Agents

COMP2002

Lauren Ansell

Introduction

Today's topics:

- Intelligent agents
- Multi-agent systems

Session learning outcomes - by the end of today's lecture you will be able to:

- Describe the architecture of an intelligent agent
- Describe the tasks undertaken by an intelligent agent
- Understand how a multi-agent system is used

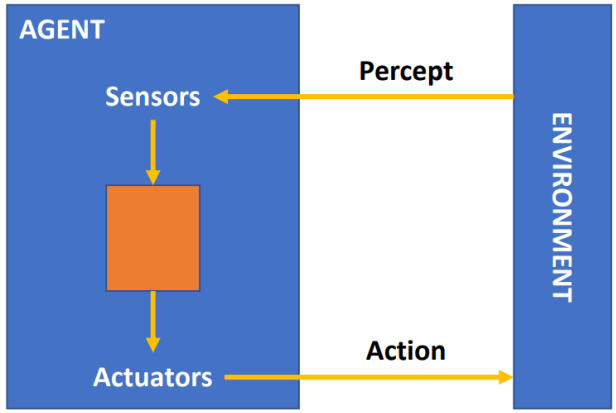
Agents

An agent perceives its environment through sensors and acts upon its environment using actuators.

A percept refers to the agent's perceptual inputs at any given instant.

Its precept sequence is the complete history of everything the agent has perceived.

An agent's action at a given time can depend on its precept sequence but not on anything it hasn't previously perceived.



Marine Autonomous Vessels

RADAR

LiDAR

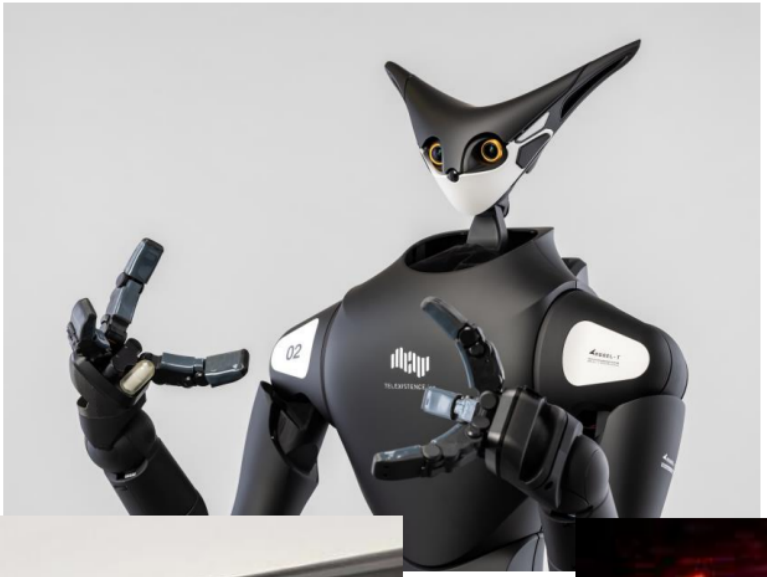
Echo sounder

CCTV/IR/multispectral cameras

Microphones

AIS

Examples Of Agents



Environments

The space in
which the agent will undertake its work.

Could
be a room, a network, software. . .
The more restrictive
the environment the easier the
agent's task – but the less realistic it is.

Environment characteristics include:

