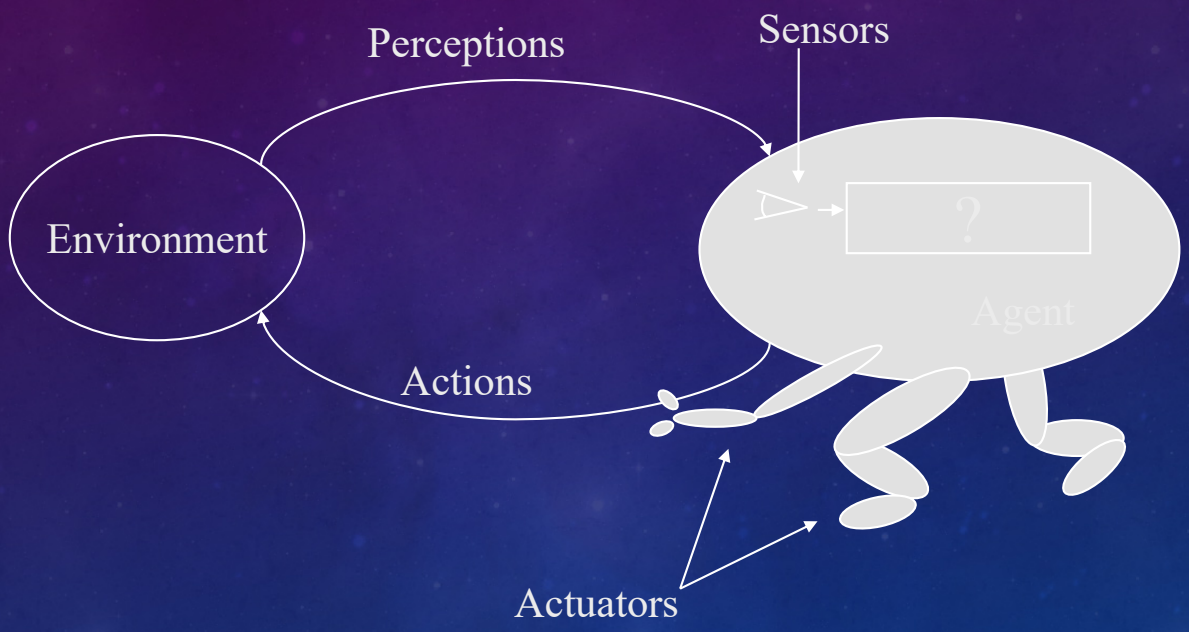

AGENTS AND
ENVIRONMENTS

CONTENTS

- What is an Agent?
- Agent description?
- Environments
- Agent structures

AGENT DEFINITION



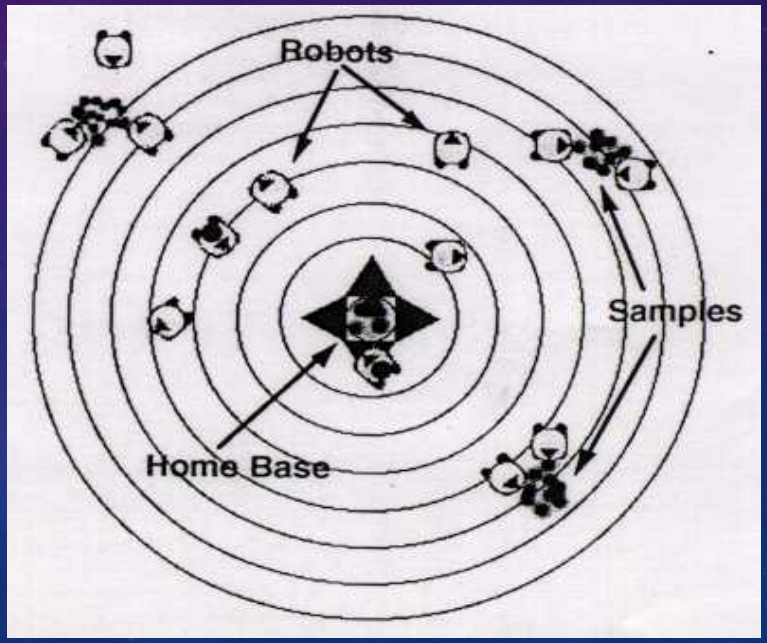
AGENT BEHAVIOR

- Agent mathematical function:
  - **Abstract Mathematical description:** We say that an agent's behavior is described by the agent mathematical function that maps any given percept to an action.
  - **Input: percepts**
  - **Output**: **actions**
- Agent program:
  - **Implementation:** Internally, the agent function for an artificial agent will be implemented by an **agent program (or a function or method).**
  - **Input: percepts**
  - **Output**: **actions**

RATIONAL AGENT DEFINITION

*"For each posible percept, a rational agent should select an action that is expected to **maximize** its performance measure, given the evidence provided by the precept sequence and whaever built-in knowledge the agent has"*

THE VACUUM WORLD



VACUUM ROBOT EXAMPLE

- **Performance**: 10 points for each sample of garbage collected.
- **Previous knowledge:** The environment is not known, robot does not where are the samples.
- **Actions**: Go forward, turn to your left or right, pick up sample, drop sample, do nothing.
- **Perceptions**: The robot can perceive its location from the base charger, it also can perceive the location of the samples.

