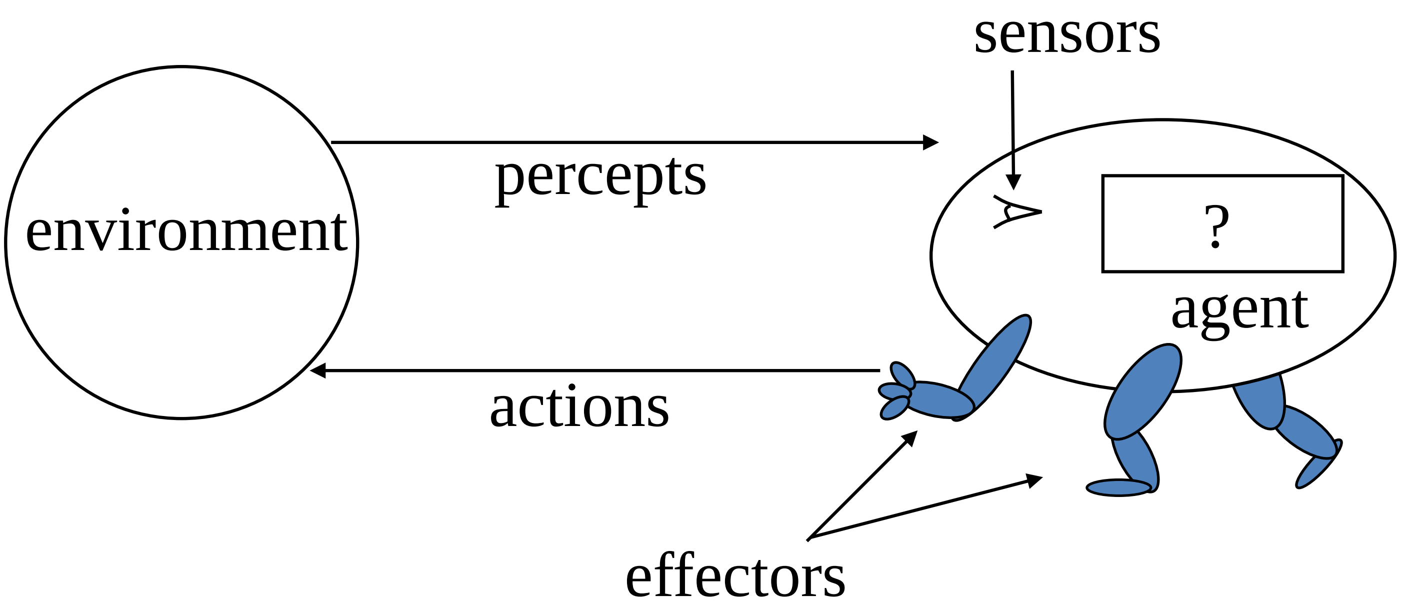
# Intelligent Agents (IA)…

- Personal View of AI
  - I want to build a robot that will
    - Clean my house
    - Cook when I don not want to
    - Wash my clothes
    - Cut my hair
    - Take a note when I am in a meeting
    - Do things the things that I don’t feel like doing

# Intelligent Agents (IA)…

- IA:- Anything that can be *viewed as* perceiving its environment through sensors and acting upon that environment through its effectors to maximize progress towards its goals.
  - The set of inputs at a given time is called a percept.
  - The sequence of percept influence the action of an agent. Action is operations involving an effectors .
  - Agent can gave a goal which is tries to achieve.
  - The performance measure is used to evaluate an agent.

# 2.1 An agent and its environment



# Intelligent Agents (IA)…

- The *agent* perceives its environment via sensors and acts in that environment with its effectors.

  **Properties**:
  - **Autonomous**
  - **Interacts** with other agents plus the environment
  - **Reactive** to the environment
  - **Goal-directed**
- An autonomous agent decides autonomously which action to take in the current situation to maximize progress towards its goals.