- Observability
- Determinism
- Dynamism
- Episodic vs sequential

Agent Programs

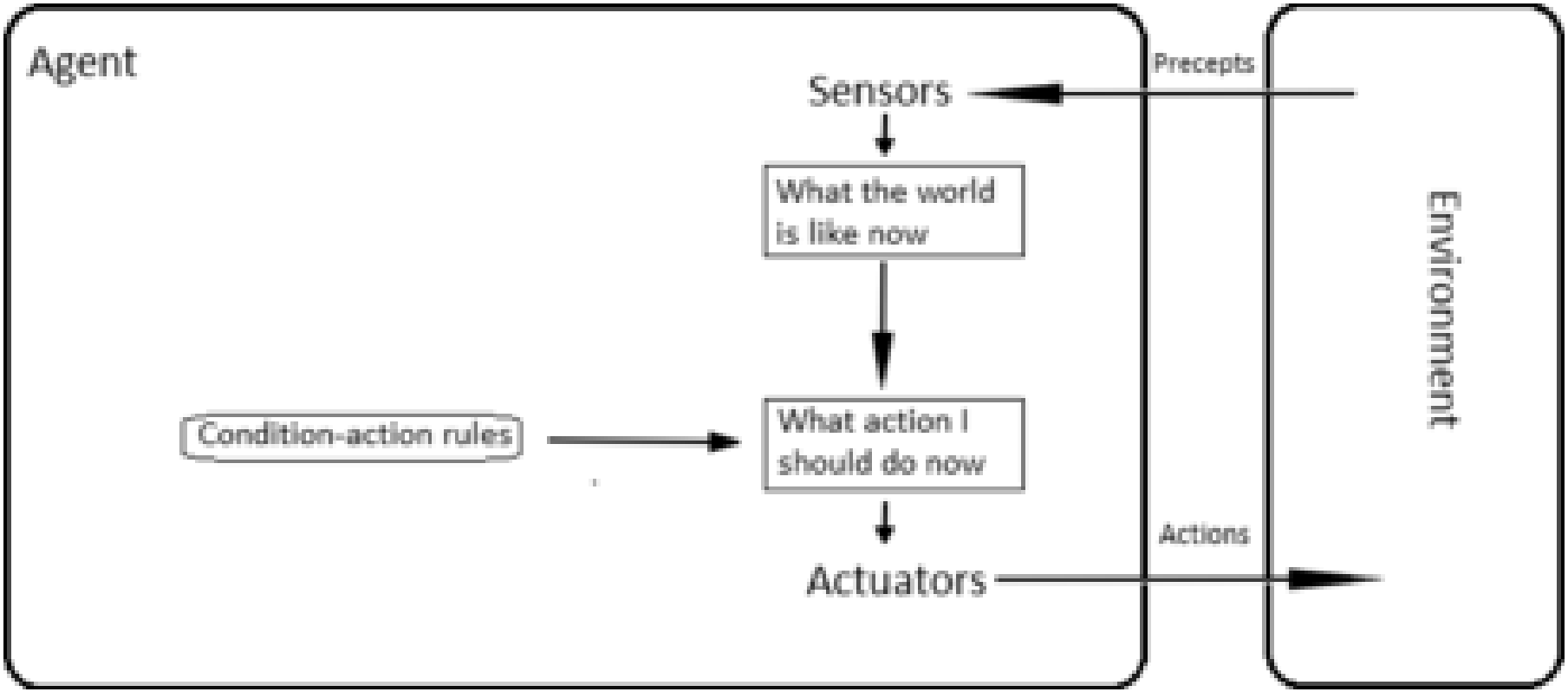
Simple reflex agents - Agents select an action on the basis of the current precept – ignoring the rest of the precept history.

Model-based reflex agents - Agents keep internal state to track aspects of the world not evident in current precept.

