MASTER WEB INTELLIGENCE

Multi-Agent Systems

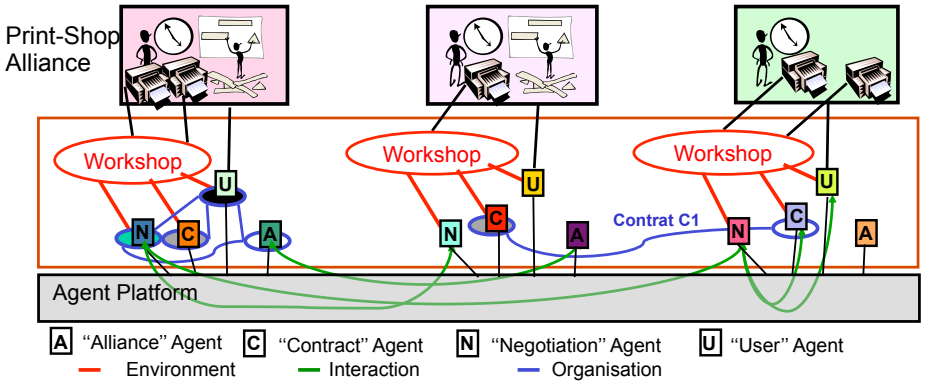
Introduction

Olivier Boissier
Olivier.Boissier@emse.fr

Plan

1. **Definitions**
2. Action Domains
3. Positioning
4. "Vowels" Dimensions
5. Multi-Agent Engineering
6. Perspectives …

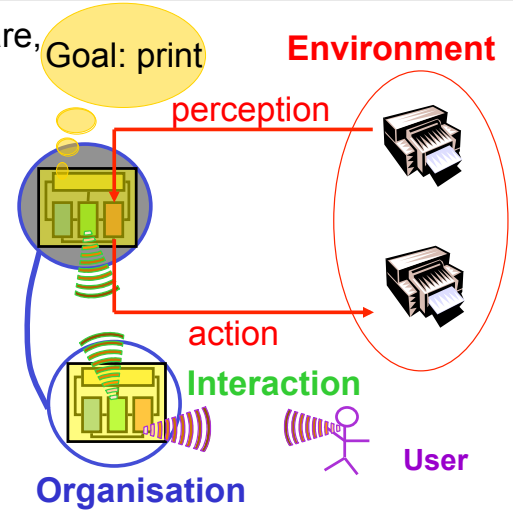
Multi-Agent Systems



Multi-Agent System (MAS) : set of agents, that interact with each other, situated in a common environment, eventually, building or participating to, an organisation

Agent (in a Multi-Agent World)

Agent : physical or software, *autonomous* entity that is pro-active, reactive, social, able to take part to an organised activity, in order to achieve its goals, by interacting with other agents and users.



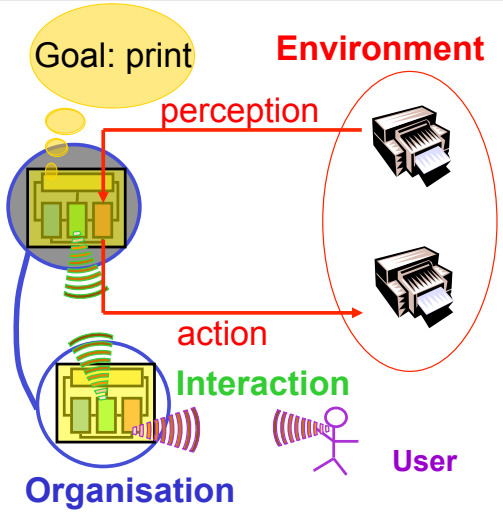
Autonomous Agent (in a Multi-Agent World)

- An agent ***X*** is autonomous with respect to ***Y*** for ***O*** in situation ***S***
  - ***Y*** can be a user, another agent, a group of agents, an organisation, …
  - ***O*** can be a *goal*, a *plan*, an *action*, a *resource*, a *norm*, a *role*, …

It means that:

- agent ***X*** can *decide locally* of the adoption of ***O*** in situation ***S***
- **And** ***Y*** *has no certainty that* ***X*** is going to adopt ***O*** in situation ***S***

→ **Loose** coupling between agents



Multi-Agent Systems Principles

- The ***Agent perspective*** (micro perspective)
  - *Reactive* & *Pro-Active* entities / Encapsulation of control
  - *Autonomy*: agents may exhibit activities that are not the one expected by the other agents in the system
  - *Delegation*: agents may receive some control over their activities
- The ***Multi-Agent System perspective*** (macro perspective)
  - *Distribution* of knowledge, of resources, of reasoning/decision capabilities
  - *Decentralization* (loose coupling) of control, authority
  - Agreement technologies, Coordination models and mechanisms to install coordination between the autonomous agents
  - Emergent / Social order / Normative functioning

Plan

1. Definitions
2. **Action Domains**
3. Positioning
4. "Vowels" Dimensions
5. Multi-Agent Engineering
6. Perspectives …

MAS Action domains

- **Socio-technical Systems**
  - Integration of software applications, with humans, organizations and the physical world
  - Making them interoperate, interact, cooperate in a flexible and consistent manner with each other
- **Problem Solving**
  - Modeling and solving problems by cooperation between local solvers
  - Installing top-down and/or bottom-up (emergent) solving process
- **Simulation**
  - Modeling and reproducing complex phenomena of interacting entities in the real world in order to understand or to explain their behavior

Socio-Technical Systems (1)

- Industries, services, IT applications are getting global
  - Placed at the centre of multiple networks
  - Developing Knowledge intensive processes
  - Based on large scale underlying IT platforms such as Internet, Web, Internet of Things
- Industries, services, IT applications are situated in an ever-evolving environment
  - Requiring efficient collaboration processes
  - While keeping flexibility and agility
- Users are more and more at the centre of the cooperation and collaboration taking place in these socio-technical systems

Socio-Technical Systems (2)

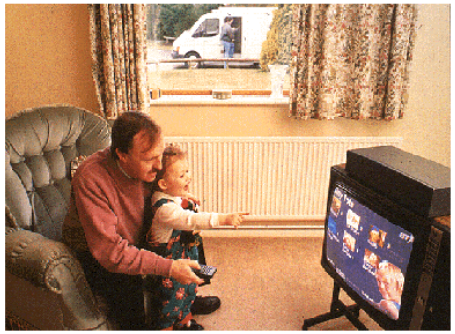
- Properties of the targeted applications:
  - Absence of monolithic vision
  - Incremental development, by different teams
  - Multi-* (sites, expertise, domains, points of view, decisions, goals, motivations, …)
  - Continuous execution and adaptation
  - User-Centred
- Main requirements:
  - Openness, permeability, scalability in size or structure
  - Distribution, no central control, control and interaction are local
  - Autonomous Interacting entities loosely coupled with others or applications
  - Knowledge Intensive processing and sharing
  - Users may delegate their decisions to the application

Example (1/3)

Service Personnalisation

Tonight's Suggested Viewing:

7pm World News Headlines
7:15 Personal Newsround
7:30 Selected highlights of today's golf
7:50-8:00 Intermission (Video-call - it's your brother's birthday)
8:00-10:00 Film choice
Jurassic Park (VR) OR Cyberspace 2
(please select now)



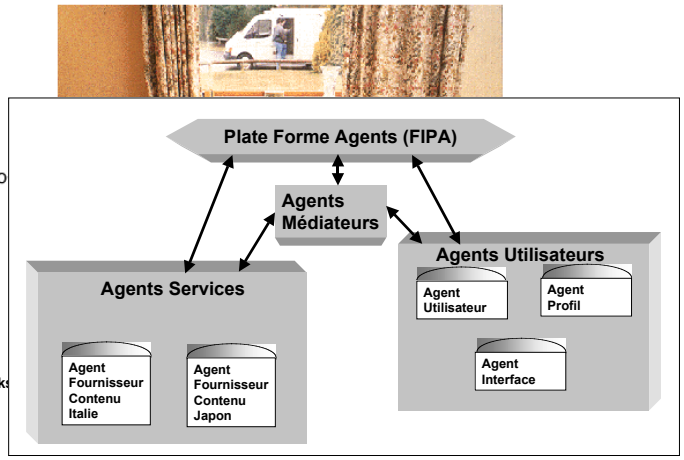
Source CLIMATE Industrial Workshop 26/4/99

Example (1/3)

Tonight's Suggested Viewing:

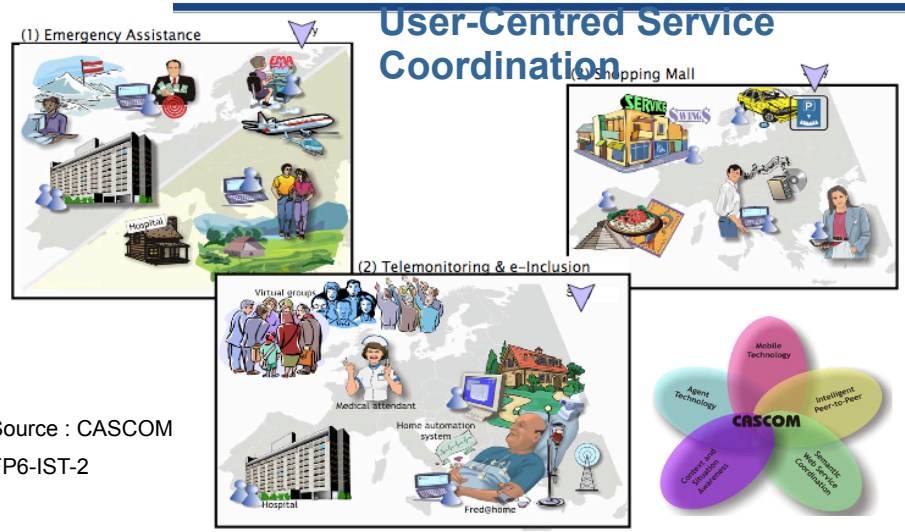
7pm World News Headlines
7:15 Personal Newsround
7:30 Selected highlights of today's golf
7:50-8:00 Intermission (Video call - it's your brother's birthday)
8:00-10:00 Film choice
Jurassic Park (VR) OR Cyberspace 2
(please select now)