PEAS ESPECIFICATION

- PEAS:
  - Performance
  - Environment
  - Actuators
  - Sensors

PEAS EXAMPLE

Type	Performance Measure	Environment	Actuators	Sensors
Taxi Driver	Safe, fast, legal, comfortable trip, maximize profits	Roads, other traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors keyboard

PROPERTIES OF THE ENVIRONMENT (ENVIRONMENT CHARACTERIZATION)

FULLY VS. PARTIAL OBSERVABLE

- **Fully observable:** if an agent's sensors give it Access to the complete state of the environment at each point in time!

- Fully observable example: Games like chess.
- Partially observable example: Autonomous car.

ESTOCHASTIC VS DETERMINISTIC

- If the next state of the environment is completely determined by the current state and the action executed by the agent, then we say that it is a **deterministic environment.**
- Environments in games such as chess are deterministic.
- Autonomous car environment is stochastic.

EPISODIC VS. SEQUETIAL

- An episode is a pair perception-action.
- In an **episodic** environment, the next episode does not depend on the actions taken in previous episodes.
- Classic example of episodic environment: Assembly line robot.
- Sequential environment: Games.

DYNAMIC VS STATIC

- If the environment can change while the agent is deliberating, then we say the environment is **dynamic**.
- If the environment does not chage with the passage of time but the agent's performance score does, the we say the environment is **semi-dynamic**.
- Dynamic environment example: Autonomous car.
- Static example: Chess

DISCRETE VS CONTINUOS

- For example, chess is a discrete envifonment because it has a finite number of different states.
- Chess also has a discrete set of percepts and actions.
- An autonomos car has a continuos-state and continuos-time problem.

SINGLE VS MULTIPLE AGENTS

- It seems to be simple.
- For example, an agent solving a crossword puzzle by itself is clearly a single-agent environment.
- Whereas an agent playing chess is a two-agent environment.
- To be a multiagent environment, it mus be a sort of explicit interaction by colaboration or cooperation.

SOME OTHER EXAMPLES

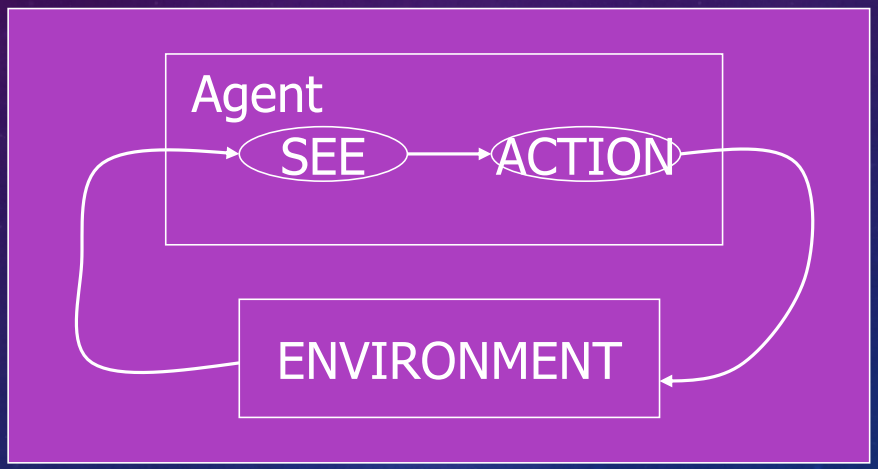
Environment	Parcially?	Sthocastic?	Episodic?	Dynamic?	Continuos?	Mutiple?
Crossword	Fully	Deterministic	Sequential	Static	Discret	Single
Chess with clock	Fully	Deterministic	Sequential	Semi	Discret	Multi
Poker	Parcially	Stochastic	Sequential	Static	Discret	Multi
Bagkgammon	Fully	Stochastic	Sequential	Static	Discret	Multi
Taxi driver	Parcially	Stochastic	Sequential	Dynamic	Cont.	Multi

AGENT ARCHITECTURES

IMPLEMENTION OF AGENTS

- **Function**: mapping from percepts to actions
- **Program**: implementation of this mapping.
- **Architecture**:
  - Agent = Architecture + Program

