

# PRÉSENTATION DES SYSTÈMES MULTI-AGENTS

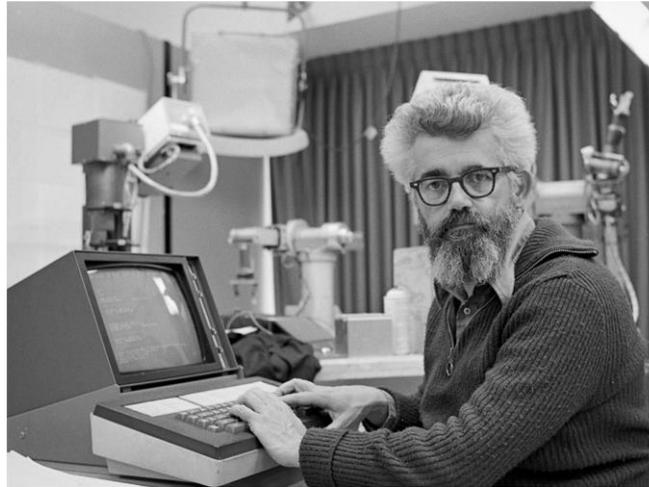
---

Cours 1

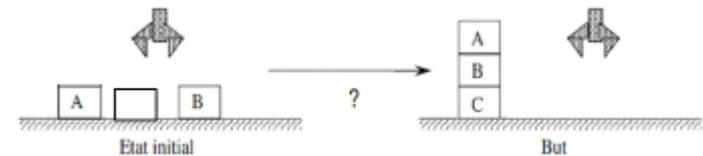
# Intervenants

- Philippe Caillou
  - MCF LRI
  - caillou@lri.fr
  - 5 cours
  - Cours disponibles sur [www.lri.fr/~caillou/master](http://www.lri.fr/~caillou/master)
- Nicolas Sabouret
  - PR LIMSI
  - sabouret@limsi.fr
  - 3 cours

# Dans les années 60, le début de l'intelligence artificielle...



John MacCarthy, qui proposa le premier Workshop sur l'intelligence artificielle en 1956

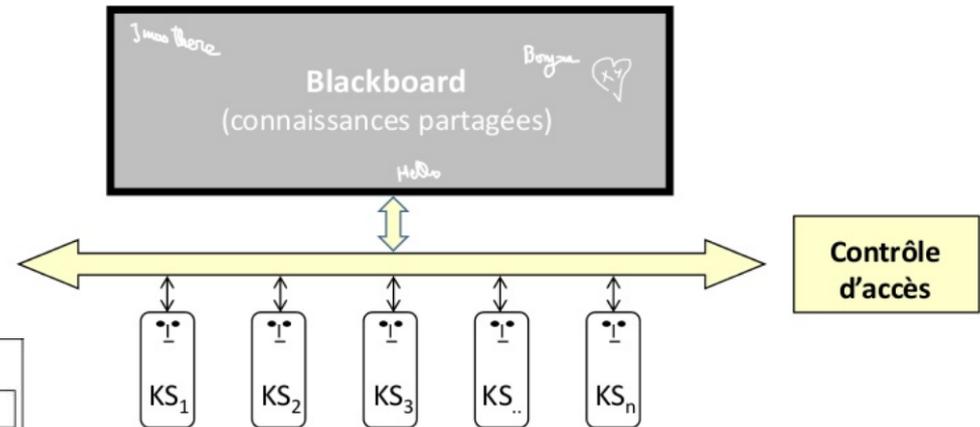
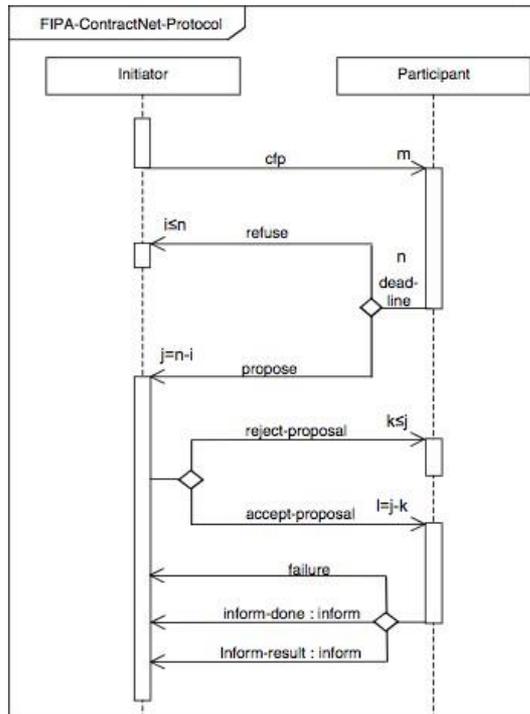


1971: STRIPS

```
(:action stack
:parameters (?x - block ?y - block)
:precondition (and (holding ?x) (clear ?y))
:effects (and (not (holding ?x)) (not (clear ?y))
           (on ?x ?y) (clear ?x) (handempty))
```

# Années 70: le début des agents

1973: Hearsay  
1980: Hearsay II  
Modèles de  
Tableaux noir



1977 Hewitt introduit the actor model: un acteur est social, réactif, communique par messages

1977: Contract Net Protocol

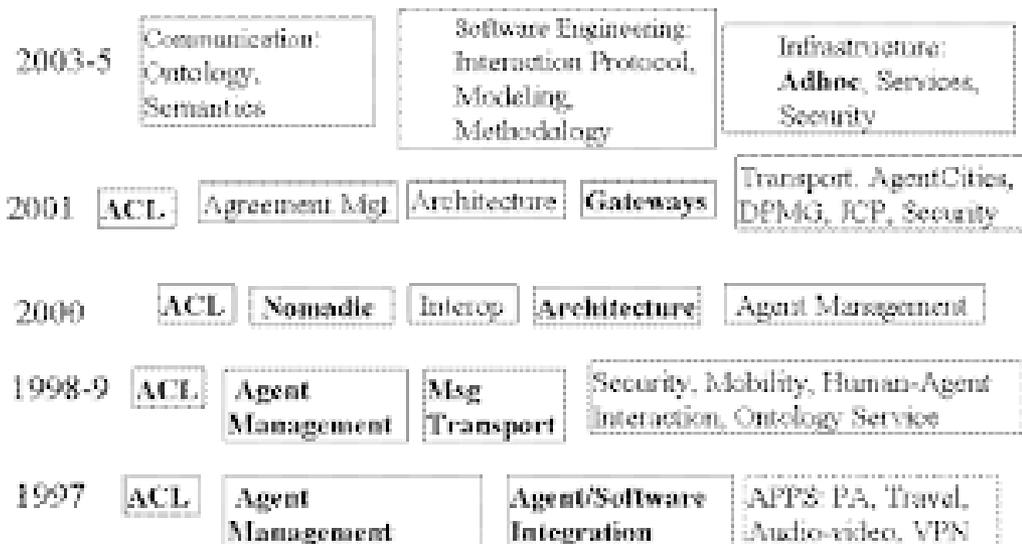
# Années 90: la normalisation

1992: KQML

```

<performative> ::= (<word> {<whitespace> :<word> <whitespace> <expression>}*)
<expression>  ::= <word> | <quotation> | <string> |
                  (<word> {<whitespace> <expression>}*)
<word>        ::= <character><character>*
<character>   ::= <alphanumeric> | <numeric> | <special>
<special>     ::= < | > | = | + | - | * | / | & | ^ | ~ | _ |
                  @ | $ | % | : | . | ! | ?
<quotation>   ::= '<expr>' | '<comma-expr>'
<comma-expr> ::= <word> | <quotation> | <string> | ,<comma-expr> |
                  (<word> {<whitespace> <comma-expr>}*)
<string>      ::= "<stringchar>*" | #<digit><digit>*"<ascii>*
<stringchar> ::= \<ascii> | <ascii>\-<double-quote>
  
```

97-05 :Foundation for  
Intelligent Physical  
Agents (FIPA)  
normalisation



# Debut 2000: les applications

1997

- ...



2001



# 2005-2015: l'analyse et le formalisme

## Prize in Game Theory and Computer Science of the Game Theory Society

### Call for nominations - Prize in Game Theory and Computer Science

(mailed to GTS members by Roger Myerson on 25 April 2013)

The Prize in Game Theory and Computer Science of the Game Theory Society was established in 2008 by a donation from Yoav Shoham in recognition of Ehud Kalai's role in promoting the connection of the two research areas. The **last time the Prize was awarded** at the Third World Congress of the Society in 2008 in Evanston. This time, the Prize will in 2013 be awarded at the **Workshop on Computational Game Theory in Stony Brook**, New York (July 16-18, 2013).

	Wednesday 7	Thursday 8	Friday 9
9h00	Norms • Verification I <b>level 3</b>	Social Networks II • Verification II <b>level 3</b>	Learning II • Challenges <b>level 3</b>
	CGT I • H&A I <b>level 2</b>	CGT II • Voting II <b>level 2</b>	Auctions • H&A IV • Path Planning <b>level 2</b>
	Voting I <b>level 0</b>	H&A II <b>level 0</b>	<b>level 0</b>
10h20	Learning I <b>level -1</b>	Planning I <b>level -1</b>	<b>level -1</b>
10h50	AoSE I • Logic I <b>level 3</b>	Dissertation <b>level 3</b>	Planning II • Mechanism Design II (11h40) <b>level 3</b>
	AGT I • Information I <b>level 2</b>	Emotions • Energy • ANAC <b>level 2</b>	Social Choice • AGT II • Learning III (11h40) <b>level 2</b>
	Social Networks I <b>level 0</b>	<b>level 0</b>	ACM Talk (10h40 - ...) <b>level 0</b>
12h10	TaGa I <b>level -1</b>	CGT III <b>level -1</b>	<b>level -1</b>
Lunch	Poster session <b>level 0</b>	Poster session <b>level 0</b>	Community meeting (13h) <b>level 0</b>
	Demos <b>level -1</b>	Demos <b>level -1</b>	<b>level -1</b>
14h40	Argumentation and Negotiation <b>level 3</b>	H&A III • Mechanism design I <b>level 3</b>	
	Teams • TaGa II <b>level 2</b>	Logic II • TaGA III <b>level 2</b>	
	Adversarial Search • Crowdsourcing <b>level 0</b>	<b>level 0</b>	
16h00	Information II <b>level -1</b>	AoSE II <b>level -1</b>	
16h30	Keynote: Couzins <b>level 0</b>	Keynote: Luck <b>level 0</b>	
17h30			

ons) who have published  
 id computer science in the  
 es of age 45 or less at the  
 straint. The amount of the  
 up to USD 2,500 to attend

# Un agent...

- nom masculin
  - (latin médiéval *agens*, participe présent de *agere*, agir)
  - Corps, substance, force qui détermine quelque chose, qui est à l'origine d'un phénomène ou d'un processus ; cause, facteur : Les enzymes sont les agents des fermentations.
  - Personne qui exerce une action d'une certaine sorte, qui joue un rôle déterminant dans la production d'un fait humain ou social ; cause, moteur : Il a été l'agent de nombreuses réformes sociales.
- Une entité autonome qui agit dans un environnement

# Une définition parmi d'autres...

- Wooldrige [08] Un agent est un système informatique situé dans un environnement, capable d'actions autonomes dans cet environnement afin d'atteindre des objectifs prédéfinis.
- Ferber [95]: On appelle agent une entité physique ou virtuelle
  - a. qui est capable d'agir dans un environnement,
  - b. qui peut communiquer directement avec d'autres agents,
  - c. qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser),
  - d. qui possède des ressources propres,
  - e. qui est capable de percevoir (mais de manière limitée) son environnement,
  - f. qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune),
  - g. qui possède des compétences et offre des services,
  - h. qui peut éventuellement se reproduire,
  - i. dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle recoit.

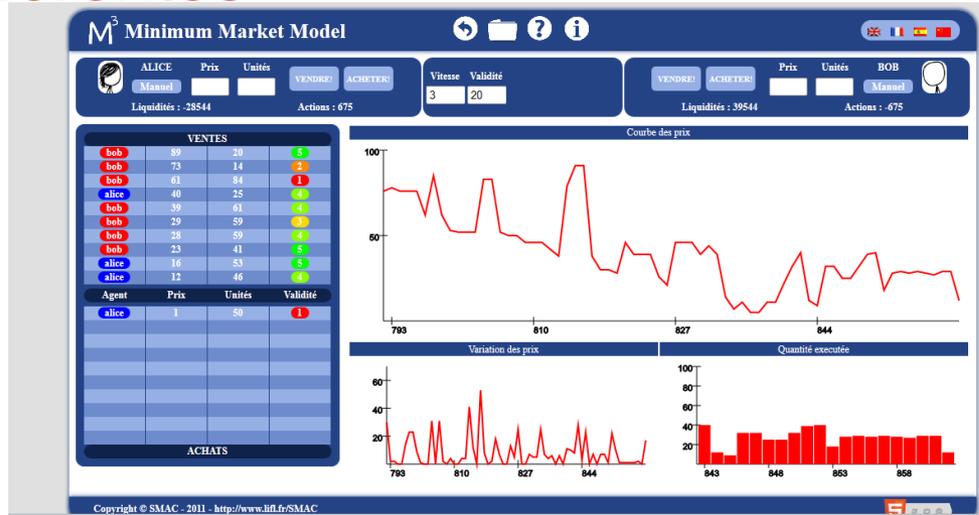
# Un Système...

- nom masculin
  - (bas latin *systema*, du grec *sustêma*, composition)
  - Ensemble organisé de principes coordonnés de façon à former un tout scientifique ou un corps de doctrine : Système philosophique.
  - Ensemble d'éléments considérés dans leurs relations à l'intérieur d'un tout fonctionnant de manière unitaire : Le système nerveux. Les différents systèmes politiques.
  - Ensemble de procédés, de pratiques organisées, destinés à assurer une fonction définie : Système d'éducation.
  - Moyen, plan employé pour obtenir un résultat : Le système de défense de l'accusé. Un bon système pour faire fortune.
  - Appareillage, dispositif formé de divers éléments et assurant une fonction déterminée : Un système de fermeture. Système optique.
  - Société considérée comme un ensemble structuré et rigide : Entrer dans le système.
  - Tendance à penser et à agir selon un ensemble de valeurs rigide et dogmatique : Il y a une part de système dans sa défense.

# Un Système multi-agents...

- Un ensemble d'entités autonomes interagissant dans un environnement

# Un ensemble d'entités autonomes interagissant dans un environnement... avec des objectifs et des problématiques très différentes



# Le cinema et la television offrent une application visible au grand public

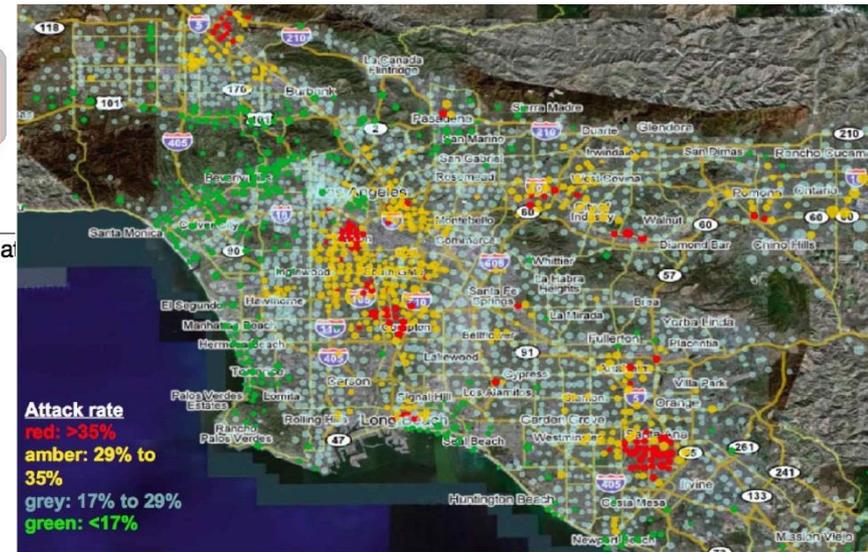
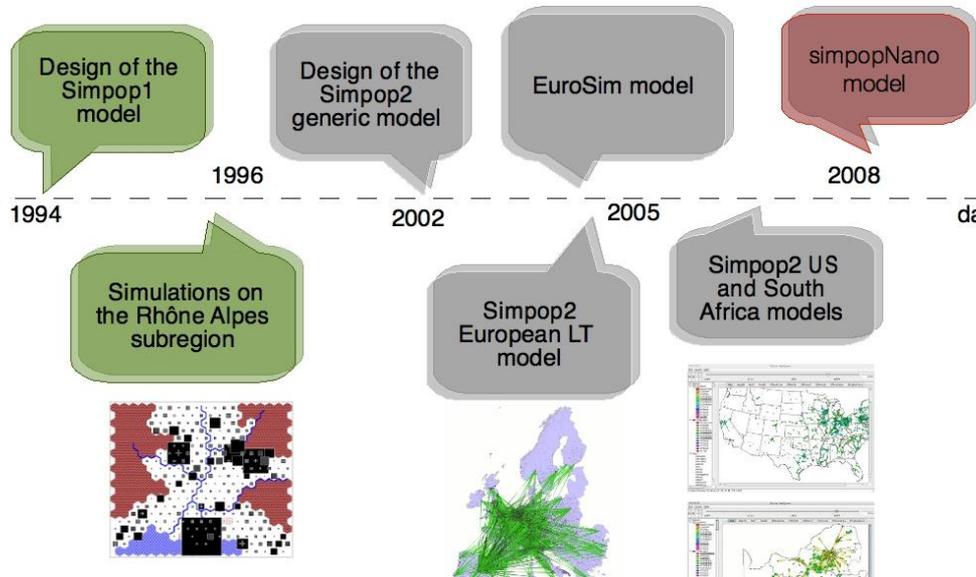
- <http://www.massivesoftware.com/>

Objectif: le réalisme individuel et collectif à court terme



# Les sciences sociales (étude de phénomènes complexes) sont la principale application de la simulation

- Simpop [Pumain 1994+] EpiSim [Epstein 07]
- Terra dinamica [2012+]



# Les jeux sont particulièrement adaptés au SMA de part leur contexte (interaction joueur/PNJ)

- <http://aigamedev.com/open/highlights/top-ai-games/>



Objectifs divers (efficacité, réalisme, ...)