Source CLIMATE Industrial Works



Example (2/3)

User-Centred Service Coordination

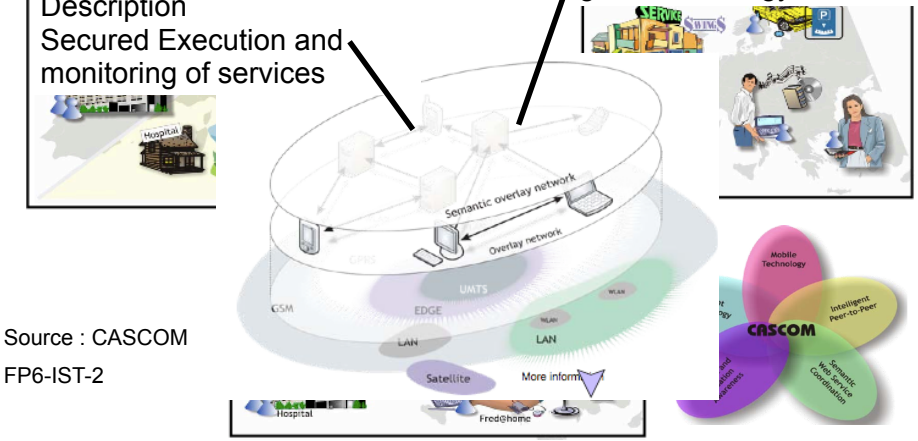


Source : CASCOM
FP6-IST-2

Example (2/3)

Service Composition
Semantic Services Description
Secured Execution and monitoring of services

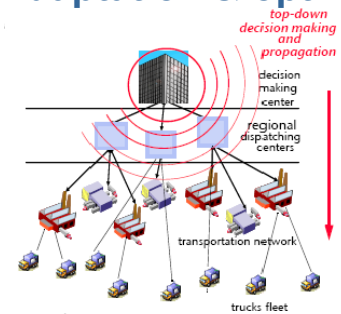
Semantic Web Agent Technology



Source : CASCOM
FP6-IST-2

Example (3/3)

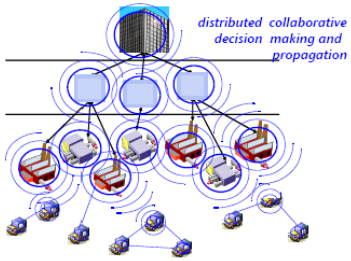
Adaptation & optimisation



Planification, coordination, optimisation along a top-down approach :
• *Centralised collect and processing of informations and events*
• Propagation of plans & decisions
• No realtime decision.

Source Whitestein Agent Technology Conference 2004

Planification, coordination, optimisation along a bottom-up approach
• Responsibility Delegation
• Communication between the nodes
• Real time detection & reaction to changes
• Adaptation to changes & continuous optimisation



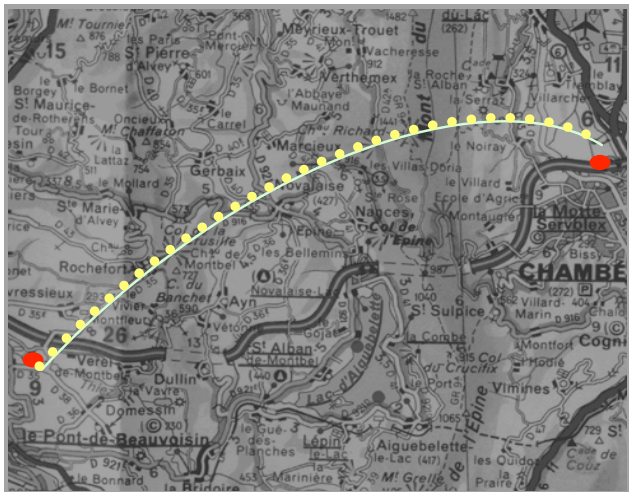
MAS Action domains

- **Socio-technical Systems**
  - Integration of software applications, with humans, organizations and the physical world
  - Making them interoperate, interact, cooperate in a flexible and consistent manner with each other
- **Problem Solving**
  - Modeling and solving problems by cooperation between local solvers
  - Installing top-down and/or bottom-up (emergent) solving process
- **Simulation**
  - Modeling and reproducing complex phenomena of interacting entities in the real world in order to understand or to explain their behavior

Problem Solving

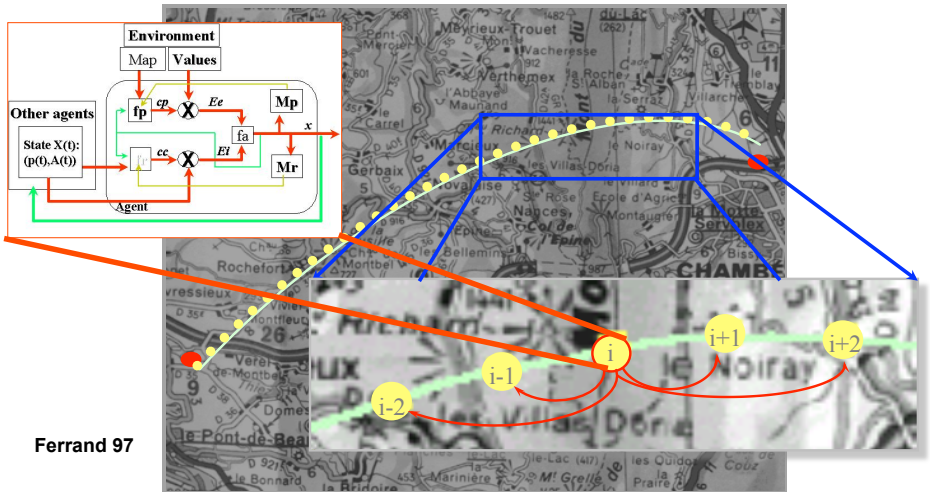
- Properties of the targeted applications:
  - Absence of global strategies, of global solving method
  - Interaction between local strategies, between local solving methods
  - Solution is the result of the interaction between local processes (points of view, decisions, goals, motivations, …)
  - Continuous functioning and evolution
- Main requirements:
  - Decentralisation, local control, interactions
  - Openness, permeability, scalability in size or structure
  - Shared and dynamic environment
  - Emergence of the solution

Example (1/2)



Ferrand 97

Example (1/2)

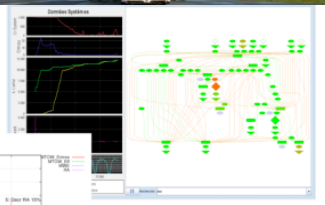


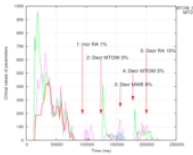
Ferrand 97

Example (2/2)

Design of Complex Systems

- Multi-Disciplinary Simulation & optimisation (ID4CS)
- Design of complex system :
  - Multi-level, Multi-disciplinary
  - Multi-methods
  - Multi-objectives, Multi-attributes
  - Uncertainty
- Cooperation methods between optimisation technics,
- Management of uncertainty
- Multi-* problem solving
- Emergence



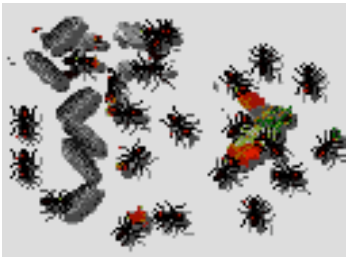


MAS Action domains

- **Socio-technical Systems**
  - Integration of software applications, with humans, organizations and the physical world
  - Making them interoperate, interact, cooperate in a flexible and consistent manner with each other
- **Problem Solving**
  - Modeling and solving problems by cooperation between local solvers
  - Installing top-down and/or bottom-up (emergent) solving process
- **Simulation**
  - Modeling and reproducing complex phenomena of interacting entities in the real world in order to understand or to explain their behavior

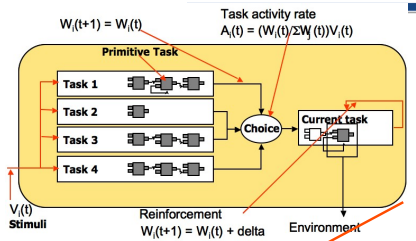
Example (1/2)

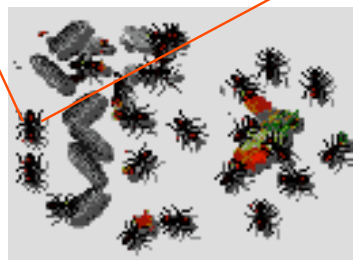
In order to:

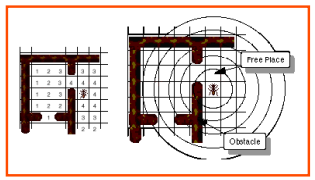


Understand, Explain
Discover, …, Help,

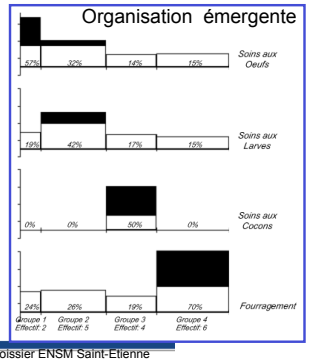
Example (1/2)







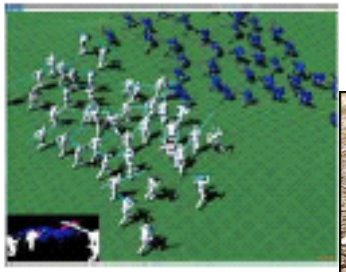
MANTA [Drogoul 93]



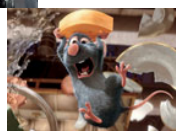
Example (2/2)

http://www.massivesoftware.com/

The Return of the King (2003)
The Two Towers (2002)
The Fellowship of the Ring (2001)



I, Robot (2004)



Ratatouille (2007)

…, Entertainment

Conversational Zeno Robot
http://hansonrobotics.com/

Plan

1. Contexte
2. Definitions
3. Action Domains
4. **Positioning**
5. "Vowels" Dimensions
6. Multi-Agent Engineering
7. Perspectives …

History – Major Steps

- 1980 : Agents in the Artificial Intelligence (AI) area
  - From AI to **Distributed** AI …

  … to Multi-Agent Systems
- 1990 : Agents are invading other domains
  - Personal Assistants, avatars,
  - Mobile Agents,
  - Reactive Agents, ….
- 1995 : Agents spread in other domains, Application domains are enlarging
  - Artificial Life, Economic Agents, …,
  - …, Web, Ambient Computing, …

History – Evolutions

- 1973 - 1980:
  - Hearsay II (1973): blackboard architecture for speech recognition
  - Actor Languages (1973): messages as control structures
  - Beings (1975), Society of Minds (1978)
- 1980 - 1990:
  - Contract Net (1980): hierarchical decentralized control
  - DVMT (1984): Distributed Interpretation
  - Subsumption architecture (1986) : Reactive Robots
  - MACE (1987): multi-agent platforms
- 1990 - … :
  - Self-organisation, emergence, Interactions, organisations, reputation, trust, Agent Oriented Software Engineering, …
  - In 1995, first international conference ICMAS,
  - since 2002, Autonomous Agents + MAS -> AAMAS

Inter-Disciplinary Domain…

- Direct Links with:
  - Programming, Objects...
  - Artificial Intelligence,
  - Distributed Systems, Parallelism,
- But also:
  - Complex System (physics, …, ethology, ecology, …)
  - Artificial Life, Neural networks, …
  - Social Psychology, Sociology, Activity Theory, Economy, ...

