

## *Bases de l'Intelligence Artificielle Distribuée*

# Principes de base : de l'agent aux systèmes multi-agents

Mr. KHEBBACHE Mohib Eddine

2<sup>ème</sup> année SDIA

25 septembre 2022

[mohibeddine-khabache@univ-eloued.dz](mailto:mohibeddine-khabache@univ-eloued.dz)

## Plan

### 1 Introduction

## Plan

### 1 Introduction

### 2 Agent

## Plan

1 Introduction

2 Agent

3 SMA

## Plan

- 1 Introduction
- 2 Agent
- 3 SMA
- 4 Évaluation des SMA

## Plan

- 1 Introduction
- 2 Agent
- 3 SMA
- 4 Évaluation des SMA

Deux questions seront abordées

- Que sont **les agents** et quels sont **les liens** avec les autres **paradigmes logiciels**, en particulier, **les systèmes experts et la programmation orientée objets**
- Pourquoi les agents constituent une nouvelle **approche de choix de conception et d'implémentation** de certaines applications logicielles réparties, coopératives et intelligentes.

## Plan

- 1 Introduction
- 2 Agent
- 3 SMA
- 4 Évaluation des SMA

### Définition selon J. Ferber

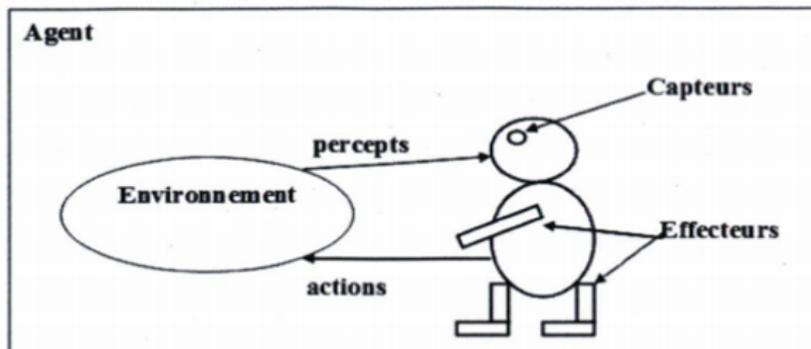
Un Agent peut être défini comme :

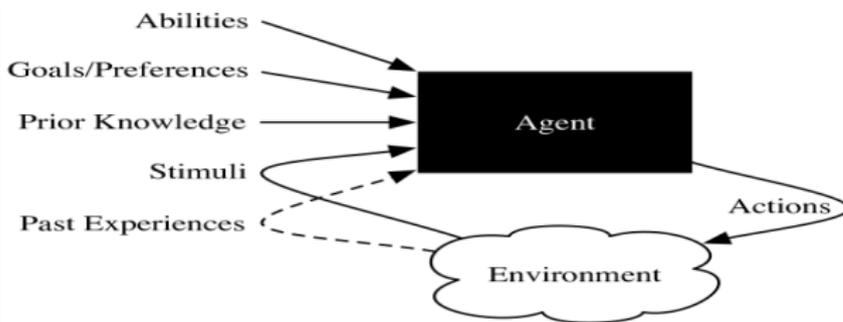
- une entité **autonome** (réelle ou abstraite)
- évoluant dans un environnement,
- capable de **le percevoir** et **d'agir** sur son environnement
- disposant d'une représentation partielle de cet environnement
- pouvant **communiquer** avec d 'autre agents (dans un univers multi-agent)
- ayant **un comportement autonome** qui est la conséquence de **ses observations**, de **ses connaissances** et **des interactions** avec les autres agents.

### Définition selon Russell et Norvig

Un Agent peut être défini comme :

- toute entité
- ayant la capacité **de percevoir** son environnement
  - à travers **ses capteurs "sensors"** (entrées sensorielles)
- et **d'agir** sur un tel environnement
  - via **ses effecteurs "actuators"**
  - où les actions provoquent un **changement dans l'état du système**





- Agent = perception + reasoning + acting + environment, which depends on :
  - inputs :
    - perceptions : stimuli, observations or actions received from the current environment
    - prior knowledge (connaissances), goals (butes) , preferences, abilities (primitive actions) → fournis par agent designer
  - outputs : actions
  - feedback : past experiences ( previous actions and stimuli, or other data, from which it can learn)

- **Agent humain “Human agent”** : eyes as sensors, hands as actuators
- **Agent robot “Robotic agent”** : computational engine + physical sensors, and actuators and environment
- **Agent logiciel “Software agent”** : program that acts in a purely computational environment (eg Internet) → a graphical user interface as sensor and actuator
- **expert system** : advice-giving computer with a human who provides perceptual information and carries out the task

## Définition selon Sycara et Wooldridge

Un agent intelligent :

- est un système informatique **situé** dans un environnement
- et qui **agit** d'une façon **autonome et flexible**
- pour **atteindre les objectifs** pour lesquels il a été conçu.