# La finance a utilisé très tôt des « agents autonomes » qui ont un impact réel sur l'économie

- <http://atom.univ-lille1.fr/js/>

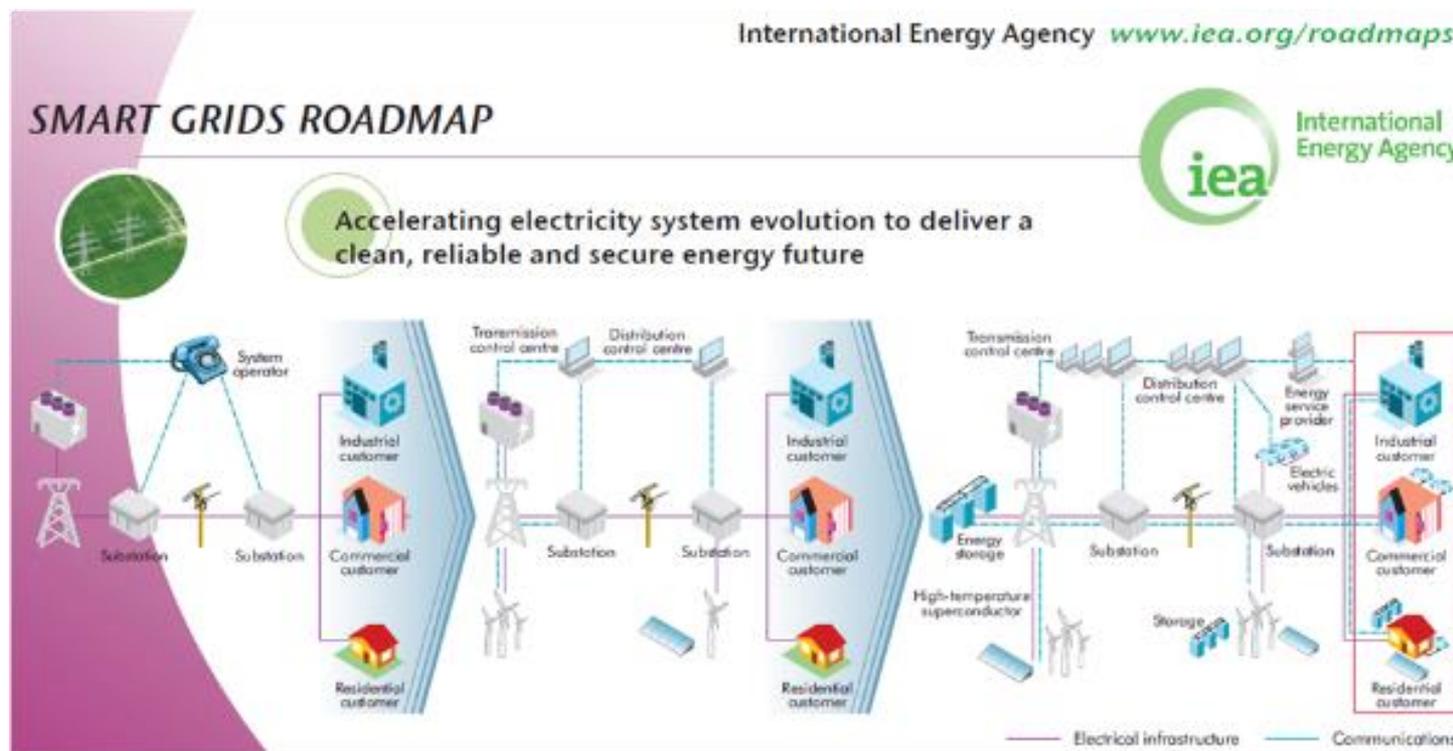
Objectif individuel: efficacité (profit individuel)

Objectif du système: efficacité (surplus global)

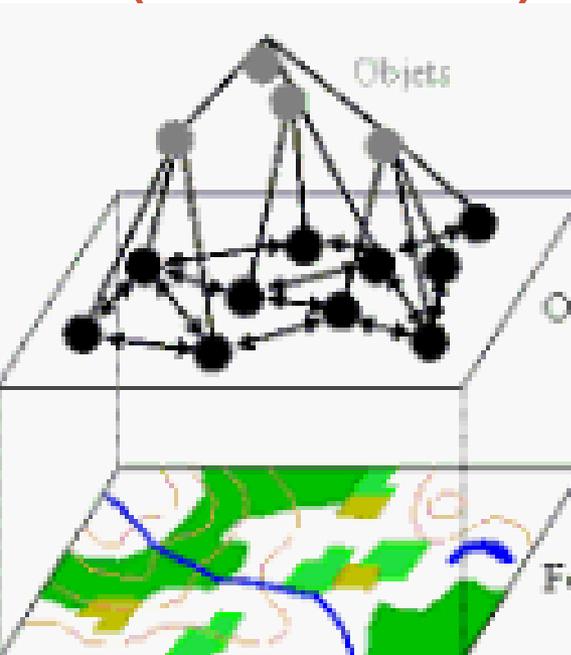


# Les smart-grids, les véhicules électriques, les objets connectés, ... doivent tous interagir et se coordonner

- Transmission plus ou moins limitée des préférences
- Objectif global et individuel
  - Chaque agent maximise son profit
  - Comment concevoir le protocole d'interaction pour que le résultat final soit « bon »?



# L'IGN utilisent des agents pour construire (certaines) cartes



*Vers une prise en compte  
dans les modèles AGE*

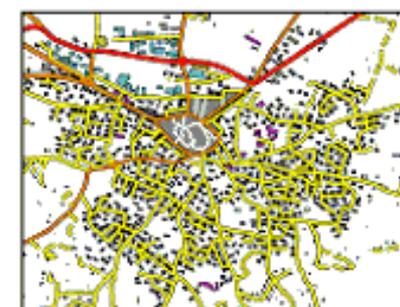
Objectif global commun



*Données initiales,  
symbolisation au 1:15 000*



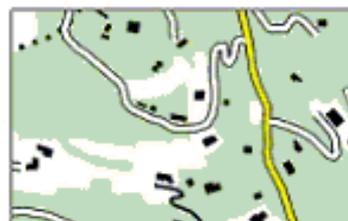
*Données initiales,  
symbolisation au 1:50 000*



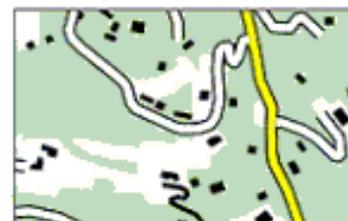
*Après généralisation,  
symbolisation au 1:50 000*

Résultats obtenus fin 2000 avec le prototype issu du projet européen AGENT : données BD Topo généralisées pour le 1 : 50 000 (ville de Trets, région d'Aix-en-Provence)

Généralisation au 1 : 25 000



Généralisation au 1 : 35 000



Généralisation au 1 : 50 000



Résultats obtenus avec le prototype CartACom issu de la thèse de Cécile Duchêne.

# Les ACA sont de plus en plus utilisés pour communiquer avec les utilisateurs

Anna peut vous aider

Bien plus de...  
Tou...  
CUISI...  
C...  
E...  
in...  
Il...  
P...  
E...  
A...  
A...  
Pr...  
C...  
C...  
S...  
M... Terminé

Vous me demandez: **cuisine**

Si vous êtes intéressé par nos cuisines, vous pouvez consulter la page internet qui vient de s'ouvrir.

validez

fermer la fenêtre

© Inter IKEA Systems B.V. 1999 - 2008

categories/departments/kitchen?dinsight=2847&kwsl=18234965

☆ | | Google

Mon compte  
Nos magasins  
IKEA FAMILY  
Nos services

Chercher  OK

Vente à Distance Newsletter IKEA

Besoin d'aide? Cliquez ici

Salon | Chambre | Cuisine | Enfant | Textiles | > Tous nos univers

**DES CUISINES POUR TOUS**

NOUVELLE CUISINE

1 LAISSEZ-VOUS INSPIRER 2 3 4 5

# Contrôle de l'agent (Type)

Reactif

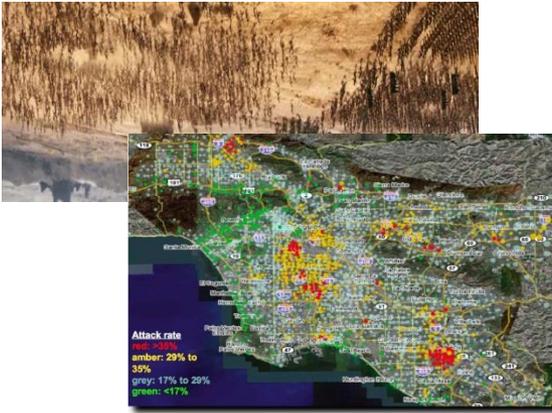
Cognitif

Apprenant

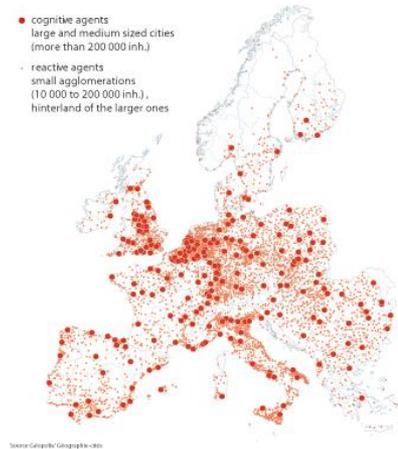
Shelling  
Massive

SimPop  
Bot

AIBO  
BW



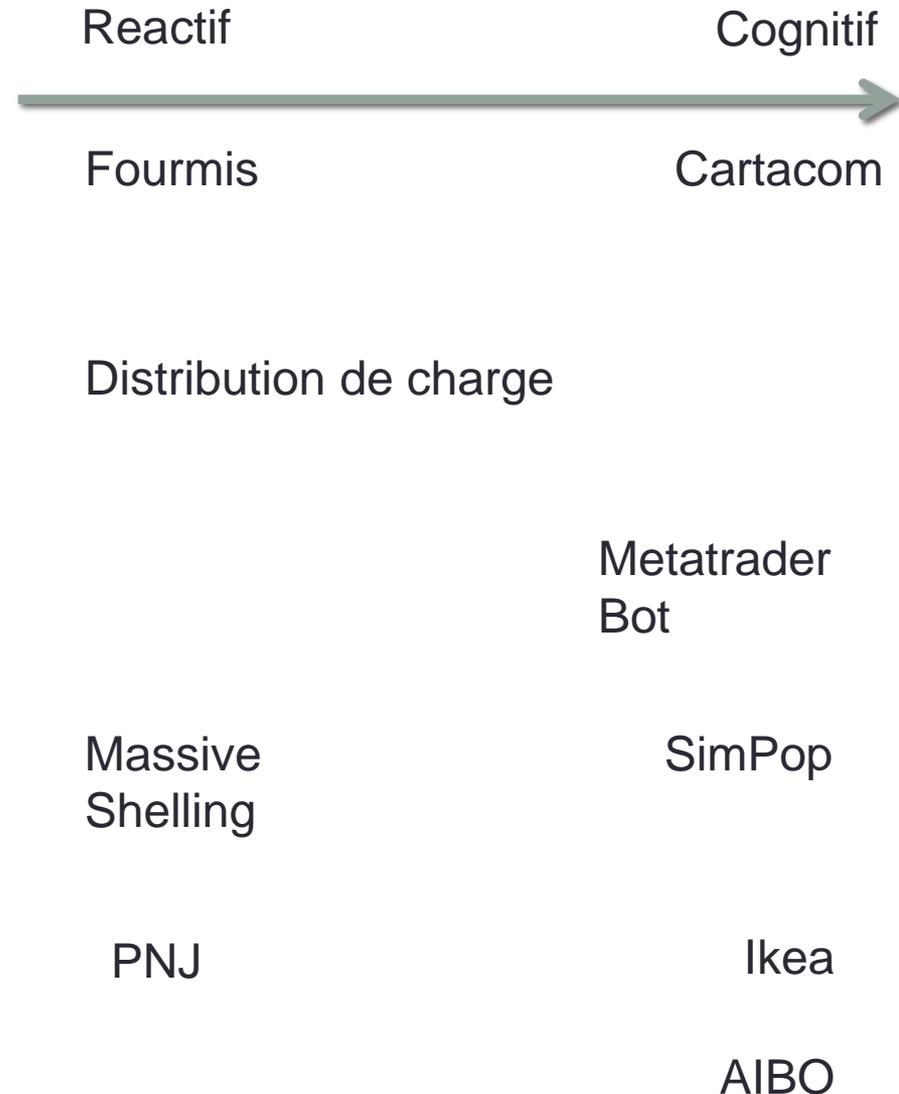
- cognitive agents  
large and medium sized cities  
(more than 200 000 inh.)
- reactive agents  
small agglomerations  
(10 000 to 200 000 inh.),  
hinterland of the larger ones



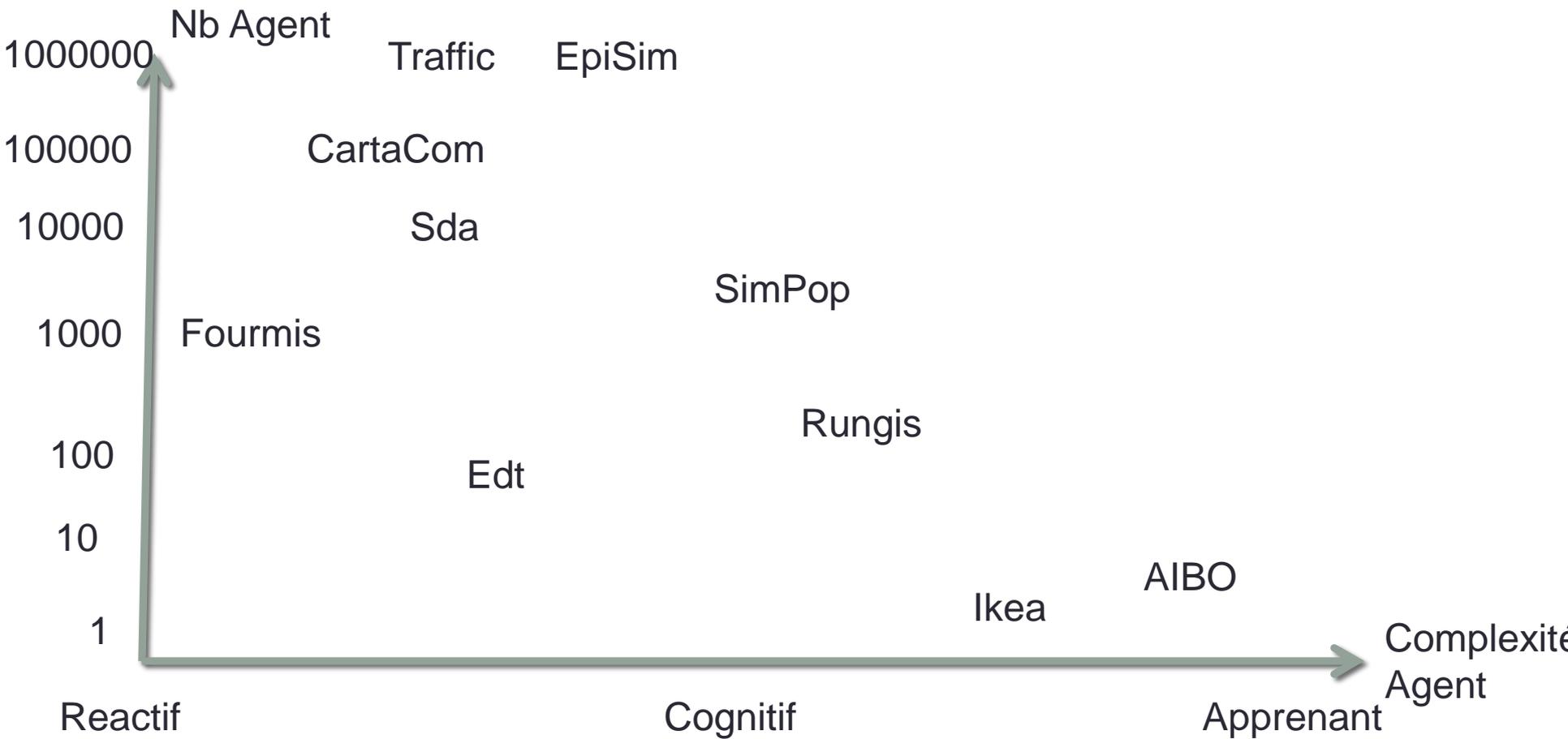
- « intelligent » dans Agent Intelligent provient de [Wooldrige et Jennings 95]
  - Reactivity => facile
  - Proactivity => facile
  - Les deux: complexe
  - + Sociability

# Objectif du SMA

- Intelligence Artificielle Distribuée (IAD)
  - Objectif commun
  - Le but du SMA est de résoudre un problème de façon distribuée
- SMA coopératif (MAS)
  - Chaque agent a son propre objectif
  - Les agents se font confiance et peuvent décider collectivement d'un objectif commun
- SMA compétitif
  - Chaque agent à son objectif propre
- Simulation à base d'agents (MABS)
  - L'objectif est de reproduire un phénomène extérieur au SMA
- Agent conversationnel
  - L'objectif est de pouvoir interagir
- Vie artificielle
  - L'objectif est d'observer un phénomène émergent



# Taille / Cognition



# Description d'un SMA

- Introduction
- Description d'un SMA
  - Composants d'un SMA
  - L'intérêt des SMA
- Modélisation d'un SMA
- Conclusion



# Les éléments du SMA

Dans un environnement



Agents: les entités autonomes



Contrôle: comment elles pensent /  
décident de ce qu'elles font



Perception / Senseurs: avec quoi  
et ce qu'elles perçoivent



Actions / Actuateurs: avec quoi et  
ce qu'elles font



Interaction / Communication : les  
échanges et contenus éventuels



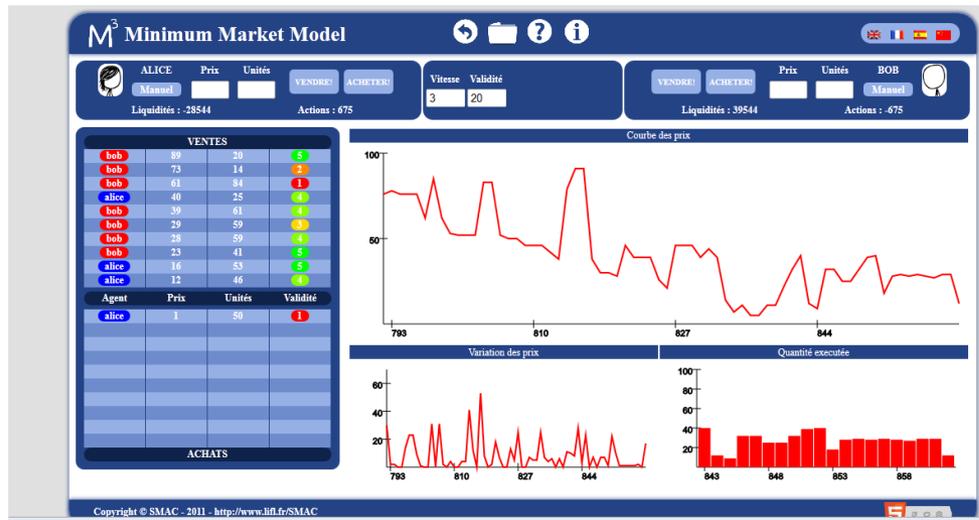
Organisations: les groupes,  
temporels ou permanents



Rôles, Services, Comportements:  
de la sémantique pour décrire ce  
qui est offert et comment



# Les caractéristiques se retrouvent dans toutes les applications



# Pourquoi c'est différent?

# Agent et Objet

- « Objects do it for free, Agents do it because they want to » Wooldrige [09]
- Notion de contrôle sur son comportement
- Notion de réactivité/adaptabilité à l'environnement
- Un thread de contrôle

# SMA et Calcul Distribué

- Notion d'action autonome
- La coordination est dynamique et en ligne alors qu'elle est généralement codée en calcul distribué (pas besoin de négocier, on sait qui va interagir, quand et comment)
- Chaque agent a ses objectifs lors de l'interaction.

# SMA et IA

- Les branches de l'IA s'intéressent aux composants de l'intelligences, les SMA à leur intégration
- L'IA s'intéresse peu aux dimensions sociales: interactions, négociation, communication, organisations, ...

# SMA et sciences sociales

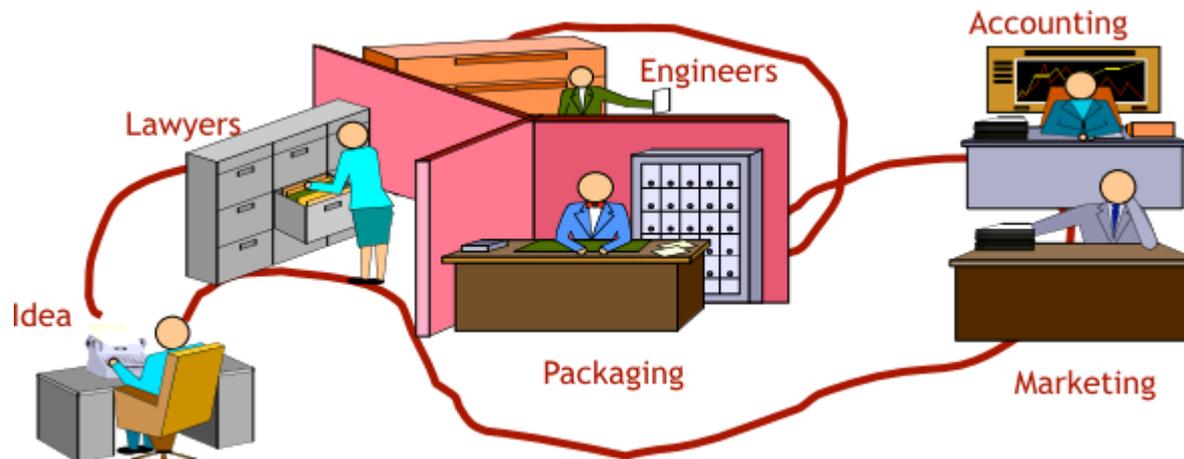
- Les sciences sociales s'intéressent (peu) à la complexité et à la façon de résoudre les problèmes des acteurs
- Les sciences sociales étudient des agents (généralement!) humains. Les objectifs et donc les comportements/interactions/organisations d'agents virtuels peuvent être très différents.