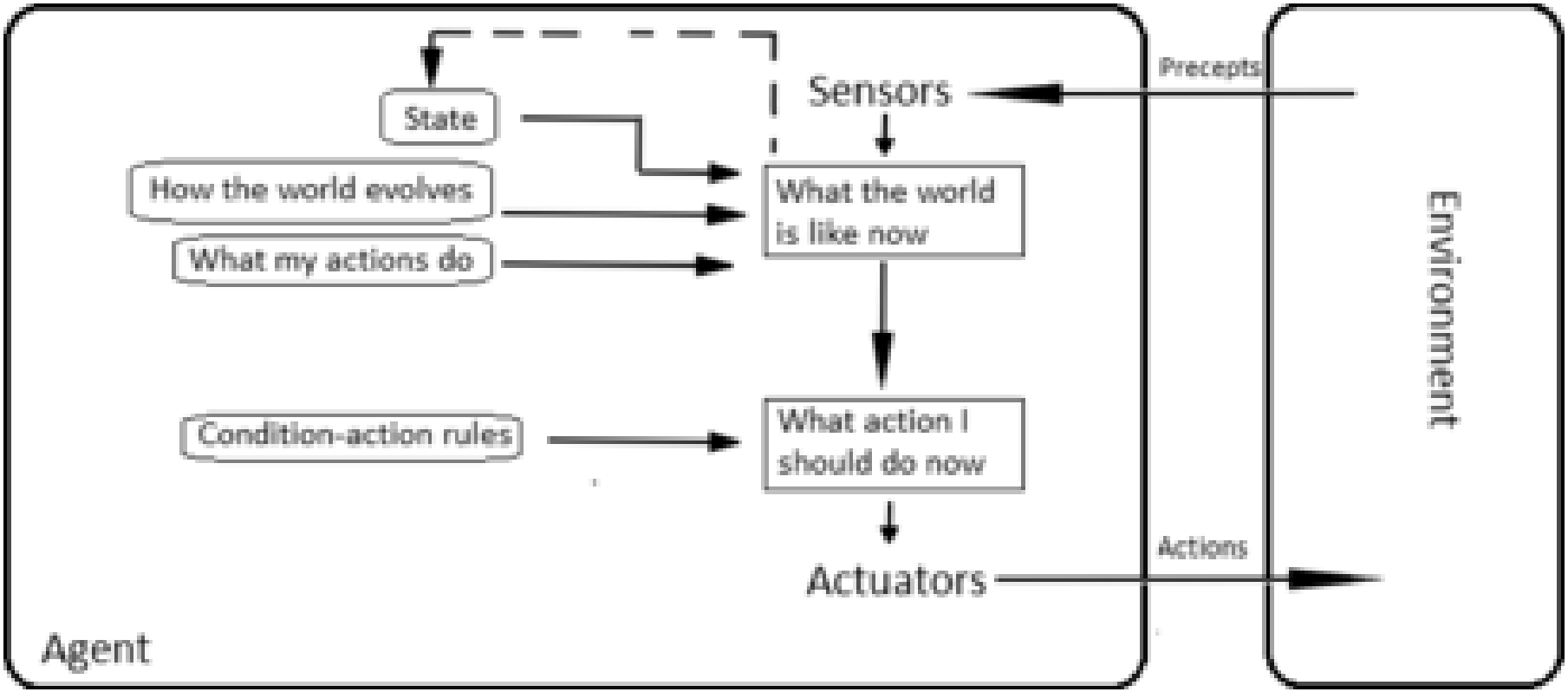
Goal-based agents - Agents consider the current precepts and precept history along with the ultimate goal.

Utility-based agents - Agents consider the utility of following a particular strategy to achieve their goal.

