

## Agents et Systèmes Multiagents: Thèmes, Approches et Défis

Brahim Chaib-draa  
Département D'informatique  
Université Laval Ste-Foy

## Buts de la présentation

- Donner un aperçu général sur l'évolution rapide de la technologie agent
- Présenter les défis et les bénéfices potentiels de la recherche «agent»

## Plan de Présentation

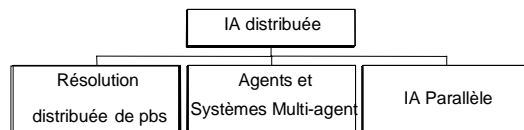
- placer les agents dans le contexte
- présenter une topologie des agents (BT), en donnant pour chaque type d'agent :
  - Raison d'être, hypothèses, buts,
  - défis, état de l'art
  - exemples

## Agents Logiciels: Historique

- La recherche sur les agents date du début de l'IA
- Carl Hewitt's actor model (1977)  
*"a self-contained, interactive and concurrently-executing object, with some encapsulated internal state and which could respond to messages from other similar objects"*

## Historique (suite)

### Contexte de la recherche sur les agents



## Motivations pour la recherche sur les agents

- Au départ, et du point de vue IA distribuée
  - ✓ modularité — pour réduire la complexité
  - ✓ vitesse — due au parallélisme
  - ✓ fiabilité — due à la redondance
  - ✓ flexibilité — nouvelles tâches peuvent être gérées plus facilement grâce à la modularité

## Motivations (suite)

- Du point de vue IA
  - intégration au niveau de la recherche
  - apport au langage naturel
  - applications proches de la réalité
  - richesse du point de vue recherche

## Applications

- gestion des workflows
- gestion des réseaux
- contrôle de trafic aérien
- gestion des affaires
- smart databases
- commande et contrôle
- éducation
- trafic routier
- bibliothèques digitales
- personal digital assistants (PDA)
- gestion des rendez-vous
- filtrage du e-mail
- gestion de l'information
- data mining
- Commerce électronique

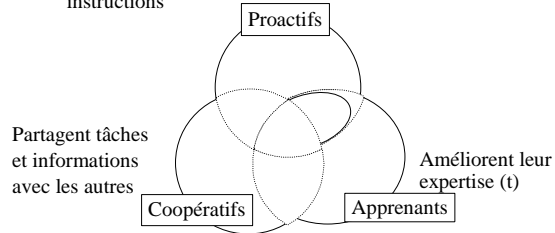
## Applications des agents (suite)

- *“in 10 years time most new IT development will be affected, and many consumer products will contain embedded agent-based systems”*  
(Guilfoyle 1995)

## Qu'est ce qu'un agent?

- *A component of software and/or hardware which is capable of acting exactly in order to accomplish tasks on behalf of its user* (plusieurs chercheurs)
- Un chapeau qui couvre une panoplie d'agents spécifiques

Ne nécessitent pas des instructions      Capables d'agir seuls



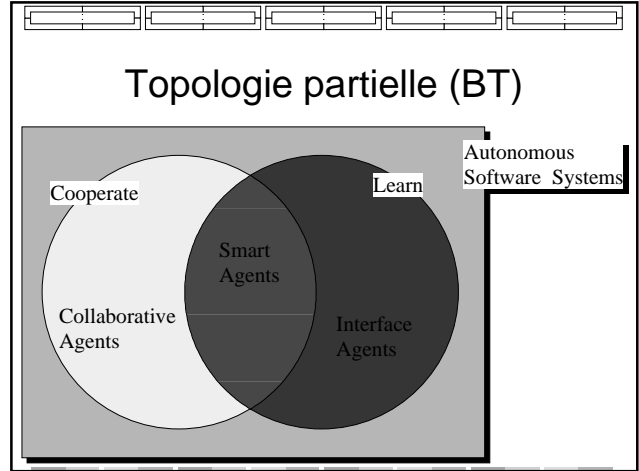
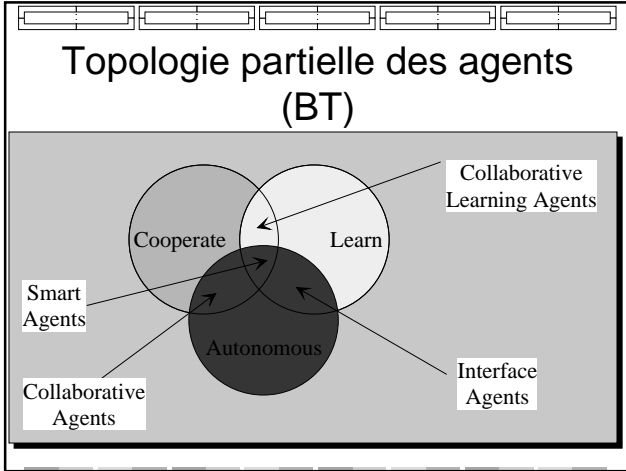
## Agents Logiciels (BT)