# Pourquoi les utiliser?

Drogoul [05]

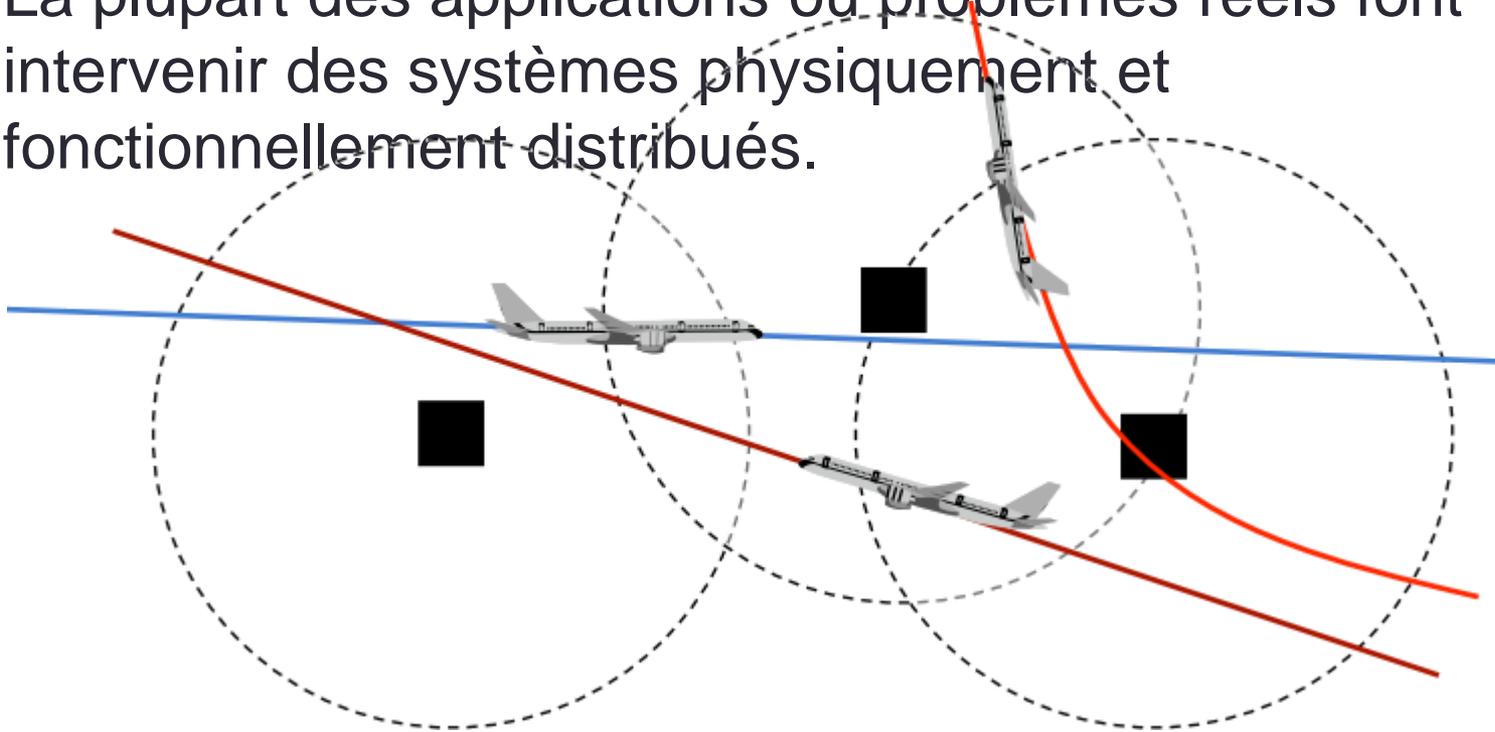
# Distribution fonctionnelle des activités humaines

- Distribution fonctionnelle dans les activités humaines (comme la conception d'un produit, par exemple) :
  - Points de vue et langages (ontologies) variés
  - Décomposition du problème en fonction des spécialités



# Distribution physique des acteurs

- Distribution physique dans de nombreuses situations:
  - Problèmes intrinsèquement distribués : réseaux, contrôle aérien, etc..
  - Robotique
- La plupart des applications ou problèmes réels font intervenir des systèmes physiquement et fonctionnellement distribués.

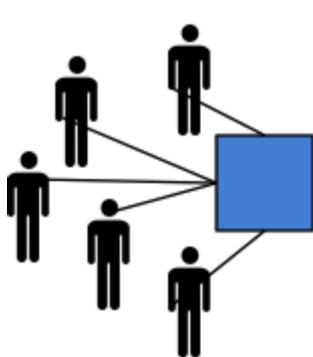


# Complexité croissante du développement

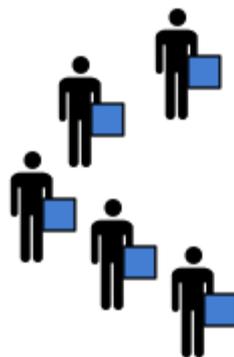
- Complexité des développements logiciels nécessite une modularité et une inter-opérabilité croissantes

# Évolution de l'informatique

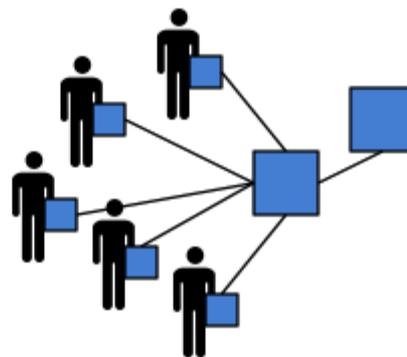
- Evolution de l'informatique vers le « Pervasive Computing » (informatique diffuse)
- Nouveaux « systèmes » informatiques
  - Construits de façon ascendante,
  - Logiciels et matériels hétérogènes,
  - Aucune contrôle global possible,
  - Configurations et fonctionnalités évolutives,
  - Situés dans des environnements dynamiques



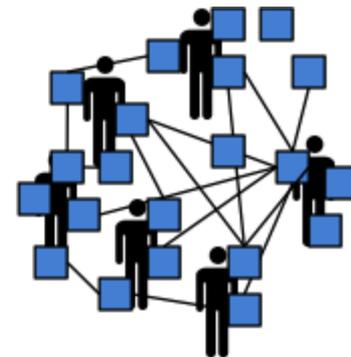
1950: Mainframe



1980: Micro-ordinateur



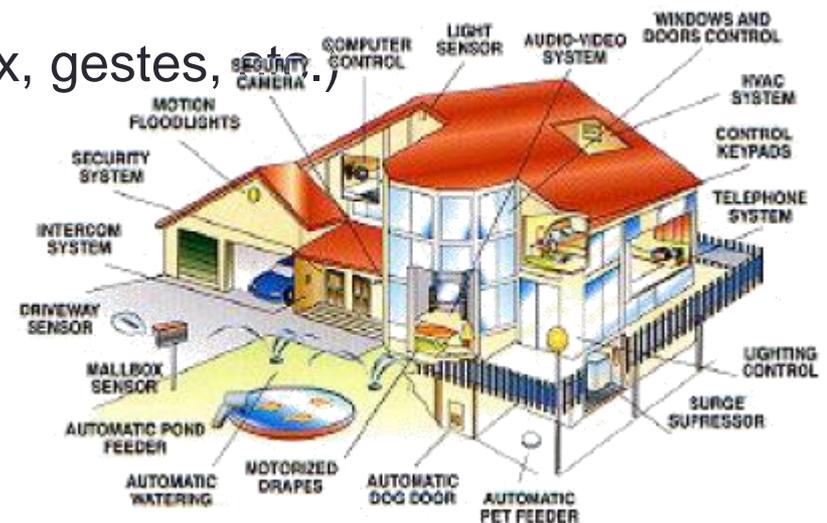
1990: Internet



200?: Informatique diffuse

# Évolution des objets usuels

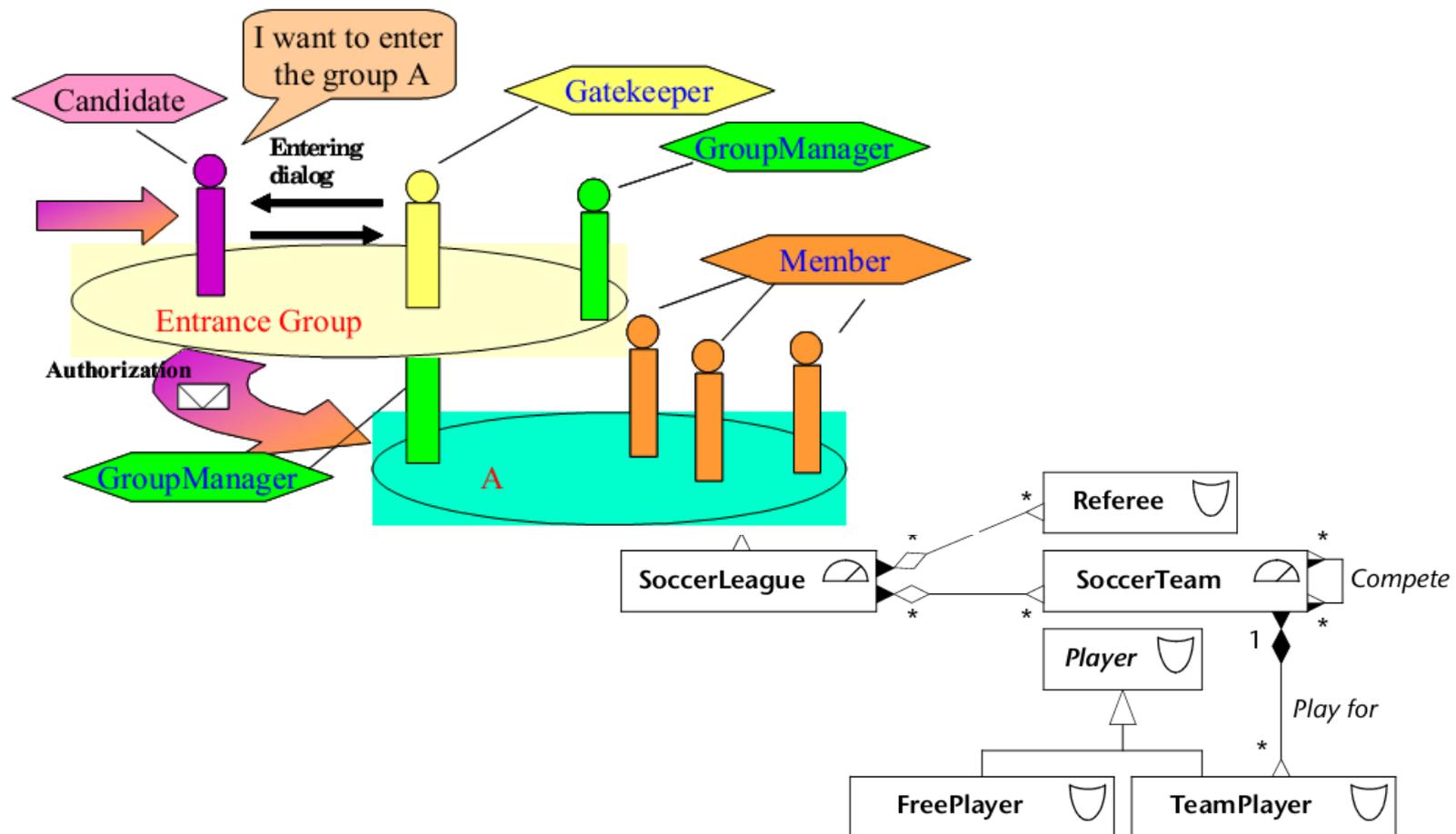
- Les objets « usuels » (électroménager, outils, vêtements, lunettes, etc.) seront « augmentés » avec des senseurs, actuateurs, microprocesseurs et leurs logiciels embarqués
  - Mobiles ou pas
  - Communicants (Wifi, BlueTooth)
  - (Semi-) autonomes (par nécessité)
  - Interfaces utilisateur avancées (voix, gestes, etc.)
  - Sensibles au contexte



# Pour le chercheur, une vision globale

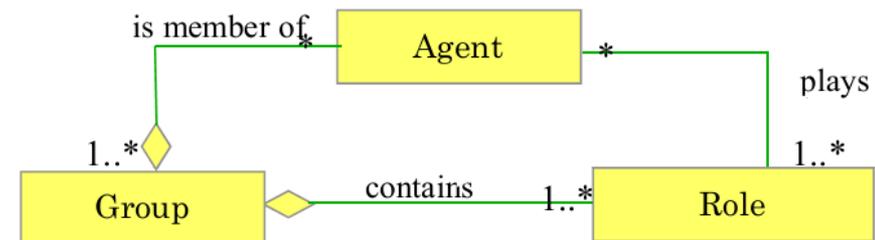
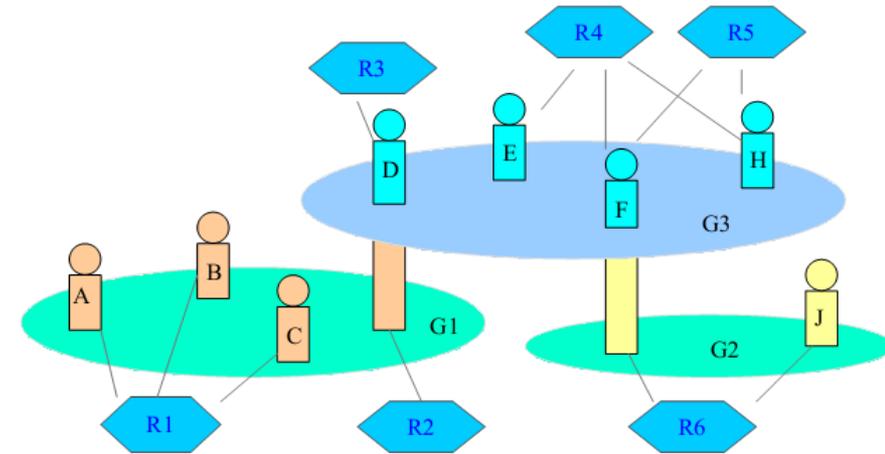
- Intégration des problématiques classiques de l'IA
  - Planification
  - Représentation des connaissances
  - Apprentissage
  - Perception
  - Action
  - Communication
- Et quelques nouvelles
  - Organisation
  - Émergence

# Modélisation de SMA

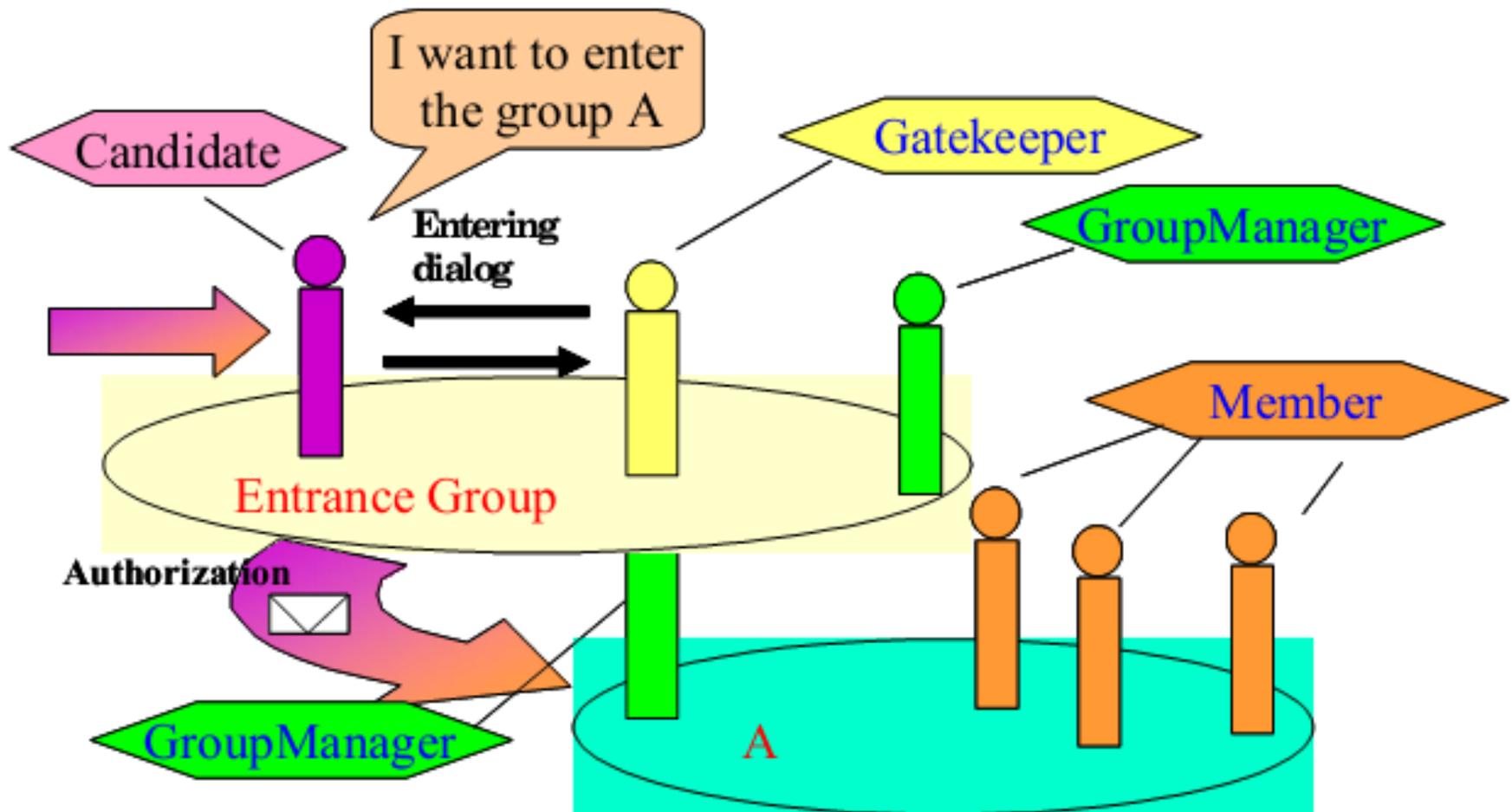


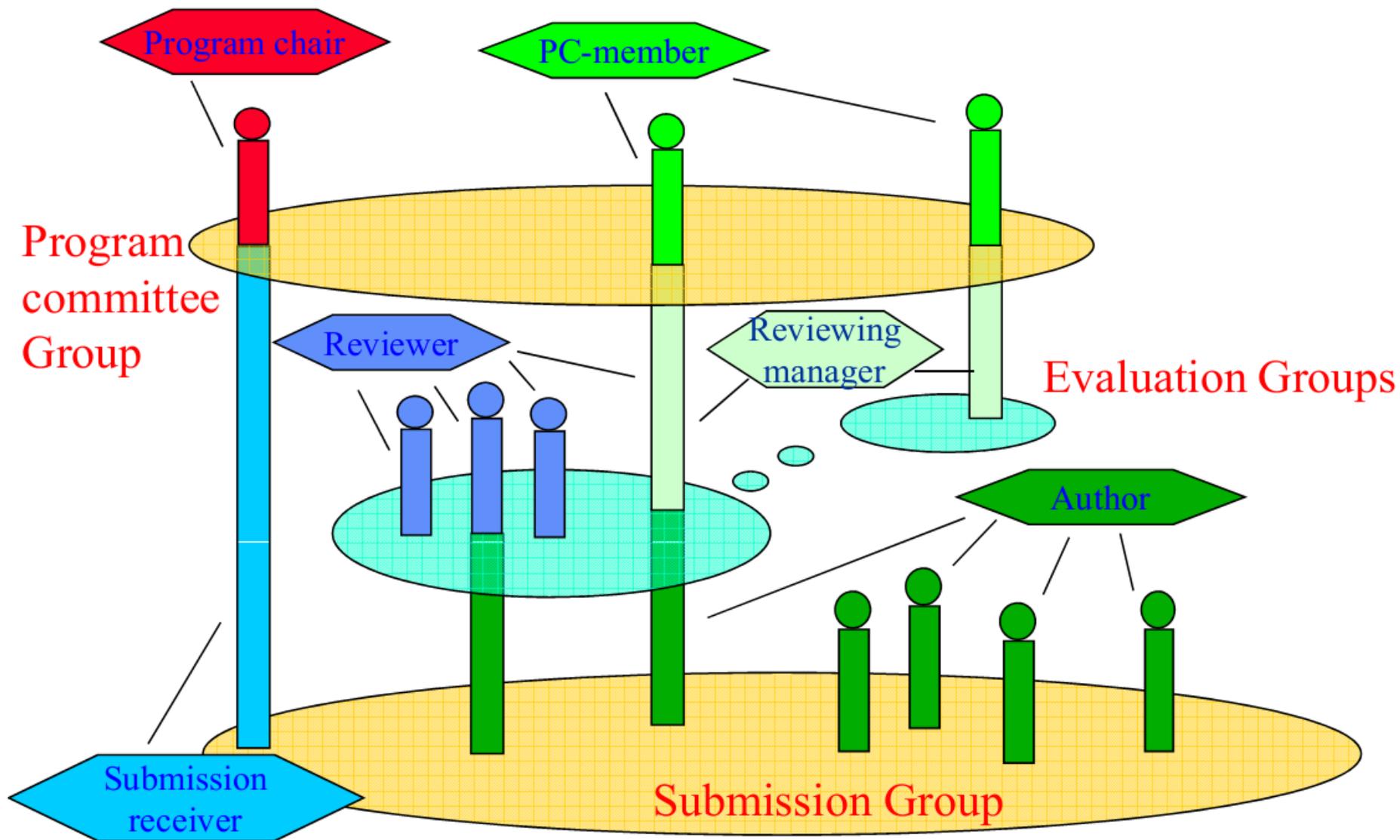
# Modèle AGR [Ferber 2002+]

