

La Programmation Orientée Agent Introduction

Emmanuel ADAM

Université de Valenciennes et du Hainaut-Cambrésis



UVHC/ISTV-LAMIH



1 Rappels sur les agents, les systèmes multi-agents

- Définition d'un agent
- Système multi-agent
- Les types d'agents
- Couplage de types d'agents
- Communication
- Organisation
- Agent vs Objet

Définition d'un agent

Définition générale

- Un agent est une entité adaptative, rationnelle, autonome, capable de communication et d'action.

Selon [Gasser 89], [Demazeau 95], [Ferber 95], [Nwana 96], [Chaib-Draa 96], ...

- Un agent est :
 - autonome
 - agir
 - capable de perception
 - adaptatif
 - apprendre
 - modifier le comportement
 - coopérant
 - communiquer
 - négocier

Définition d'un agent

Définition générale

- Un agent est une entité adaptative, rationnelle, autonome, capable de communication et d'action.

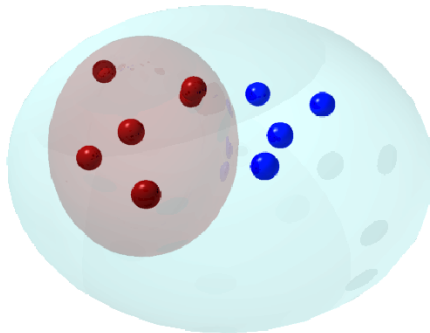
Selon [Gasser 89], [Demazeau 95], [Ferber 95], [Nwana 96], [Chaib-Draa 96], ...

- Un agent est :
 - autonome
 - agir
 - capable de perception
 - adaptatif
 - apprendre
 - modifier le comportement
 - coopérant
 - communiquer
 - négocier

Système Multi-Agent

SMA, Monde et Environnement

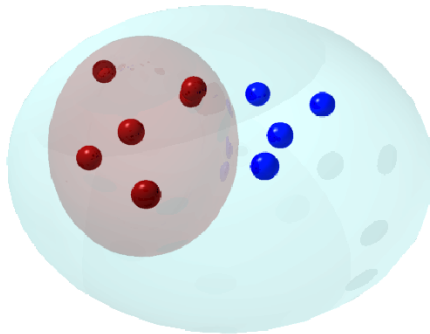
- Un *système multi-agent (SMA)* est constitué d'agents.
- Les entités en interaction avec le SMA forment l'*Environnement*.
- L'ensemble SMA + Environnement est appelé *Monde*



Système Multi-Agent

SMA, Monde et Environnement

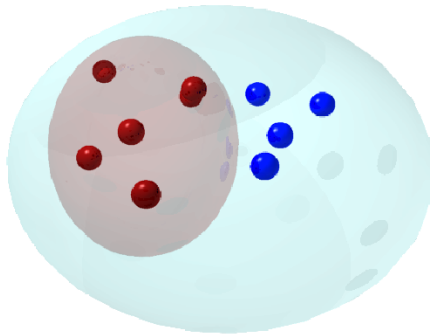
- Un *système multi-agent (SMA)* est constitué d'agents.
- Les entités en interaction avec le SMA forment l'*Environnement*.
- L'ensemble SMA + Environnement est appelé *Monde*



Système Multi-Agent

SMA, Monde et Environnement

- Un *système multi-agent (SMA)* est constitué d'agents.
- Les entités en interaction avec le SMA forment l'*Environnement*.
- L'ensemble SMA + Environnement est appelé *Monde*



Structure d'un agent

Grille d'analyse fonctionnelle [Ferber 95]

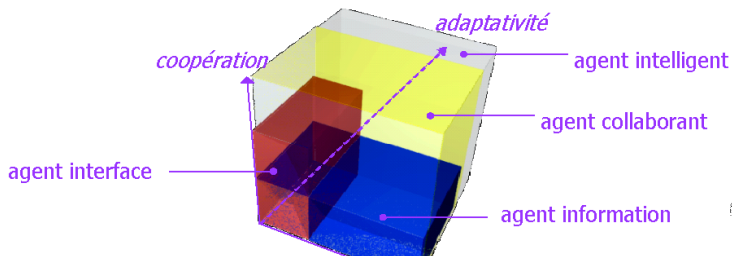
Les agents possèdent des fonctions décrites selon différents axes :

...	<i>Personnel</i>	<i>Environnemental</i>	<i>Social</i>	<i>Relationnel</i>
<i>Représentationnel</i>	Connaissances			
<i>Organisationnel</i>	Planification			
<i>Conatif</i>	But, désirs, contraintes			
<i>Interactionnel</i>	Perception, coopération			
<i>Productif</i>	Action			
<i>Conservatif</i>	Conservation, protection			