# Behavior and performance of IAs

- The **behavior** and the **performance** of an agent is evaluated in terms of the agent function.
- The agent function implements a mapping from perception history to action
- Perception (sequence) to Action Mapping: $f : P^* \rightarrow A$
  - Ideal mapping: specifies which actions an agent ought to take at any point in time
  - Description: Look-Up-Table, Closed Form, etc.
  - Performance measure: a *subjective* measure to characterize how successful an agent is (e.g., speed, power usage, accuracy, money, etc.)
  - *(degree of) Autonomy*: to what extent is the agent able to make decisions and take actions on its own?

# Intelligent Agents (IA)…

- Knowledge of Environment (World)
  - Different to sensory information from environment
- World knowledge can be (pre)-programmed in
  - Can also be updated/inferred by sensory information
- Using knowledge to inform choice of actions:
  - Use knowledge of current state of the world
  - Use knowledge of previous states of the world
  - Use knowledge of how its actions change the world
- Example of agent :
  - **Chess agent**
    - World knowledge is the board state (all the pieces)
    - Sensory information is the opponents move
    - It’s moves also change the board state
  - **Human** : sensor(eye, ears, skins, taste). effectors (Hand, fingers , legs, mouth)
  - **Robot**: sensor (camera, infrared,..), effectors (gripper, wheel, speaker,.)
  - **Soft ware agent** : function as sensor and actuator

# Task of the Environment

- The are four elements to take account when designing an agent to solve a particular problem
- PEAS
  - **P**erformance: measure the success of an agents behaviour
  - **E**nvironment:  where an agent operates
  - **A**ctuator: agent acts with in its environment
  - **S**ensor: the agent senses its environment
- E.g. Taxi driver
  - P: safe, fast, legal, maximize profit
  - E:road, other traffic, customer, pedestrian
  - A: brake, accelerator, signal, horn, steering wheel
  - S: camera, sonar, speedometer, GPS, odometer, engin sensor, keyboard.

# Details of the Environment

- Must take into account:
  - some qualities of the world
- Imagine:
  - A robot in the real world
  - A software agent dealing with web data streaming in
- considerations:
  - Accessibility, Determinism
  - Episodes
  - Dynamic/Static, Discrete/Continuous

# Accessibility of Environment

- Is everything an agent requires to choose its actions available to it via its sensors?
  - If so, the environment is fully accessible
  - E.g. Chess
- If not, parts of the environment are inaccessible
  - Agent must make informed guesses about world
  - E.g. Poker ,taxi driver, medical, image analysis

# Determinism in the Environment

- Does the change in world state
  - Depend only on current state and agent’s action?
  - E.g. Chess, image analysis
- Non-deterministic environments
  - Have aspects beyond the control of the agent
  - Utility functions have to guess at changes in world
  - People moved chairs to block its path
  - E.g. Card game, medical diagnosis, taxi-driver

# Episodic Environments

- **Episodic**: The agent's experience is divided into independent "episodes," each episode consisting of agent perceiving and then acting. Quality of action depends just on the episode itself, because subsequent episodes do not depend on what actions occur in previous episodes. → Do not need to think ahead
- E.g. image analysis
- Is the choice of current action
  - Dependent on previous actions?
  - If not, then the environment is episodic
- In non-episodic/ sequential environments:
  - Agent has to plan ahead:
    - Current choice will affect future actions
  - E.g. Taxi driver, puzzle, poker, medical

# Static or Dynamic Environments

- Static environments don’t change
  - While the agent is deliberating over what to do
  - E.g. puzzle
- Dynamic environments do change
  - So agent should/could consult the world when choosing actions
  - Alternatively: anticipate the change during deliberation
  - Alternatively: make decision very fast
  - E.g. Taxi driver, medical, image analysis