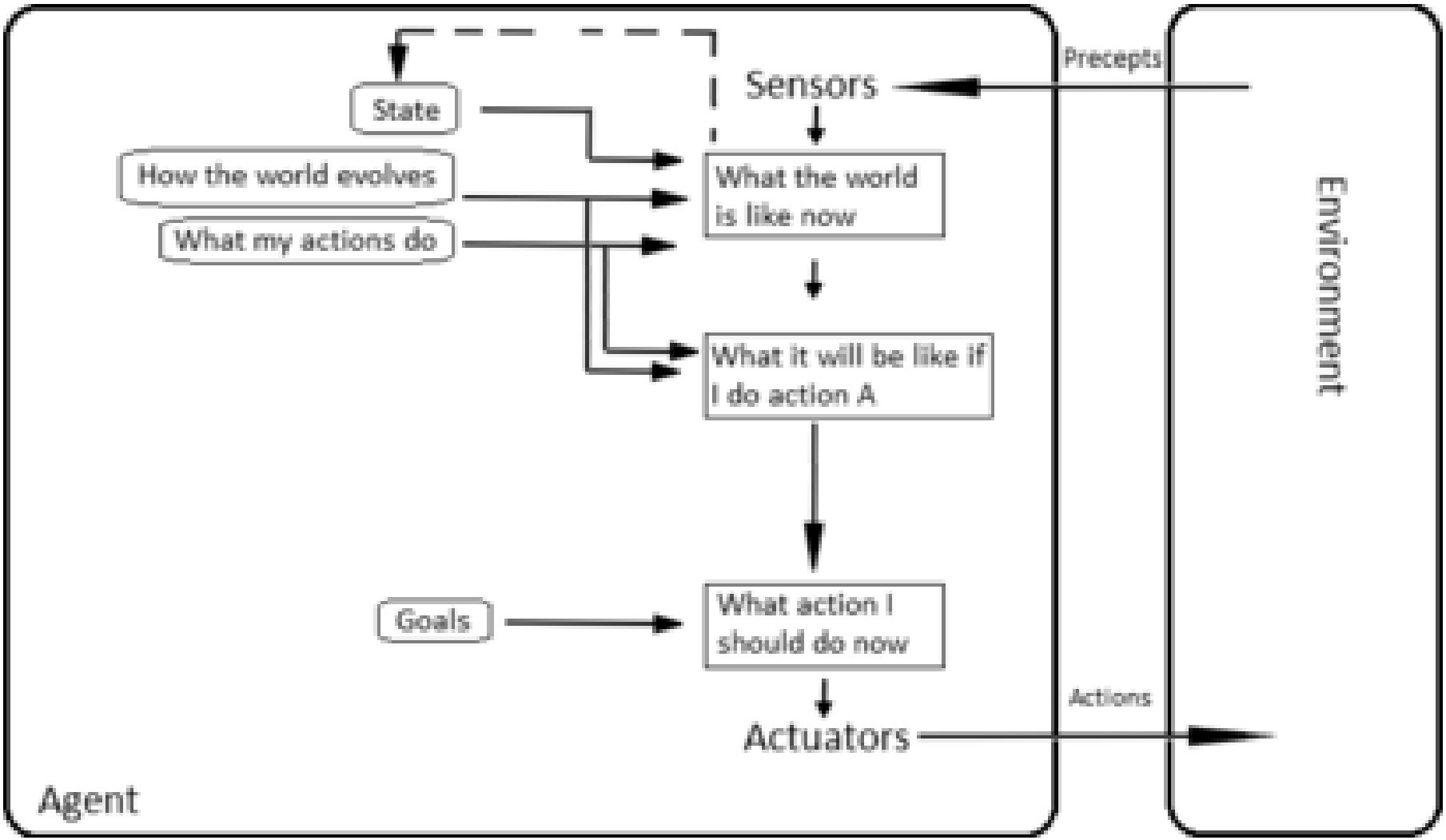
Simple Reflex Agents



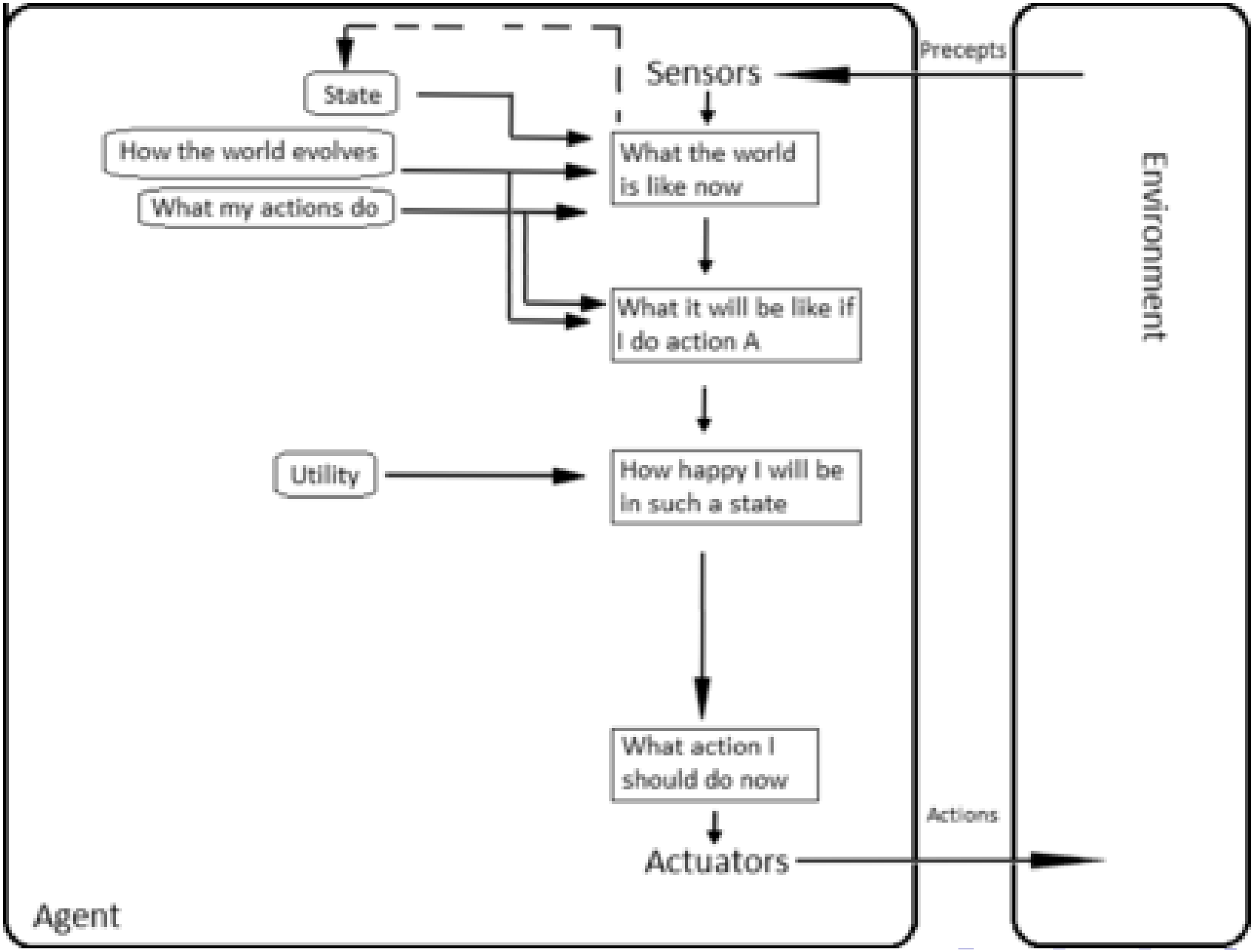
Model-based Reflex Agents