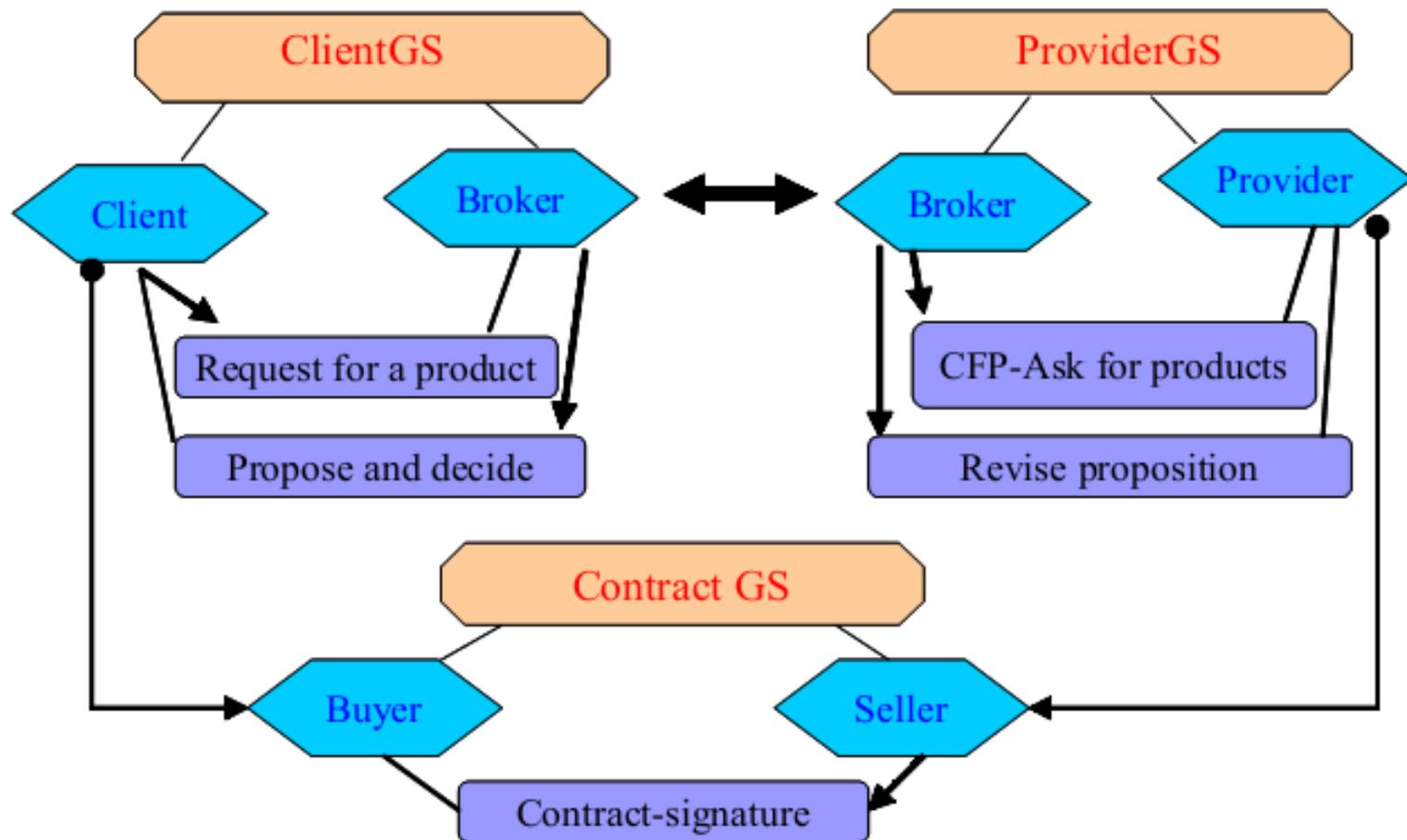
- **Agent:**
  - Une entité autonome et communicante
  - Joue des rôles dans des groupes. Un agent peut avoir plusieurs rôles et être membre de plusieurs groupes.
- **Rôle:**
  - la représentation abstraite de la fonction d'un agent.
  - Les rôles sont locaux aux groupes. Un rôle peut être joué par plusieurs agents
- **Groupe:**
  - un ensemble d'agents partageant une caractéristique
  - Deux agents ne peuvent communiquer que s'ils sont membres du même groupe

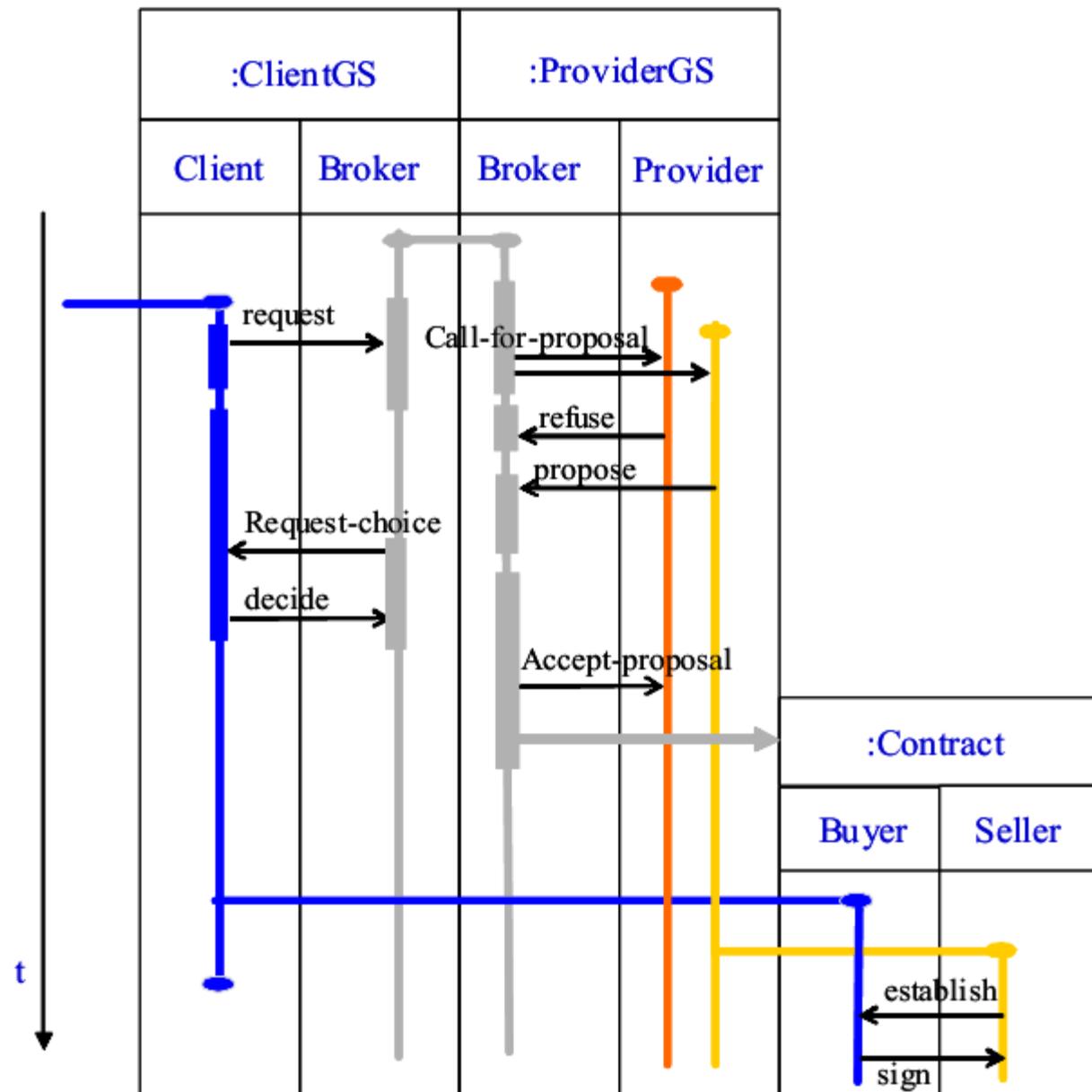
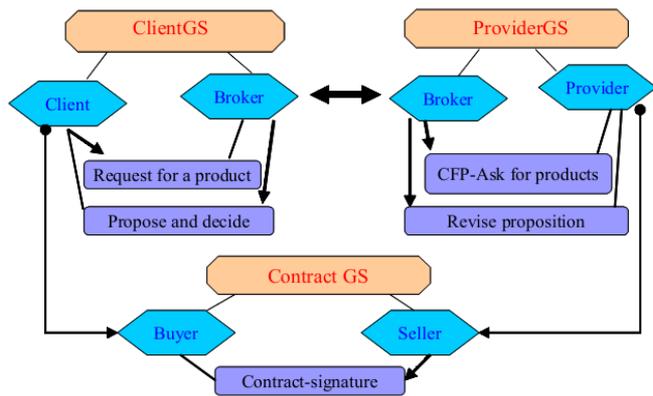


# Modèle AGR









# MadKit

- <http://www.madkit.org/>
- Plateforme très légère en Java
- Modèle AGR pour gérer la communication et les interactions

The screenshot displays the MadKit desktop environment. On the left, a file explorer shows a directory structure including 'autoload', 'scripts', 'seditfiles', 'schemefiles', 'interaction.scm', 'jessfiles', 'test and others', 'scriptlib5.1.clp', 'warbot', 'jesstravel', 'madkitlib.clp', 'notagent.jess.files', 'marketorg', 'madkit.standard', 'PingPong.clp', 'warbot', 'jeditfiles', 'jEditSchemeEval.s', 'pythonfiles', 'test.py', 'PythonAgent.py', 'tutorial', 'essai1.py', 'LaunchGuiLayout.', 'LaunchGuiLayout.', 'LaunchPingPong.p', 'PingPong.py', 'beanshellfiles', 'formalisms', 'jessagents.jar', 'beanshellagents.jar', 'system.jar', 'EditorAgent', 'JEditAgent', 'AgentLister', 'GroupObserver', 'DirectoryAgent', 'GroupMessageTracer', 'MinimalChatAgent', 'WebBrowserAgent', 'OrganizationTracer', 'MemoryMonitorAgent', 'toolagents.jar', 'turtledemos.jar', 'scheme.jar', and 'dynamic-bees.jar'.

In the center, a 'GroupObserver-2' window shows a tree view of agents: public, communications, jedit, system, ping-pong, python, TERMITES, and communities. Below this, a 'Messages' table is visible with columns for 'Sender', 'Receiver', and 'Message Content'. A 'Memo...' window shows a graph of memory usage, with '1.7988K allocated' and '1.5802K used'.

On the right, a 'jEdit - PingPong.py' window displays the following Python code:

```

# PingPong agent in Python
# Author: J. Ferber
# (c) 2001 Madkit team

from madkit.kernel import Agent
from madkit.lib.messages import StringMessage
import java.util

def activate():
    self.println("PythonAgent ping pong activated")
    # self.pause(1000)
    self.println("looking for a ping pong group")
    if self.isGroup("ping-pong"):
        self.println("Yeah I am coming")
        # self.joinGroup("ping-pong")
    else :
        self.println("I create a new group")
        self.createGroup(1,"ping-pong",None,None)
        self.requestRole("ping-pong","player")

def live():
    other = None
    self.println("looking for a ping pong partner")
    while (other == None) :
        v = self.getAgentsWithRole("ping-pong","player")
        # self.println("OK1")
        for x in v :
            self.println(x.toString())
            if not x.equals(self.getAddress()) :
                other = x
        # self.println(v):
        self.pause(1000)
    self.println("other is : " + other.toString())
    self.sendMessage(other, StringMessage("Balle"))
    for y in range(0,5):
        ans = self.waitNextMessage()
        self.println("OK, my turn " + java.util.Date().toString())
        self.pause(1000)

```

At the bottom, a 'PythonAgent' window shows a list of agents: 'mka: PythonAgent, 423@aquinas:K1023'. Below this, a console window displays the following output:

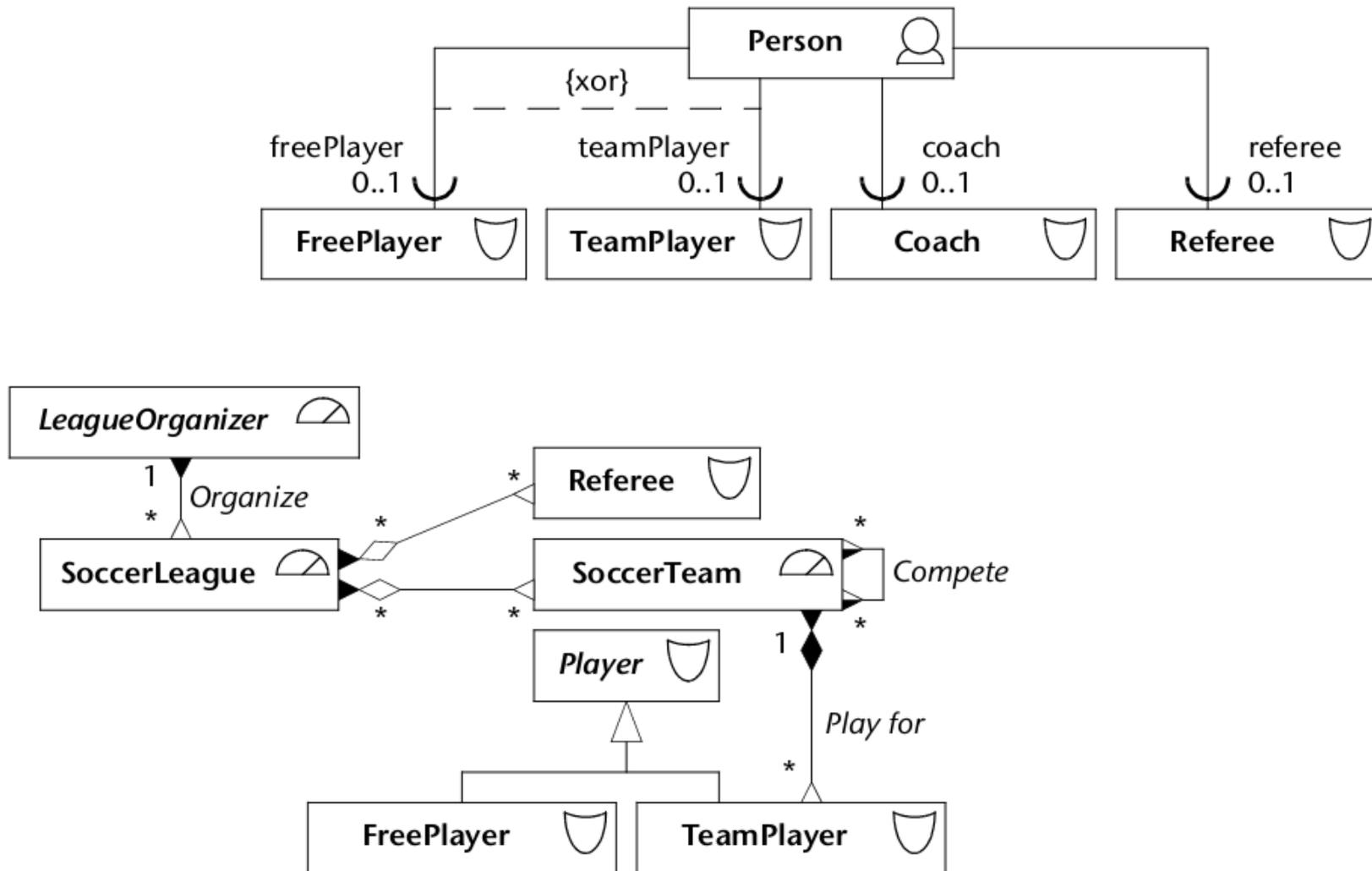
```

Probe on <TERMITES;world>, agents are 0
removing community : travel
<Kernel> : [JEditAgent] trying to launch jEdit
<Kernel> : [JEditAgent] jEdit launched
[warning] options.DragAndDrop.label property not
[warning] options.DragAndDrop.label property not

```

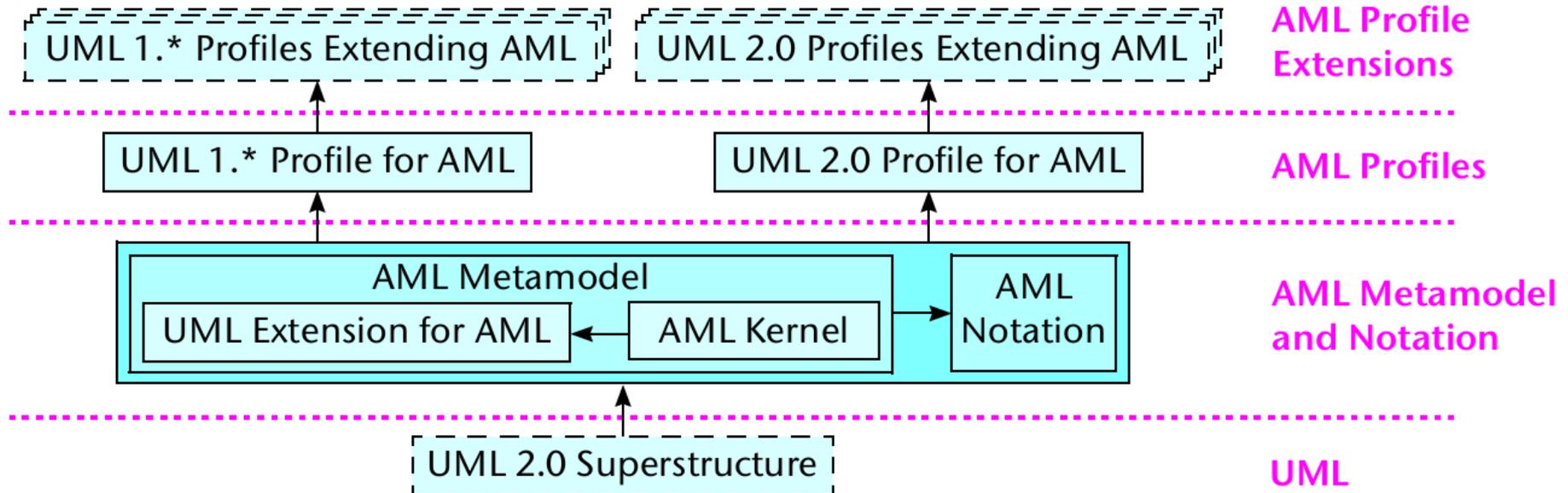
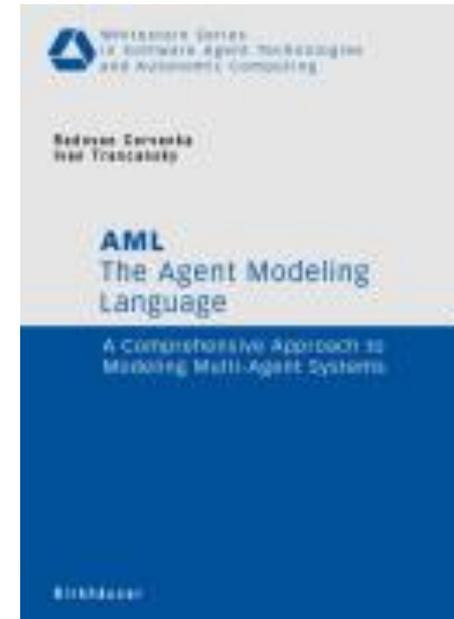
The bottom status bar shows 'Activity Log', '1,1 Top', 'python Cpl1252 none single ins 15Mb/17Mb'.