## Flexibilité ?

- **Réactif** : capacité de percevoir l'environnement et de répondre à temps (élaborer une réponse dans les temps requis) aux changements qui peuvent affecter l'environnement ;
- **Pro-actif** : comportement orienté objectifs en prenant des initiatives au "bon" moment ;
- **Social(Aptitude sociale)** : capacité d'interaction avec les autres agents de façon coopérative ou compétitive pour atteindre ses objectifs.

Il ressort de ces définition certaines propriétés :

- **La nature** : agents physiques ou virtuels
- **situé** : capacité d'agir sur le même environnement qu'il perçoit à partir des entrées sensorielles → e.g systèmes de contrôle de processus, systèmes embarqués... ;
- **L'autonomie** : capacité **d'agir et de contrôler** ses propres actions ainsi que son état interne (ressources) **sans l'intervention** d'un tiers (humain ou agent) → autonomous vehicle with automation levels ;
- **La perception** : capacité d'observer l'environnement par les capteurs ;
- **La communication** : capacités à communiquer avec les autres agents ainsi qu'avec des utilisateurs humains.

- **Le raisonnement** : l'agent peut être lié aux mécanismes de raisonnements plus ou moins complexes (dotés de règles inférentielles).
- **L'environnement** : c'est l'espace dans lequel l'agent va agir ; celui-ci peut se réduire au réseau constitué par l'ensemble des agents.
- **La capacité représentationnelle** : l'agent peut avoir une vision très locale de son environnement, et aussi capacité de représenter des notions mentales.
- **Intentionnalité** : un agent intentionnel est un agent guidé par ses buts. Une intention exprime donc la volonté d'un agent d'atteindre un but ou d'effectuer une action.
- **Rationalité** : les agents rationnels disposent des critères d'évaluation de leurs actions, et sélectionnent selon ces critères les meilleures actions pour atteindre le but.

- **Adaptabilité** : un agent adaptable est un agent capable de contrôler ses aptitudes ( communicationnelles, comportementales, etc. ) selon l'environnement.
- **Engagement** : La notion d'engagement est l'une des qualités essentielles des agents coopératifs.
- **Apprentissage** : un agent peut mémoriser ses expériences et en conséquence, adapter son comportement et/ou créer des nouveaux connaissances → machine learning, deep learning ;
- **Mobilité** : capacité de se déplacer d'une machine à une autre et éventuellement se dupliquer.

- Objectif de l'autonomie : la robustesse
- Les agents **ont certains niveaux de contrôle local sur leurs décisions, leurs comportements et leurs propres états internes** (degrés d'autonomie) :
  - **autonomie de décision (has autonomy) :**
    - Prendre ses décisions « soi-même » en délibération (decision theory).
    - Avoir le pouvoir de dire non !
    - indépendamment des autres
    - → avoir de l'autonomie
  - **Comportement autonome (is autonomous) :**
    - être autonome avec propre contrôle de comportement et ressources (énergie, CPU, quantité de mémoire, accès à certaines sources d'informations, connaissances...)
- e.g : autonomous vehicle

- l'adaptation **structurelle** et **comportementale** d'un ensemble d'agents
  - **adaptation structurelle** :
    - caractéristique **individuelle** des agents
    - on parle alors d'**apprentissage** → par des **techniques symboliques ou neuronales**
  - **adaptation comportementale** :
    - un processus **collectif** mettant en jeu des **mécanismes reproductifs**, ce que l'on appelle **évolution** → en utilisant des **algorithmes génétiques**.

- les agents peuvent **observer des choses différentes** de l'environnement entouré par leurs capteurs
- les observations(percepts) se diffèrent :
  - **spatialement** (apparaître à différents endroits),
  - **temporellement** (arriver à des moments différents),
  - ou **sémantiquement** (nécessite des interprétations différentes)
- → rend le monde partiellement observable à chaque agent

- thermostat avec détecteur  
de température

# **situé**

# **autonome**

- calendrier électronique

# **proactif**

- courrier électronique trié  
par expéditeurs

# **capable de répondre à  
temps**

- courrier électronique trié  
par importance

# **social**

# **buts**

# **coordination**

- le système de control  
aérien de l'aéroport X

# **communication**

# **organisation adaptation**

- **Objet**, entité informatique qui :
  - encapsule un certain état( propriétés ou attributs) et des méthodes décrivant son comportement.
  - et communique par envoi de message
- **Agent**, entité réelle ou virtuelle qui :
  - encapsule un certain état interne et un comportement.
  - et communique par des modes complexes (langage de communication)