# Examples of agents in different types of applications

| Agent type | Percepts | Actions | Goals | Environment |
|---|---|---|---|---|
| Medical diagnosis system | Symptoms, findings, patient's answers | Questions, tests, treatments | Healthy patients, minimize costs | Patient, hospital |
| Satellite image analysis system | Pixels of varying intensity, color | Print a categorization of scene | Correct categorization | Images from orbiting satellite |
| Part-picking robot | Pixels of varying intensity | Pick up parts and sort into bins | Place parts in correct bins | Conveyor belts with parts |
| Refinery controller | Temperature, pressure readings | Open, close valves; adjust temperature | Maximize purity, yield, safety | Refinery |
| Interactive English tutor | Typed words | Print exercises, suggestions, corrections | Maximize student's score on test | Set of students |

## 2.3 Acting of Intelligent Agents (Rationality)

- An Intelligent Agent must sense, must act, and must be autonomous (to some extent). It also must be rational.
- AI is about building rational agents. An agent is something that perceives and acts. A rational agent always does the right thing.
  1. What are the functionalities (goals)?
  2. What are the components?
  3. How do we build them?

# Rationality

- Perfect Rationality assumes that the rational agent knows all and will take the action that maximizes her utility. Human beings do not satisfy this definition of rationality.
- **Rational Action is the action that maximizes the expected value of the performance measure** given the percept sequence to date.

# Rationality…

- However, a rational agent is not omniscient. It does not know the actual outcome of its actions, and it may not know certain aspects of its environment. Therefore rationality must take into account the limitations of the agent.
- The agent has too select the best action to the best of its knowledge depending on its percept sequence, its background knowledge and its feasible actions. An agent also has to deal with the expected outcome of the actions where the action effects are not deterministic.

# Bounded Rationality

- In 1957, Simon proposed the notion of Bounded Rationality: that property of an agent that behaves in a manner that is nearly optimal with respect to its goals as its resources will allow.
- Under these promises an intelligent agent will be expected to act optimally to the best of its abilities and its resource constraints.

# 2.4 Structure of Intelligent Agents

agent = architecture + **program**

**Agent program:** the implementation of
$f$ : P* $\rightarrow$ A, the agent's perception-action mapping

- **Architecture:** a device that can execute the agent program (e.g., general-purpose computer, specialized device, etc.)

**Ideal Rational Agent**: For each possible percept sequence, such an agent does whatever action is expected to maximize its performance measure, on the basis of the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

Physical agents vs. software agents
(software agents = softbots)

# 2.5 Agent types

1. Table-driven agent
2. Simple reflex agent
3. Reflex agent with internal state
4. Agent with explicit goals
5. Utility-based agent

# 1. Table-driven agent

**function** TABLE-DRIVEN-AGENT (percept) **returns** action
static: *percepts*, a sequence, initially empty
*table*, a table, indexed by percept sequences, initially fully specified

