Competency 13

Explores new trends and future directions of ICT

13.2 Explores the fundamentals and applications of agent technology

Time: 4 periods

Learning Outcomes

- Briefly describes software agents and their characteristics
- Briefly describes multi-agent systems and their characteristics
- Identifies the applications of agent systems

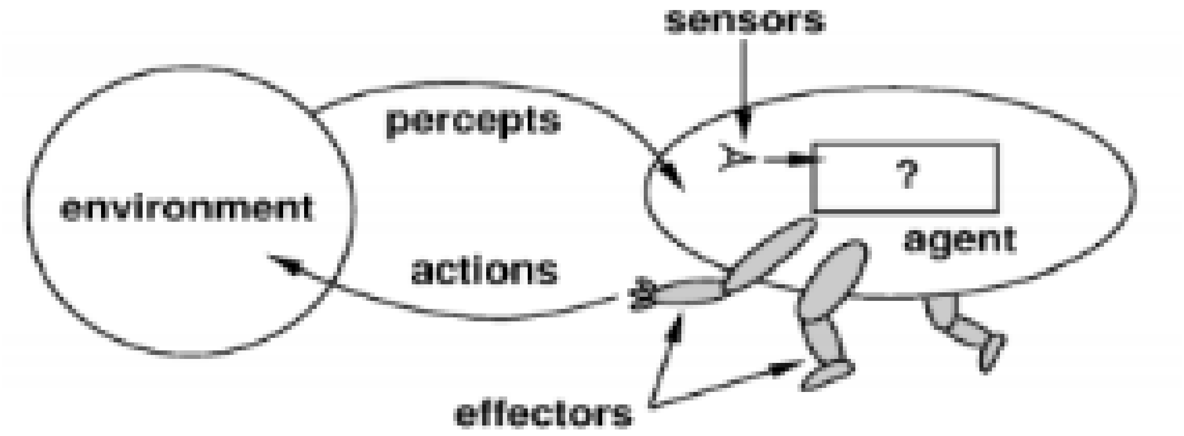
New Challenges for Computer Systems

- Traditional Design Problem
    - How can we build a system that produces the correct output given some input?
- Modern-day Design Problem
    - Modern-day, many of the systems we need to build in practice have a ’**reactive’** flavor, in the sense that they have to maintain a long-term, ongoing interaction with their environment, they do not simply compute some function of an input and then terminate.
    - The main role of reactive systems is to maintain an interaction with their environment, and therefore must be described in terms of their on-going behaviors, such as OS, process control systems, online banking systems, etc.
    - A still more complex class of systems is a subset of reactive systems that we call agent.
    - Agent is a reactive system that exhibits some degree of autonomy (independence) in the sense that we delegate (agent)some task to it, and the system itself determines how best to achieve this task

Agent-Based Systems

Most widely accepted definition: An agent is anything that can perceive its environment (through its sensors) and act upon that environment (through its effectors)

Definition from the agents/MAS area (Wooldridge & Jennings): An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives.



In computer systems, a software agent is a piece of software that works on behalf of a human user or another computer application.

Example: Cortana, in Windows

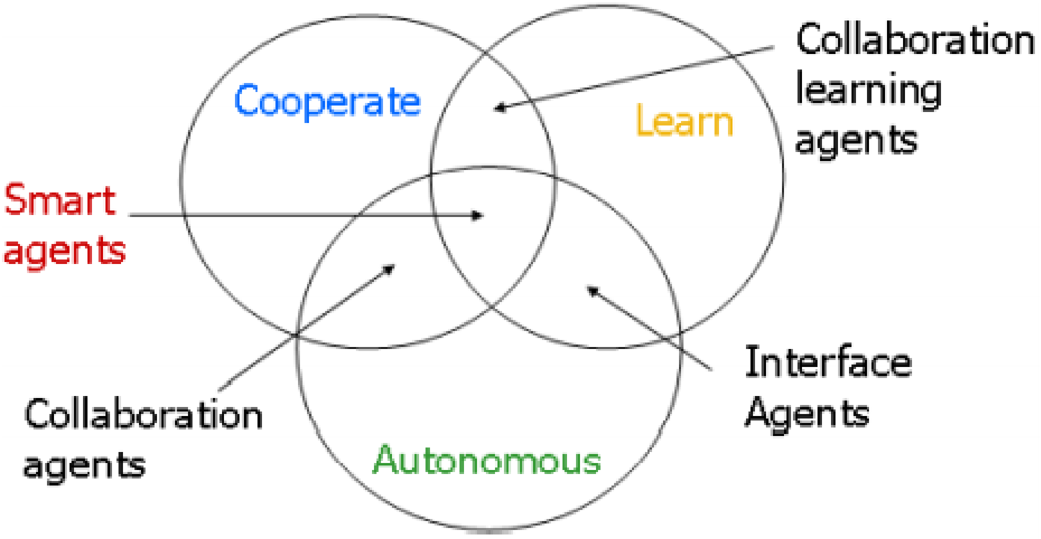
It is a piece of software inside the Windows operating system that acts as a virtual assistant. It provides personalized services to the human user such as searching for apps and files, recognizing voice commands, giving reminders, etc.