### ● Degré d'autonomie

- programmation orienté objet, déclaration des attributs (ou des méthodes) **privées** : accessibles uniquement de l'intérieur de l'objet, ou **Publiques** : accessibles de n'importe où
  - → Un objet a un contrôle absolu sur son propre état interne : il a une autonomie sur son état
- programmation orienté objet, les méthodes sont exécutés sans restriction lorsqu'elles invoquées.
  - ⇒ le comportement d'un objet est contrôlé par élément extérieur (pas de contrôle absolu) → la décision de l'exécution dépend de l'objet qui invoque la méthode ⇒ pas de comportement autonome .
- L'agent a le pouvoir de prendre des initiatives, négocier ou refuser d'obéir à une requête ⇒ l'invocation de méthodes des objets est remplacée par la négociation → avoir une autonomie de décision et a un comportement autonome.



### ● flexibilité : réactif & Proactif & capacité sociale

- pas d'intention d'agir lui même sans faire passer par une demande =  
Un objet est une entité purment réactive → n'est pas flexible.
- les objets ne disposent que de schémas d'interaction prédéfinis et rigides ⇒ pas de capacité sociale
  - ⇒ les objets ne sont pas des entités intelligentes.
- Un agent possède des processus internes qui fonctionnent lui-même ou en réponse à des évènements externes ⇒ Un agent est réactif et Proactif ⇒ flexible.
- un agent dispose de schémas d'interaction complexes (ni prédéfinis ni rigides) ⇒ capacité sociale



- **comportement actif**

- les agents sont des **entités actives** engagées indéfiniment dans une boucle perception-action  $\implies$  un agent est permanent actif,
- les objets sont des **entités passives** pouvant s'activer ponctuellement suite à l'invocation d'une de leurs méthodes.



- **communication et interaction**

- les agents peuvent s'engager dans des interactions de haut niveau en se basant sur **la théorie des actes du langage**
- les objets échangent des messages de **manière ad-hoc (envoi de messages)**.

## Donc

les langages orientés objets peuvent aider à implémenter des agents mais pas à les rendre intelligents (flexibles et autonomes).

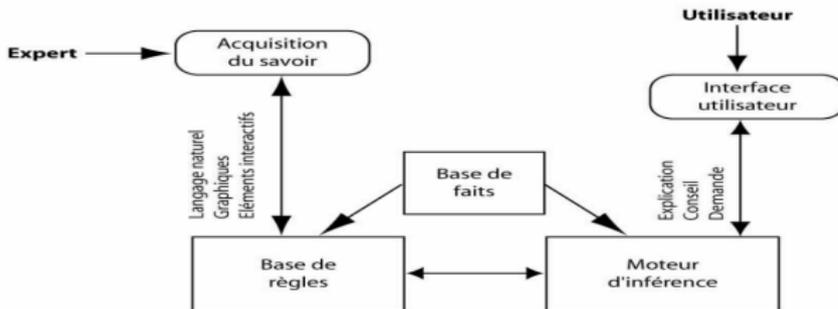
```
import jade.core.Agent;
public class BookBuyerAgent extends Agent {
    @Override
    protected void setup() { .....}
    @Override
    protected void beforeMove() { ..... }
    @Override
    protected void takeDown() { ..... }

}
```

### Définition

Un système expert est un système à base de connaissances capable de répondre à des questions, en effectuant un raisonnement à partir de faits et de règles connus. Il peut servir notamment comme outil informatique d'aide à la décision et au diagnostic.

- Un système expert se compose de 3 parties :
  - 1 Une base de faits ;
  - 2 Une base de règles ;
  - 3 Un moteur d'inférence.





### ● un système expert (SE)

- n'agit sur aucun environnement, fournit des conseils
- raisonne, mais n'agit pas en perception de ..., ni en action sur ...
- n'obtient pas des informations à travers des capteurs, mais à travers un utilisateur agissant comme intermédiaire.
- pas autonome : nécessitent des interventions et des instructions des experts.
- Pas pro-actifs, ni adaptatifs, ni distribués.
- SE n'est pas supposé de coopérer avec d'autres agents (sauf certains SE temps réels comme ARCHON)
- L'évolution des SE vers les systèmes multi-experts → l'origine des SMA.