Direct Inheritance

- Object Oriented Programming:
  - Encapsulation, modularity : an object encapsulate data and methods that manage them (ex : C++, Java, Smalltalk),
  - Distribution : Distributed objects, CORBA, DCOM
  - → Actor Languages Development
- Artificial Intelligence:
  - Symbolic Reasoning Models (Expert systems, Knowledge Representation), logic, …
  - distribution : Blackboard Architectures
- Distributed Systems

Multi-Agent vs Objects

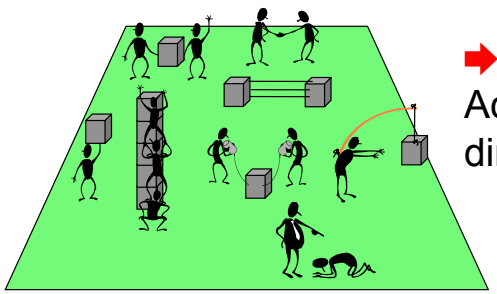
- An agent, as an object, encapsulates a state and behaviors

BUT:

- An agent encapsulates its control over its behaviors; an object has only control over its state
- Interactions among agents have a broader scope than the method calls between objects. Interactions consist in goals, plans, actions, hypothesis exchanges
- An agent may have different control cycles (data-directed, goal-directed, interaction-directed, …)
- A MAS has several control flows. An Object system has, a priori, only one control flow.

Multi-Agent vs Artificial Intelligence (1)

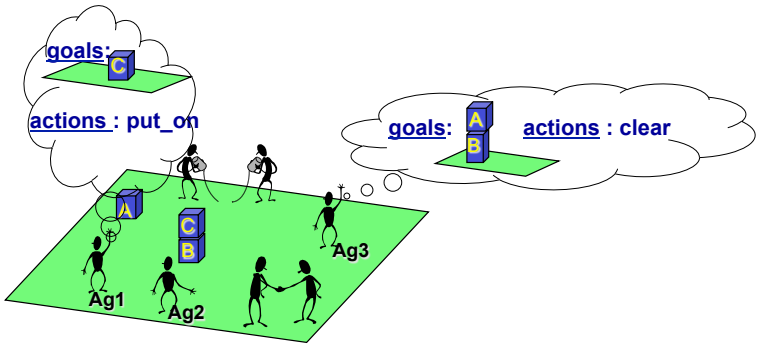
Mono-agent perspective of Artificial Intelligence is pushed away



➡ Knowledge, Goals, Actions gain a **social** dimension

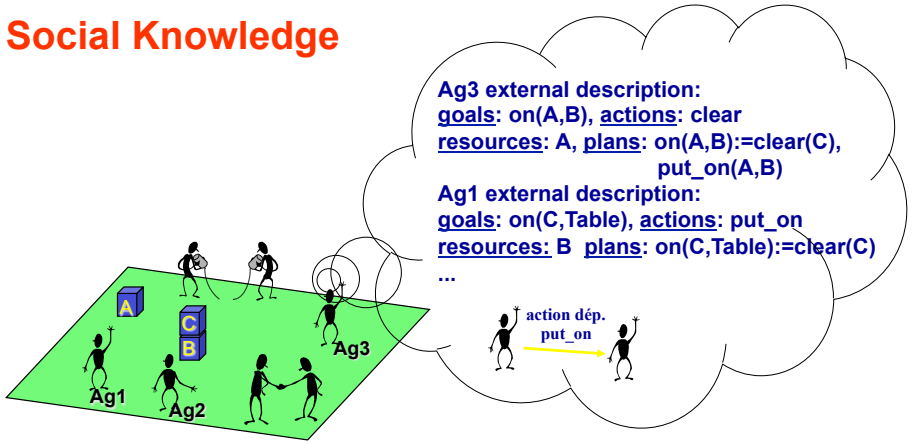
Multi-Agent vs Artificial Intelligence (2)

Ex. dependence networks



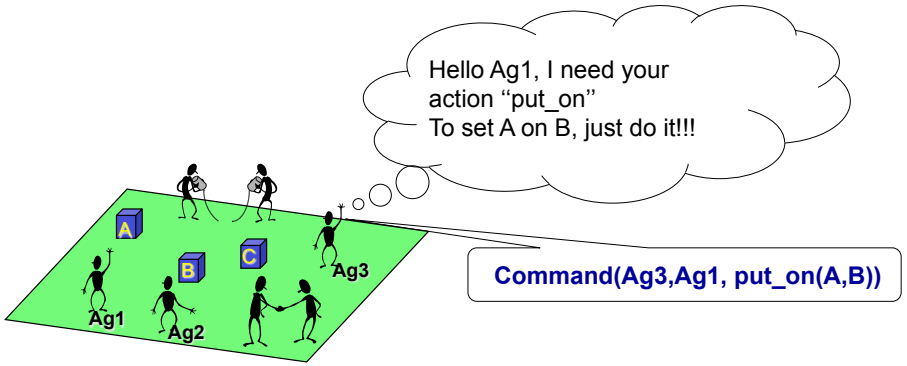
Multi-Agent vs Artificial Intelligence (3)

Social Knowledge



Multi-Agent vs Artificial Intelligence (4)

Social Interaction



Multi-Agent vs Distributed Systems

- Both take into account interconnection and distribution
- In MAS, Interconnection and Distribution are concerned by:
  - The requirement of taking into account the agent autonomy, of developing synchronization and coordination mechanisms to coordinate their activities
  - The requirement to represent and take into account the user interests
  - The requirement to cooperate and to achieve agreements (or even compete) with other systems aiming at achieving their own interests.

A Large Domain!!!

From *Autonomous Agents* to *Multi-Agent Systems*

- Autonomous Robots
- Personal Assistants
- Desktop Agents
- Softbots, Knowbots
- Mobile Agents
- Reactive Agents
- Intelligent Agents, Cooperative Agents, Conversational Agents
- Autonomous Agent in a multi-agent world

Plan

1. Definitions
2. Action Domains
3. Positioning
4. **"Vowels" Dimensions**
5. Multi-Agent Engineering
6. Perspectives …

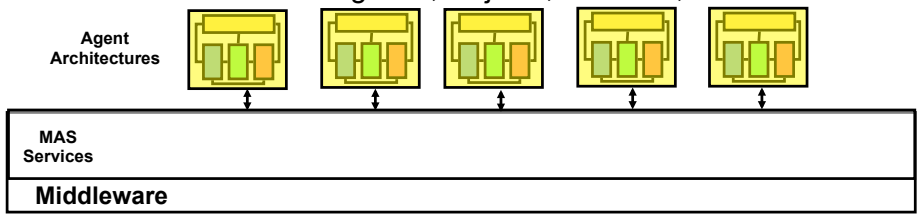
Multi-Agent Models

- Several multi-agent models exist in the literature !
- One possible structuration of the different models:
  - Along four dimensions (''Vowels'' [Demazeau 95]) :
    - **A**gent, **E**nvironment, **I**nteraction, **O**rganisation
  - Taking two main points of view:
    - **global** (System centred), **local** (Agent centred)

Agent Model

Global point of view

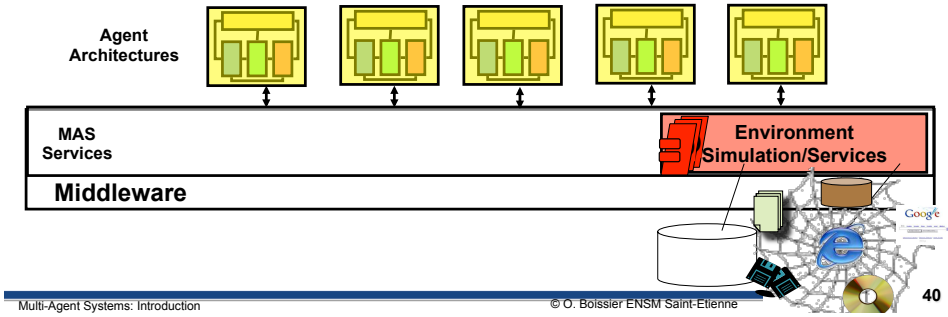
- To define the autonomous entities in the system (internal architectures, knowledge representation, reasoning mechanisms…)
  - BDI Agents (Belief Desire Intention) [Rao 95]
  - Rational Agents [Russel 91]
  - Situated Agents [Agre 87], [Maes 90]
- Sources : Artificial Intelligence, Objects, Robotics, ...