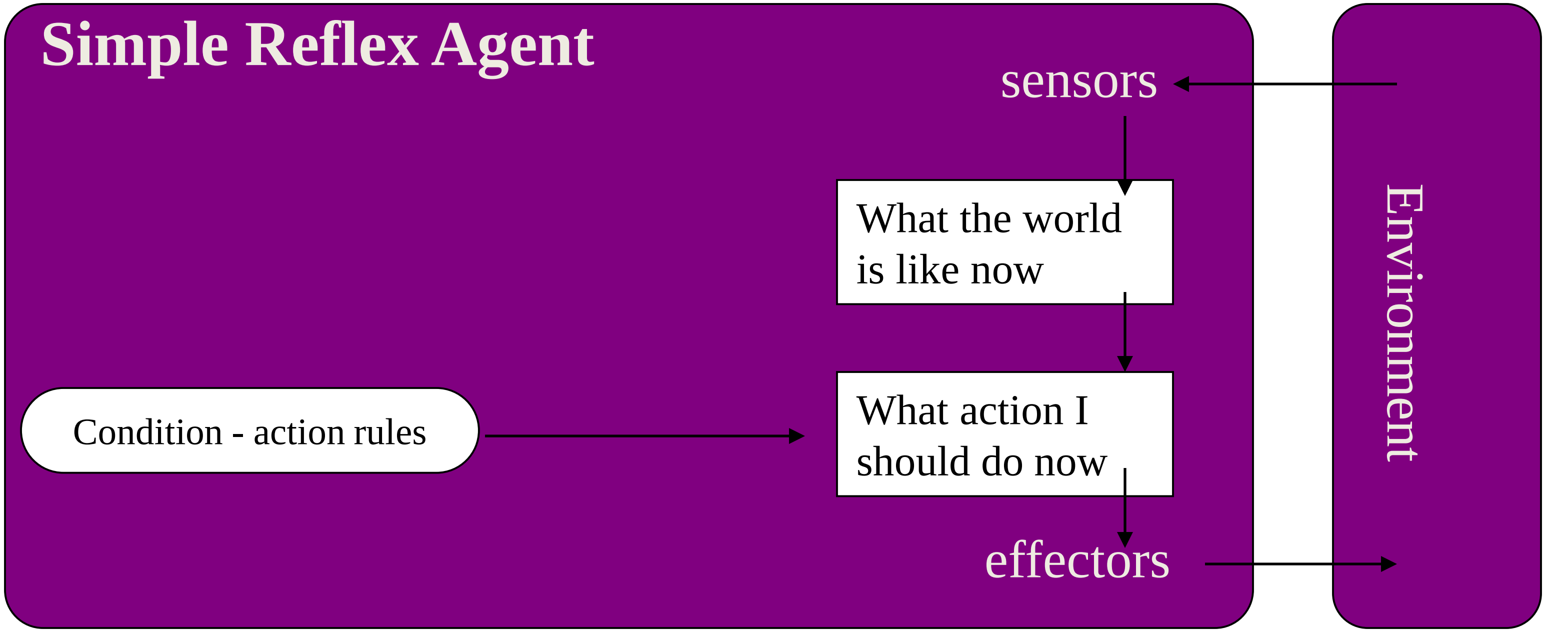
append *percept* to the end of *percepts*
*action* ← LOOKUP(*percepts*, *table*)
**return** *action*

An agent based on a prespecified lookup table. It keeps track of percept sequence and just looks up the best action

- **Problems**
  - – Huge number of possible percepts (consider an automated taxi with a camera as the sensor) => lookup table would be huge
  - – Takes long time to build the table
  - – Not adaptive to changes in the environment; requires entire table to be updated if changes occur

# 2. Simple reflex agent

- Differs from the lookup table based agent is that the condition (that determines the action) is already higher-level interpretation of the percepts
  - Percepts could be e.g. the pixels on the camera of the automated taxi



