

University of Msila  
FACULTY OF MATHEMATICS AND  
INFORMATICS  
DEPARTMENT OF COMPUTER  
SCIENCE

# Agent models and architectures



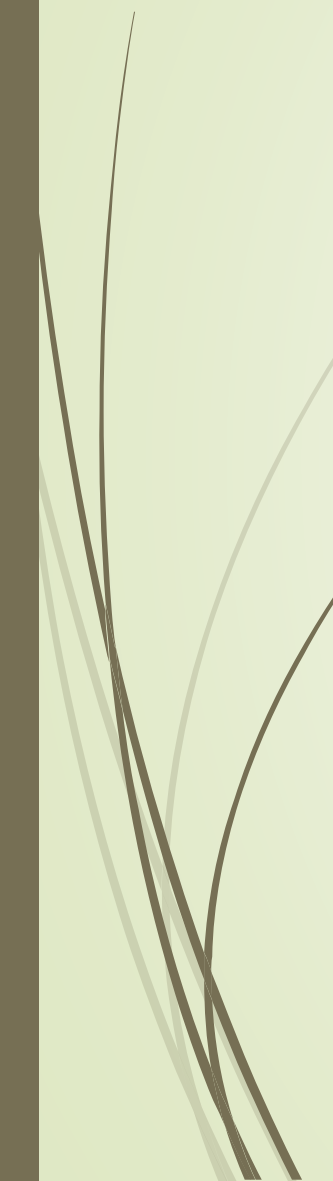


# Outline

- 1. Agent concept**
  - 2. Multi-agent systems**
  - 3. Specification of SMAs**
  - 4. MSA architecture**
  - 5. Communication in SMAs**
  - 6. MSA development environment**
- 

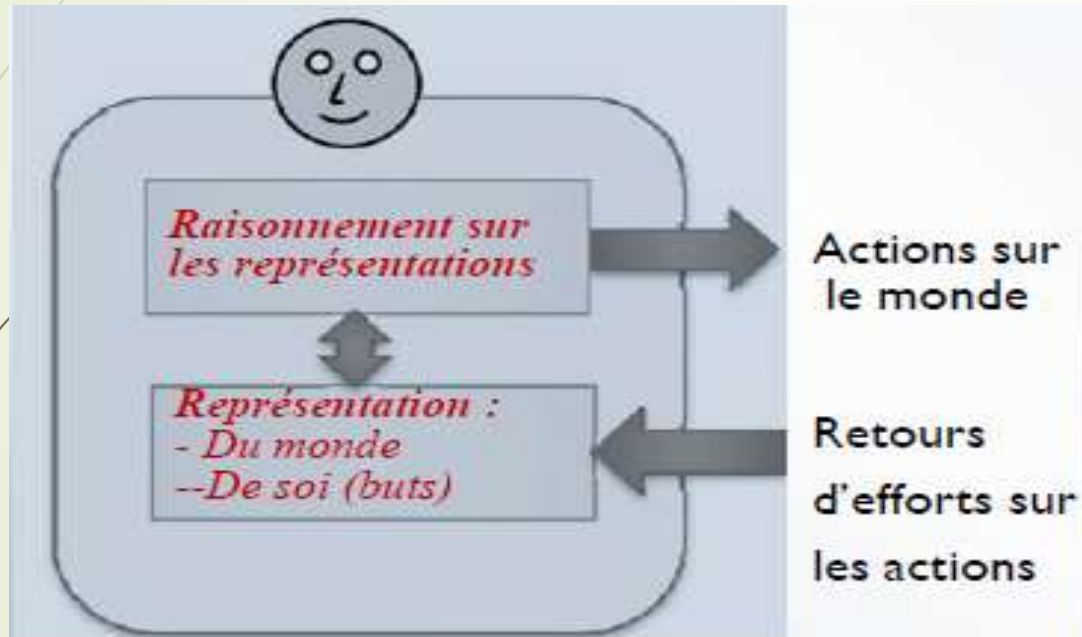


# Agent concept

- ▶ An agent: An agent is an entity interacting with other agents in a social field".
  - Population
  - Interaction
  - Intention
  - Learning
  - Reasoning
- 

# Cognitive agent

- each agent is specialized in a field and knows how to communicate with the others. They have explicit goals and plans for achieving them. Explicit representation of self, environment and other agents. Explicit organization. Explicit and elaborate interaction.