Environment Model

Global point of view

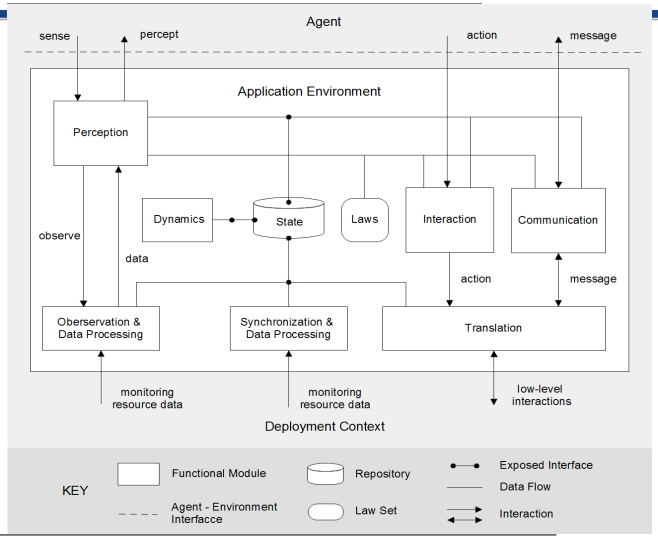
- To define the common space shared among the agents
  - Required explicit abstraction so that agents can be part of a MAS (execution space, coordination space, information space, etc)
  - Active entity that mediates the interaction among agents and provides access to resources
- Sources : Simulation, Physics‡, …



Reference Model [Weyns 06]

Global point of view

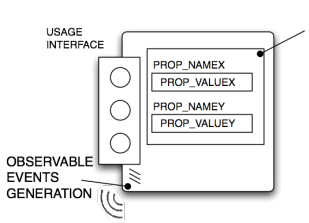
- Perception & Action Modalities,
- Resources, Coordination Artifacts
- Topology
- Proper Dynamic



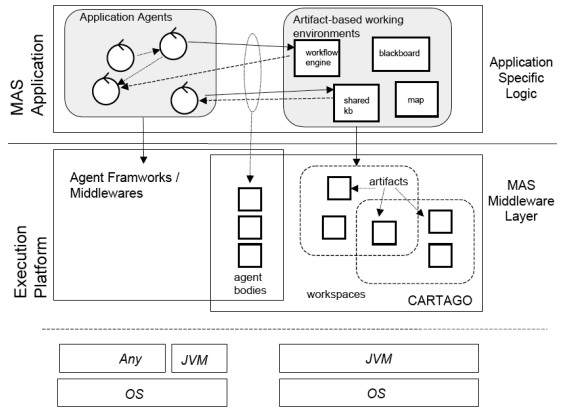
CARTAGO [Ricci 07]

Global point of view

http://apice.unibo.it/xwiki/bin/view/CARTAGO/



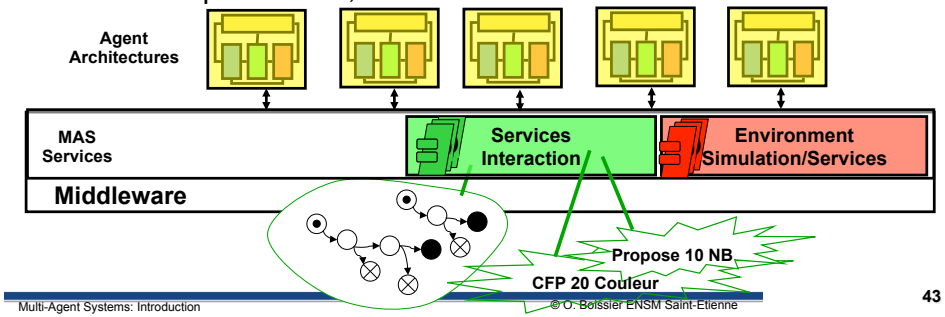
Artefact



Interaction Model

Global point of view

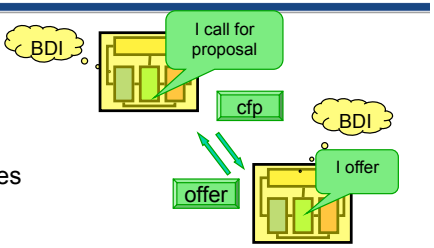
- To define and structure the **dynamic relation** between two or several agents through **reciprocal actions** that may draw patterns of activity (**conversations**)
  - Agent Communication Languages (ACL FIPA, KQML, …), Content Languages, Ontologies
  - Interaction Protocols, Conversations, …
- Sources : Speech Acts, Conversations

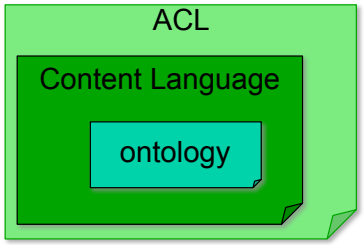


Agent Communication Language

Global point of view

- High level language to share propositional attitudes, to collaborate, Negotiate, ...
  - FIPA-ACL [FIPA 02], KQML [Finin 97], etc.
- Set of performatives based on mental states
  - Inform, request, cfp, agree, understood
- Content Languages
  - Ex: KIF, FIPA-SL, FIPA-CCL, etc.
  - To express actions, objects, propositions
  - Based on ontologies, i.e. common vocabularies relative to a domain *(*weather-ontology, cinema-ontology, etc)





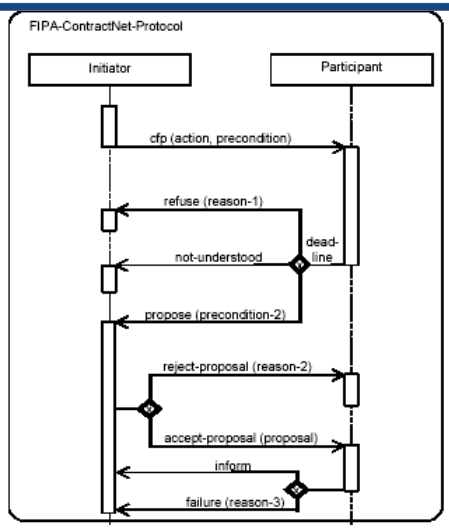
ACL FIPA Message Example

Global point of view

```
(inform                                  communication act
    :sender A                            used for routing the message
    :receiver B                          used for routing the message
    :content (price (bid gooood02) 150)  content
    :in-reply-to round-4
    :reply-with bid04
    :encoding 1000
    :language fipa-sl1                   content language
    :ontology hpl-auction
    :reply-by 10                         deadline for answering
    :protocol offer                      interaction protocol
    :conversation-id conv02              conversation id
)
```


Interaction Protocol Example

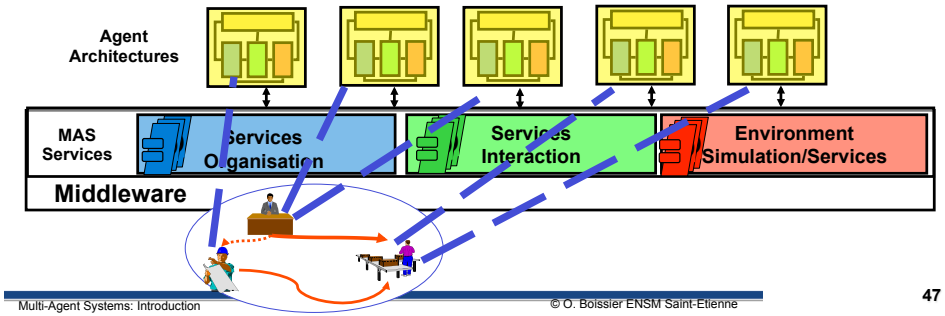
Global point of view



Organization Model

Global point of view

- To define the coordination pattern for achieving a given goal and To make these coordination patterns accessible to the agents
  - Organizational Structures : roles, groups, plans, scenes
- To control and regulate the autonomous activities of the agents
  - RBAC, Norms, Obligations, Permissions, Laws, …
- Sources : sociology, social psychology, CSCW,

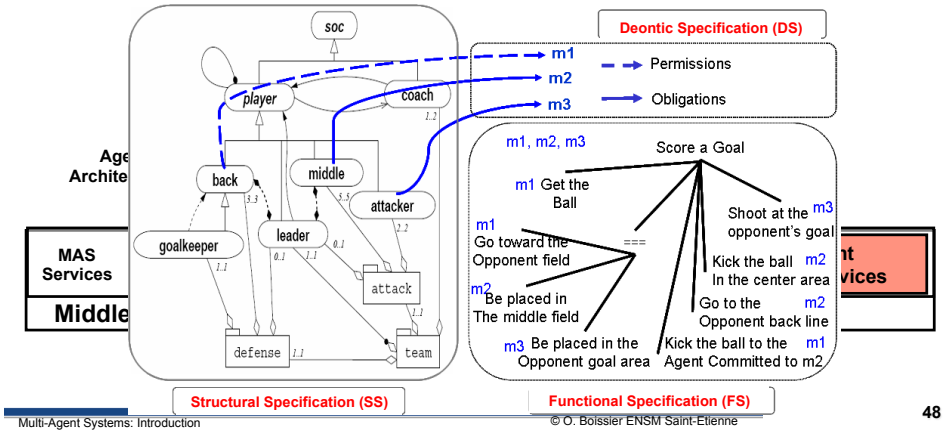


Organization Modeling Language

Global point of view

Declarative definition of the agents organization expressing coordination patterns to achieve a common global goal

Ex : AGR [Ferber 98], Teamwork [Tambe 98], Islander [Esteva 01], Moise+ [Hubner 02], …



Organization Management Platform

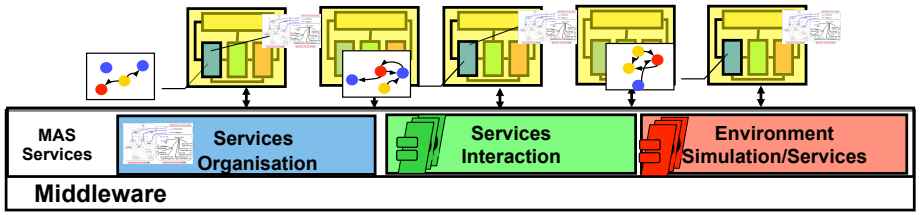
Global point of view

Organization specification written with the organization modeling language

Management of the **organizational entity**

- Within the agents ex : Jason-MOISE [Hubner 07],
- Within components accessible to the agents
  ex : Madkit [Gutknecht 00], Karma [Pynadath 03], Ameli [Esteva 04], S-Moise+ [Hubner 05], SYNAI [Gateau 06], …

http://moise.sourceforge.net

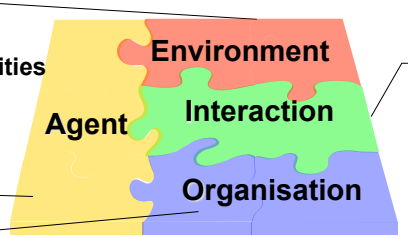


Synthesis

Global point of view

Common space shared among the agents (Signals, Influences, Dynamics, …)

Autonomous entities (Internal Architectures, Knowledge, …)



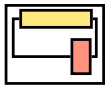
Interactions among the agents (ACL, Interaction Protocols, …)

Coordination Patterns (Organization Modeling Languages, Norms, …)

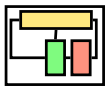
Agent Model

Local point of view

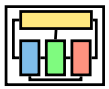
Agent : software or hardware encapsulating processing mechanisms and data, which is able to control its decisions and actions (internal & external), to perceive and act on the environment, to interaction with the other agents and to manage and reason on the relations with other agents and norms.