## Attributs primaires

Autonomie

Coopération

Apprentissage



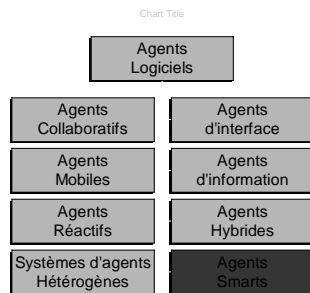
### Autres classes d'agents (BT)

hybrid agents	combine two or more agent philosophies in a single agent e.g. deliberative and reactive philosophies
heterogeneous agent systems	an integrated multi-agent set-up, with agents belonging to two or more different agent classes  may also contain hybrid agents

### Attributs secondaires (BT)

Attribute	
versatility	does the agent have many goals or can it engage in a variety of tasks?
benevolence	is the agent benevolent or non-helpful, antagonistic or altruistic?
veracity	does the agent lie knowingly or is it always truthful?
persistence	is the agent temporally continuous?
emotional attitudes	e.g. does the agent get 'fed up' being asked to do the same thing time and time again?
mental attitudes	does the agent have mentalistic attitudes such as beliefs, desires and intentions?

## Topologie des agents



## Agents collaboratifs

- Mettent l'accent sur l'*autonomie* et la *coopération* entre agents
- Ils opèrent presque tout le temps dans des environnements multiagents ouverts et contraints par le temps
- Ils *négocient* avec les autres agents en vue d'atteindre des ententes lors de la résolution distribuée de problèmes
- Capacités limitées en apprentissage

## Agents collaboratifs: raison d'être

“to create a system that interconnects separately developed collaborative agents, thus enabling the ensemble to function beyond the capabilities of any of its members” (Huhns & Singh 1994)

## Raisons d'être (suite)

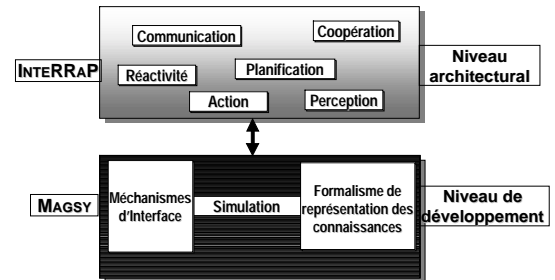
- Résoudre des problèmes qui sont trop grands ou trop complexes pour un seul agent
- Permettre l'interconnectivité et l'interopérabilité des legacy systems
- Résoudre des problèmes «naturellement» distribués

## Simulation avec InteRRAP

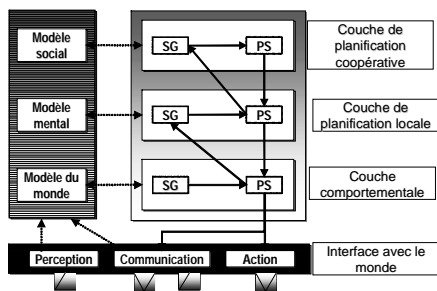
Simuler = *copier* la réalité

- Prédire le comportement d'un système complexe
- Simuler avec des agents, pourquoi ?
  - Interaction entre entités autonomes
  - Distribution : connaissances, contrôle...
  - Complexité
  - Méthodes qualitatives