## Plan

- 1 Introduction
- 2 Agent
- 3 SMA**
- 4 Évaluation des SMA



# Systemes Multi-Agents(SMA)

## Pourquoi les SMA (motivations) ?

Principes  
de base :  
de l'agent  
aux  
systemes  
multi-  
agents

Mr.  
KHEB-  
BACHE  
Mohib  
Eddine

Introduction

Agent

SMA

Évaluation  
des SMA

SMA peut offrir certaines propriétés :

- Efficacité
  - Décomposition et répartition des connaissances et des traitements (Unité base : agent).
  - La vitesse, avec le parallélisme.
- Souplesse
  - portabilité, adaptabilité et possibilité d'apprentissage des agents.
  - Fiabilité et tolérance aux défaillances due à la redondance et à la distribution du contrôle  $\implies$  résistance à des environnements évolutifs ou instables

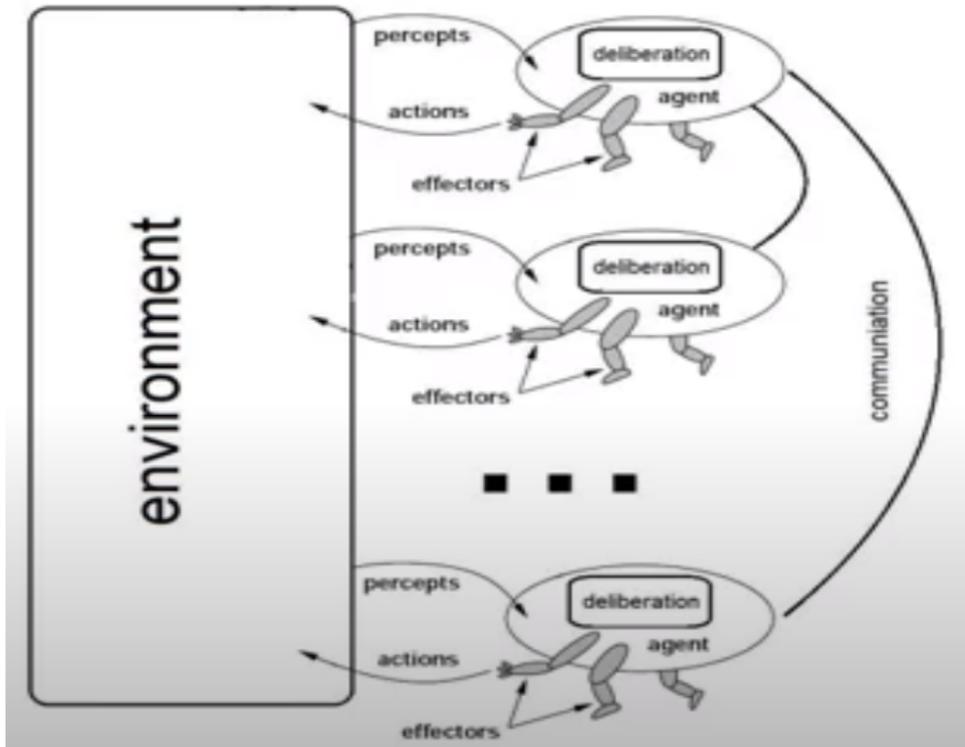
- Paradigme de conception
  - Systèmes et architectures ouvertes
  - Adéquation à des applications distribuées et coopératives
  - Modularité, réutilisabilité, interopérabilité, l'extensibilité
  - Traitement symbolique au niveau de connaissances.
  - L'intervention des schémas d'interaction sophistiqués (coopération, coordination, négociation).
- offre une vue naturelle des systèmes intelligents
- offre une grande qualité de service dans un travail collectif

### point de vue systémique

Un système est un ensemble **d'éléments en interaction** ; les **interactions** contribuent à **maintenir la structure du système** et à la faire **évoluer**. Le système est **dynamique** lorsque les interactions provoquent des **changements d'états**

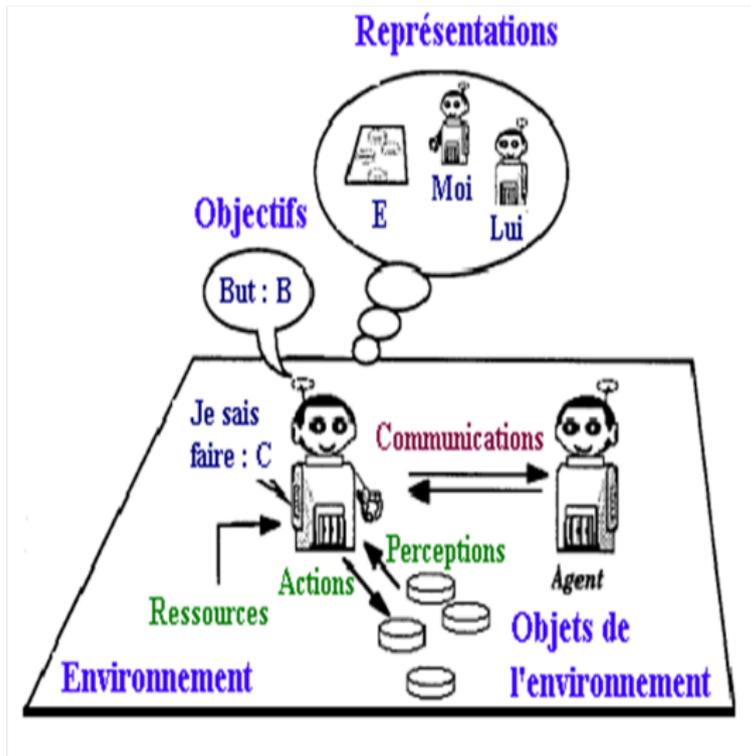
### point de vue d'approche orientée-agent