Les types d'agents

Agents cognitifs et réactifs

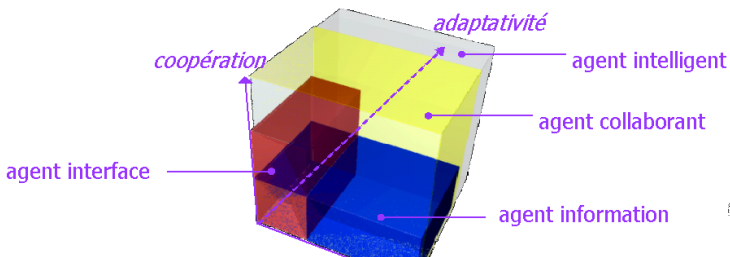
- Les *agents cognitifs* possèdent une représentations partielle du monde, communiquent avec les autres agents et élaborent de stratégies.
- Les *agents réactifs* ont un perception limitée et fonctionnent par action/réaction.
- Certains besoins nécessitent l'utilisation de SMA *hybrides* couplant ces types d'agents.



Les types d'agents

Agents cognitifs et réactifs

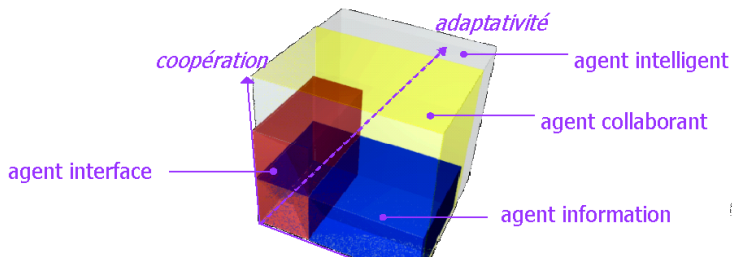
- Les *agents cognitifs* possèdent une représentations partielle du monde, communiquent avec les autres agents et élaborent de stratégies.
- Les *agents réactifs* ont un perception limitée et fonctionnent par action/réaction.
- Certains besoins nécessitent l'utilisation de SMA *hybrides* couplant ces types d'agents.



Les types d'agents

Agents cognitifs et réactifs

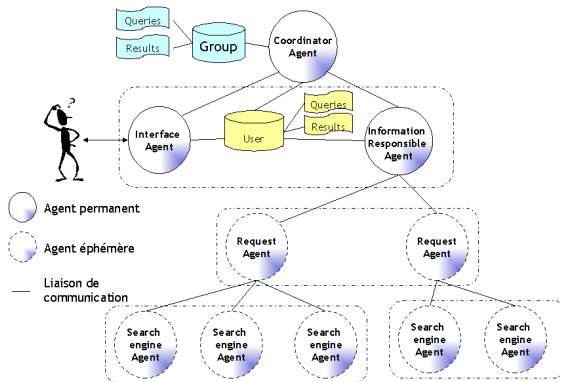
- Les *agents cognitifs* possèdent une représentations partielle du monde, communiquent avec les autres agents et élaborent de stratégies.
- Les *agents réactifs* ont un perception limitée et fonctionnent par action/réaction.
- Certains besoins nécessitent l'utilisation de SMA *hybrides* couplant ces types d'agents.



Couplage de types d'agents

Exemple : la recherche d'informations

Cet exemple dans la recherche d'information fait interagir des agents d'interfaces, des agents d'informations et des agents de collaboration.



Langage de communication agent

ACL : Agent Communication Language

La nécessité de posséder un langage commun entre agents est claire. Plusieurs propositions :

- KIF : Knowledge Interchange Format (Langage formel de dialogue interne)
- KQML : Knowledge Query Meta Language
- *FIPA ACL : extension de KQML*

FIPA-ACL 1/2

structure d'un message FIPA-ACL

Un message en FIPA ACL est constitué de champs obligatoires et facultatifs :

- performative** : type de l'acte communicatif
 - sender** : emetteur du message
 - receiver** : destinataire(s) du message
 - reply-to** : destinataire de la réponse au message
 - content** : contenu du message
- language** : type de langage utilisé
- encoding** : type de codage du message
- ontology** : ontologie sur laquelle est basée le message
- protocol** : type de protocole utilisé
- conversation-id** : identifiant de la conversation
 - reply-with** : type de réponse souhaitée
 - in-reply-to** : nom de la requête
 - reply-by** : type de réponse

FIPA-ACL 2/2

Liste de performatifs de FIPA ACL :

Les performatifs possibles sont :

- *Accept Proposal, Agree, Cancel, Call for Proposal, Confirm, Disconfirm, Failure, Inform, Inform If, Inform Ref, Not Understood, Propagate, Propose, Proxy, Query If, Query Ref, Refuse, Reject Proposal, Request, Request When, Request Whenever, Subscribe*

Protocoles

Protocoles de communications

Des normes de communications fixant les règles d'interactions peuvent être utilisées, par exemple :

AchieveRE : un Initiateur envoie un message, le receveur peut répondre par not-understood, refuse ou agree. Suite à l'accord (agree), le receveur retourne un message de type inform (réponse) ou failure.