- Situated Agents
  - agents that reason on themselves and on the environment



- Social Agents
  - situated agents that reason also on the other agents



- Organization Aware Agents
  - social agents that also reason on the organizations/norms in which they are involved

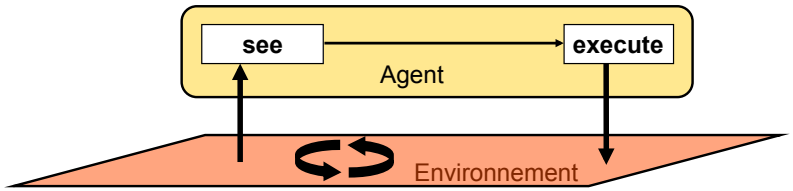
Agent types wrt Control Architecture

Local point of view

- Strength of the coupling of the decision mechanism with external events (environment, interactions, organisation/normes)
  - **Reactive agent**

Reactive Agent Model

- Control Cycle: closed loop involving two operations "execute" and "see" (Stimulus/Response)
- reaction to the evolutions of the environment
- No explicit representation of the environment, of the other agents, of its skills, …
- Decision taken using neither some past history nor future (no planning)



Reactive Agent Model (2)

- Reactive approaches (developed in opposition to AI symbolic reasoning)
- Different approaches based on :
  - behaviours
    - [Brooks 86], [Steels 89], (robotic)
    - [Drogoul 93] (ethology)
  - interactions
    - [Demazeau 93] (image analysis, cartography, …)
    - [Bura 91] (games)
  - situations
    - [Agre 87] (games)
    - [Wavish 90] (design, manufacturing)

Reactive Agent Model (3)

Control Cycle Example (implemented using production rules)

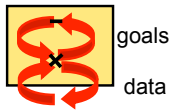
condition-action rules
set of percepts

```
do {
    percepts := see();
        state := interpret_input(percepts);
    rule := match(state,rules);
    execute(rule[action]);
} while (1);
```


Agent types wrt Control Architecture

- Strength of the coupling of the decision mechanism with external events (environment, interactions, organisation/normes)
  - Reactive agent

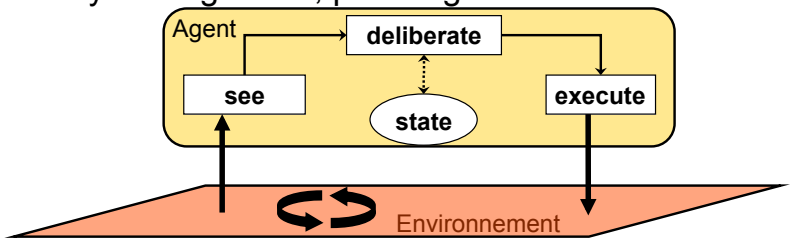
  - **Deliberative agent**



Deliberative Agent Model

Local point of view

- Introduction of “deliberate” function in the control cycle between the "see” and "execute" functions in order to reason and choose the proper action
- Explicit Representation of the environment, of the other agents, of its skills, …
- History management, planning



Deliberative Agent Model (2)

Local point of view

- Example of control cycle

```
s : state,
eq : event queue

s := initialize();
do {
    options := option_generator(eq,s);
    selected := deliberate(options, s);
    s := update_state(selected,s);
    execute(s);
    eq := see();
} while(1);
```


Deliberative Agent Model (3)

Local point of view

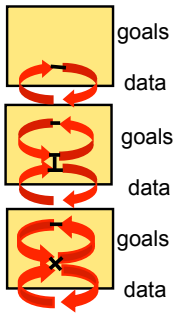
- BDI Control Cycle
  - *b* : beliefs, *g* : desires, *i* : intentions, *eq* : event queue

```
(b,g,i) := initialize();
repeat
    options := option_generator(eq,b,g,i);
    selected := deliberate(options, b,g,i);
    i := selected ∪ i;
    execute(i);
    eq := see();
    b := update_beliefs(b,eq);
    (g,i) := drop_successful_attitudes(b,g,i);
    (g,i) := drop_impossible_attitudes(b,g,i);
forever
```


Agent types wrt Control Architecture

Local point of view

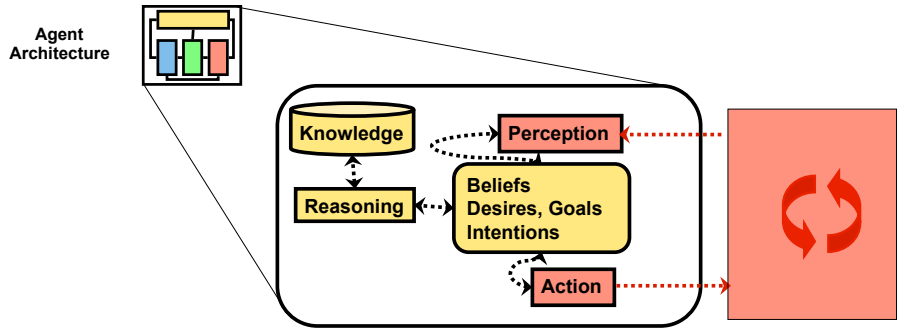
- Strength of the coupling of the decision mechanism with external events (environment, interactions, organisation/normes)
  - Reactive agent
  - **Hybrid agent**
  - Deliberative agent



Situated Agent

Local point of view

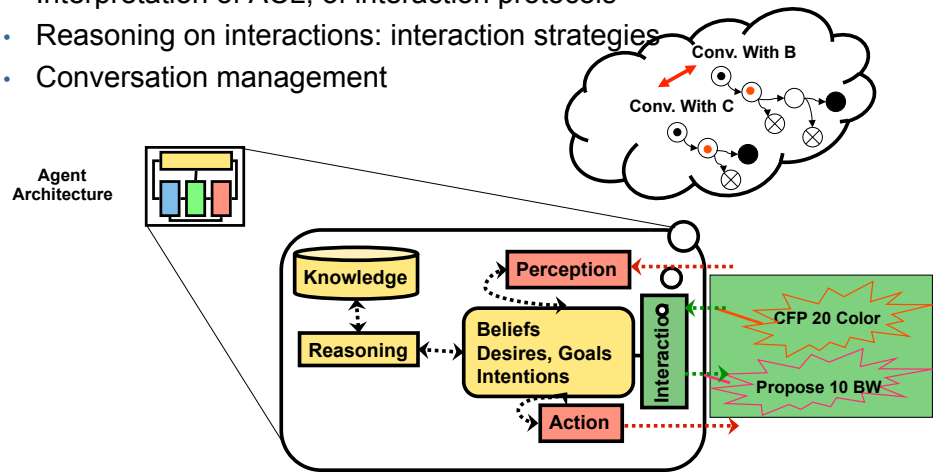
- Action,
- Perception (ex: access to DataBase, internet, …, physical world, virtual world)



Social Agent

Local point of view

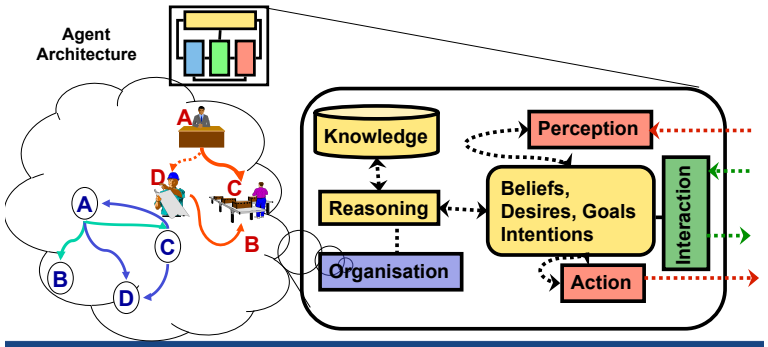
- Interpretation of ACL, of interaction protocols
- Reasoning on interactions: interaction strategies
- Conversation management



Organization Aware Agent

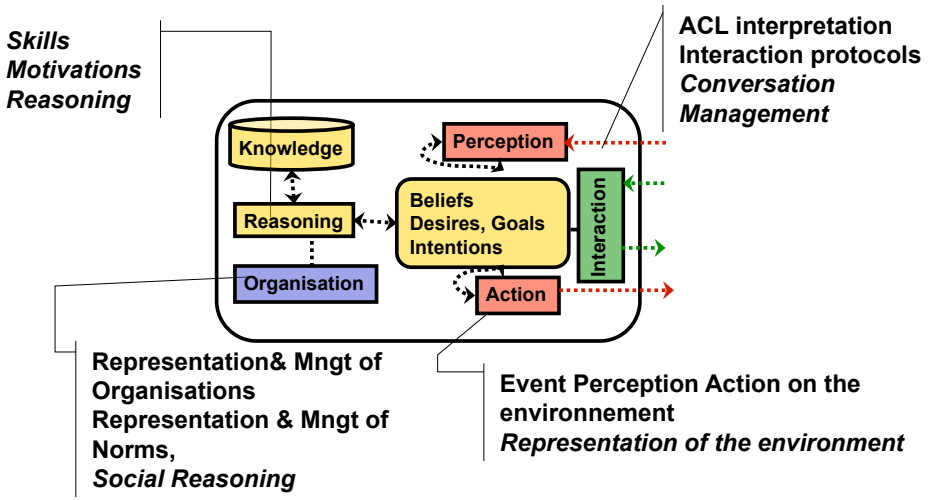
Local point of view

- Reasoning on norms, on organizations [Carabelea 04]
- Contracts, Dependence Networks [Sichman 94]
- Reputation, Trust [Muller 06]