REFLEX AGENT



$see : S \rightarrow P$

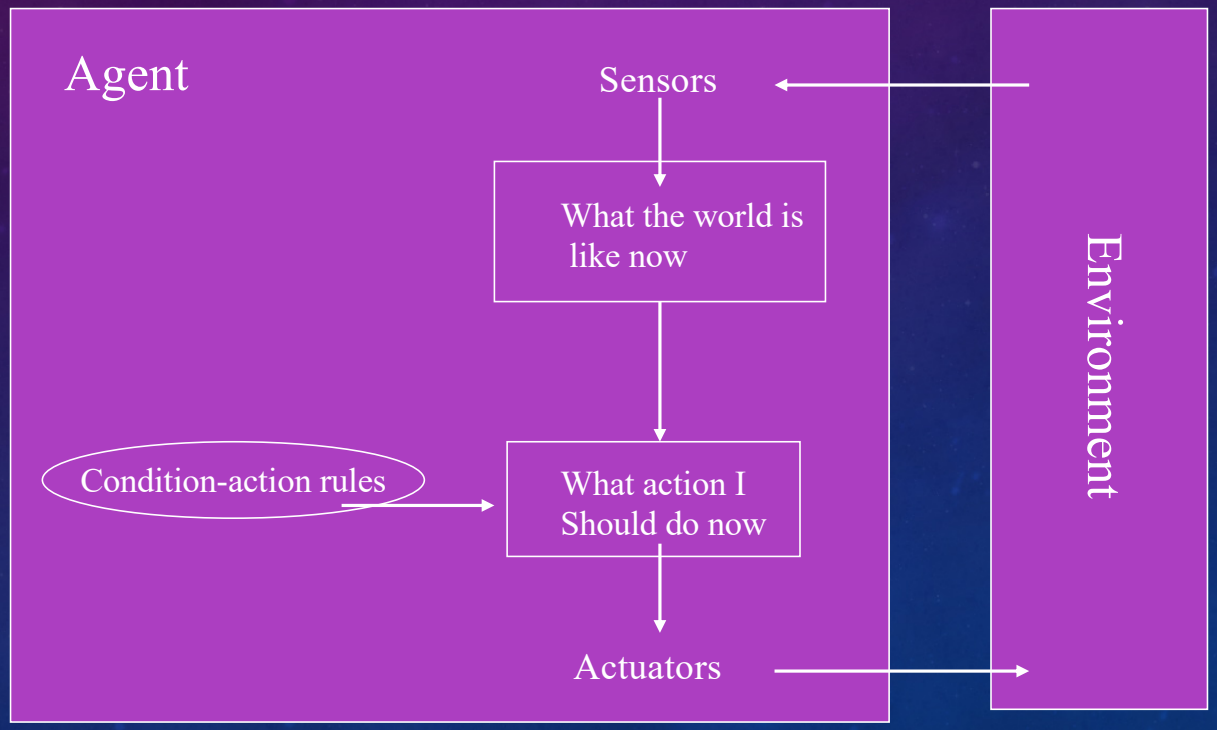
$action : P^* \rightarrow A$

P = Percepts

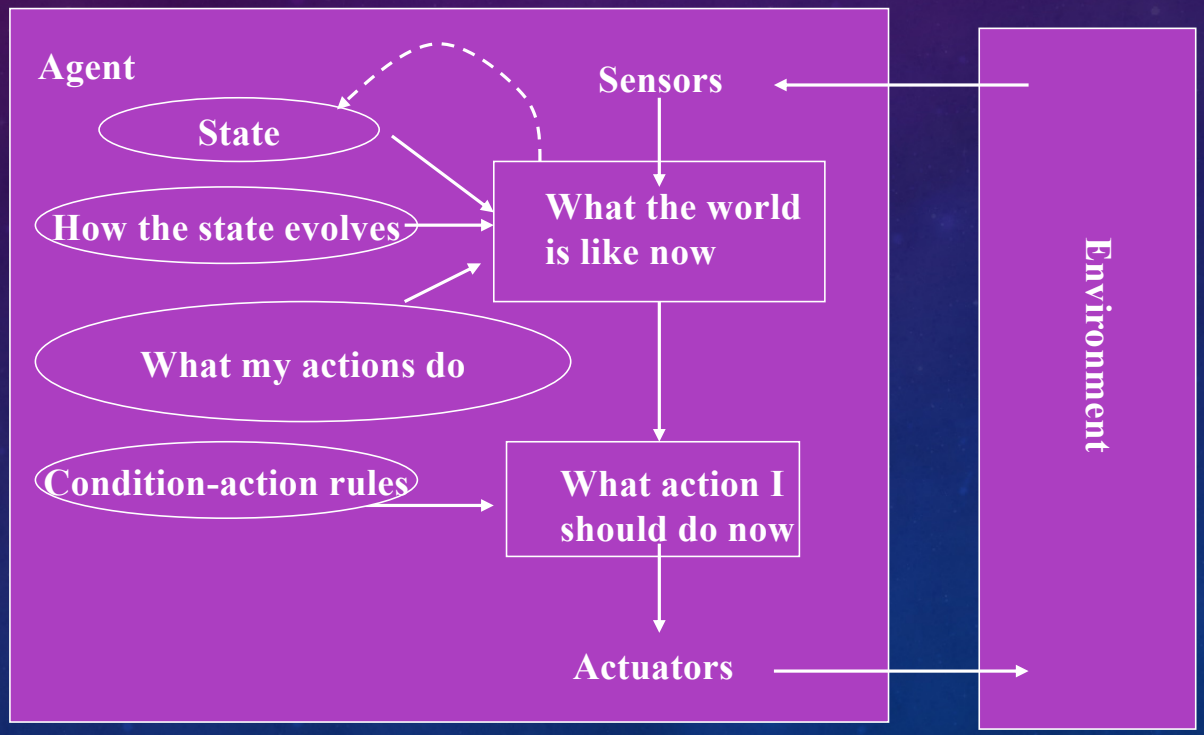
BASIC ARCHITECTURES

- Simple reflex agent