- ✓ **Représentation explicite** (Soi, Environnement, autres agents)
- ✓ **Architectures Complexes** (Ex, BDI)
- ✓ **Stimulus -> réponse**
- ✓ **Organisation explicite** (Partage ressources, protocoles,...)
- ✓ **Communication explicite**, point à point, élaborée (ex : KQML)
- ✓ Petit/moyen nombre d'agents  
Certaines **validations** formelles possibles.



# Cognitive agent

➤ **Example**

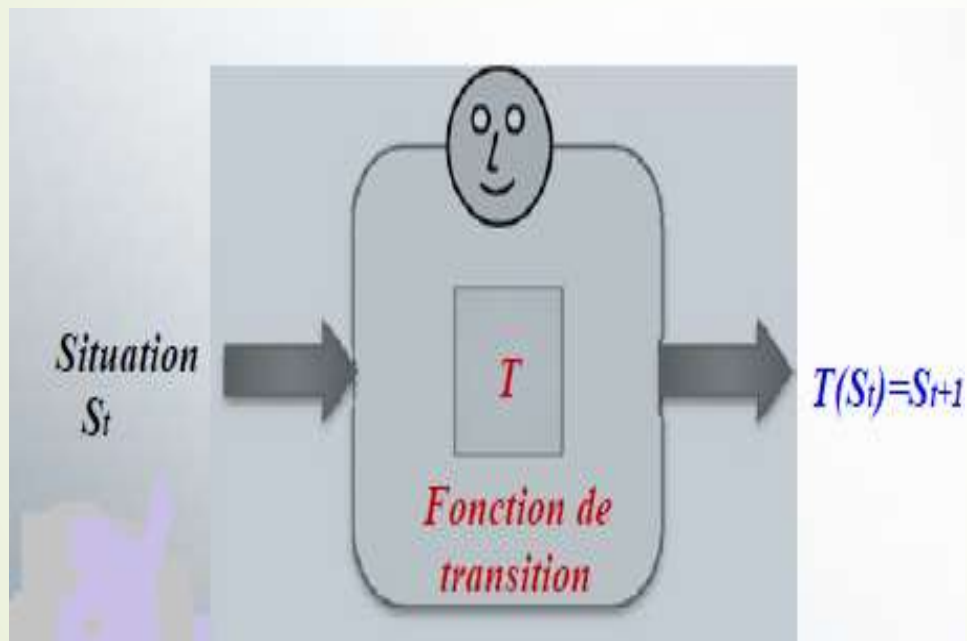
➤ ability to construct a plan such as :

- Plan open door :
- go to the location of the key
- take the key- go to the door
- open the door with the key



# Reactive agent

- agents with no intelligence (no anticipation, no planning) that react by stimulus-response to the current state of the environment.



- ✓ Architectures **simples**
- ✓ **Stimulus -> réponse**
- ✓ Auto-organisation, ex : colonie de fourmis
- ✓ **Communication** via l'environnement ex : **perception/actions** sur l'environnement, phéromones de fourmis
- ✓ Grand ou très grand nombre d'agents (**redondance, robustesse**)