FIPA - Contract NET : Protocole d'établissement d'un contrat (de vente par exemple).

FIPA - Propose : un Initiateur propose à Participant d'effectuer une action .

FIPA - Subscribe : un Initiateur envoie un message à un Participant de souscription de service.

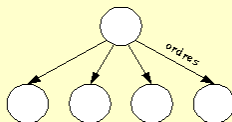
Structures de SMA

Organisations

Les SMA peuvent être :

organisés a priori : en structure hiérarchique, de marché, de communauté, de société [Grislin95], [Mandiau99]

organisés par émergence : la structure de l'organisation apparaît suite aux interactions entre agents



Structure hiérarchique
[Ito 98], [Odubiyi 97], ...

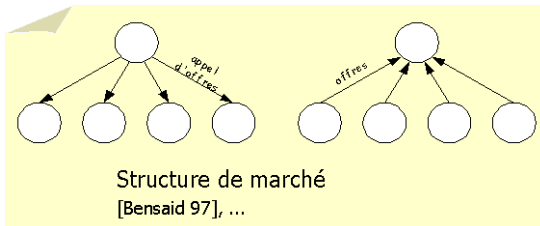
Structures de SMA

Organisations

Les SMA peuvent être :

organisés a priori : en structure hiérarchique, de marché, de communauté, de société [Grislin95], [Mandiau99]

organisés par émergence : la structure de l'organisation apparaît suite aux interactions entre agents



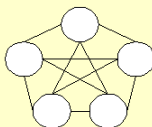
Structures de SMA

Organisations

Les SMA peuvent être :

organisés a priori : en structure hiérarchique, de marché, de communauté, de société [Grislin95], [Mandiau99]

organisés par émergence : la structure de l'organisation apparaît suite aux interactions entre agents



Structure de communauté
[Minar 98], ...

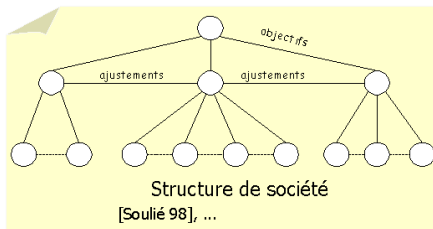
Structures de SMA

Organisations

Les SMA peuvent être :

organisés a priori : en structure hiérarchique, de marché, de communauté, de société [Grislin95], [Mandiau99]

organisés par émergence : la structure de l'organisation apparaît suite aux interactions entre agents



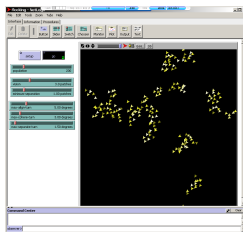
Structures de SMA

Organisations

Les SMA peuvent être :

organisés a priori : en structure hiérarchique, de marché, de communauté, de société [Grislin95], [Mandiau99]

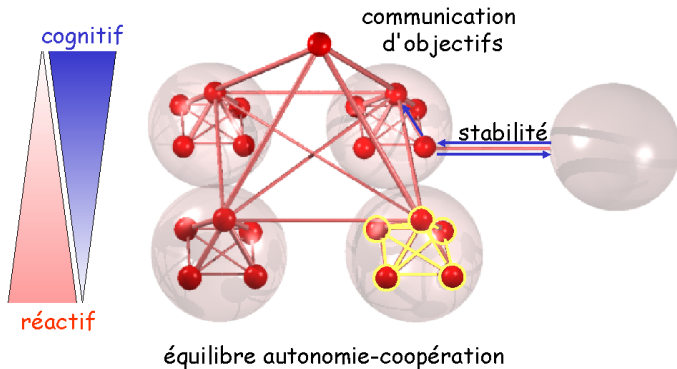
organisés par émergence : la structure de l'organisation apparaît suite aux interactions entre agents



Système MultiAgent Holonique 1/3

SMA Holonique

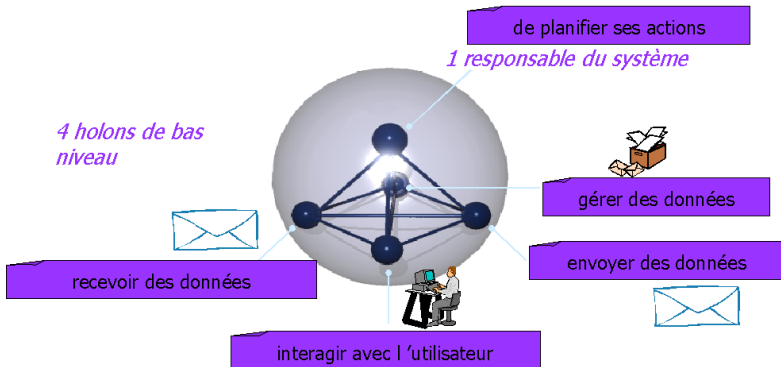
Un sma holonique possède une structure réursive de société. Les agents (holons) sont stables, autonomes et coopérants.