- Model-based reflex agent
- Goal-based agent
- Utility-based agent

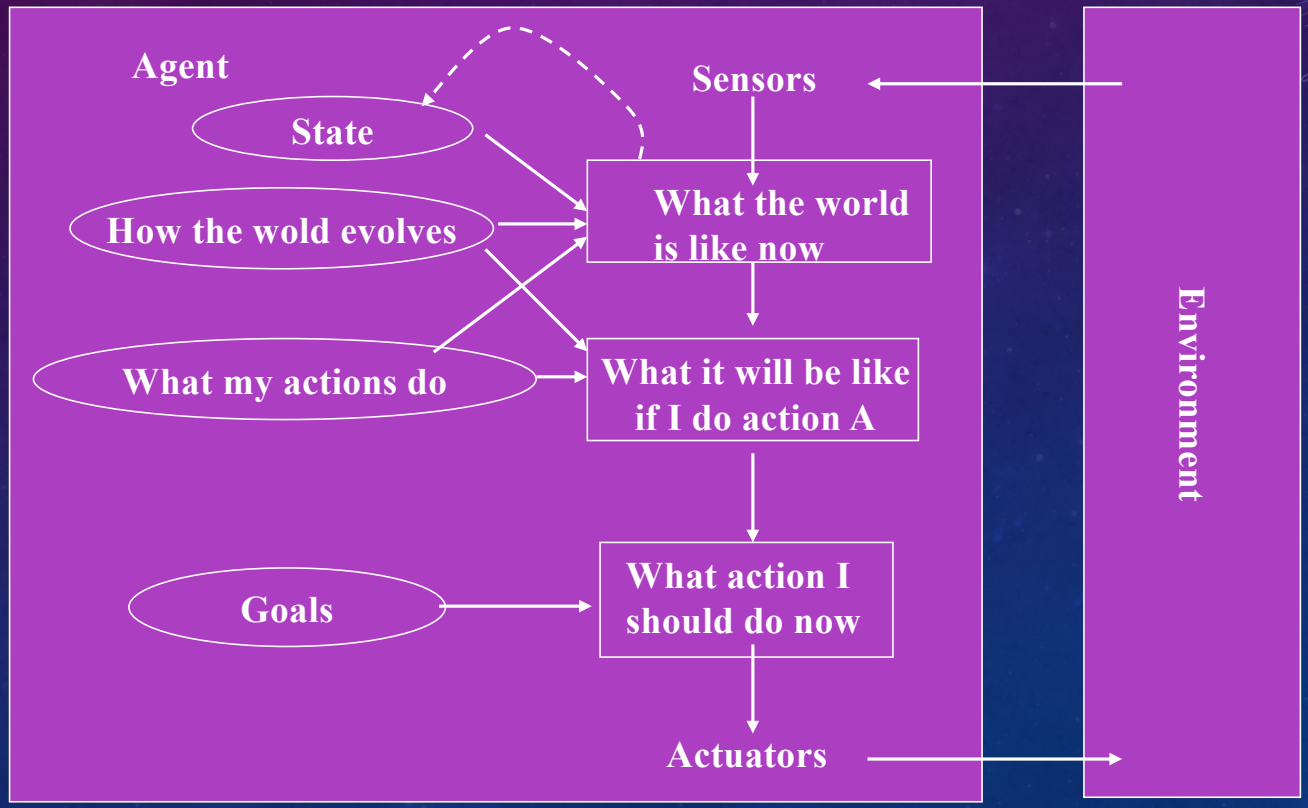
SIMPLE REFLEX AGENT



MODEL-BASED REFLEX AGENT



GOAL-BASED AGENT



SOME OTHER EXAMPLES