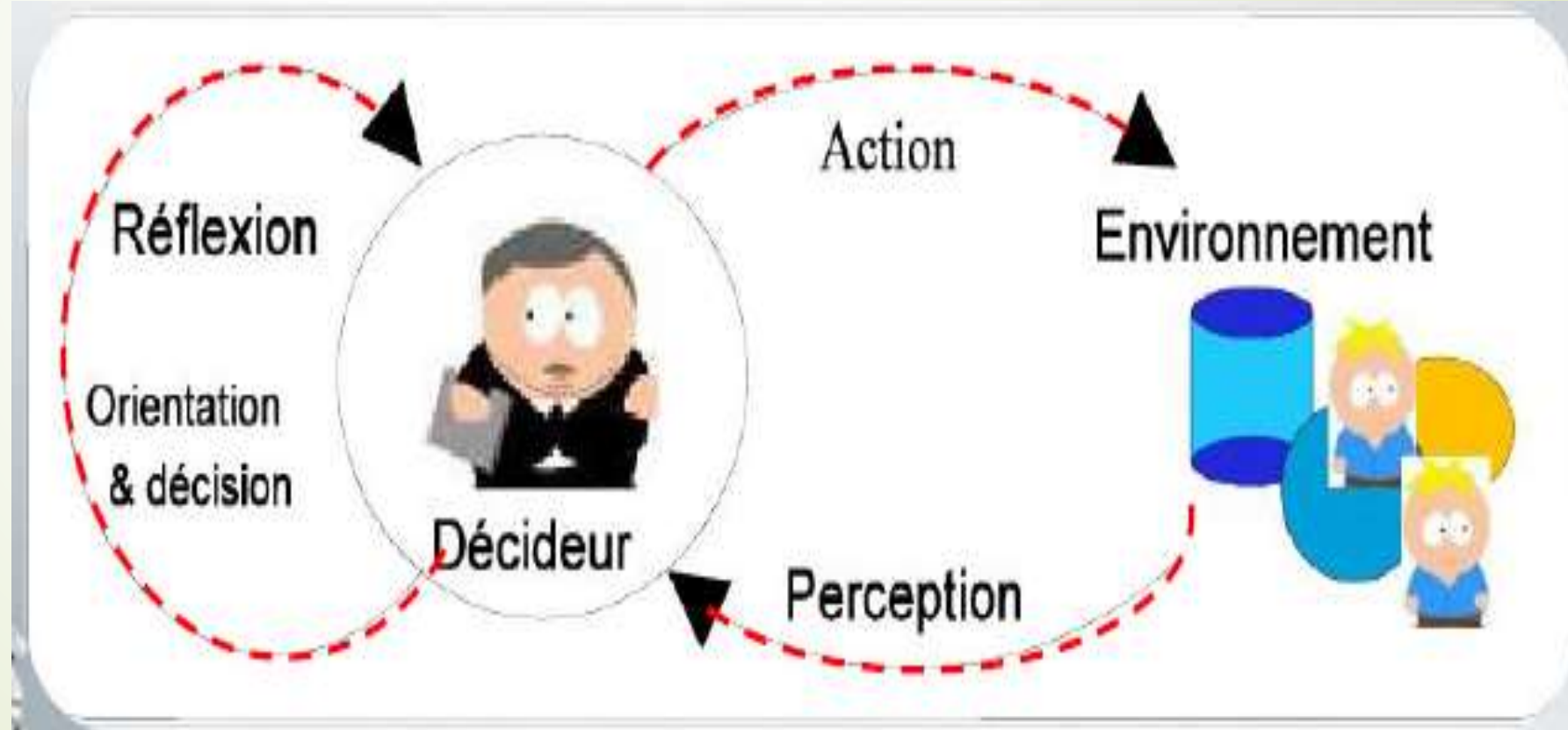
# Reactive agent

➤ **Example of reactive behavior to open a door:**

- R1: if I'm in front of the door and I have a key, then open it
- R2: if I'm in front of the door and don't have a key, then try to open it
- R3: if the door doesn't open and I don't have the key, then go and find the key
- R4: if I'm looking for a key and there's a key in front of me, then take the key and go to the door.

# Agent rationnel Délibératif

- Perception
- Analysis
- Decision
- Action





# Agent Intentionnel BDI

- Architecture BDI ➔ **Belief-Desire-Intention model of intelligent agent rationality**
- **Beliefs:** the information the agent possesses about its environment and about other agents existing in the same environment.  