## L'architecture InteRRAP



## Architecture d'un agent d'InteRRAP

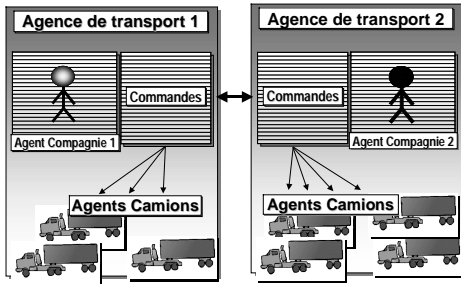


## MAGSY

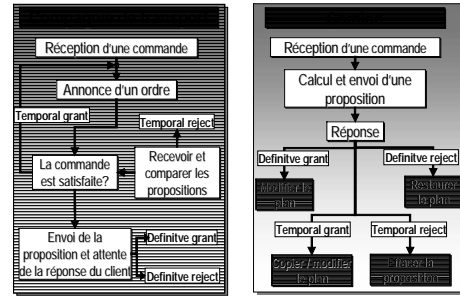
### Le niveau de développement

- Représentation orientée objet des connaissances
- Inférence :
  - Chaînage avant pour l'activation des conditions dans BBL
  - Chaînage arrière pour la planification
- Simulation :
  - Création d'agents
  - Visualisation et contrôle
  - Expérimentations et tests

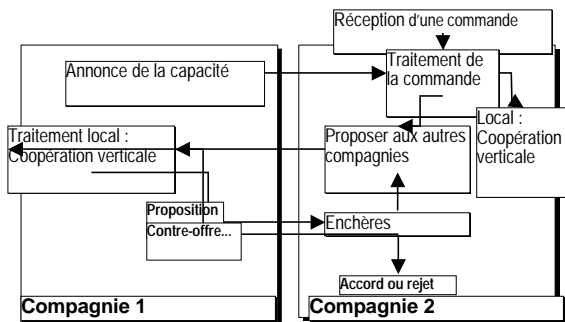
## MARS: le domaine d'application



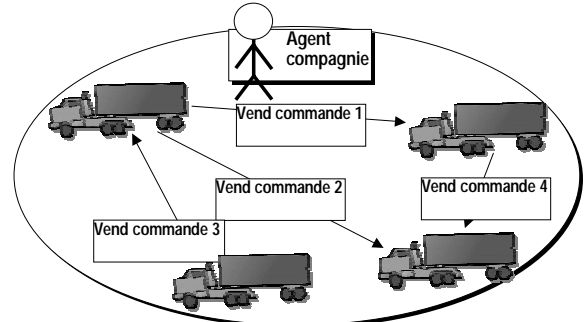
## Coopération verticale: Extended Contract Net Protocol