- comme approche moderne de l'IAD, un SMA est un ensemble d'entités actives  $\implies$  **agents artificiels**,
- capables d'agir de manière autonome (**performs independent actions**),
- ayant le contrôle de leurs propres comportements et états internes,
- et pouvant interagir et s'organiser de manière flexible dans un environnement commun (**shared environment**)
- pour aboutir à un objectif global : la résolution de problèmes.



**Ferber** propose une description générale de la structure d'un système Multi-agents :

- ❶ **un environnement** : c'est à dire un espace disposant généralement d'une métrique.
- ❷ **un ensemble d'objets situés**, c'est à dire qu'il est possible d'associer à chaque objet, à un moment donné une position dans l'environnement.
- ❸ **un ensemble d'agents** représentant des entités actives du système.
- ❹ **un ensemble de relations** qui unissent des objets (ou des agents ) entre eux.
- ❺ **un ensemble d'opérations** permettant aux agents de percevoir, produire, consommer, transformer, manipuler des objets.
- ❻ **des opérateurs** chargés de représenter l'application de ces opérations et la relation du monde à cette tentative de modification



- Un SMA peut-être :
  - ouvert : les agents y entrent et en sortent librement (ex : un café)
  - fermé : l'ensemble d'agents reste le même (ex : un match de football)
  - homogène : tous les agents sont construits sur le même modèle (ex : une colonie de fourmis)
  - hétérogène : des agents de modèles différents, de granularité différentes
- Un SMA peut offrir :
  - L'autonomie, La distribution, La décentralisation, le calcul asynchrone...

- Dans les SMA, les agents :
  - coexistent dans un environnement commun (**shared environment**)
  - sont partiellement indépendants :
    - ayant des **rôles et tâches(tasks) spécifiques** à accomplir
    - ayant des **points de vue partiels(locales)** du système → **perception distribuée**
    - ayant des **objectifs (goals) individuels**
    - ⇒ des capacités limitées de résolution de problèmes (e.g. agents interfaces, agents d'information),
  - ont de **nature** :
    - **homogènes** : same tasks and goals
    - **hétérogènes** : different tasks and goals
  - devront interagir
    - pour : coopérer (contrôle), collaborer (allocation de tâches), négocier (résolution de conflits), se coordonner (synchronisation) ou d'entrer en conflit,
    - afin d'atteindre un objectif collectif ou de favoriser la stabilité du système



# Système multi-agents

## Caractéristiques d'un SMA

Principes  
de base :  
de l'agent  
aux  
systèmes  
multi-  
agents

Mr.  
KHEB-  
BACHE  
Mohib  
Eddine

Introduction

Agent

**SMA**

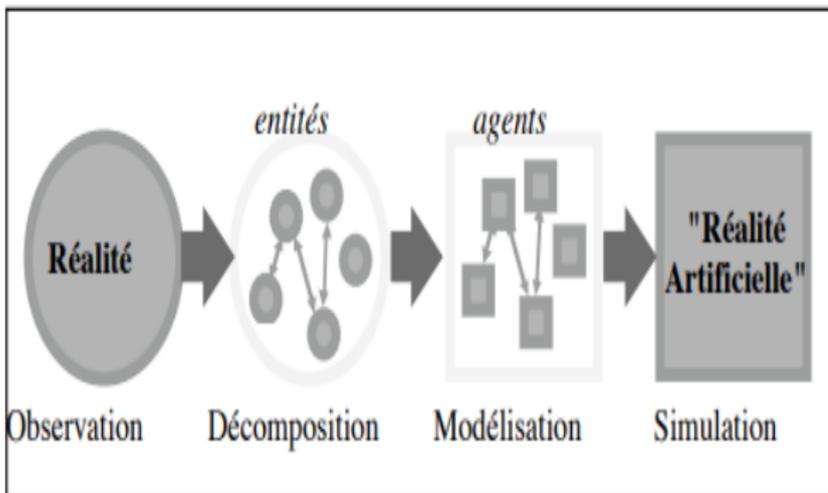
Évaluation  
des SMA

- Dans les SMA, l'environnement :
  - dynamique, incertain et avec distribution totale des connaissances et du contrôle
  - → **absence de contrôle central du système global**,
    - → le comportement et la dynamique globale du système sont les résultats des interactions entre ses entités.