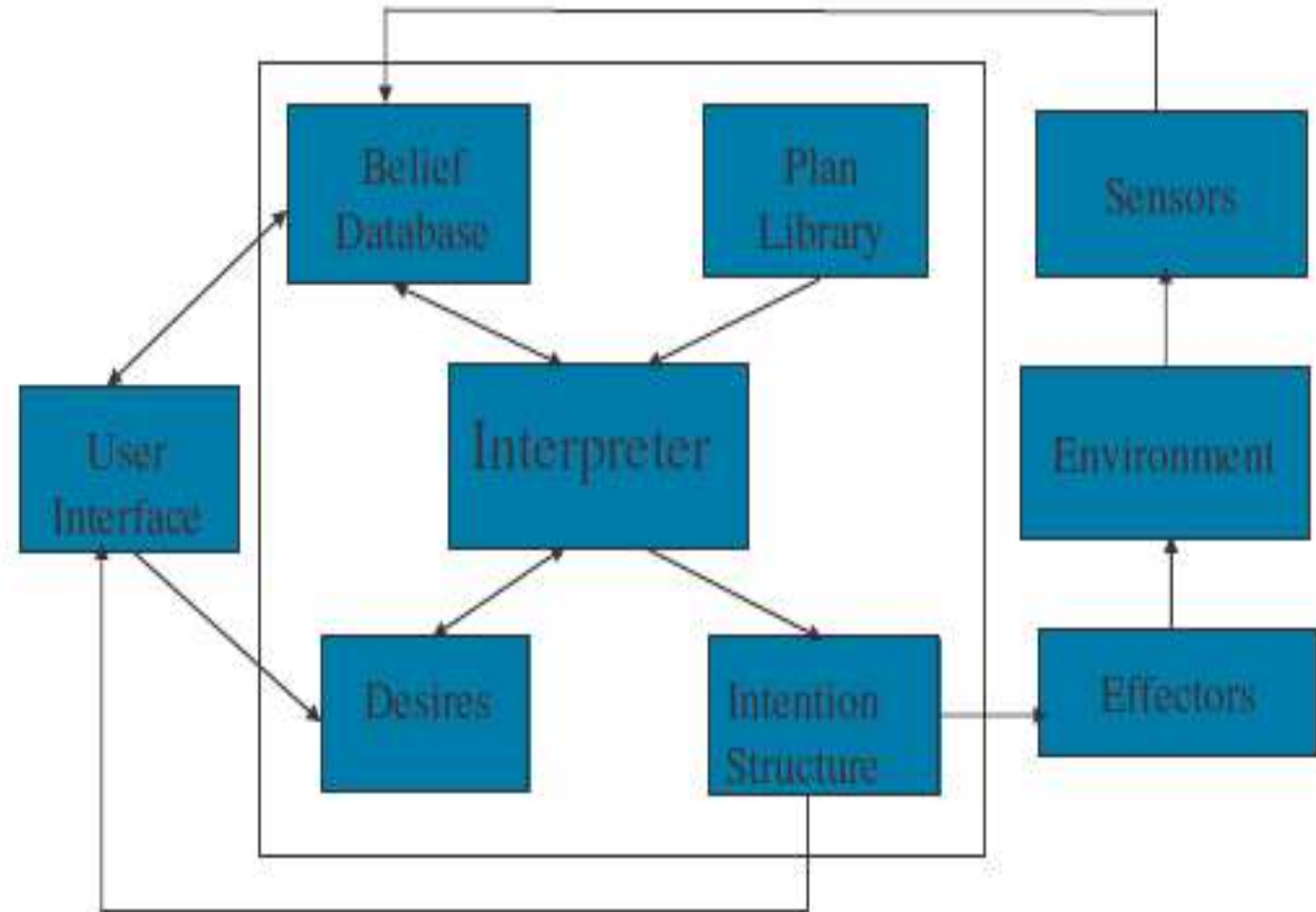
These beliefs may be incorrect, incomplete or uncertain.
- **Desires:** the states of the environment that the agent would like to achieve. An agent may have contradictory desires, in which case he chooses a consistent subset of his desires. This subset corresponds to the agent's goals.
- **Intentions:** these are the actions the agent has decided to take to achieve its goals.
- The agent decides what actions to take based on his internal states, which are expressed in the form of beliefs (Belief), desires (Desire) and intentions (Intention).