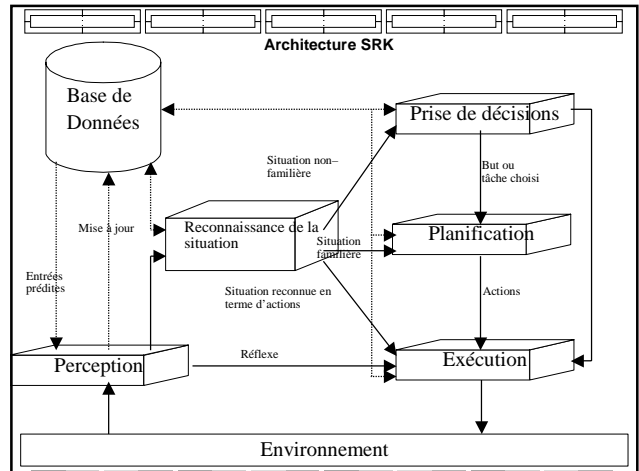
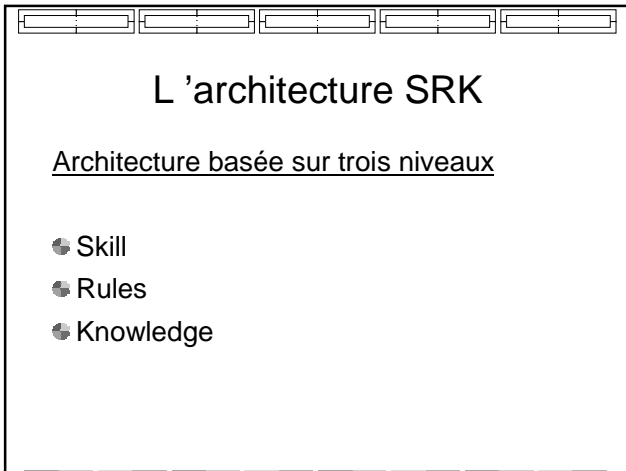
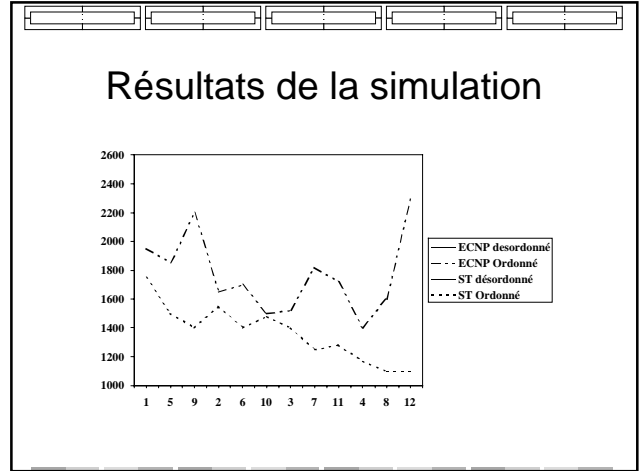
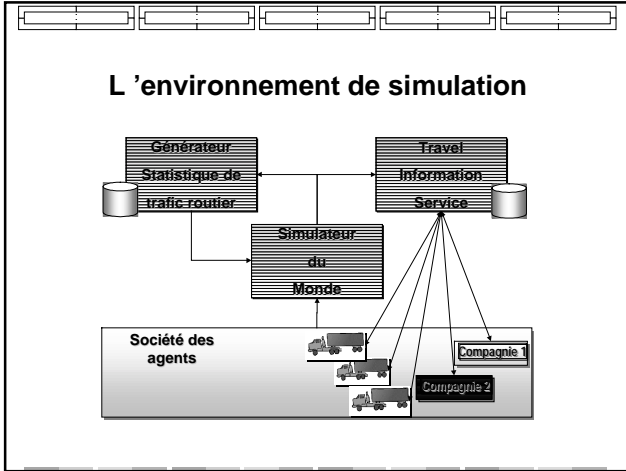


## Coopération horizontale



## Le commerce simulé





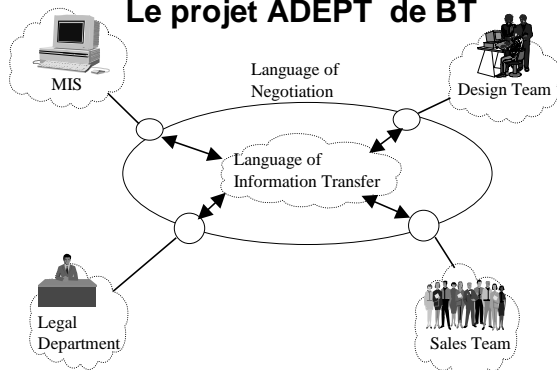
## Utilisation de l'architecture SRK

- ✓ Simulation et modélisation du trafic routier
- ✓ Application au contrôle-commande
- ✓ Prise de décisions distribuées
- ✓ Interfaces personnes-machines