# AML : Agent Modeling Language

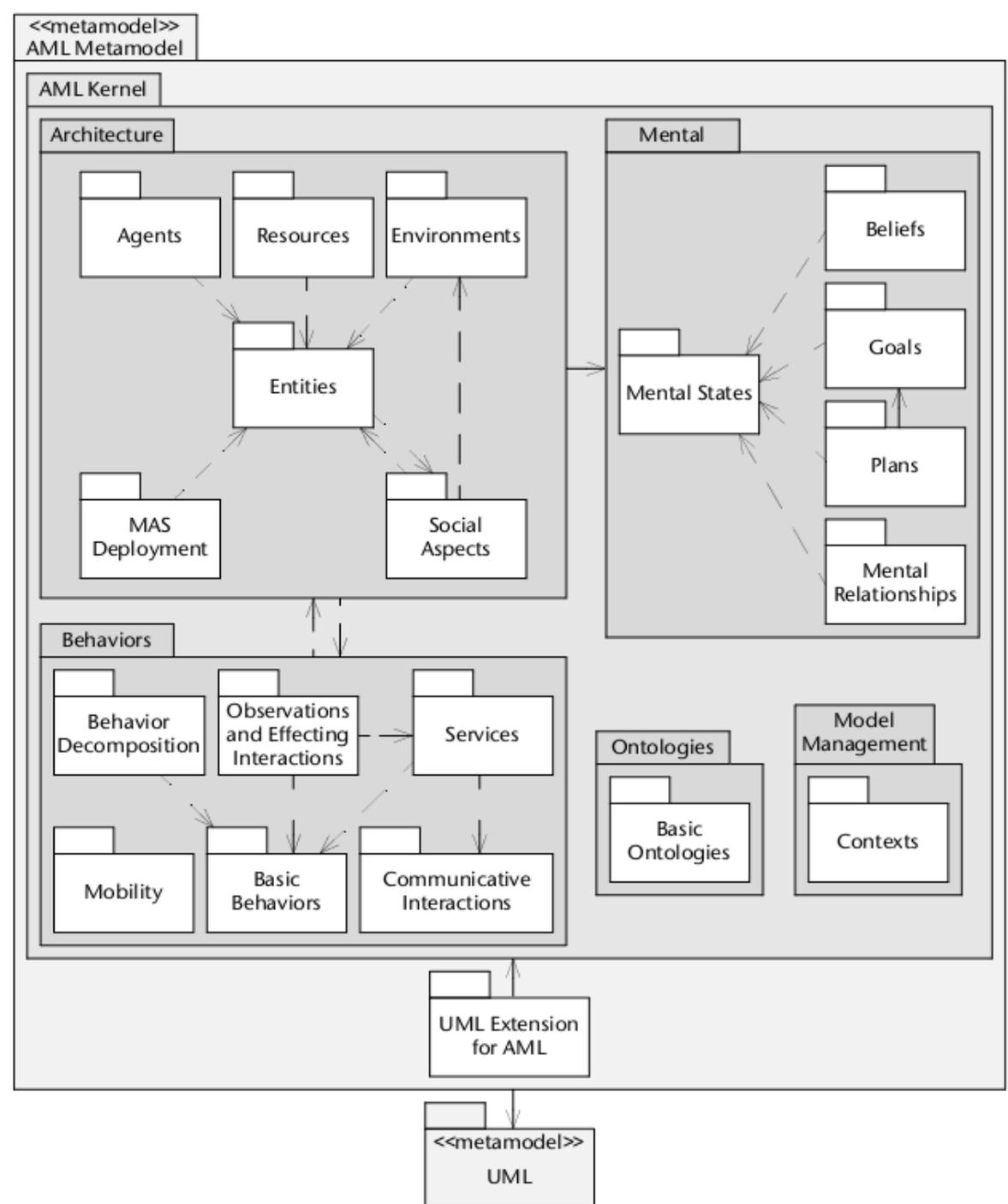


# AML: Un formalisme de modélisation agent

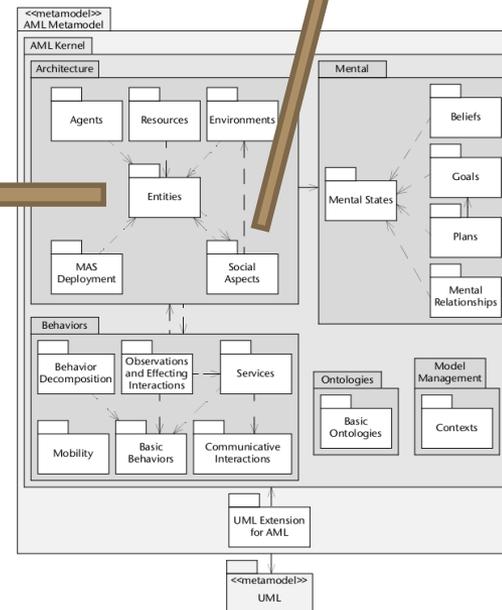
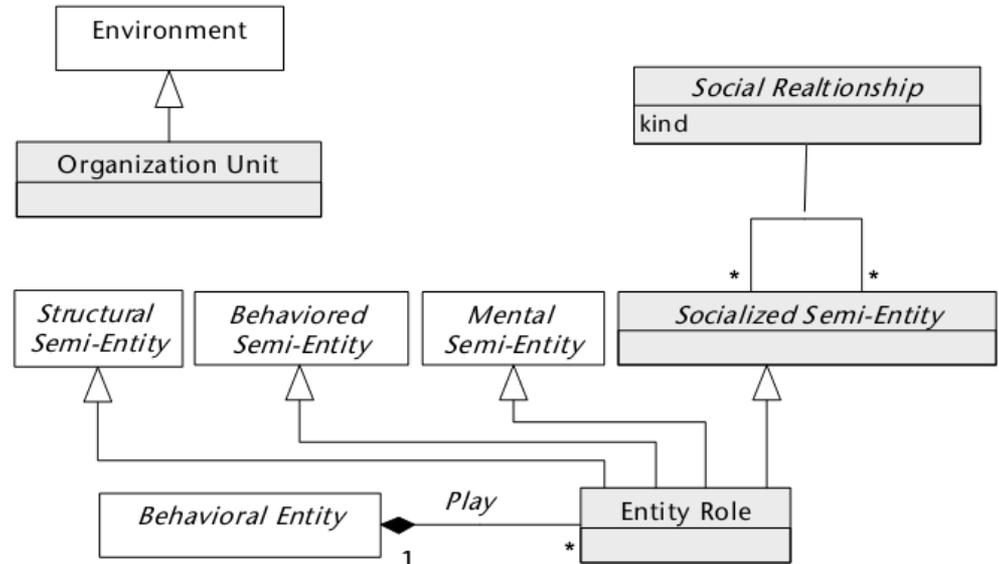
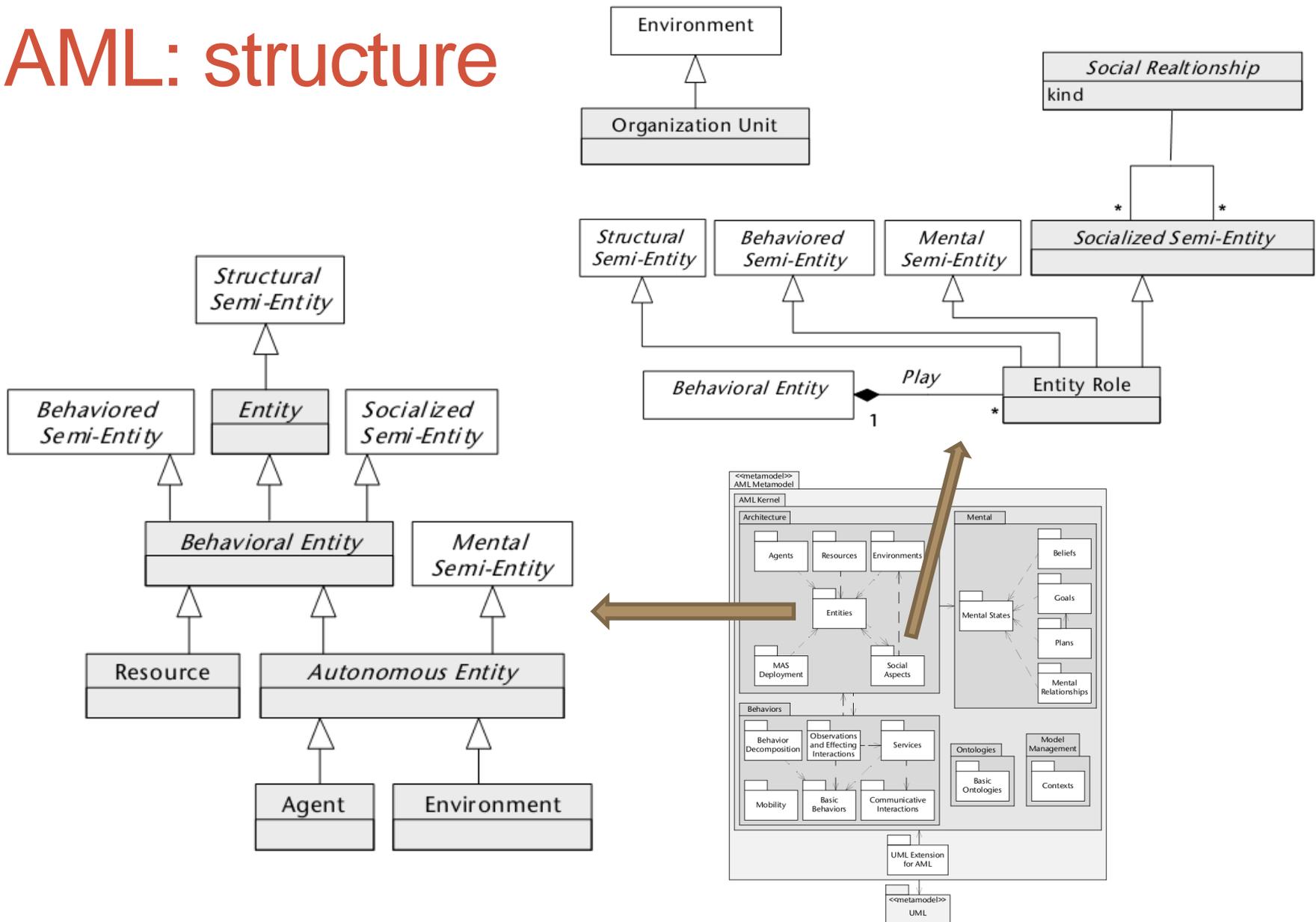
- [Radovan Cervenka, Ivan Trencansky 07]



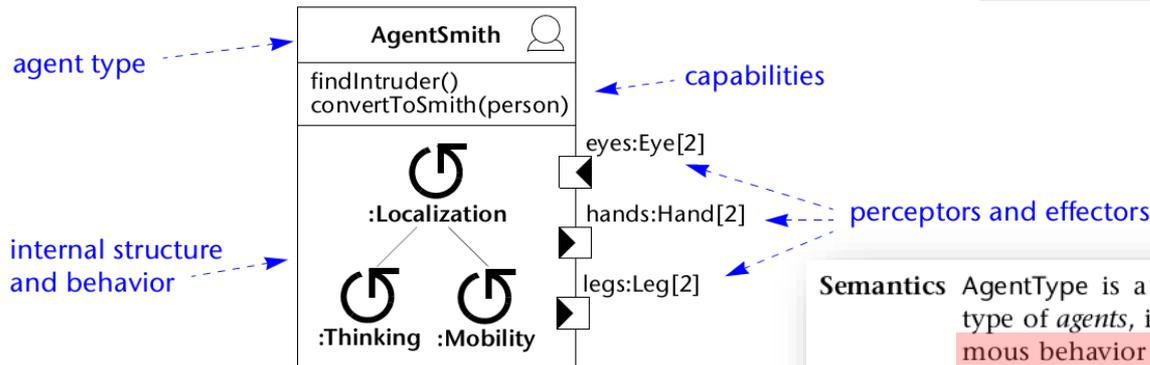
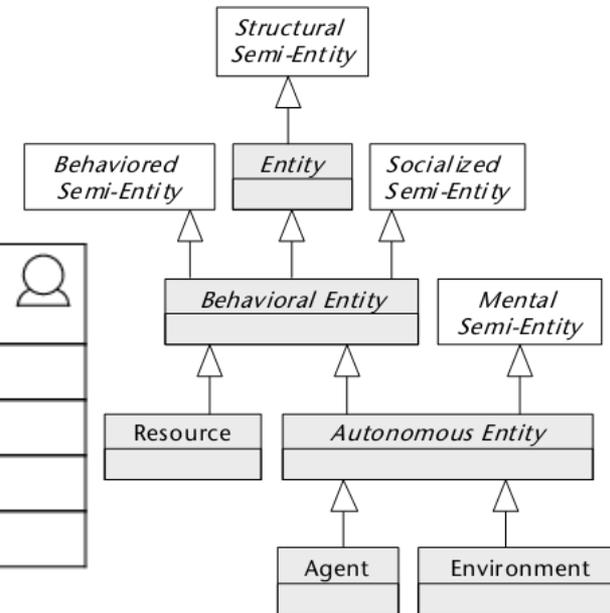
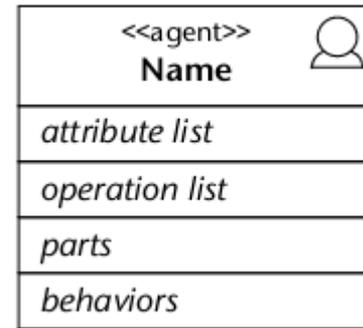
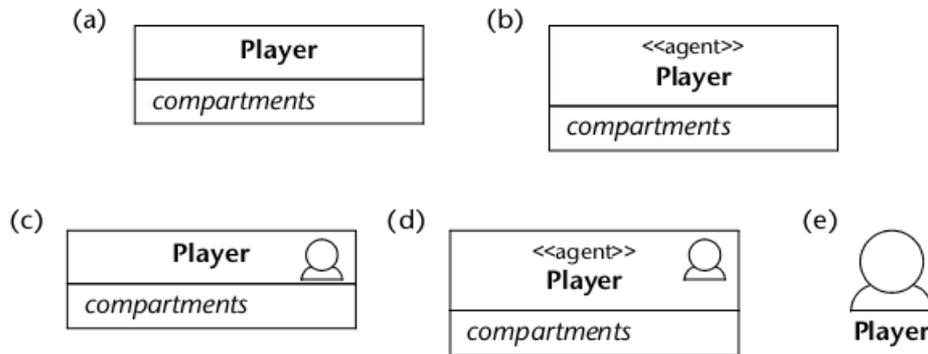
# AML: composants



# AML: structure



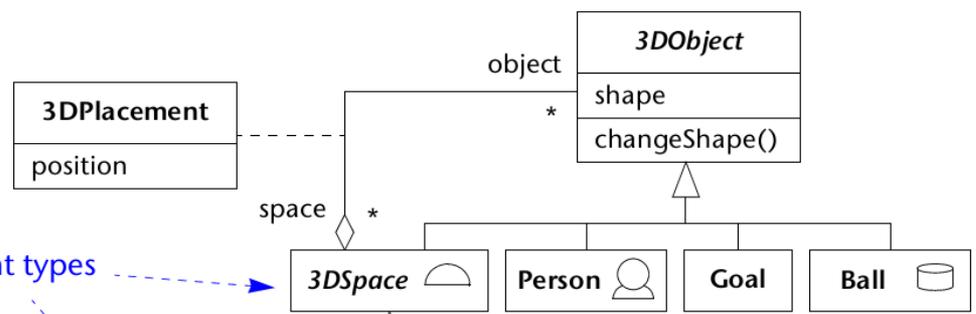
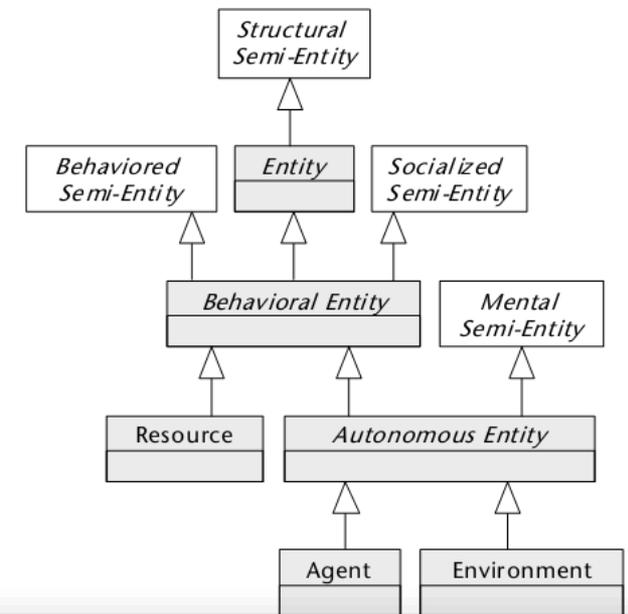
# AML: agents



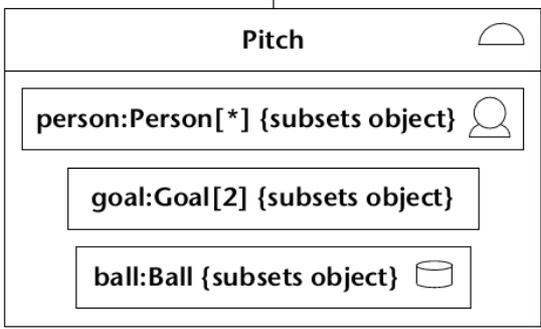
**Semantics** AgentType is a specialized AutonomousEntityType used to model a type of *agents*, i.e. **self-contained entities that are capable of autonomous behavior within their environment**. An agent (instance of an AgentType) is a special object (which the object-oriented paradigm defines as an entity having identity, status and behavior; not narrowed to an object-oriented programming concept) having at least the following additional features<sup>27</sup>:

- ❑ *Autonomy*, i.e. control over its own state and behavior, based on external (reactivity) or internal (proactivity) stimuli, and
- ❑ *Ability to interact*, i.e. the capability to interact with its environment, including perceptions and effecting actions, speech act based interactions, etc.

# AML: ressource et environnement



environment types

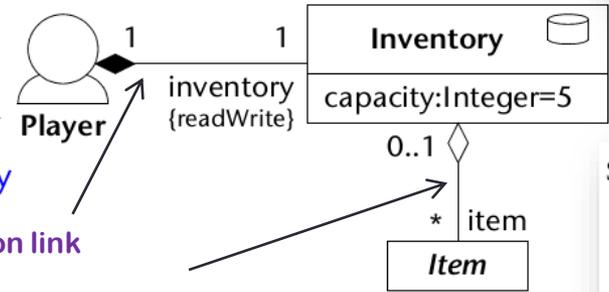


**Semantics** EnvironmentType is a specialized AutonomousEntityType used to model types of *environments*, i.e. the logical and physical surroundings of entities which provide conditions under which those entities exist and function. EnvironmentType thus can be used to define particular aspects of the world which entities inhabit, its structure and behavior. It can contain the space and all the other objects in the entity surroundings, and also those principles and processes (laws, rules, constraints, policies, services, roles, resources, etc.) which together constitute the circumstances under which entities act.

accessing entity

Composition link

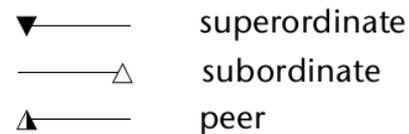
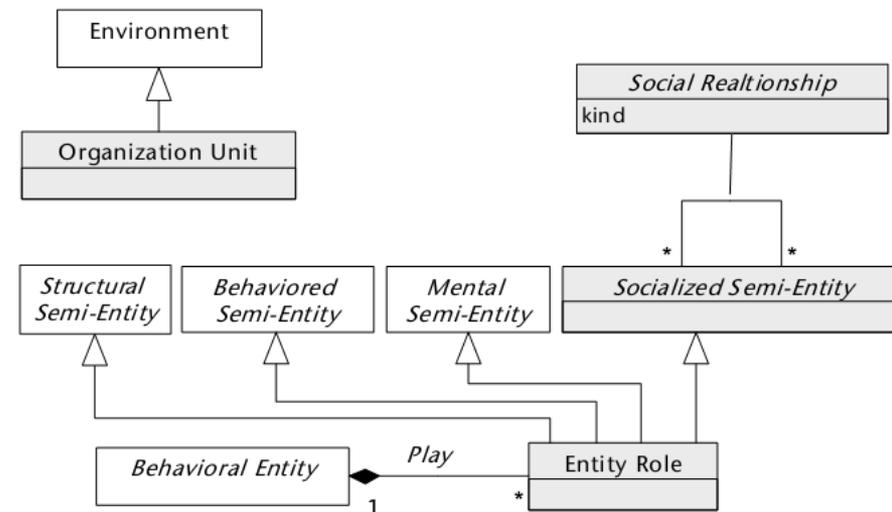
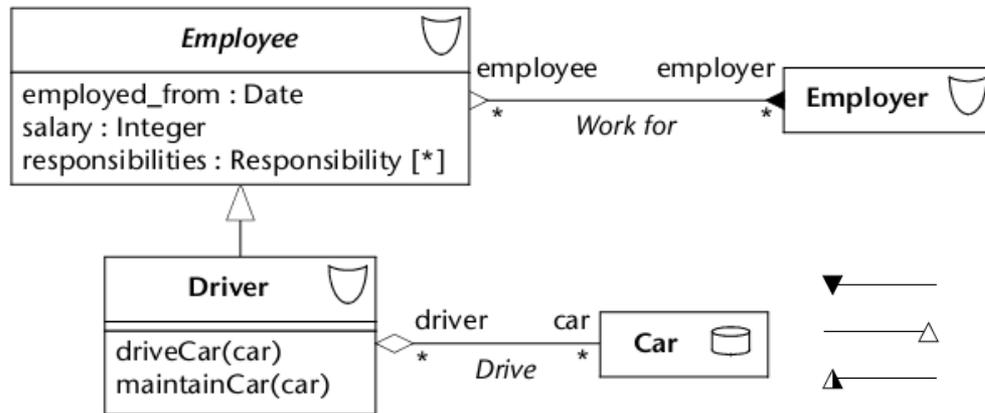
Agregation link



resource type

**Semantics** ResourceType is a specialized BehavioralEntityType used to model types of resources contained within the system<sup>28</sup>. A resource is a physical or an informational entity, with which the main concern is its availability and usage (e.g. quantity, access rights, conditions of consumption).