Système MultiAgent Holonique 2/3

Récursivité

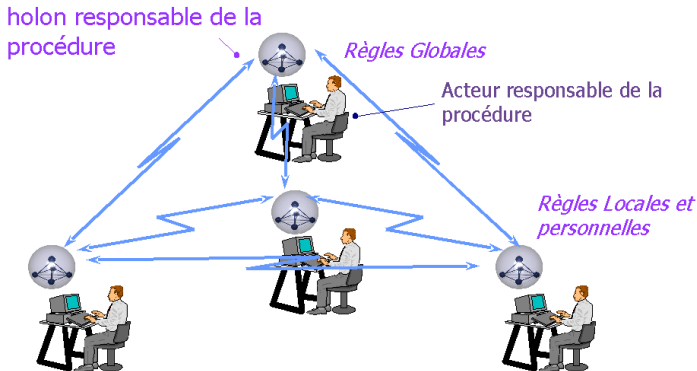
Chaque holon du système possède les mêmes caractéristiques que l'ensemble et possède une spécialité.



Système MultiAgent Holonique 3/3

SMA Holonique

Un ensemble de sma holoniques (agents holoniques), est composé de sma holoniques.



Agent vs Objet

Un agent est il un objet ?

- Un agent est un objet qui est adaptatif, rationnel, autonome, capable de communication et d'action.
- Un agent logiciel est de préférence créé sous forme d'un objet ayant les caractéristiques d'un processus
- Un objet $o1$ ne peut qu'appeler une méthode existante d'un objet $o2$ qui doit l'exécuter
- Un agent $a1$ peut demander l'exécution d'une méthode à l'agent $a2$ qui ne la connaît pas ou qu'il ne souhaite pas exécuter (dans l'instant ou jamais !)
- Un agent agit en fonction de son but et de ses contraintes.

Agent vs Objet

Un agent est il un objet ?

- Un agent est un objet qui est adaptatif, rationnel, autonome, capable de communication et d'action.
- Un agent logiciel est de préférence créé sous forme d'un objet ayant les caractéristiques d'un processus
- Un objet $o1$ ne peut qu'appeler une méthode existante d'un objet $o2$ qui doit l'exécuter
- Un agent $a1$ peut demander l'exécution d'une méthode à l'agent $a2$ qui ne la connaît pas ou qu'il ne souhaite pas exécuter (dans l'instant ou jamais !)
- Un agent agit en fonction de son but et de ses contraintes.

Agent vs Objet

Un agent est il un objet ?

- Un agent est un objet qui est adaptatif, rationnel, autonome, capable de communication et d'action.
- Un agent logiciel est de préférence créé sous forme d'un objet ayant les caractéristiques d'un processus
- Un objet $o1$ ne peut qu'appeler une méthode existante d'un objet $o2$ qui doit l'exécuter
- Un agent $a1$ peut demander l'exécution d'une méthode à l'agent $a2$ qui ne la connaît pas ou qu'il ne souhaite pas exécuter (dans l'instant ou jamais !)
- Un agent agit en fonction de son but et de ses contraintes.

Agent vs Objet

Un agent est il un objet ?

- Un agent est un objet qui est adaptatif, rationnel, autonome, capable de communication et d'action.
- Un agent logiciel est de préférence créé sous forme d'un objet ayant les caractéristiques d'un processus
- Un objet $o1$ ne peut qu'appeler une méthode existante d'un objet $o2$ qui doit l'exécuter
- Un agent $a1$ peut demander l'exécution d'une méthode à l'agent $a2$ qui ne la connaît pas ou qu'il ne souhaite pas exécuter (dans l'instant ou jamais !)
- Un agent agit en fonction de son but et de ses contraintes.

Agent vs Objet

Un agent est il un objet ?

- Un agent est un objet qui est adaptatif, rationnel, autonome, capable de communication et d'action.
- Un agent logiciel est de préférence créé sous forme d'un objet ayant les caractéristiques d'un processus
- Un objet $o1$ ne peut qu'appeler une méthode existante d'un objet $o2$ qui doit l'exécuter
- Un agent $a1$ peut demander l'exécution d'une méthode à l'agent $a2$ qui ne la connaît pas ou qu'il ne souhaite pas exécuter (dans l'instant ou jamais !)
- Un agent agit en fonction de son but et de ses contraintes.