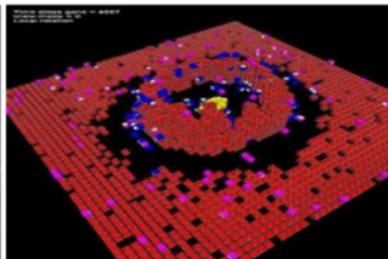
## Simulation multi-agent "ABS" (Agent Based Simulation) :

- l'Études et la simulation de phénomènes complexes (sociaux ou naturels)
  - faisant intervenir un grand nombre d'«individus»
  - par la construction et l'analyse des modèles, sous forme d'agent  
⇒ décomposition structurelle.
- Exemples :
  - Ecosystèmes, Société d'insectes ( une colonie de fourmis)
  - La robotique collective
  - La construction de mondes synthétiques
  - Compréhension des interactions entre humains : **simulations comportementales**
    - Étude de clientèles, Intégration de malades mentaux
  - Modéliser des environnements virtuels :
    - jeux vidéo, cinéma

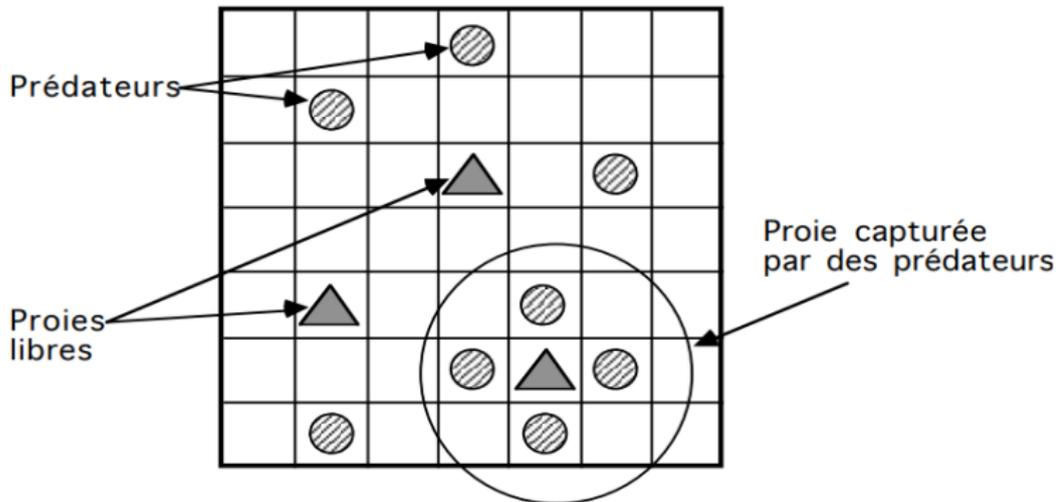
## Simulation multi-agent "ABS" (Agent Based Simulation) : décomposition structurelle



## Simulation multi-agent "ABS" (Agent Based Simulation) : simulation de systèmes complexe



### Simulation multi-agent "ABS" (Agent Based Simulation) : La construction de mondes synthétiques la poursuite (le problème des proies et des prédateurs)



## Simulation multi-agent "ABS" (Agent Based Simulation) : La robotique collective Plusieurs robots qui coopèrent pour ac- complir une mission



## Simulation multi-agent "ABS" (Agent Based Simulation) : Compréhension des interactions entre humains & Modéliser des environnements virtuels



## La résolution de problèmes complexes par des agents logiciels :

- La résolution distribuée de problèmes où le domaine n'est pas distribué:
  - Systèmes de production : ordonnancement d'ateliers, conduite et contrôle de processus industriels, systèmes multi-capteurs, ...
  - Diagnostic : diagnostic médical, diagnostic de réseaux
- La résolution distribuée de problèmes distribués:
  - Taches de contrôle : contrôle du trafic routier, trafic aérien, distribution d'énergies, supervision d'un atelier de production (plusieurs machines, plusieurs pièces, plusieurs usinages) ...
  - Télécommunications, systèmes de transports, réseaux : routage, équilibrage de charges, recouvrement d'erreurs, management et surveillance de réseaux de télécommunications (détection, intervention, réparation) : agents de suivi, agents de diagnostic de panne, agent opérateur de maintenance

**l'intégration et l'inter-opérabilité** : visent à intégrer des logiciels existants, des systèmes mécaniques et des êtres humains et à faire interopérer l'ensemble d'une manière cohérente :

- Commerce électronique et processus métier : applications B2C, B2B...
- Recherche et filtrage d'informations sur le Web : agrégation de services
- intelligence ambiante « pervasive computing » : vêtements intelligents, réseaux ad hoc, IoT, Data Mining, Cloud, BigData...