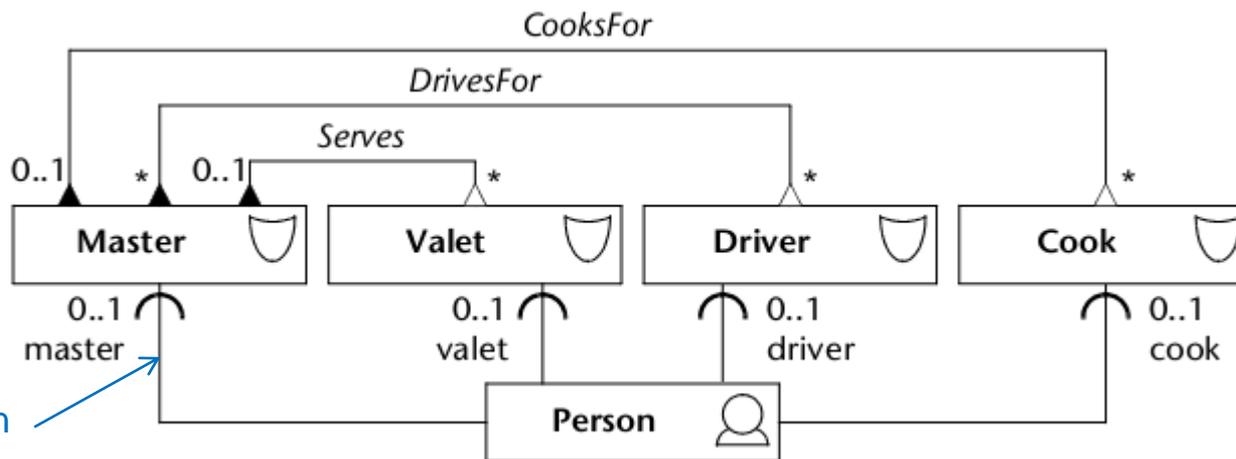
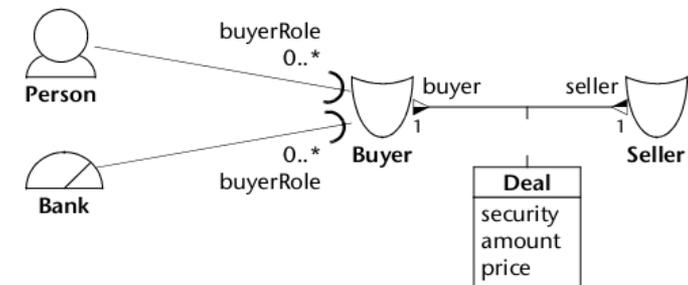
# AML: roles



**Semantics** EntityRoleType is a specialized BehavoredSemiEntityType, MentalSemiEntityType, and SocializedSemiEntityType, used to represent a coherent set of features, behaviors, participation in interactions, and services offered or required by BehavioralEntityTypes in a particular context (e.g. interaction or social).

Each EntityRoleType thus should be defined within a specific larger behavior (collective behavior) which represents the context in which the EntityRoleType is defined together with all the other behavioral entities it interacts with. An advisable means to specify collective behaviors in AML is to use EnvironmentType or Context.

# AML: play association



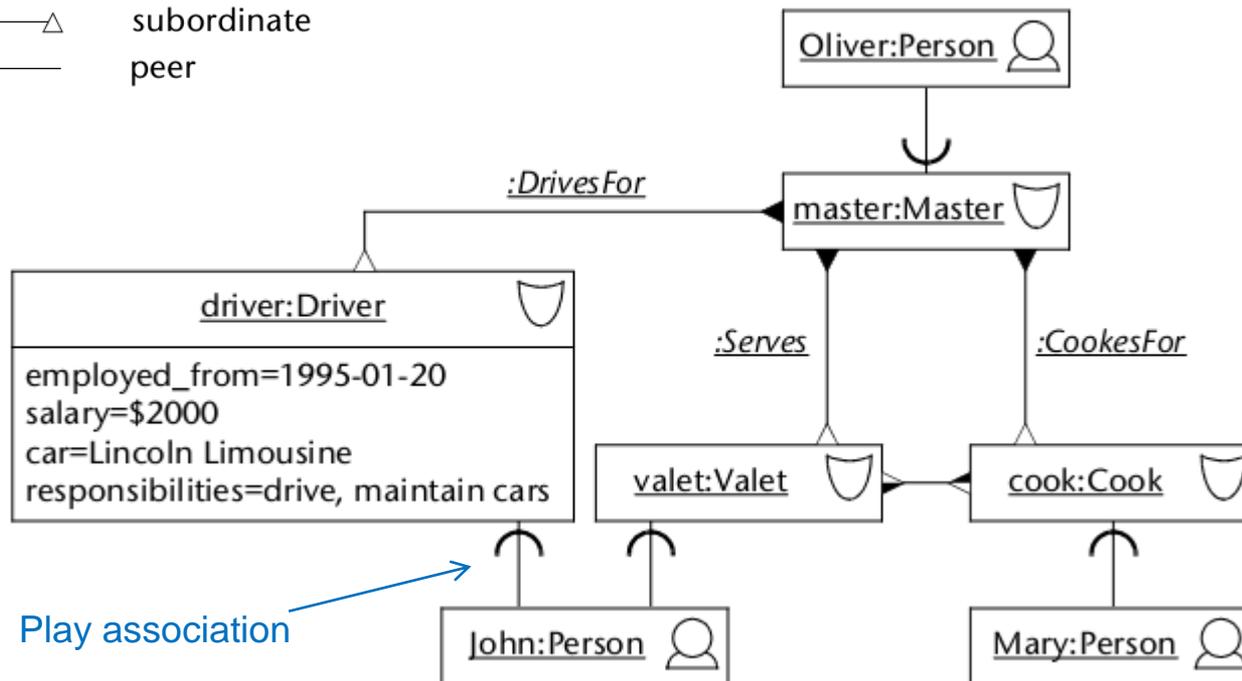
Play association

- ▼ — superordinate
- △ subordinate
- ▲ — peer

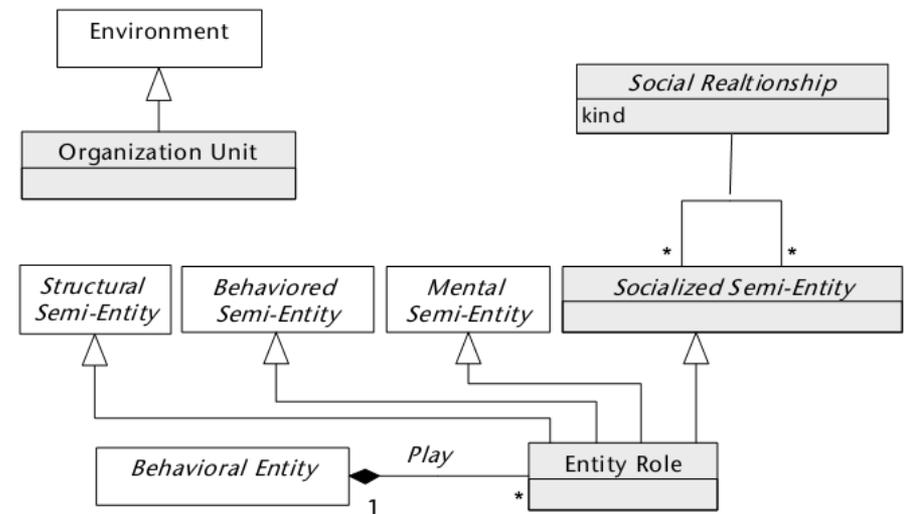
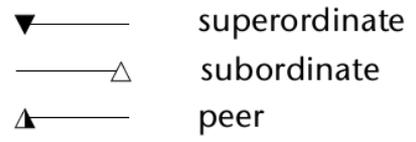
**Semantics** PlayAssociation is a specialized Association (from UML) used to specify RoleProperty in the form of an association end. It specifies that entity roles of a roleMemberEnd's type (which is an EntityRoleType) can be played, i.e. instantiated by entities of the other end type (which are BehavioralEntityTypes).

# AML: roles (instances)

- ▼—— superordinate
- △ subordinate
- ▲—— peer

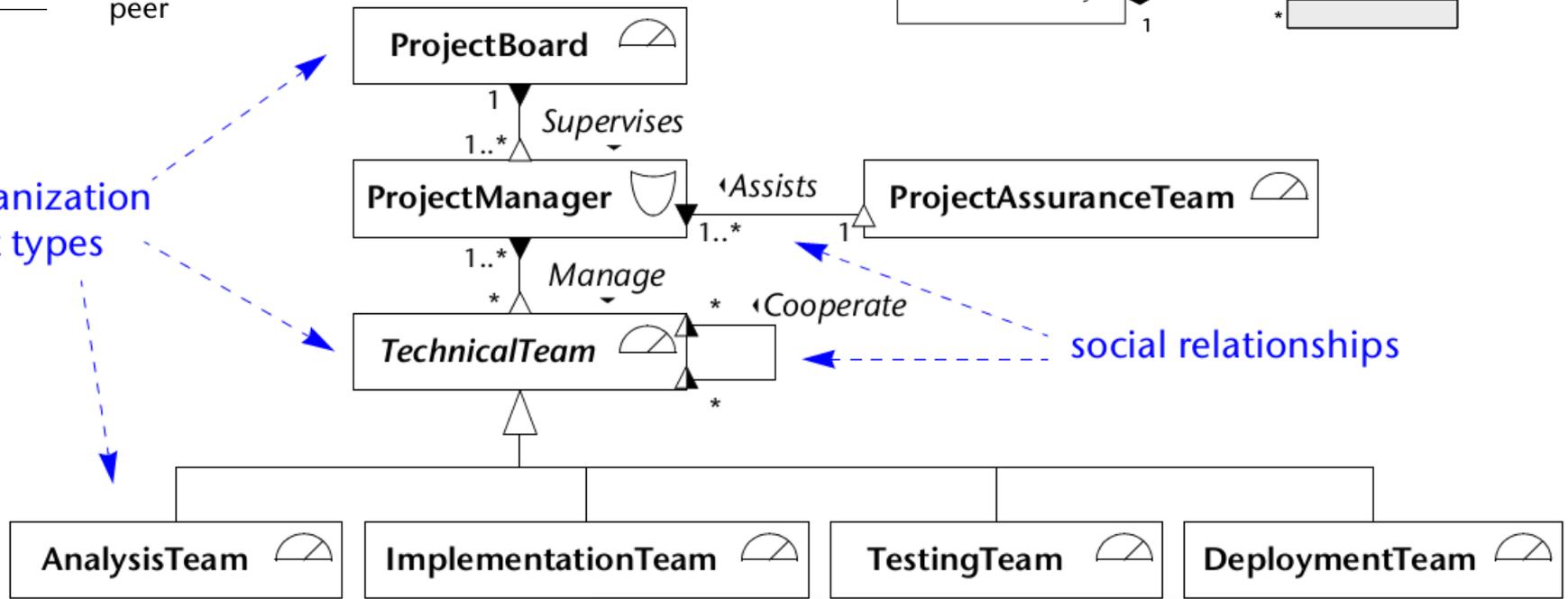


# AML: organisation

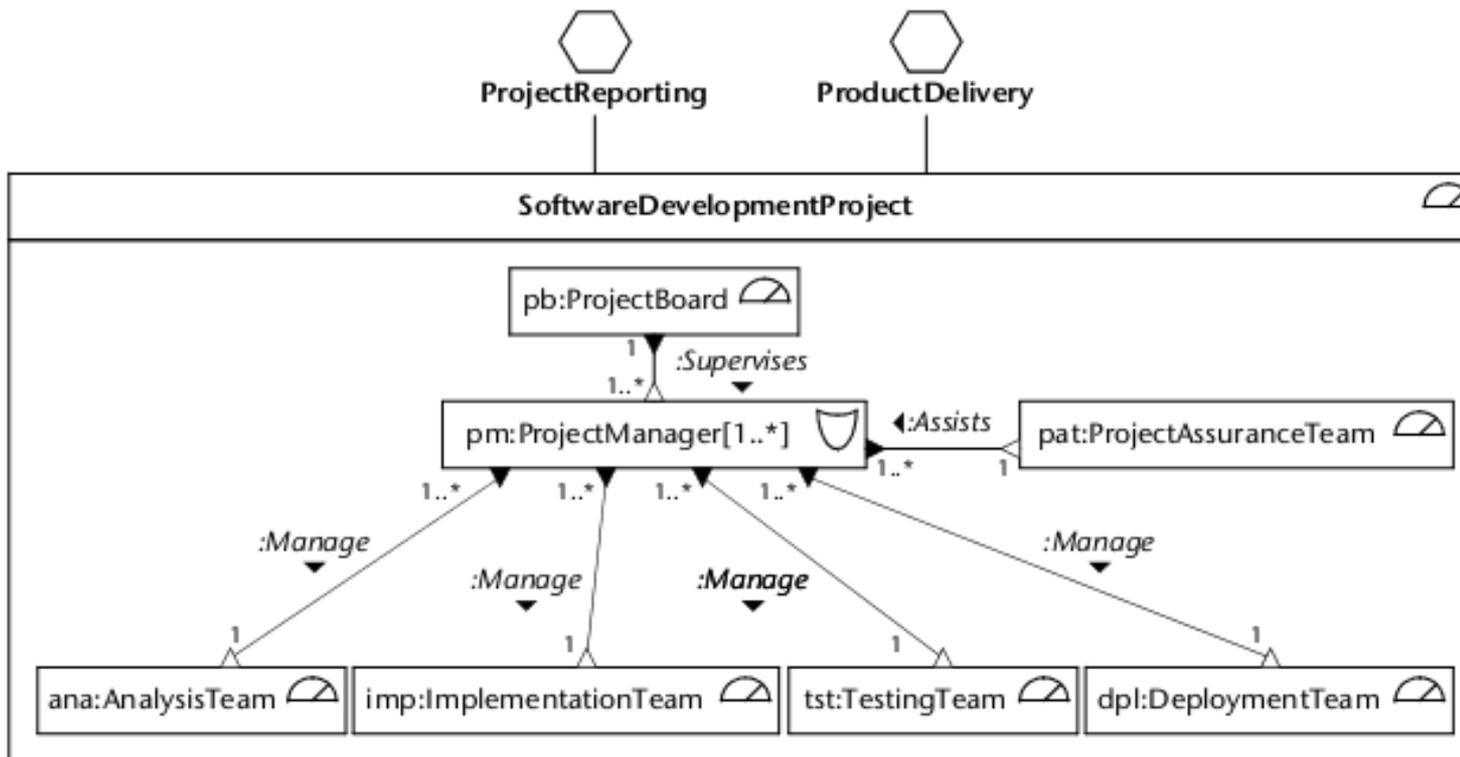
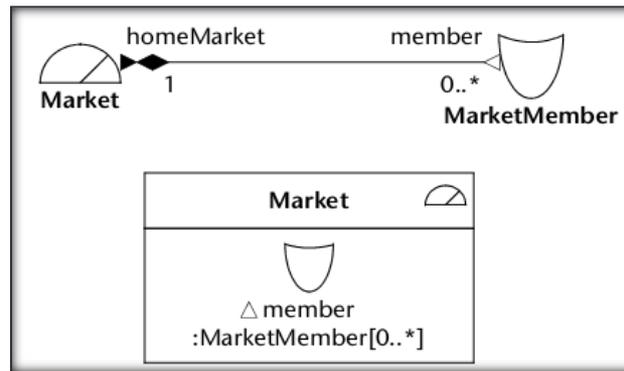


organization unit types

social relationships



**Semantics** OrganizationUnitType is a specialized EnvironmentType used to model types of organization units, i.e. types of social environments or their parts.



# AML: exercice

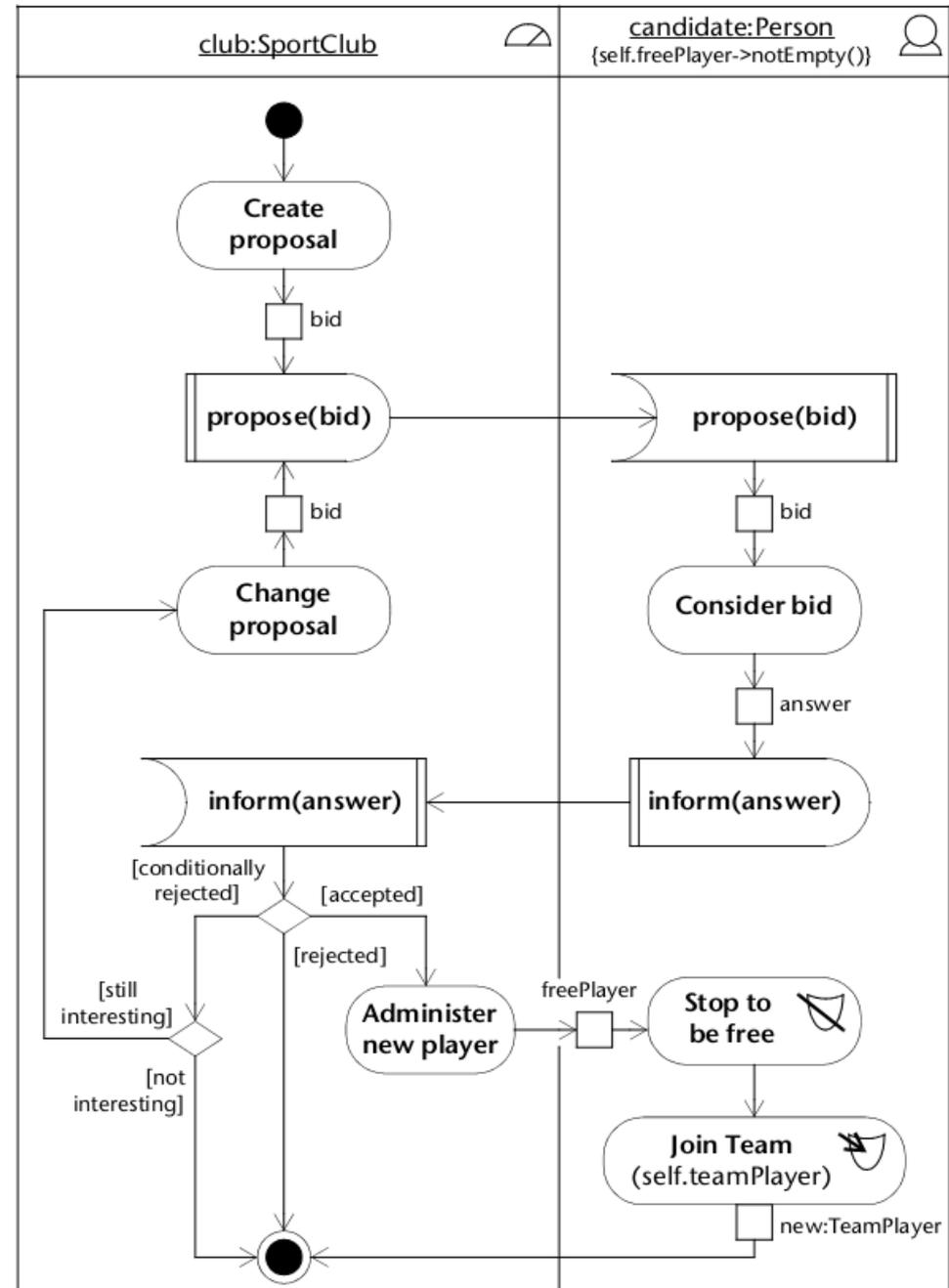
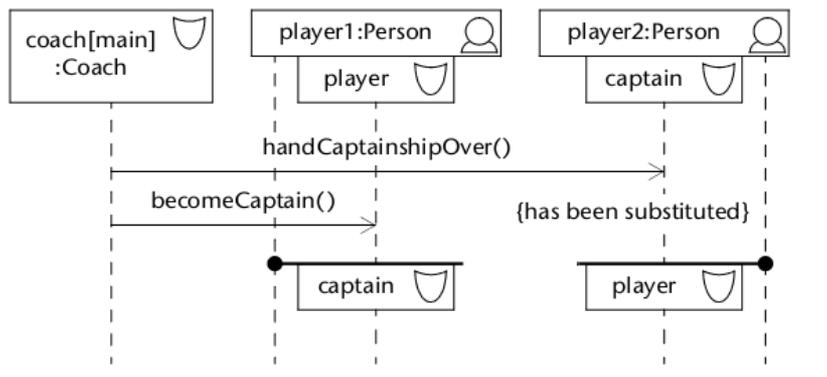
Modélisation de jeu de foot avec un agent humain et un agent bot