## The Pleiades System (Sycara 1995)

- Des agents collaboratifs participent à la prise de décision à travers l'infosphère
- Le système est utilisé pour développer le CMU's *Visitor Hosting System*  
agents qui coopèrent afin de créer et gérer l'horaire d'un visiteur à CMU

## Le projet ADEPT de BT



## Le consortium d'ADEPT

Funding : DTI (ISIP), EPSRC, BT, ICI

### Consortium

- BT Laboratories
- ICI Engineering Technology
- Loughborough University of Technology
- Queen Mary & Westfield College

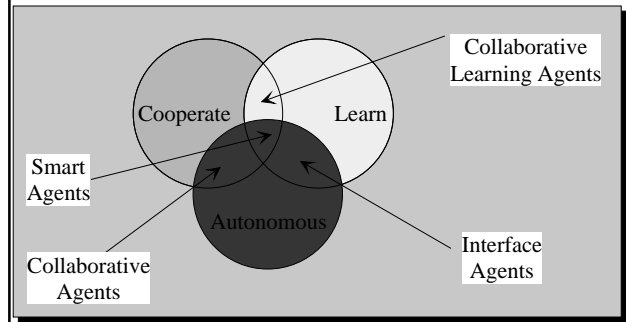
3 Year Project

Total effort = 33 man-years

## Les agents collaboratifs : Défis

- À Utiliser + dans les applications industriels
- À construire selon les méthodes issues du «génie» voire du génie logiciel
- Pousser encore plus loin la coordination, la communication et la négociation
- S'intéresser à la stabilité, à la scalabilité et aux problèmes de performance
- Introduire davantage d'apprentissage
- Trouver les «bonnes» méthodes d'évaluation

## Vue partielle de la topologie agent



## Agents interfaces

- Ils mettent l'accent sur l'*autonomie* et l'*apprentissage* en vue d'exécuter des tâches pour les utilisateurs
- Ils fournissent une assistance *proactive* à l'utilisateur pour une application précise
- Leurs interactions avec les autres agents sont en général limitées

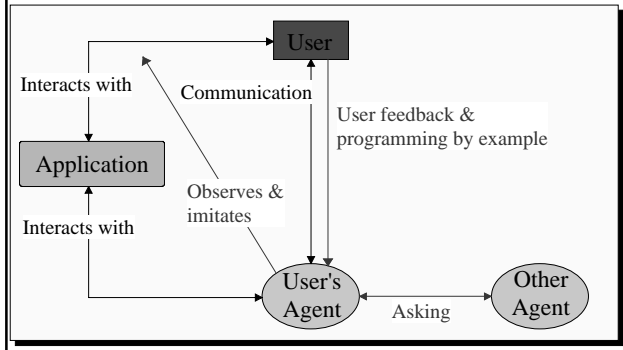
## Agent interface = secrétaire

C'est un *assistant personnel* qui *collabore* avec l'utilisateur (un genre de secrétaire)

## Agents interfaces : raisons d'être

- Moins de travail pour l'utilisateur et le développeur
- L'agent peut s'adapter, à travers le temps, aux habitudes de l'utilisateur et à ses préférences
- Peuvent aider plusieurs utilisateurs à se partager leur savoir-faire

## Comment ces agents fonctionnent



## Agents interface : marché?

- Agents simples mais *utiles*
- Opèrent dans des domaines limités
- Nécessitent peu ou pas de coopération avec les autres agents