Goal-based Agents



Utility-based Agents



Learning Agents

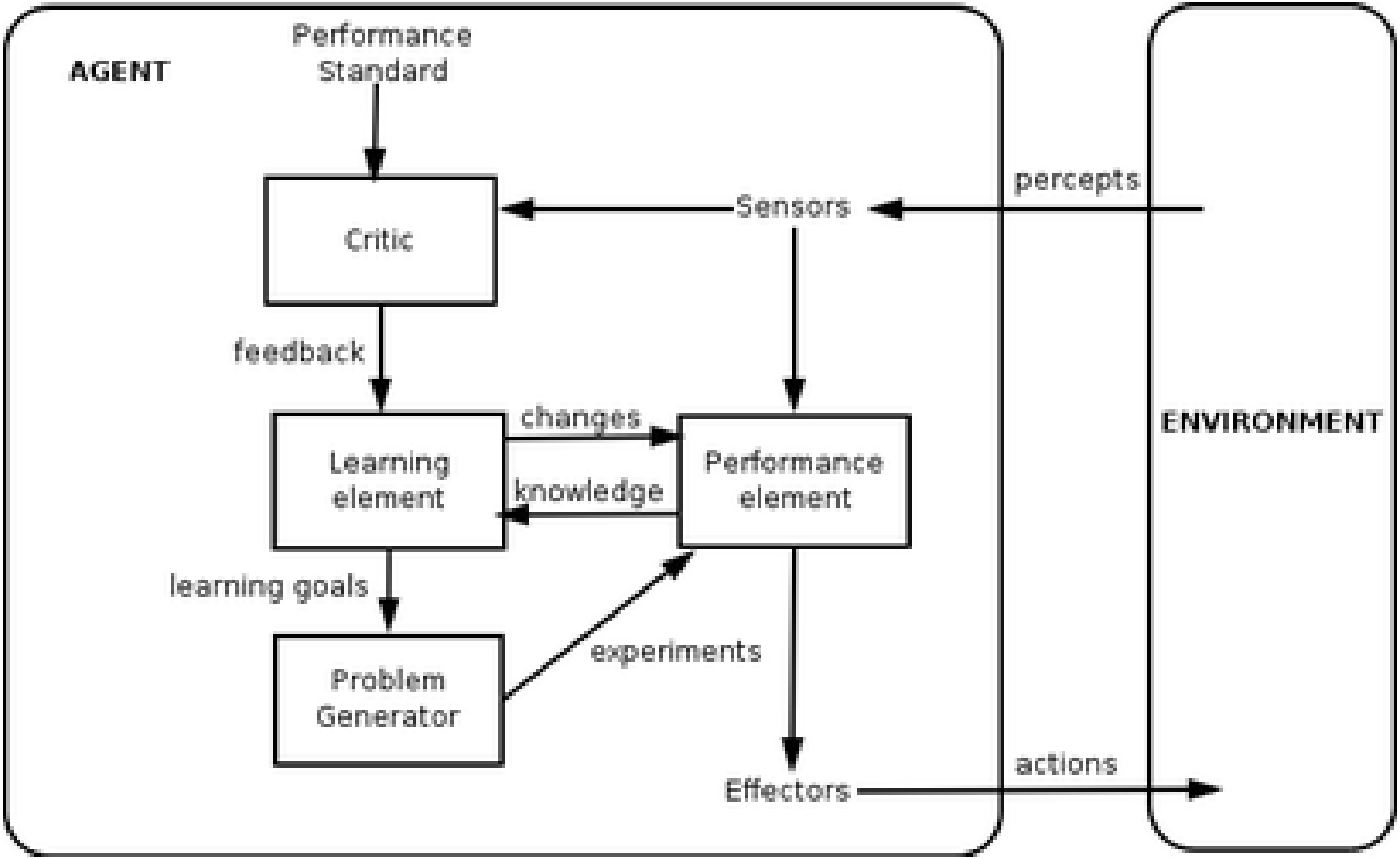
How do agents come into being?