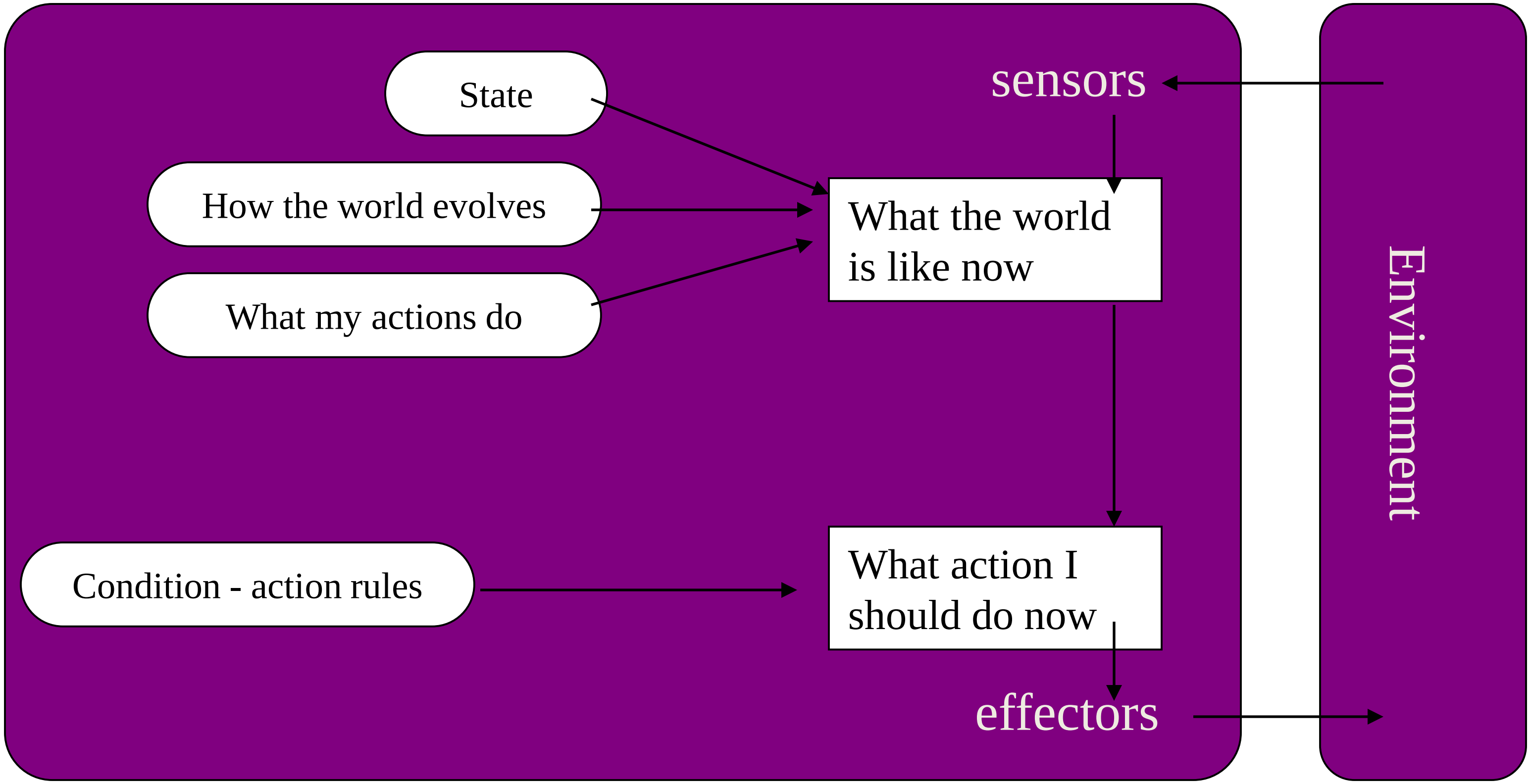

**function** SIMPLE-REFLEX-AGENT(*percept*) **returns** action
static: *rules*, a set of condition-action rules

*state* ← INTERPRET-INPUT (*percept*)
*rule* ← RULE-MATCH (*state*,*rules*)
*action* ← RULE-ACTION [*rule*]
**return** *action*

First match.
No further matches sought.
Only one level of deduction.

A simple reflex agent works by finding a rule whose condition matches the current situation (as defined by the percept) and then doing the action associated with that rule.