# BDI architecture

Function act (p:P): A  
Start

B:=review\_beliefs (B, p)  
D:=determine\_new\_goals  
(B,D,I)  
I:=select\_goals\_to\_attempt  
(B,D,I)  
return (action(unplan (I)))

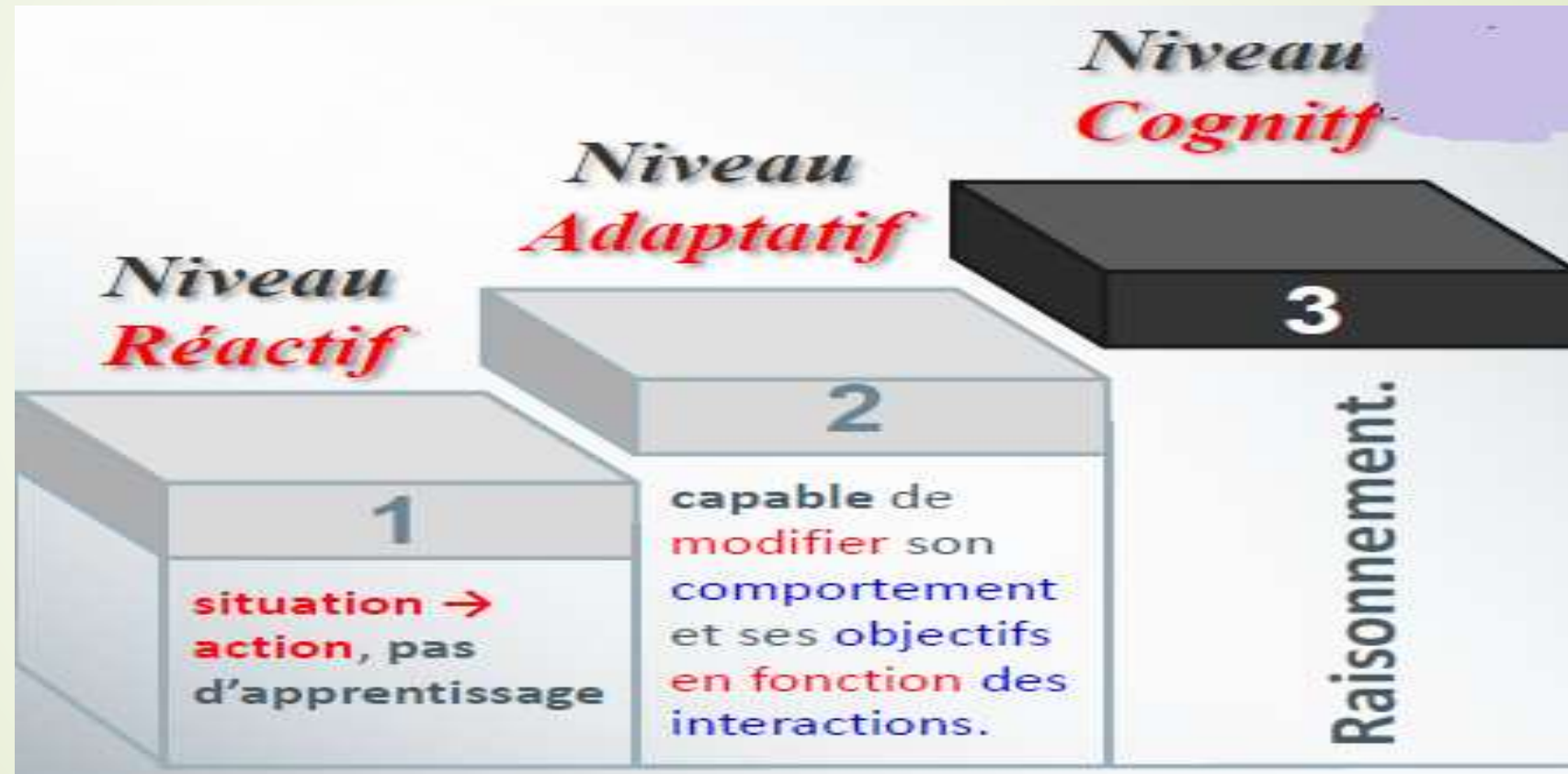
End function act.



BDI agent architecture (selon Kiny).