Turing (1950) proposed building intelligent agents by hand but concludes “some more expeditious method seems desirable”.

Build learning machines and teach them.

Four components of a learning element

- Learning element – makes improvements to an agent
- Performance element – selects external actions
- Critic – provides feedback on how the agent is doing
- Problem generator – suggests actions that will lead to new informative experiences


AGENT
Performance Standard
Critic
Sensors
percepts
feedback
changes
Learning element
knowledge
Performance element
ENVIRONMENT
learning goals
Problem Generator
experiments
Effectors
actions

Weiss Classification

Weiss defines four classes of agents:

- Logic-based agents
- Reactive agents
- Belief-desire-intention agents
- Layered architectures

Multi-Agent Systems

Use multiple agents to solve more complex questions

Flexibility – can collaboratively solve problems and still learn independently to make autonomous decisions.

Can be competitive or co-operative.

Scalable – agents only communicate with direct neighbours reducing comms overhead.

Many software environments for training them

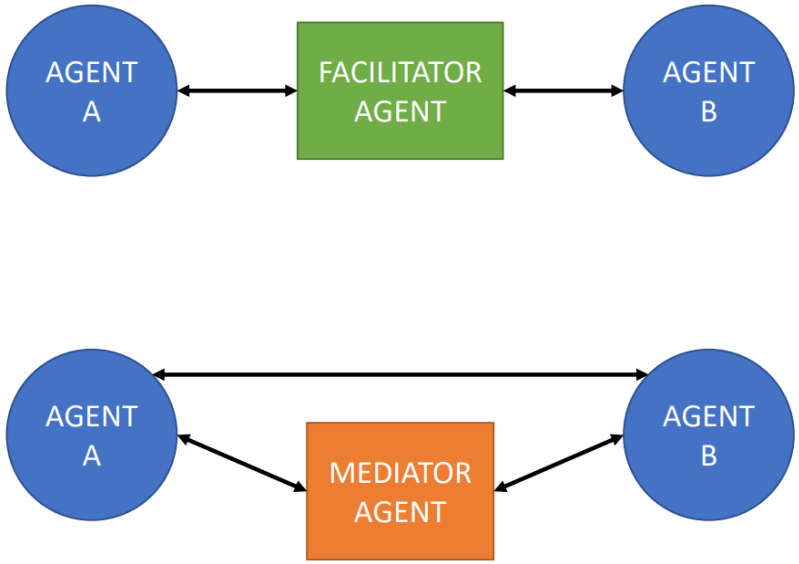
Wide range of applications – autonomous cars, multi-robot factories, automated trading, commercial games. . .