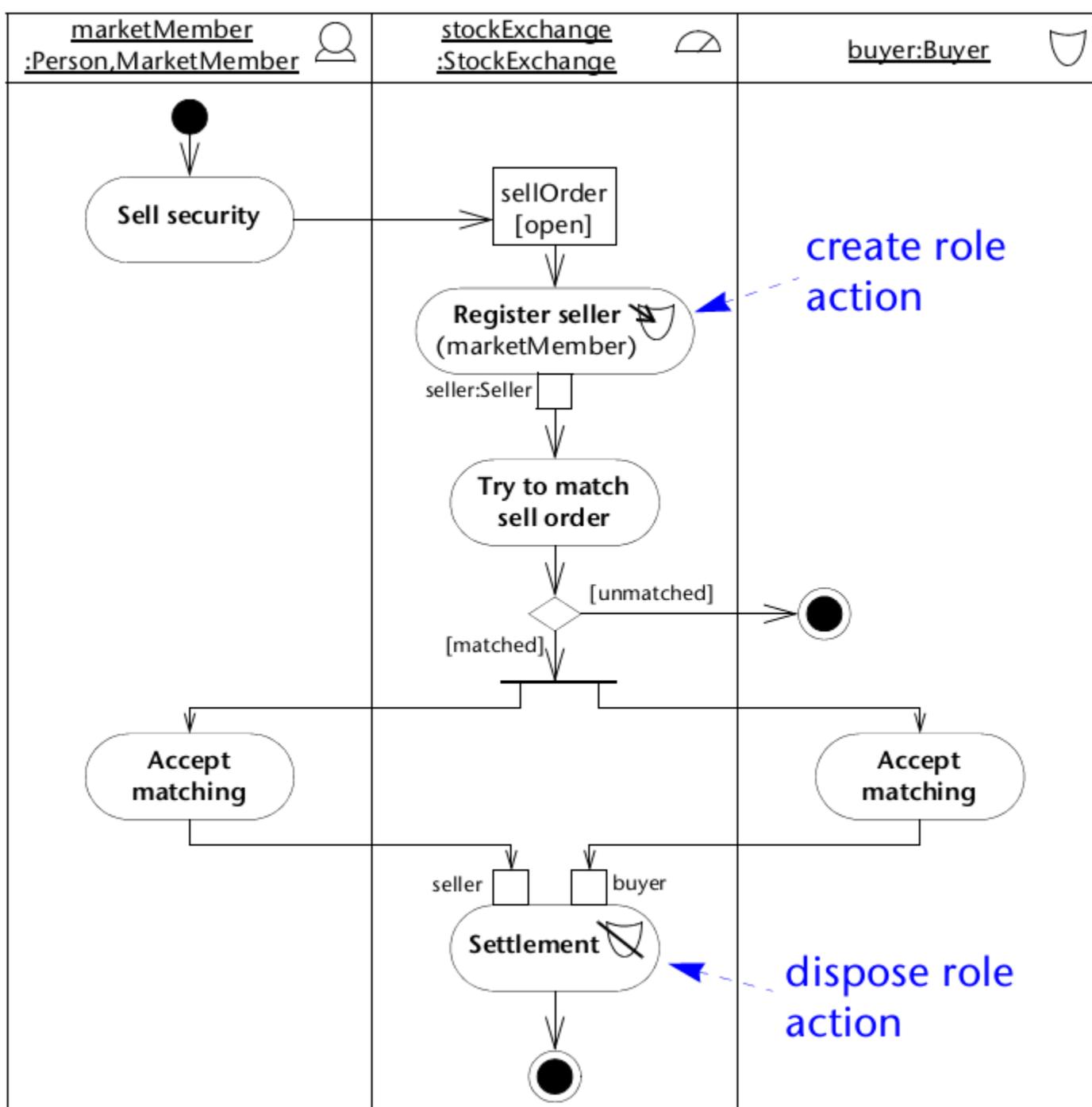
- 1- Ecrire le diagramme correspondant à une SoccerTeam, composée d'1 à 3 coach encadrant 11 à 16 joueurs qui coopèrent entre eux
- 2- Ajouter le terrain, sur lequel se jouent plusieurs matchs. Chaque match a 3 arbitres qui contrôlent les deux équipes (toujours composées de coach et joueurs). Par ailleurs les joueurs peuvent être des goals, des défenseurs, des milieux ou des attaquants
- 3- Ajouter le fait que le Bot1 joue un gardien alors que le P1 (joueur humain) joue un attaquant.

# AML: exercice

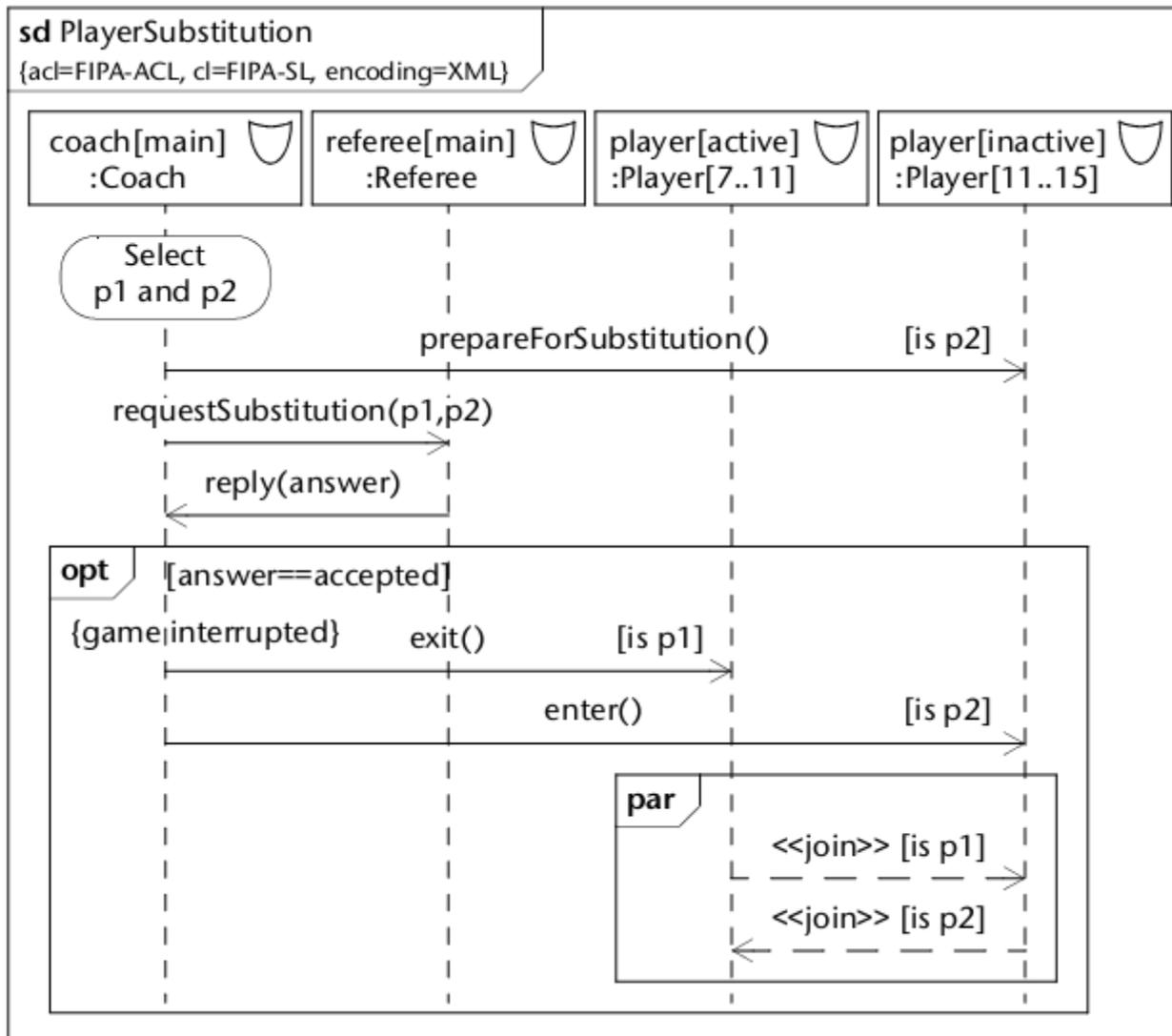
---

# AML: manipulation de rôles

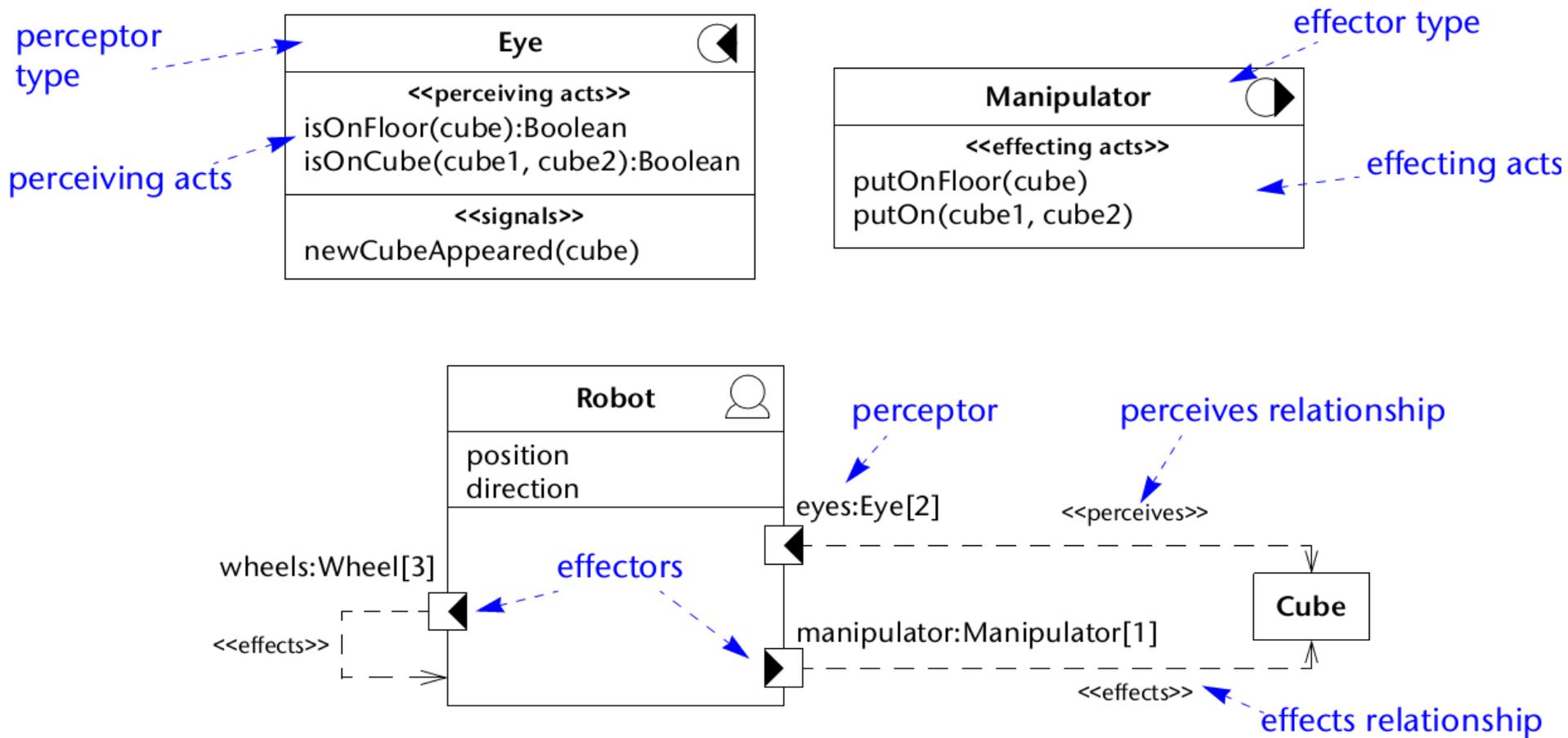




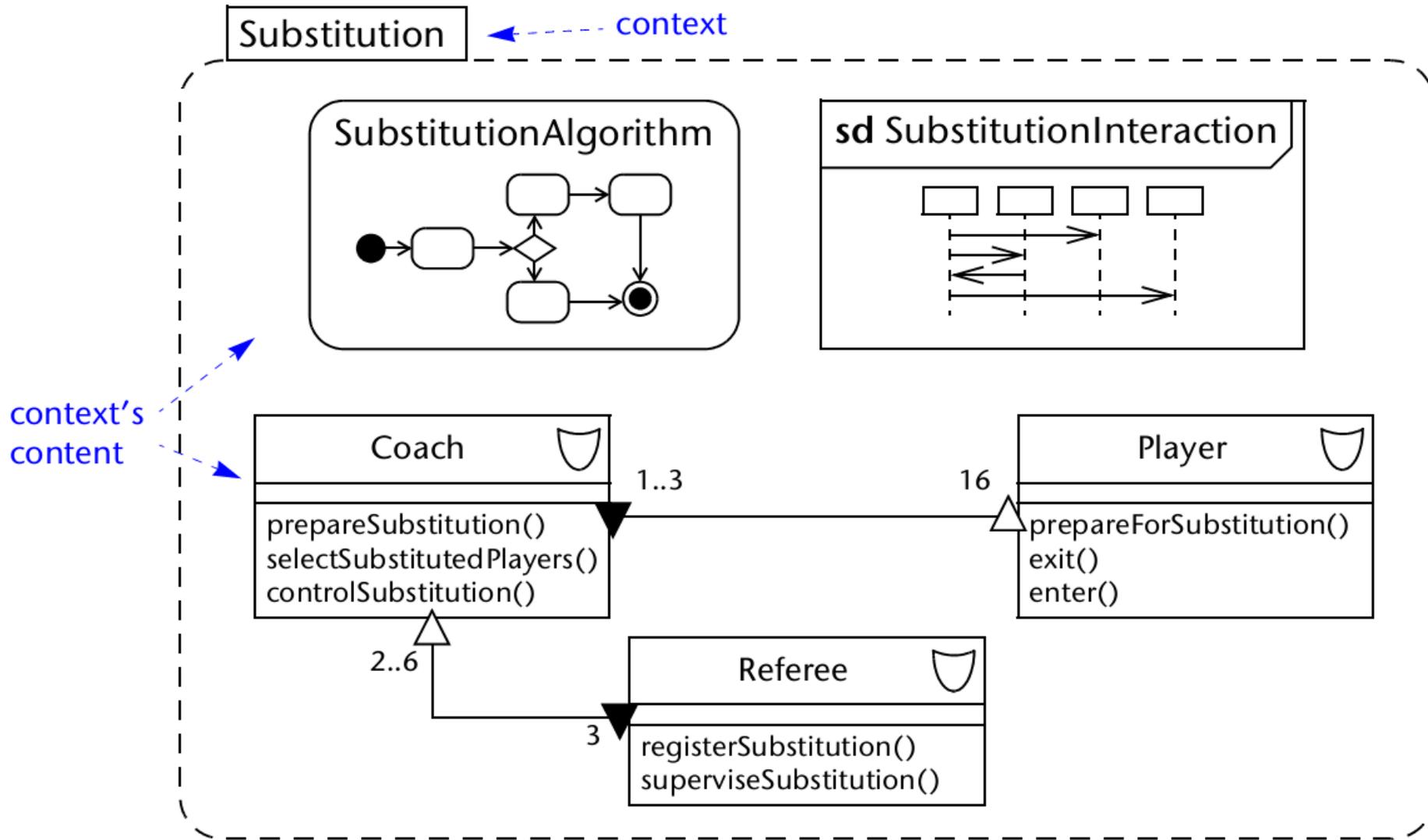
# AML: interaction



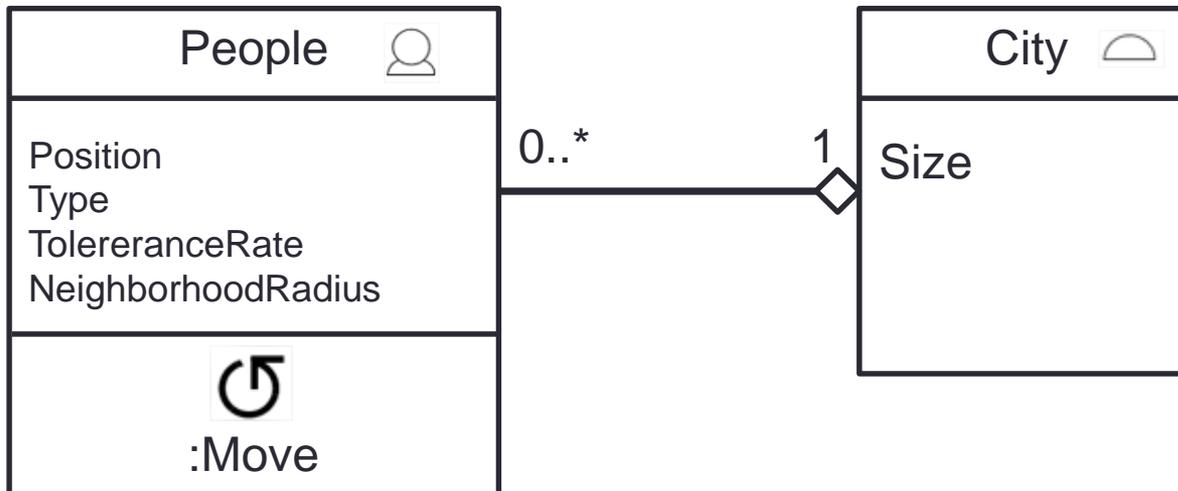
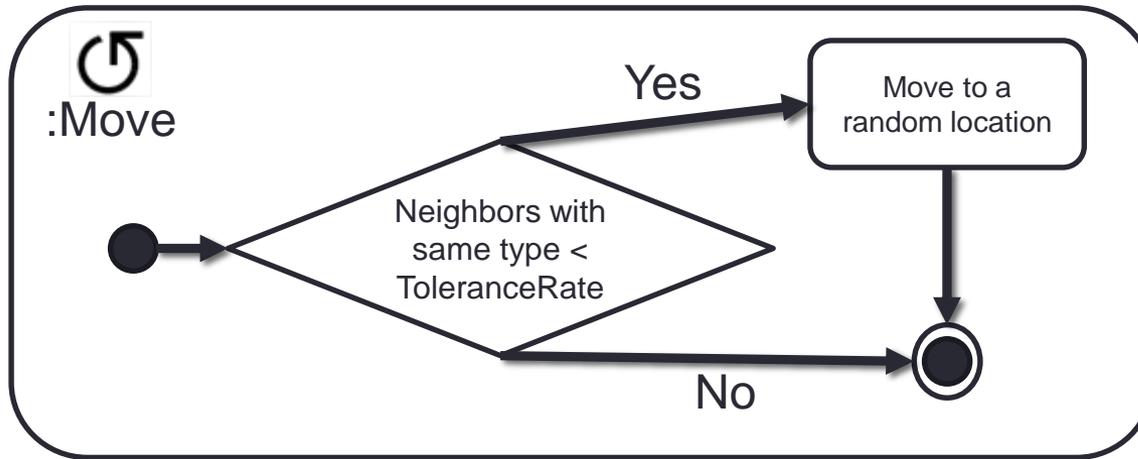
# AML: perceptions et actions