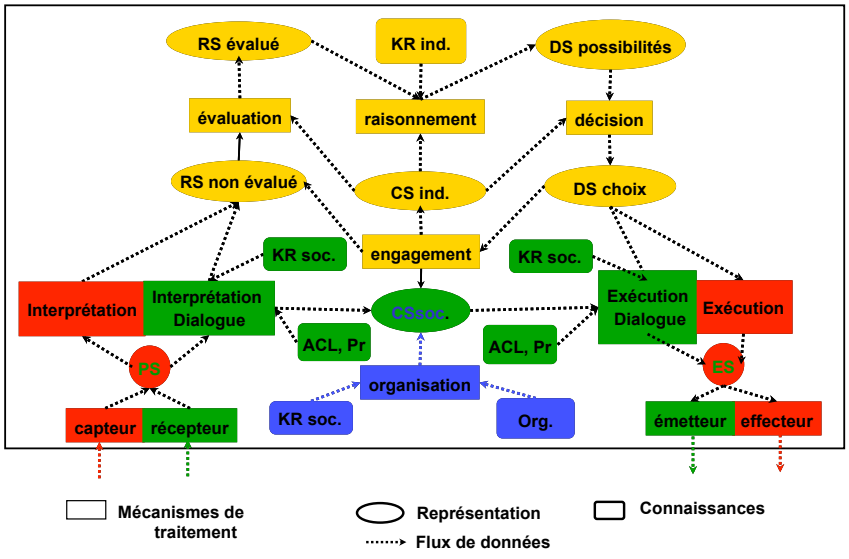


Synthesis

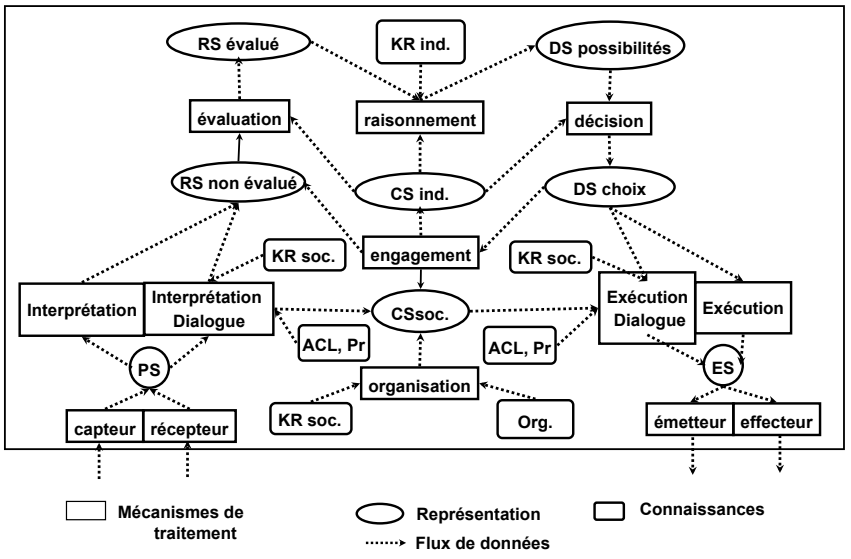
Local point of view



Synthesis



Synthesis

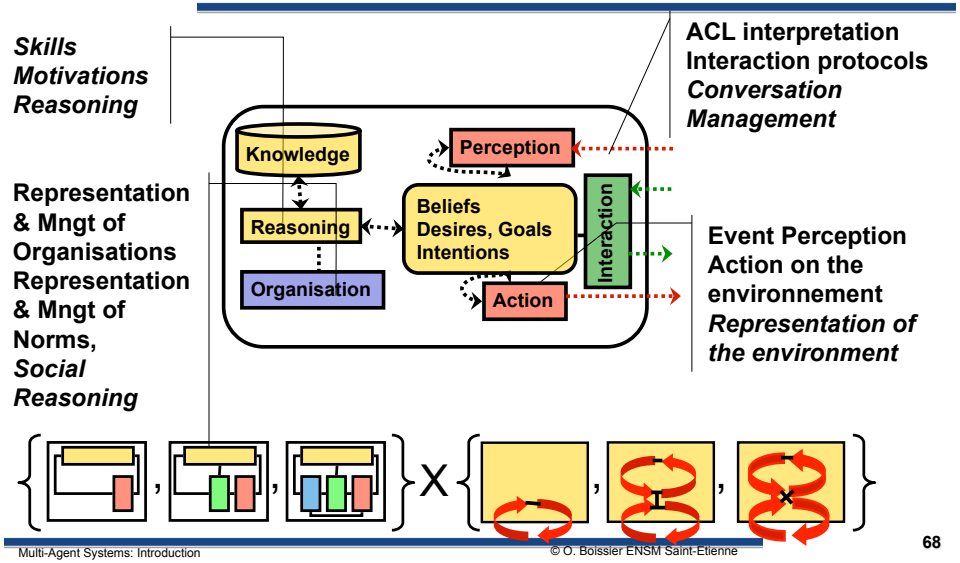


Synthesis

```
Compiled from AgentCore.java
public abstract class mast.core.AgentCore extends java.lang.Thread implements java.io.Serializable
{
    public boolean RUNNING;
    java.io.File log;
    java.lang.StringBuffer logString;
    long timeout;
    java.lang.String name;
    mast.facet.Facet facet[];
    java.lang.Object facetParams[][];
    java.util.Hashtable eventQueue;
    java.util.Hashtable expectingAnswer;
    java.util.Vector subscribeList;
    public mast.core.AgentCore();
    public mast.core.AgentCore(java.lang.String);
    public mast.core.AgentCore(java.lang.String,long);
    public mast.core.AgentCore(java.lang.ThreadGroup,java.lang.String,long);
    public void setFacets(mast.facet.Facet[], java.lang.Object[][]) throws
java.lang.IllegalThreadStateException;

    public abstract void start(java.util.Hashtable, java.io.File);

    public void start(java.io.File);
    public void run();
    public void postEvent(mast.facet.FacetEvent);
    void forwardEvent(mast.facet.FacetEvent) throws mast.facet.InvalidEventException;
    mast.facet.FacetDescription getNextReceiver(mast.facet.FacetEvent) throws
mast.facet.InvalidEventException;
    java.util.Vector getReceiversFor(mast.facet.FacetEvent);
    boolean match(mast.facet.FacetDescription, mast.facet.FacetEvent);
    public void setAnswerTo(mast.facet.FacetEvent, mast.facet.FacetEvent, boolean);
    public void setNoAnswerTo(mast.facet.Facet, mast.facet.FacetEvent);
    public void subscribe(mast.facet.Facet, mast.facet.FacetEventFilter);
    public void unsubscribe(mast.facet.Facet, mast.facet.FacetEventFilter);
    public void unsubscribeAll(mast.facet.Facet);
    public void appendToLog(java.lang.String);
    public void saveLogFile() throws java.io.IOException;
    static {};
}

Method mast.core.AgentCore()
   0 aload_0
   1 aconst_null
   2 ldc #1 <String "Agent Core">
   4 getstatic #2 <Field long DEFAULT_TIMEOUT>
   7 invokespecial #3 <Method mast.core.AgentCore(java.lang.ThreadGroup,java.lang.String,long)>
  10 return

Method mast.core.AgentCore(java.lang.String)
   0 aload_0
   1 aconst_null
   2 aload_1
   3 getstatic #2 <Field long DEFAULT_TIMEOUT>
   6 invokespecial #3 <Method mast.core.AgentCore(java.lang.ThreadGroup,java.lang.String,long)>
   9 return

Method mast.core.AgentCore(java.lang.String,long)
   0 aload_0
   1 aconst_null
   2 aload_1
   3 lload_2
   4 invokespecial #3 <Method mast.core.AgentCore(java.lang.ThreadGroup,java.lang.String,long)>
   7 return
```


Synthesis

Local point of view



Plan

1. Definitions
2. Action Domains
3. Positioning
4. “Vowels” Dimensions
5. **Multi-Agent Engineering**
6. Perspectives …

Multi-Agent Engineering

- Developing multi-agent applications is often a difficult task
  - implementation, distribution, communications, …
- There exists
  - Multiple technologies focused on particular points of a MAS
  - Multiple agent programming languages, dedicated or general purpose based on existing programming languages
  - Multiple multi-agent programming platforms, involving often a specific agent architecture, proposing or not, first order abstractions for the environment, the organisation, the interaction.
  - Multiple standards
  - Multiple software engineering methods for the analysis and design of MAS.

→ Multiple languages, platforms, methods are available …

→ But often limited to a very focused set of applicative domains.

→ Which one to choose? How to choose? How to compare?

Multi-Agent Technologies

Technologies

- Agent Architectures and Theories
- Coalition formation mechanisms
- Multi-Agent Planning
- Agent Communication Languages, Interaction Protocols
- Auction mechanisms
- Negotiation strategies and mechanisms, Argumentation
- Electronic Institutions, Organisations, Norms
- Reputation, Trust
- Mono & multi-agent Learning
- Self-organisation, emergence, …

Declarative Approach

Languages

- CLAIM (Computational Language for Autonomous Intelligent and Mobile Agents)
  - Cognitive Agent Programming Language
  - Belonging to Himalaya Framework (Hierarchical Intelligent Mobile Agents for building Large-scale and Adaptive sYstems based on Ambients)
  - Based on process algebra in order to represent concurrency and agent mobility
  - Based on SyMPA platform implemented in JAVA respecting the MASIF standard
- FLUX :
  - Cognitive Agent Programming Language
  - Fluent Calculus implementation (Action representation formalism)
  - http://www.fluxagent.org

Imperative Approach

- JACK Agent Language (JAL)
  - Developed by Agent Oriented Software
  - Based on PRS, BDI model (similar to hybrid languages such as Jason, 3APL, Jadex)
  - JAL is an extension of Java allowing to create plans, beliefs base, …
  - Possibility to use team of agents, organisation of agents
  - http://www.agent-software.com

Hybrid Approach

- 3APL (An Abstract Agent Programming Language « triple-a-p-l »)
  - Programming language for the development of cognitive agents:
    - By defining structures for beliefs, goals, plans, actions (internal, external or communication) and reasoning rules (modification of plan bases),
    - By reasoning methods to generate plans, revise plans, to achieve goals
    - Integration of Prolog and Java
    - http://www.cs.uu.nl/3apl
- Jason : extended version of AgentSpeak(L) interpreter, agent oriented programming language based on logic. Introduced by Rao.