Bring with them many complexities that aren't present with single agent systems

Middle Agents

Facilitators

Facilitator
agent acts as an intermediary
between requester and responder.
Bottleneck – so have
multiple collaborative facilitators

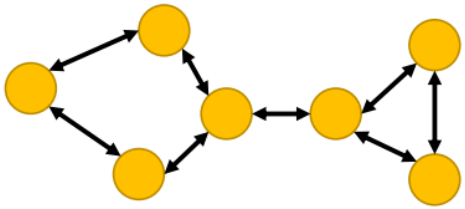


Mediators

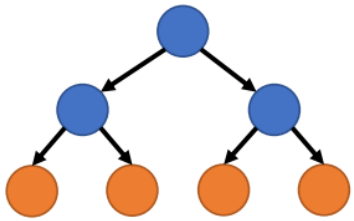
Mediator sets up
a connection between two agents
who then communicate directly

Agent Organisation

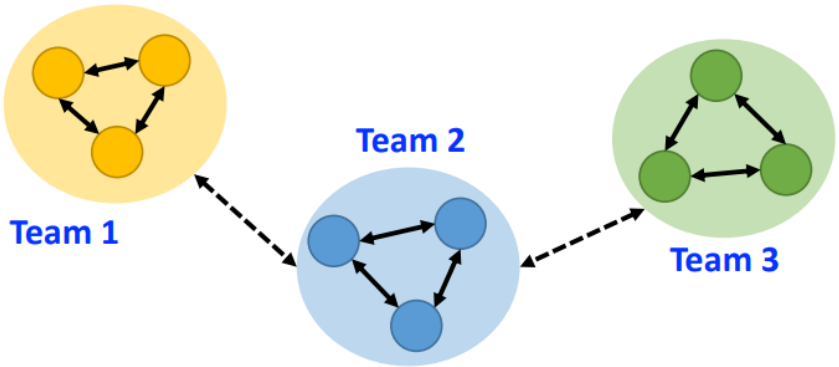
Flat



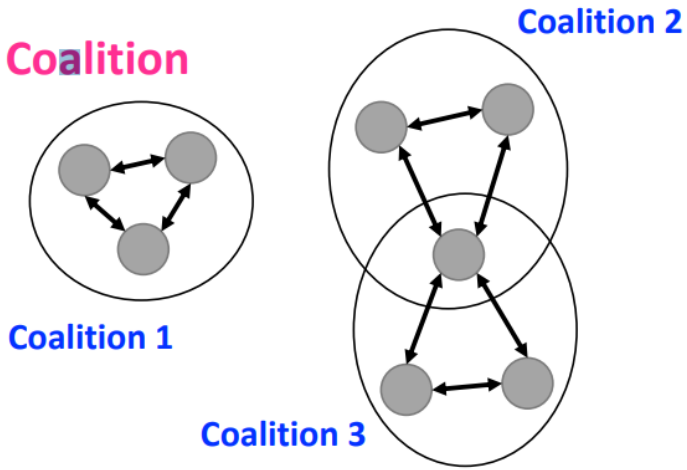
Hierarchical



Team



Coalition



Task Allocation

Agent talent – the total resources available to an agent

Agents are assigned tasks according to their available resources.

Should prevent overloading.

Agent position

The position of an agent affects communications.

Reduce overhead by considering agent position – assign a task requiring certain resources to an agent that is close to agents providing those resources.

Communications

Speech act – an agent performs an action that is perceived by other agents as an utterance that updates the environment's .

Message passing – an agent sends a direct message to each of its neighbours using an agreed protocol.

Blackboard – agents share data using a central blackboard where it can be accessed by the other agents

Fault Detection

Important to detect and isolate faulty agents in case they infect other agents

Fault detection methods are often centralized – problematic as it introduces a single point of failure.

Monitor communications between agents and classify according to expected or unexpected – then examine the agents responsible for the unexpected communications.

Focus is often on homogeneous agents – agents are often heterogeneous.

Approaches are often resource hungry.

How should a faulty agent be isolated?

Security

Security is a challenge in multiagent systems because of decentralization, sociability and mobility.

Security requirements

Authentication – ensure each agent is who they claim to be.
Authorization – ensure an agent should have access to what they're requesting.
Integrity – ensure that a message has not been altered.
Availability – ensure that the services/resources are available to authenticated and authorized agents.
Confidentiality –only those agents allowed to read specific data can do so.

Summary

Intelligent agents

- Operate within an environment using percepts and an agent program

Multiagent systems

- Use agents to solve more complex problems by working together
- Additional complexity – how should the agents collaborate?
- Must consider fault detection and security