# AML: contexte



## ShellingModel



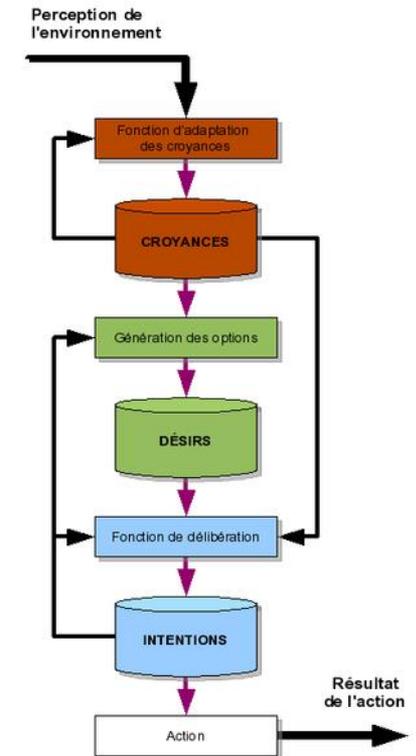
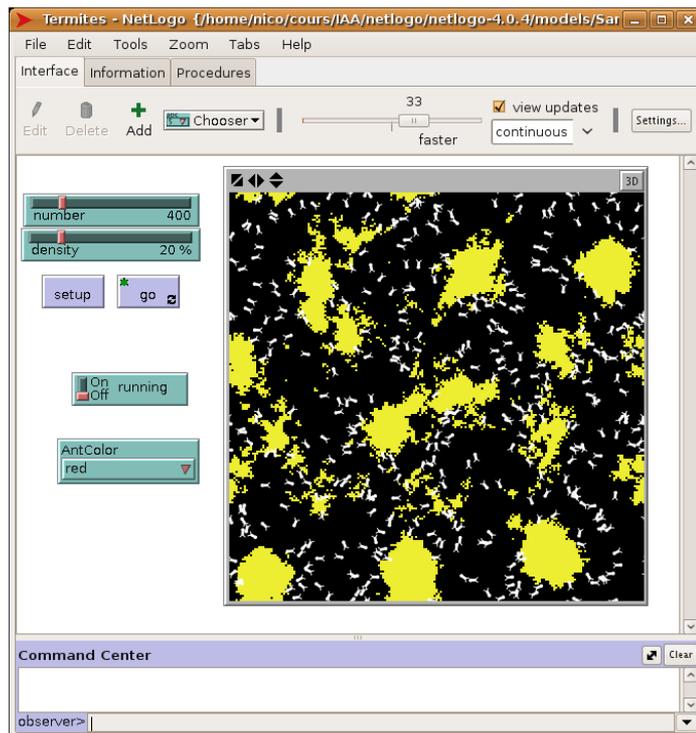
# Plan du cours

- Introduction
- Description d'un SMA
- Modélisation d'un SMA
- Conclusion
  - Plan
  - Bibliographie

# Présentation et implémentation des systèmes multi-agents (SMA)

## C1 : Présentation

- Qu'est ce qu'un agent et un SMA?
- A quoi ca sert?
- Comment le programmer?
- Application à GAMA



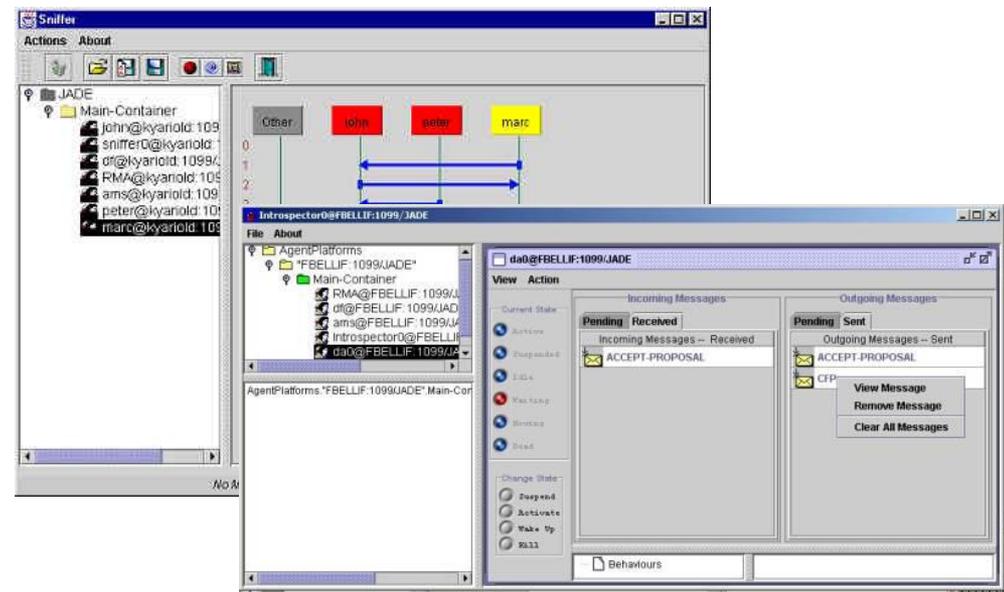
## C2 : Modèles d'agents

- Comment fonctionne un agent?
  - Modèles réactifs à base de règles simples
  - Modèles cognitifs
  - Modèles à base d'interaction
  - ...

# Communications dans un SMA

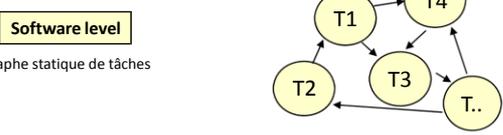
## C3 : Communication

- Les types de communications
- Les actes de langages



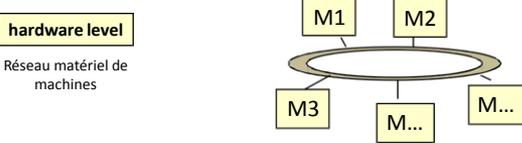
Langages et Protocoles d'interaction  
Echanges d'information et de services

## C4 : TP Jade



Multitâches  
Communication  
Protocoles  
Routage

- Programmation avec la plateforme JADE

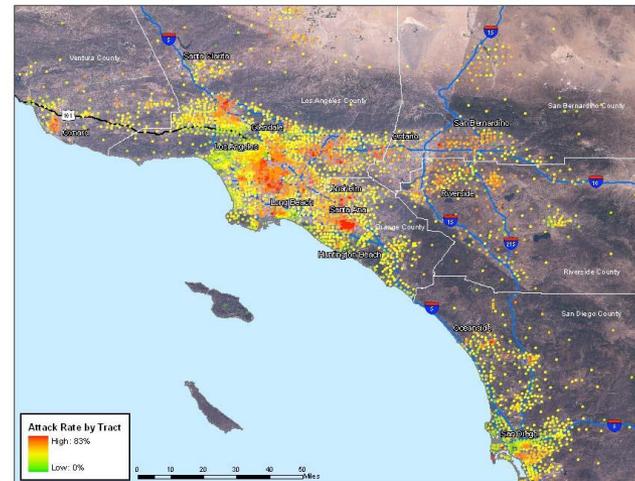
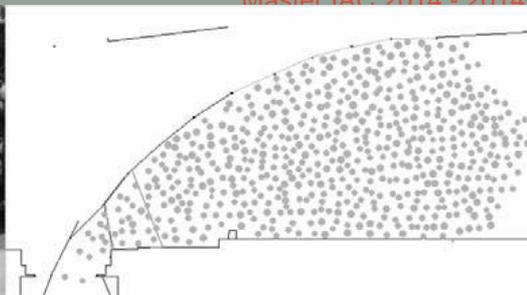


Réseaux intranet  
Internet global (Web)

# C5 : Simulations à base d'agents

Simulations de phénomènes sociaux

- **Définition:** utilisation d'agents autonomes pour reproduire un phénomène économique ou social
- **Applications:** Diffusion de virus, évacuation de stade, marché du travail, ...
- **Principes:** définition du comportement à un niveau micro, observation à un niveau macro
- **Futur:** simulation de comportements individuellement réalistes (évolution d'une négociation sur un marché)



## Méthodologie de la simulation

- **Modélisation d'un système**
- **Implémentation dans un simulateur**
- **Calibration, validation et analyse du modèle**

## Projet

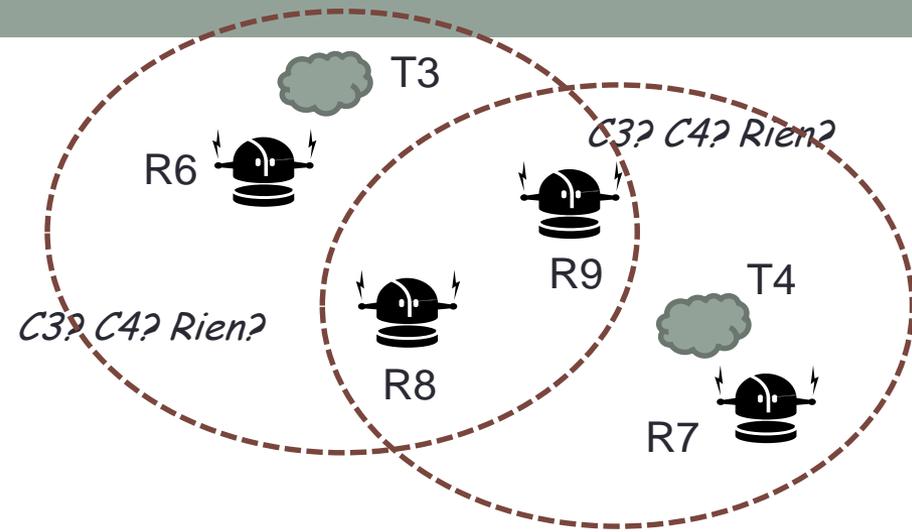
- **Modélisation, implémentation et analyse d'un modèle en binôme**
- **Modèle de la littérature ou choisi par les étudiants**



# Groupes et organisations d'agents

**C6: Théorie des jeux: comment prévoir le résultat des interactions individuelles?**

- **Définition:** Analyse et organisation des interactions avec des agents ayant un objectif propre
- **Méthodes:** théorie des jeux



		Prisoner 2	
		cooperate	defect
Prisoner 1	cooperate	(3,3)	(0,5)
	defect	(5,0)	(1,1)



## C7 : Méthodes de coordination

- **Définition:** Comment faire se coordonner des groupes d'agents et prévoir leur interactions?
- **Méthodes:** bien-être social, méthodes de vote, coalitions d'agents, méthodes d'enchère
- **Applications:** optimisation de chaîne de production, répartition de tâches dans un réseau informatique, négociation automatique

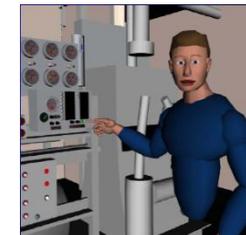
# C8 Agents Conversationnels Animés

## Agents Conversationnels

- **Définition** Agents autonomes chargés de l'interface conversationnelle entre un système et les usagers
- **Domaine** Web, Jeux, E-learning, Réseaux Sociaux
- **Rôles** Présentateur, assistant, compagnon, tuteur, coach
- **Spécificité** Munis d'une personnalisation



têtes parlantes



corps-bras



réalité virtuelle

## Agents Assistants

- **Définition** Capacité de raisonnement intelligent (IA) sur la structure et le fonctionnement du système assisté
- **Interaction** Langue Naturelle (questions de l'utilisateur)
- **Avenir** Agents assistants dans l'ambient.

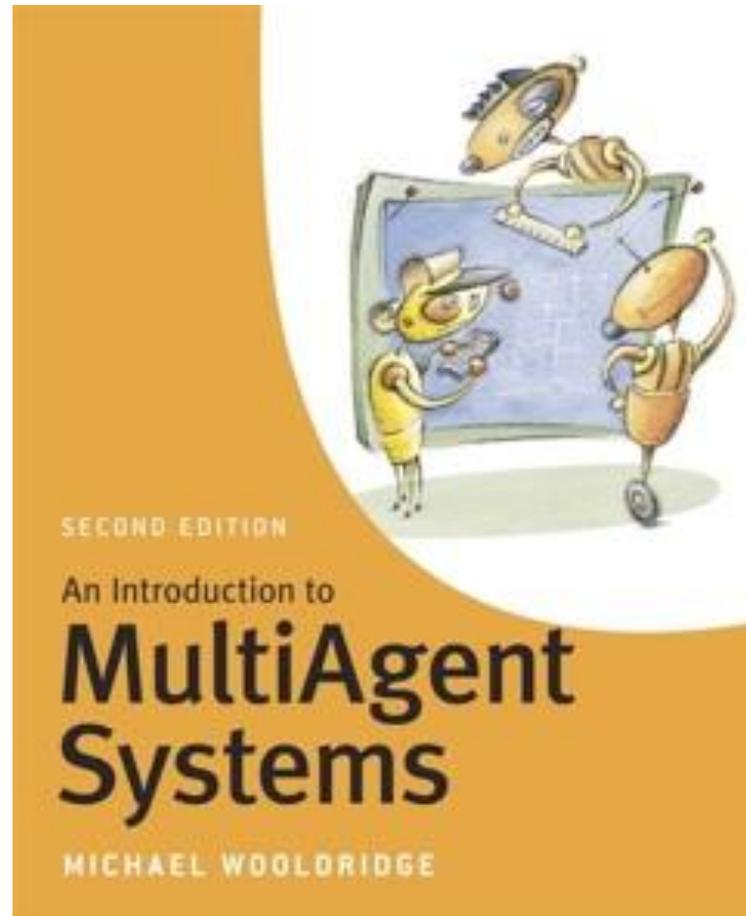


# Notation

- 50% Projet
  - Projet JADE (cours 4)
  - + Simulation multi-agent
    - Modélisation
    - Implémentation
    - Analyse
- 50% examen final
  - Questions de cours et exercices

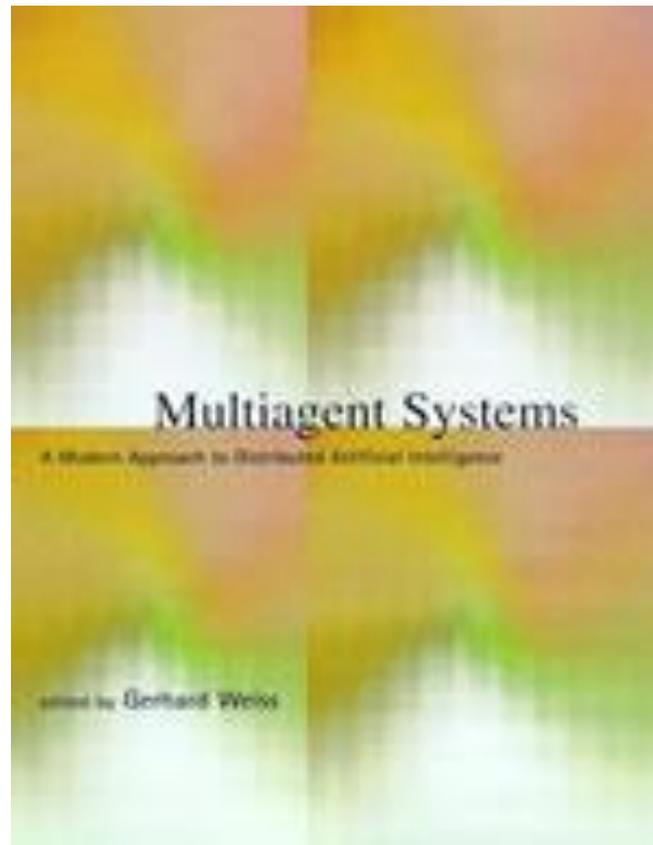
# Biographie

- An introduction to MAS (2009) M. Wooldridge



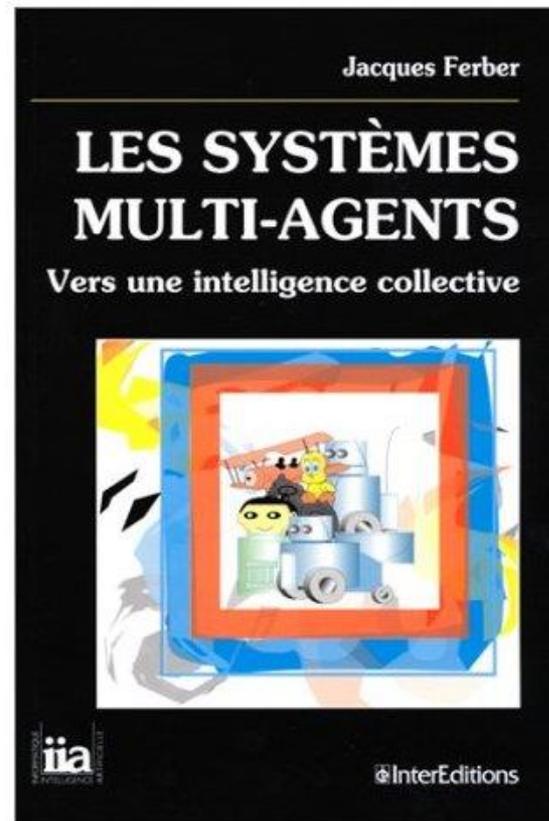
# Biographie

- Multiagent Systems. **A Modern Approach to Distributed Artificial Intelligence (2000)** Edited by Gerhard Weiss



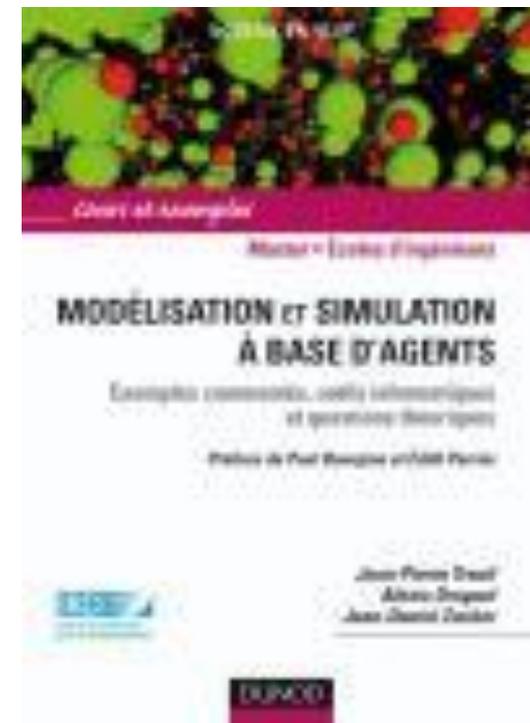
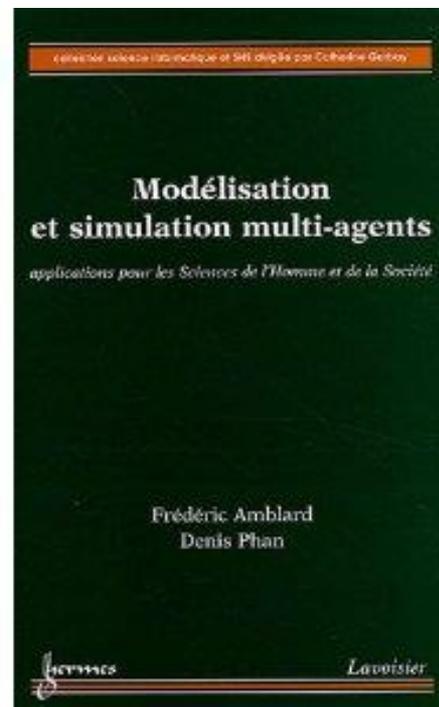
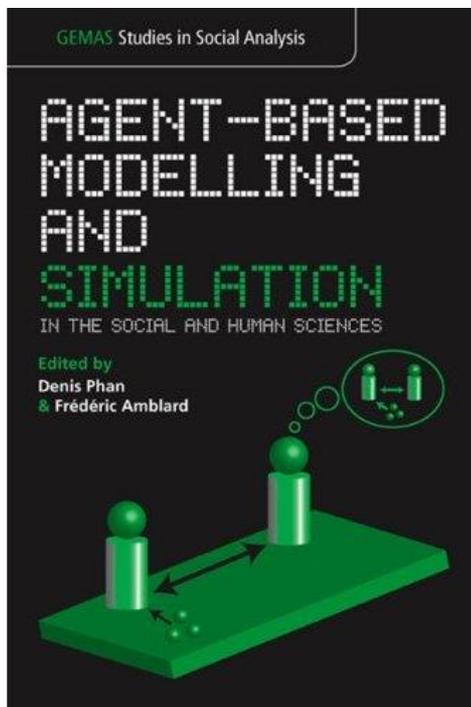
# Biographie

- Jacques Ferber: Les systèmes multi-agents. Vers une intelligence collective. InterEditions, Paris, 1995



# Biographie

- **Agent-based Modelling and Simulation in the Social and Human Sciences (2007)** Frederic Amblard & Denis Phan
- **Modélisation et simulation à base d'agents : Exemples commentés, outils informatiques et questions théoriques (2008)** JP Briot, Alexis Drogoul, JD Zucker



# Biographie

- Revues
  - JAAMAS (Journal of Autonomous Agent and MAS)
  - JASSS (Journal of Artificial Societies and Social Simulation)
    - Disponible gratuitement sur internet
- Actes de conférences:
  - AAMAS
  - PRIMA
  - IAT
- Francophone:
  - JFSMA

# Les grandes problématiques : les sessions AAMAS

- ***Plateformes et developpement***
  - Agent-Based System Development
  - Verification
- ***Modeles d'agents et d'interaction***
  - Agent Reasoning (3)
  - Learning (2)
  - Robotics (2)
- ***Simulation***
  - Simulation
- ***Economie artificielle et competition***
  - Game Theory (2)
  - Trust
  - Agreement Technologies
  - Economic Paradigms (2)
  - Agent Societies
- ***IAD et coordination***
  - Coordination and cooperation (3)
  - Social Choice (2)
  - Distributed Problem Solving
- ***Agents conversationels***
  - Virtual Agents