## Agents interfaces: Défis

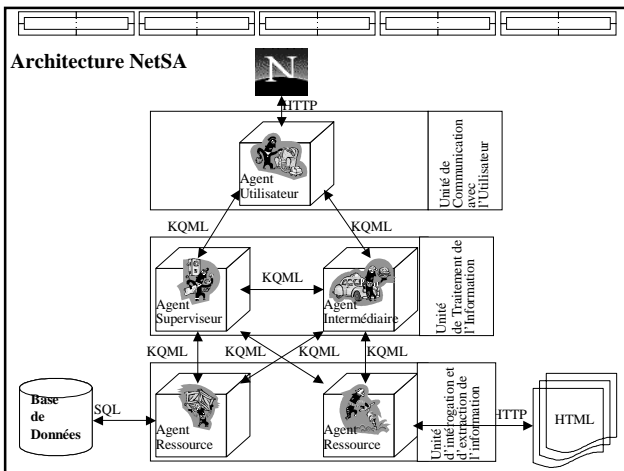
- Mener des expérimentations en vue de déterminer *quelles* sont les techniques d'apprentissage qui sont préférables, pour *quels* domaines, et *pourquoi*
- Garantir la «vie privée» des utilisateurs
- Étendre si possible le champ d'application

## Agents Mobiles

- Pièces logicielles qui peuvent :
  - migrer d'une machine à une autre
  - interagir avec différentes machines
  - exécuter des tâches pour les utilisateurs
  - retourner à leur 'point de départ'

## Agents pour l'Internet

- L'explosion du Web nécessite des outils performants pour rechercher, filtrer, extraire, etc. les informations dont on a besoin
- La solution passe par une architecture multiagent (avec des agents collaboratifs): NetSA en est une



## Agents pour l'internet : Défis

- Communication
- Sécurité
- Intégration de la planification de l'apprentissage et du data-mining

## Agents réactifs

- Ils ne possèdent pas une modélisation symbolique interne de leur environnement
- Ils répondent par une réponse précompilée à tout état de l'environnement
- Il se basent sur l'idée d'émergence

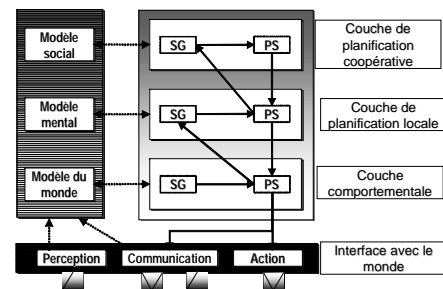
## Agents Réactifs: raisons d'être

- Robustesse et tolérance aux fautes
- Flexibilité
- Adaptabilité
- Rapides temps de réponses

## Agents hybrides

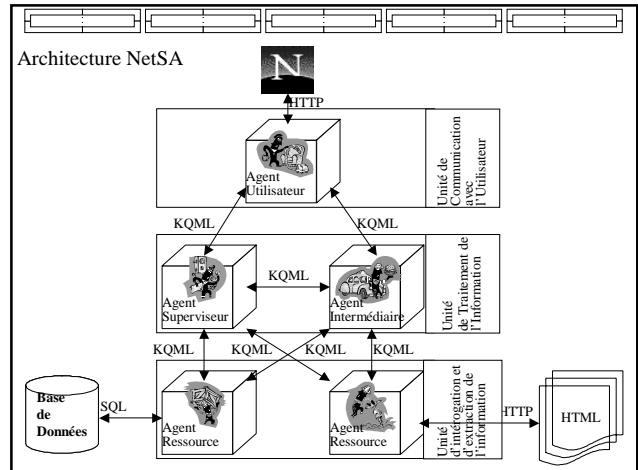
- Ils combinent deux ou plusieurs philosophies
  - InteRRAP combine délibératif et le réactif
  - SRK combine Skill, Rules et Knowledge
- Ils couvrent les différentes facettes qu'on trouve habituellement aux niveaux des applications

## Un agent hybride: InteRRAP



## Systèmes multiagents hétérogènes

Un exemple de ce type de système est donné par l'architecture NetSA (Networked Software Agent)



## Systèmes multiagents hétérogènes : Raisons d'être

- Les systèmes ouverts (ex: l'Internet)
- Interopérabilité
- Liens avec les legacy systems

## Systèmes multiagents hétérogènes: Défis

- Communication
- Ontologies
- Liens avec les legacy systems

## Aspects théoriques

- Logique
- BDI
- Les théories de jeu
- Les théories issues des sciences économiques
- Les aspects linguistiques pour les communications

## Conclusions