Software agents in most cases reflect (weak) intelligence by their abilities such as recognizing voices, interpreting the text in natural language, predicting based on past data, and reasoning based on observations.

Characteristics of Agents

- **Autonomous:** can take decisions by themselves without the interference of the human user
- **Proactive:** can initiate actions on its environment
- **Reactive:** can react to events in the environment
- **Cooperative:** can cooperate with other agents and humans
- **Able to learn:** can learn by observation, experience, and data
- **Adaption:** agents improve performance over time
- **Social ability:** can interact and communicate with others complying to the social rules and norms
- **Mobility:** the ability of an agent to move around an electronic network
- **Veracity:** an agent will not knowingly communicate false information

These features actually help to differentiate a software agent from an object in software. A good classification of software agents is shown below.



Multi-agent systems

Software agents can interact with other software agents to execute a common task.

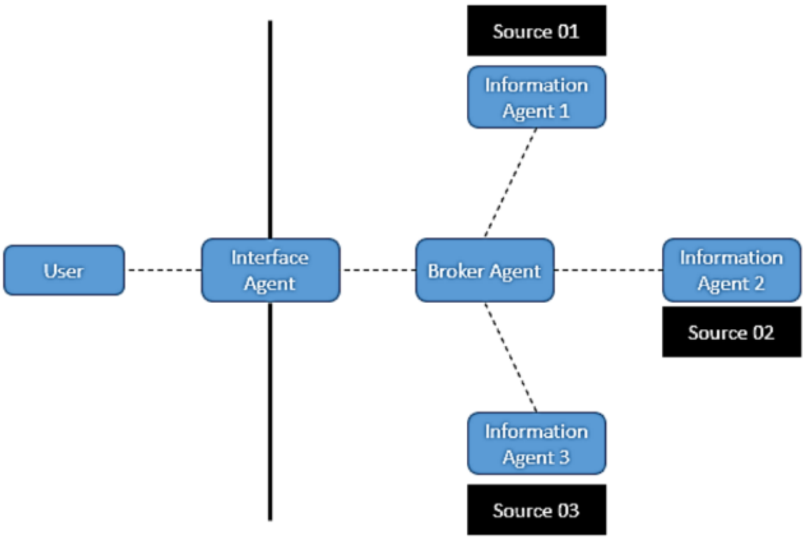
In most cases, a single software agent is not capable of solving complex problems.

Thus, the applications aiming at solving complex problems require multiple software agents working as a single unit which is called a Multi-Agent System or MAS.

A multi-agent system (MAS) is a loosely coupled network of software agents that interact to solve problems that are beyond the individual capacities or knowledge of each problem solver.

In such a multi-agent system, the agents are,

- **Autonomous**: meaning that they can make independent decisions by being self-aware
- **Having only a local view:** meaning that they only know how to execute their part and cannot see beyond their scope
- **Acting in a decentralized manner:** meaning that no agent to control the system
- A user gets some information through an information broker from multiple sources. Each source is assigned to an information agent who submits the relevant information to the broker who filters and provides the appropriate information to the user. The user accesses the system through an interface agent.



The architecture of a multi- agent system

Applications of Agent systems

Agent systems are applied in a range of applications.

Cortana in Windows and Siri on Apple iOS systems are good examples of software agents being practically implemented, such virtual assistants exist in many computer systems to make the tasks of the human users easier.

Agents also assist humans in banking, e-learning, e-commerce, booking, and various other similar systems. For example, information searching in electronic commerce applications involves a multi-agent system similar to the architecture shown above.

There, users submit their search criteria for products through their respective interface agents who then connect with information broker agents that filter information received through agents at various merchant sites and submit to the users.

Online booking systems too involve such architecture to filter information on behalf of the users, connect to the respective merchant and do the booking on behalf of the human user.

References

https://www.tutorialspoint.com/artificial_neural_network/artificial_neural_network_applications.htm

Teachers' Guide

https://portalwrite.wordpress.com/man-machine-coexistence/