- l'environnement une partie essentielle (élément) des systèmes multi-agents situés.
- L'environnement peut être dans un état « e » parmi un ensemble d'états  $E=e_1, \dots, e_n$ .
- L'environnement peut changer son état soit d'une manière spontanée soit comme résultat des actions de l'agent.
- L'environnement peut être : **environnement centralisé & environnement distribué**
- un lieu où des ressources sont disponibles :
  - la donnée du problème  $\implies$  cas de la résolution de problèmes
  - ensemble d'objets actifs et passifs que l'agent peut manipuler (ex : dans la nourriture, les larves, )  $\implies$  cas de simulation
  - des informations non contrôlées par le système (ex : web)  $\implies$  cas d'intégration et des systèmes collaboratifs
  - autre : un utilisateur via une interface graphique, autres agents ...

les S.M.A relevent beaucoup de défis :

- Comment formuler, décrire, décomposer, et allouer les problèmes et synthétiser les résultats ?
- Comment permettre aux agents de communiquer et d'interagir ? Quoi et quand communiquer ? → problématique de communication
- Comment assurer que les agents agissent de manière cohérente : en prenant leurs décisions ou actions, en gérant les effets non locaux de leurs décisions locales ou en évitant les interactions nuisibles ? → problématique de l'action
- la problématique de l'adaptation en termes d'adaptation individuelle ou apprentissage d'une part et d'adaptation collective ou évolution d'autre part.



- La problématique de l'interaction s'intéresse aux moyens de l'interaction (quel langage ? quel support ?), et à l'analyse et la conception des formes d'interactions entre agents. Les notions de collaboration et coopération (en prenant coopération comme collaboration + coordination d'actions + résolution de conflits) sont ici centrales.
- Comment trouver le meilleur compromis entre le traitement local au niveau d'un seul agent et le traitement distribué entre plusieurs agents → comment gérer la répartition des ressources limitées ?
- Comment concevoir les plates-formes technologiques et les méthodologies de développement pour les S.M.A ?

- SMA est un **domaine de recherche pluridisciplinaire** s'appuyant sur des connaissances théoriques solides, validées et partagées par une communauté
  - **une communauté scientifique** : projet européen **AgentLink**, conférences internationales annuelles **AAMAS** et **IJCAI**, revues **JAAMAS**, **IJAOS** et **JASSS**.
- des progrès spécifiques ont été réalisés **en termes d'ingénierie** permettant une industrialisation des applications par la proposition
  - de **méthodes (GAIA, MOISE...)**,
  - de **formalismes basés** sur la logique, de **notations (ex. AUML)**,
  - de langages standards (ex. FIPA-ACL)
  - de **plateformes de simulation (GAMA, NetLogo) ou de développement (Madkit, JADE)**

En s'inspirant de **l'ingénierie de systèmes** , **l'ingénierie de connaissances** et **l'ingénierie orientée objet**, *l'ingénierie logicielle à base d'agents* ou *Ingénierie logicielle orienté agent* (**Agent-based software engineering "ABSE"**) est un nouveau domaine du développement logiciel qui utilise des agents comme éléments de base ( **building blocks**) d'un système logiciel  $\implies$  (le cas d'objets de l'ingénierie logiciels orientés objet)

- trois étapes (steps or phases) de projet de développement des SMA  $\implies$  PROCESSUS DE DÉVELOPPEMENT :
  - analyse et modélisation du système (**system analysis and modeling**)
  - conception et implémentation (**design and implementation**)  
 $\implies$  commençant par une conception générale de l'architecture du système pour ensuite détailler les composants du système : gents et ressources,
  - validation du logiciel (**software validation**)

- deux grands groupes de méthodologies pour le développement du logiciel orienté agent :
  - Le premier groupe étend ou adapte les méthodologies orientées objet pour prendre en compte les caractéristiques des agents,
  - le deuxième groupe part des méthodologies de l'ingénierie de connaissances.
- Les méthodologies (méthodes de conception) les plus employées en ingénierie des systèmes multi-agents sont : méthodologie **GAIA** et **AUML**



### La méthodologie Gaia,

- proposé par Wooldridge (Wooldridge e.a., 2001),
- permet de concevoir un système multi-agents en utilisant un paradigme organisationnel.
- Les concepts (ou entités) de Gaia sont de deux types : abstraits et concrets.
- Le paradigme organisationnel utilisé par la méthodologie est centré sur le concept de rôle.
- Un rôle est défini par quatre attributs : responsabilités, permissions, activités, et protocoles.