  - Communication between agents based on Speech-act (beliefs and goals annotated by the information sources)
  - Plans annotations
  - Functions for selecting, for trust computation,
  - Functions and agent architecture may be adapted (perception, belief-revision, inter-agent communication, acting)
  - Integration by the user of existing code by the way of internal
  - Implemented in java, bound to the organisational language MOISE, interfaced with the CARTAGO platform
  - http://jason.sourceforge.net

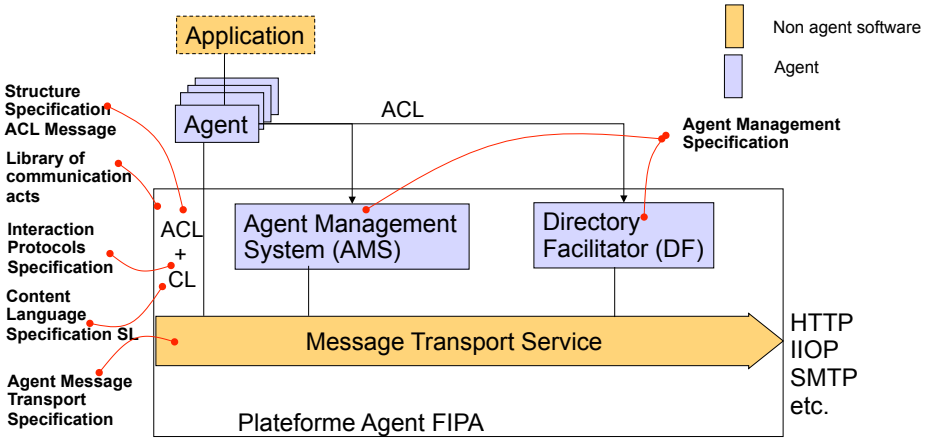
Existing Platforms

- Platforms
  - FIPA compliant
    - FIPA-OS (http://sourceforge.net/projects/fipa-os/)
    - Jade/LEAP (http://jade.tilab.com/)
  - Others :
    - SACI Simple Agent Communication Infrastructure (http://www.lti.pcs.usp.br/saci/)
- Developping Environments
  - Madkit (www.madkit.org)
  - JADEX, BDI agent model based on JADE (http://sourceforge.net/projects/jadex)
  - JACK execution environment, compiler, BDI agent model based on Procedural Reasoning System (PRS) (http://www.agent-software.com)
  - AgentBuilder based on Agent Oriented Program (AOP) (http://www.agentbuilder.com/)
  - AgentTool (http://macr.cis.ksu.edu/projects/agentTool/agentool.htm)
  - ADELFE (http://www.irit.fr/ADELFE/)
- Have a look at *Software Products for MAS*, AgentLink, June 2002

Standards

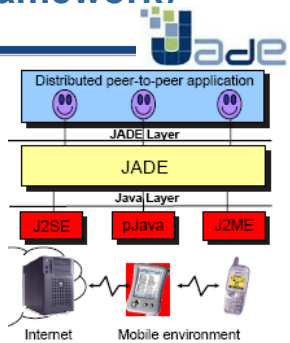
- **Knowledge Sharing Effort** The DARPA Knowledge Sharing Effort
  - http://www-ksl.stanford.edu/knowledge-sharing/
- **MASIF** - OMG (Object Management Group) : OMG effort to standardize mobile agents - middleware services and internal middleware interfaces
  - www.omg.org
- IEEE Computer Society FIPA Standards Committee (**Foundation for Intelligent Physical Agents**)
  - www.fipa.org

FIPA Plateform



ACL = Agent Communication Language

JADE (Java Agent DEvelopment Framework)



- Middleware for developing agent-based P2P application
  - On fixed platforms, smart phones, …
- Two main products :
  - Agent Platform compliant to FIPA specifications
  - API to develop agents in Java
- Open Source Project, LGPL License
- Controlled by Telecom Italia Lab, who owns the project
- Result of the joint effort of multiple actors belonging to the JADE Board (founded in 2003) which missions concern the promotion, the governance and implementation of the changes of the JADE platform

MOTOROLA · WHITESTEIN Technologies · TELECOM LAB · PROFACTOR

- Project portal : http://jade.tilab.com

Standards & Multi-Agent Systems

- Ontologies : DAML, OIL, OWL, …
  - http://www.daml.org
  - http://www.ontoknowledge.org/oil/
  - http://www.w3.org/
- Other standards (De Facto)
  - Jini (www.sun.com/jini),
  - UPnP (www.upnp.org),
  - UDDI (www.uddi.org),
  - Salutation (www.salutation.org)
  - mobility : Aglets (www.trl.ibm.com/aglets/)
  - Web Services (http://www.w3.org/)
  - …

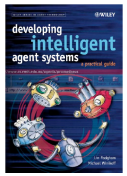
Multi-Agent Methodologies

The engineering of Multi-Agent Systems needs to take into account two levels:

- Multi-Agent System level (System-Centred)
  - Number of agents, Agent Heterogeneity?
  - What is the common medium shared by the agents (Environment)?
  - What are the communication mechanisms between agents?
  - What are the communication languages, the ontologies, the interaction protocols used by the agents?
  - What is the organisation regulating the actions of the agents? How is it established?
  - How do the agents coordinate their actions? How to ensure a consistent behavior?
- Agent level (Agent-Centred)
  - What does an agent represent? What are the kinds of actions encapsulated into an agent?
  - How do the agents represent the environment, the organisation in which they are situated?
  - How do the agents process the interaction with other agents?
  - What is the agent architecture?

Tools supporting methods

- Software Engineering Tools supporting methodologies:
  - MASE AgentTool : macr.cis.ksu.edu/projects/agentTool/agentool.htm
  - ZEUS : sourceforge.net/projects/zeusagent
  - Prometheus PDT : http://www.cs.rmit.edu.au/agents/pdt/
  - PASSI ToolKit : mozart.csai.unipa.it/passi/ptk.htm
  - INGENIAS : grasia.fdi.ucm.es/ingenias/
  - OPM : www.objectprocess.org
- Different ways to model applications:
  - *Agent* Oriented Software Engineering
  - *Environment* Oriented Software Engineering
  - *Interaction* Oriented Software Engineering
  - *Organization* Oriented Software Engineering



Plan

1. Definitions
2. Action Domains
3. Positioning
4. “Vowels” Dimensions
5. Multi-Agent Engineering
6. **Perspectives …**

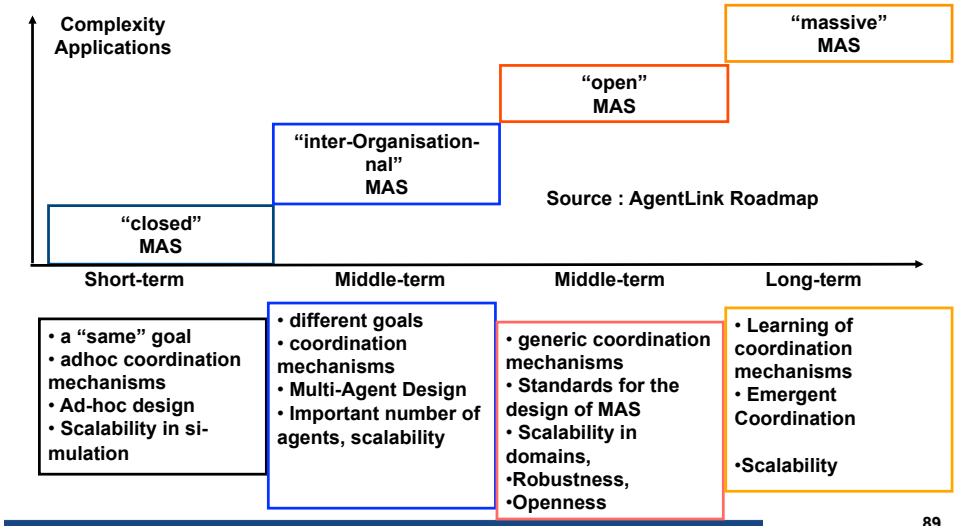
To continue …

- General references
  - Pitfalls of Agent-Oriented Development, M. Wooldridge, N.R. Jennings, Agents ’98, 1998.
  - Foundations of Distributed Artificial Intelligence, G.M.P. Hoare, N.R. Jennings, Wiley & Sons, 1996
  - Les systèmes multi-agents, J. Ferber, InterEditions, 1995
  - Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, edited by Gerhard Weiss, MIT Press, 1999. ISBN 0-262-23203-0
  - Principes et architectures des Systèmes Multi-Agents, J.P. Briot, Y. Demazeau, IC2, Hermès, 2001
- Some standards
  - Knowledge Sharing Effort http://www.cs.umbc.edu/kse/
  - OMG Agent Working Group http://www.objs.com/isig/agent.html
  - FIPA http://www.fipa.org
  - W3C http://www.w3.org
- Some general adresses
  - Collège SMA de l’AFIA : http://sma.lip6.fr
  - AgentLink : http://www.agentlink.org
  - AgentCities : http://www.agentcities.org