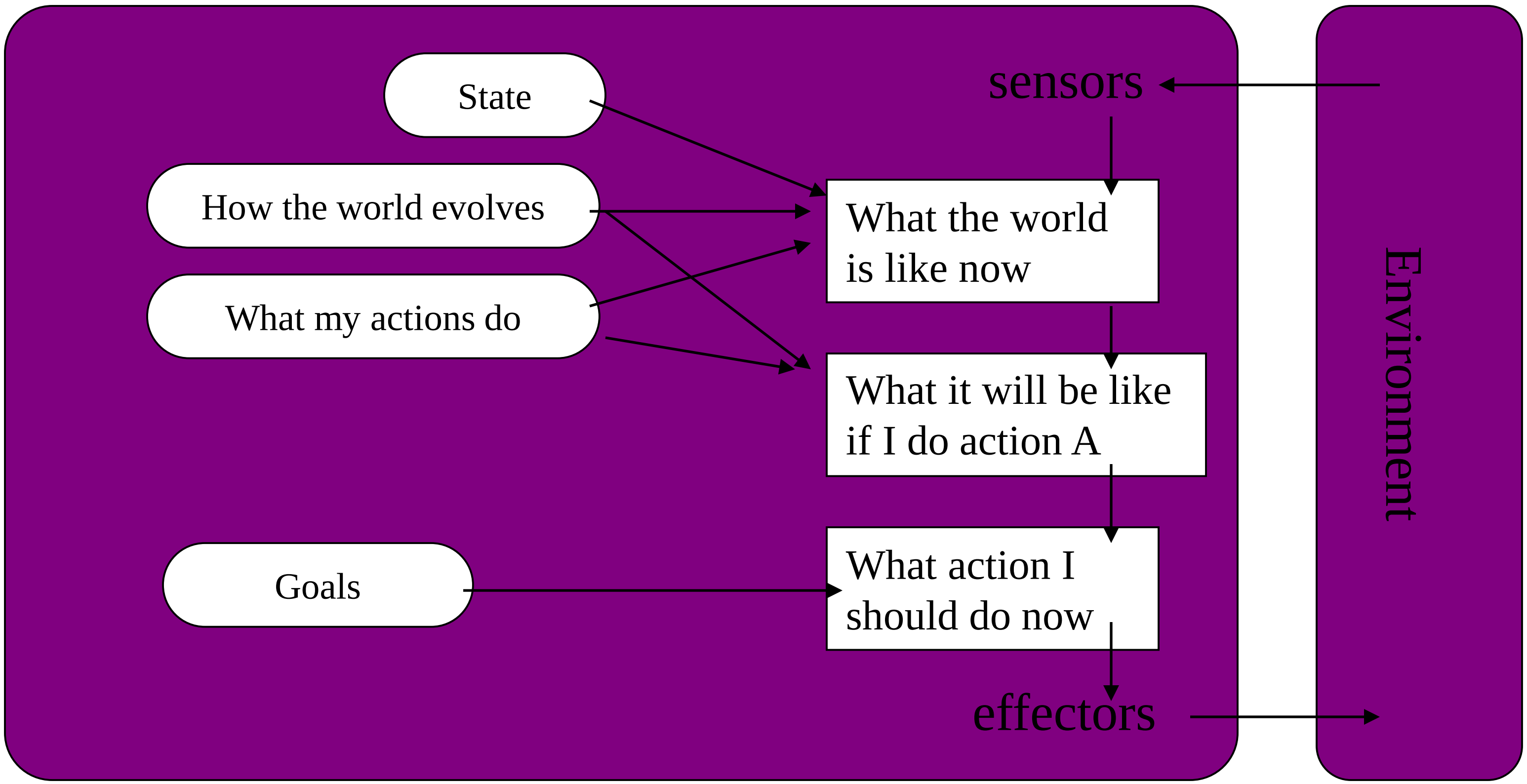
# 3. Reflex agent with internal state



# Reflex agent with internal state …

A reflex agent with internal state works by finding a rule whose condition matches the current situation (as defined by the percept and the stored internal state) and then doing the action associated with that rule.