## La méthodologie AUML (**Agent Unified Modelling Language**)

- la phase de conception de SMA nécessite des notations spécifiques
- proposé d'adapter les notations d'UML pour décrire la modélisation orientée agent.
- Les modifications proposées à UML sont :
  - la représentation des threads simultanés d'interaction (par exemple transmission de messages à plusieurs agents) permettant ainsi à UML de modéliser les protocoles d'interactions entre agents, par exemple le réseau contractuel ;
  - la notion de rôle qui étend celle fournie par UML et permet de modéliser un agent qui joue plusieurs rôles.
  - accord avec les spécifications existantes de FIPA (Foundation for Intelligent Physical Agents) et OMG (Object Management Group).

## Plan

- 1 Introduction
- 2 Agent
- 3 SMA
- 4 **Évaluation des SMA**

- l'environnement est vue comme un problème dans lequel les agents agissent comme des solutionneurs
- un agent  $\implies$  **perçoit son environnement par des capteurs**  $\implies$  séquence de percepts.

### définir le succès de l'agent

Pour chaque séquence de percepts possible, un agent rationnel devrait **sélectionner une action** qui devrait **maximiser son performance** jugé par le changement d'état de l'environnement.

Donc, **la rationalité de l'agent** peut alors être **évaluée** par **une mesure de performance**, en fonction de la représentation de l'environnement :

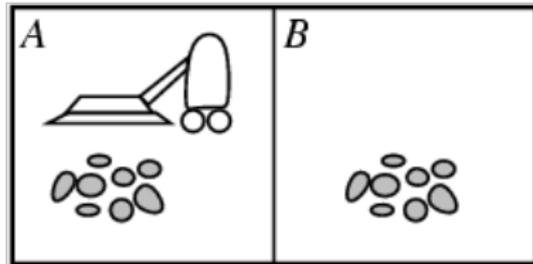
- rationalité (succès de l'agent)  $\implies$  actions que l'agent peut effectuer(A)\* = **argmax** MP(connaissance préalable de l'agent sur l'environnement , séquence de percepts de l'agent à temps).

La première étape lors de la conception d'un agent rationnel est de **spécifier l'environnement de la tâche (task environment)** par les quatre éléments suivants (PEAS) :

- **Performance** : Mesure de la performance
- **Environment** : Environnement
- **Actuators** : Effecteurs ou actionneurs
- **Sensors** : Capteurs

### Exemple : Le monde de l'aspirateur

- *Perçus* : localisation et contenus, e.g., [A,Dirty]
- *Actions* : Left, Right, Suck, NoOp
- *Mesure de performance* : Un critère objectif de succès pour un comportement d'agent
  - volume de saleté ramassée,
  - temps pris,
  - électricité consommée,
  - bruit produit, etc.



### Exemple : Agent : robot pour ranger des pièces

- *Mesure de performance* : Pourcentage de pièces mises dans des boîtes correctes
- *Environnement* : Convoyeur de pièces, boîtes
- *Actuateurs* : bras mécanique
- *Senseurs* : Camera, senseurs des angles



Type d'agent	Mesure de performance	Environnement	Effecteurs	Capteurs
Système de diagnostic médical	Rétablissement des patients, minimisation des coûts	Patient, hôpital, personnel	Affichage de questions, tests, diagnostics, traitements, orientation	Entrée au clavier de symptômes, recherches, réponses du patient
Système d'analyse d'images satellites	Catégorisation correcte des images	Liaison descendante depuis un satellite en orbite	Affichage de la catégorisation de la scène	Tableaux de pixels colorés
Robot contrôleur de pièces	Pourcentage de pièces dans les bonnes corbeilles	Tapis roulant avec pièces, corbeilles	Bras et main articulés	Caméra, capteurs articulés
Contrôleur de raffinerie	Pureté, production, sécurité	Raffinerie, opérateurs	Valves, pompes, réchauffeurs, affichages	Capteurs de température, de pression et chimiques
Répétiteur d'anglais interactif	Notes des étudiants aux contrôles	Ensemble d'étudiants, organisme faisant passer les tests	Affichages d'exercices, de suggestions, de corrections	Entrées au clavier

**Figure 2.5 :** Exemples de types d'agents et leurs descriptions PEAS.