# Typical agent model

- ▶ An architecture implemented with three levels of description:

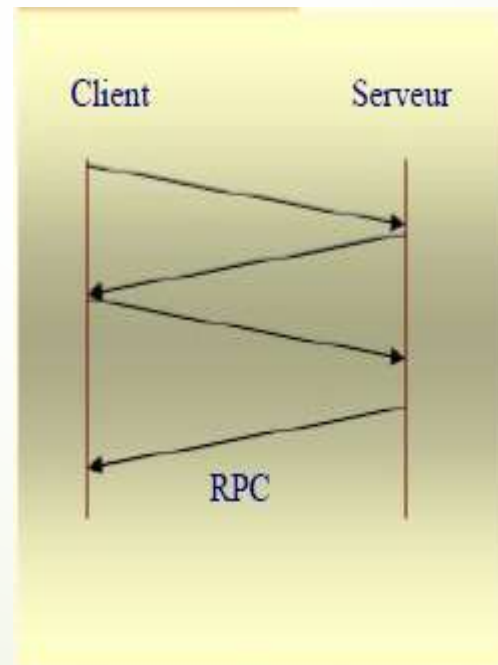


# Mobile agent

- Mobile agents are software entities that can move around the network on their own initiative, moving from machine to machine and communicating with other agents or accessing server resources.
- They make better use of resources: by transferring applications from the client to the server and executing local procedure calls instead of external ones, network traffic is reduced.
- Key attributes of a mobile agent
  - Code mobility;
  - Data mobility;
  - State mobility: ready/locked/actual execution.
  - This enables software entities to move autonomously across the network to accomplish specific tasks.
- Reduce communication costs.

# RPC vs Mobile Agents

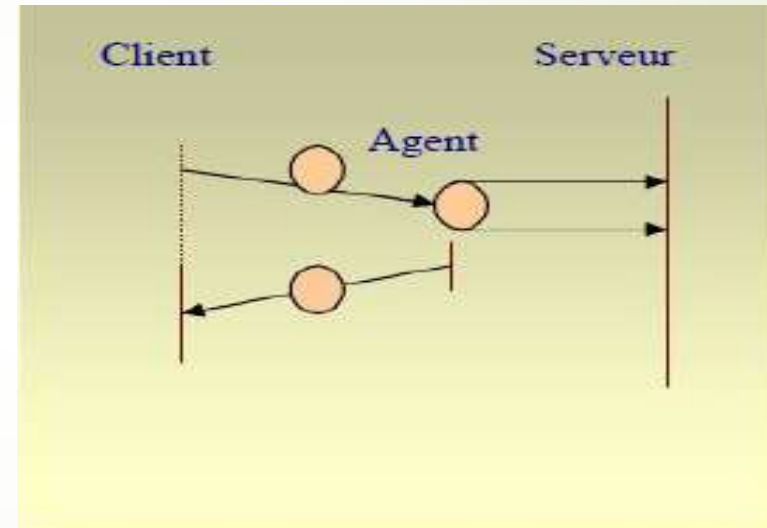
- ▶ traditional distributed computing is generally based on the client/server model, which is frequently linked to the RPC (Remote Procedure Call) mechanism.
- ▶ When an RPC is called, parameters are sent to the server to execute the procedure, and the result is sent back to the client. Each RPC requires the establishment of a communication channel over the network.





# RPC vs Mobile Agents

- With mobile agents, code mobility allows part of the client program to move to the server, act locally and return to the client once the interactions have been executed.
- Execution improvements depend on network bandwidth and the number of messages that can be optimized in transit.

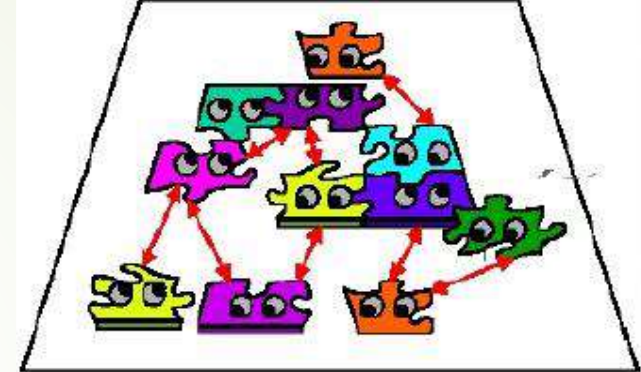


## ➤ Mobile agent applications

- ✓ *E-commerce ;*
- ✓ *Database searching ;*
- ✓ *Network management;*
- ✓ *Mobile computing.*

# Multi-agent architectures

- Reactive agent-based MAS
- Emergent collective behavior



- SMA based on cognitive agents (also known as deliberative agents)
- emphasis on action, decision and interaction in a collective context (socio-mimetic inspiration)
- Ability to learn and adapt to the environment