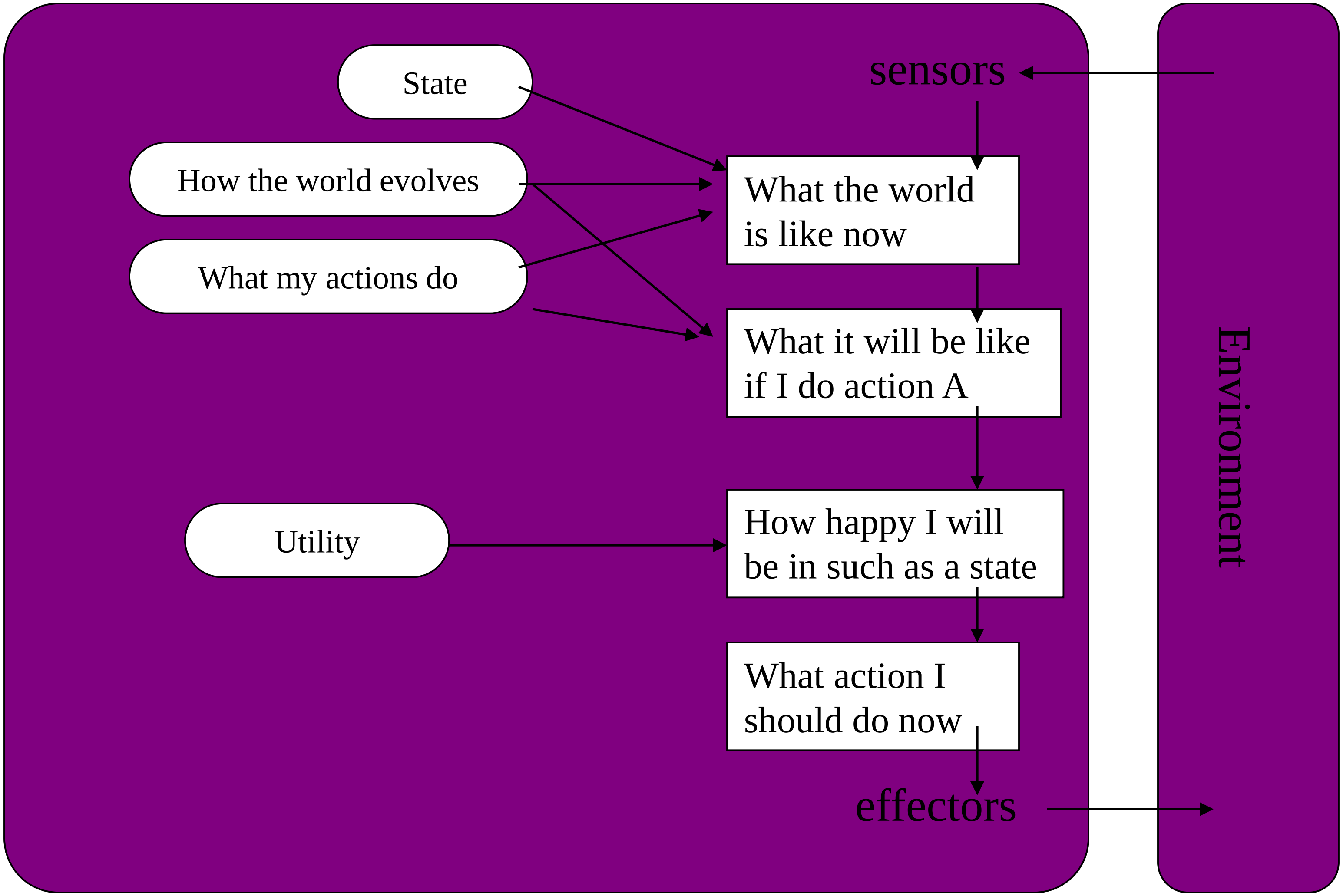
# 4. Agent with explicit goals



# Agent with explicit goals …

- Choose actions so as to achieve a (given or computed) goal = a description of desirable situations. e.g. where the taxi wants to go
- Keeping track of the current state is often not enough – need to add goals to decide which situations are good
- May have to consider long sequences of possible actions before deciding if goal is achieved – involves considerations of the future, "what will happen if I do…?" (search and planning)

# 5. Utility-based agent



# Utility-based agent …

- When there are multiple possible alternatives, how to decide which one is best?
- A goal specifies a crude destination between a happy and unhappy state, but often need a more general performance measure that describes “degree of happiness”
- Utility function U:  indicating a measure of success or happiness when at a given state
- Allows decisions comparing choice between conflicting goals, and choice between likelihood of success and importance of goal (if achievement is uncertain)

# End