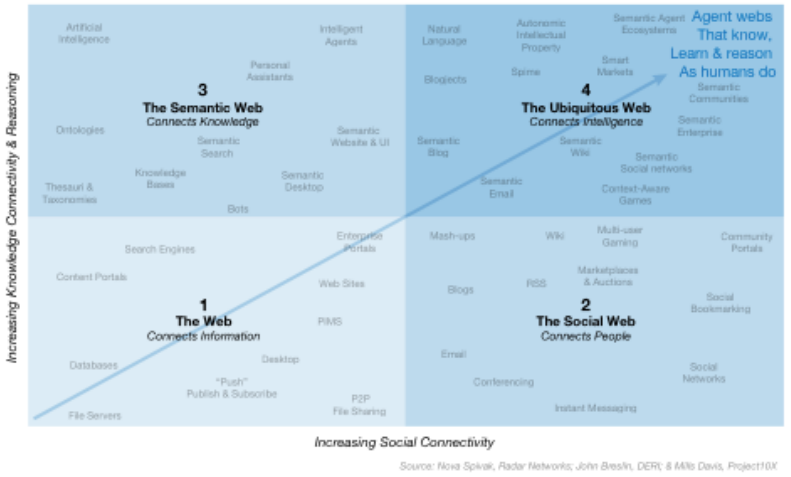
Multi-Agent Modeling

- Multi-model :
  - Articulation of different formalisms
- Multi-viewpoints :
  - Extern/intern, system centred/agent centred
  - Multiple views on a shared world
- Multi-levels
  - Via organisations, via the environment (MAS)
- Multi-scales
  - temporal, spatial, …

Scientific Challenges



Applicative Challenges



Domain Overview (1/2)

- International Conferences
  - International Conference on Multi-Agent System (ICMAS) de 1995 à 2000,
  - International Conference on Autonomous Agents and MultiAgent Systems (AAMAS) depuis 2002. (http://www.aamas-conference.org/)
- French Conferences
  - Journées Francophones SMA (http://www.cerv.fr/jfsma08/)
  - Collège SMA de l'AFIA (http://sma.lip6.fr/)
- European Projects
  - AgentLink (réseau d'excellence www.agentlink.org), Roadmap (www.agentlink.org/roadmap)
- Some ''Success Stories''
  - Brahms (agentsolutions http://agentsolutions.com/home.htm) @ NASA Ames Research Center
  - Living Systems (Whitestein technologies http://www.whitestein.com) @ ABX Logistics
  - eSTAR (http://www.estar.org.uk/) intelligent robotic telescope network
  - CalicoJack (http://www.calicojack.co.uk/)
  - Review of Industrial Deployment of Multi-Agent Systems http://agents.felk.cvut.cz/teaching/33ui2/on-aplications.pdf

Domain Overview (2/2)

- Standards
  - FIPA (Foundation for Intelligent Physical Agents) (http://www.fipa.org/)
- Competitions
  - http://www.robocup.org/
  - http://www.rescuesystem.org/robocuprescue/
  - http://tac.eecs.umich.edu/association.html
  - http://www.lips.utexas.edu/art-testbed/

Domain Overview (3/3)

- Journals
  - Autonomous Agents and Multi-Agent Systems
  - Artificial Intelligence
  - Knowledge Engineering Review
  - International Journal of Agent-Oriented Software Engineering (IJAOSE)
  - Web Intelligence and Agent Systems *An International Journal*
- News
  - Agent List
    - http://www.cs.umbc.edu/agentslist/
  - Distributed Artificial Intelligence List
    - DAI-List-Request@ece.sc.edu
  - French list
    - sma@loria.fr
    - http://sma.lip6.fr/

Bibliography

- [Agre 87] Agre, P. E. and D. Chapman (1987). " Pengi: An Implementation of a Theory of Activity" . AAAI-87. The Sixth National Conference on Artificial Intelligence, Menlo Park, CA., Morgan Kaufman, Los Altos, CA
- [Bayardo 97] R.J. Bayardo et al.. InfoSleuth: Agent-Based semantic integration of information in open and dynamic environments. Proc. ACM SIGMOD Intern. Conf. on Management of Data, 1997.
- [Brooks 86] R. Brooks, J.H. Connel: *Asynchronous Distributed Control System for a mobile robot*, SPIE 727 Mobile Robots, 1986
- [Bura 91] "Eco-Résolution: un Modèle de Résolution par Interactions" Bura S., Drogoul A., Ferber J. & Jacopin E. Actes de RFIA, Lyon, 1991.
- [Carabelea 04] Cosmin Carabelea, Olivier Boissier, Cristiano Castelfranchi: Using Social Power to Enable Agents to Reason About Being Part of a Group. ESAW 2004: 166-177
- [Casare & Sichman, 05] S. J. Casare and J. S. Sichman, Towards a functional ontology of reputation, Proceedings of AAMAS'05, 2005.
- [Castelfranchi & Falcone, 98] C. Castelfranchi and R. Falcone, Principles of trust for MAS: Cognitive anatomy, social importance and quantification. Proc of ICMAS'98, pages 72-79, 1998.
- [Chavez 96] K. Chavez, P. Maes, Kasbah: an agent marketplace for buying and selling goods. Proceedings of PAAM'96, Practical Application
- [Conte & Paolucci, 02] R. Conte and M. Paolucci, Reputation in Artificial Societies. Social Beliefs for Social Order, Kluwer Academic Publishers, G. Weiss (eds), Dordrecht, The Netherlands, 2002.

Bibliography

- [Demazeau 95] Y.Demazeau: *From interactions to collective behaviour in agent-based systems.* In Proc. of the 1st European Conf. on Cognitive Science, Saint Malo, France, April, 1995, p. 117-132.
- [Demolombe & Lorini, 08] R. Demolombe and E. Lorini, Trust and norms in the context of computer security: a logical formalization. Proc of DEON'08, LNAI, 1998.
- [Durfee 97] Durfee, E. H., Kiskis, D. L., and Birmingham, W.P., "The Agent Architecture of the University of Michigan Digital Library", IEE/British Computer Society Proceedings on Software Engineering (Special Issue on Intelligent Agents) 144(1), February 1997.
- [Esteva 01] Marc Esteva, Julian A. Padget, Carles Sierra: Formalizing a Language for Institutions and Norms. ATAL 2001: 348-366
- [Esteva 04] Marc Esteva, Bruno Rosell, Juan A. Rodríguez-Aguilar, Josep Lluís Arcos: AMELI: An Agent-Based Middleware for Electronic Institutions. AAMAS 2004: 236-243
- [Fasli 04] : M. Fasli et M. Michalakopoulos, « e-Game: a generic auction platform supporting customizable market games », IAT 2004.
- [Ferber 98] Jacques Ferber, Olivier Gutknecht: A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems. ICMAS 1998: 128-135
- [Finin 97] KQML as an agent communication language Tim Finin, Yannis Labrou, and James Mayfield, in Jeff Bradshaw (Ed.), ``Software Agents", MIT Press, Cambridge, to appear, (1997)
- [FIPA 02] http://www.fipa.org/repository/aclspecs.html

Bibliography

- [Fullam et al, 05] K. Fullam, T. Klos, G. Muller, J. Sabater-Mir, A. Schlosser, Z. Topol, S. Barber, J. Rosenschein, L. Vercouter and M. Voss, A Specification of the Agent Reputation and Trust (ART) Testbed: Experimentation and Competition for Trust in Agent Societies, Proceedings of AAMAS'05, 2005.
- [Gandon 02] Fabien Gandon, Laurent Berthelot, Rose Dieng-Kuntz, A Multi-Agent Platform for a Corporate Semantic Web, AAMAS 2002, 6th International Conference on Autonomous Agents, 5th International Conference on Multi-Agents Systems, 9th International Workshop on Agent Theories Architectures and Languages, Eds Castelfranchi C., Johnson W.L., p. 1025-1032, July 15-19, 2002, Bologna, Italy.
- [Gateau 06] Benjamin Gâteau: Using a Normative Organisational Model to Specify and Manage an Institution for Multi-agent Systems. EUMAS 2006
- [Gutknecht 00] Olivier Gutknecht, Jacques Ferber: MadKit: a generic multi-agent platform. Agents 2000: 78-79
- [Herzig et al, 08] A. Herzig, E. Lorini, J. F. Hubner, J. Ben-Naim, C. Castelfranchi, R. Demolombe, D. Longin and L. Vercouyter. Prolegomena for a logic of trust and reputation, submitted to Normas 08.
- [Hubner 02] Jomi Fred Hübner, Jaime Simão Sichman, Olivier Boissier: A Model for the Structural, Functional, and Deontic Specification of Organizations in Multiagent Systems. SBIA 2002: 118-128
- [Hubner 05] Jomi Fred Hübner, Jaime Simão Sichman, Olivier Boissier: S-MOISE+: A Middleware for Developing Organised Multi-agent Systems. AAMAS Workshops 2005: 64-78
- [Hubner 07] Jomi Fred Hübner, Jaime Simão Sichman, Olivier Boissier: Developing organised multiagent systems using the MOISE. IJAOSE 1(3/4): 370-395 (2007)

Bibliography

- [Klusch 00] M. Klusch: Information Agent Technology for the Internet: A Survey. Journal Data & Knowledge Engineering, Kluwer Academic, 36(3), 2000
- [Klusch 96] Klusch, M., Shehory, O., 1996, Coalition Formation Among Rational Information Agents, 1996, Proc. of MAAMAW, Eindhoven, LNAI Series Vol. 1038:204-217, Springer-Verlag
- [Klusch 99] M. Klusch (ed.): Intelligent Information Agents. Springer, 1999
- [Klush 01] M. Klusch (ed.) Special issue on Intelligent Information Agents: Theory and Applications, Intelligent Cooperative Information Systems, vol. 10(1&2), March 2001
- [Léger 99] Alain Léger, Marie-Pierre Gleizes, Hans Joachim Einsiedler. ABROSE : A Co-operative Multi-Agent Based Framework for Electronic Marketplace. Dans : InfoWin, ACTS Project Infowin (AC113), Vol. -, p. 21-36, 1999.
- [Maes 04] P. Maes, Agents that reduce work and information overload, Communications of the ACM, Vol. 37, no. 7, July 1994, special issue on Intelligent Agents.
- [Maes 90] Maes, P. (1990). "Situated Agents Can have Goals." Designing Autonomous Agents . Maes, P. (Ed.). Cambridge, MA., MIT Press: 49-70.
- [McBurney 03] P. McBurney et S. Parsons, « Dialogue Game Protocols », Communication in Multiagent Systems, M.-P. Huget (ed.), LNCS 2650, 2003.
- [Mui et al., 02] L. Mui and M. Mohtashemi and A. Halberstadt, Notions of Reputation in Multi-agent Systems: A Review, Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS'02), p. 280-287, 2002, C. Castelfranchi and W.L. Johnson (eds), Bologna, Italy, July, ACM Press, New York, NY, United States of America.

Bibliography

- [Muller & Vercouter, 05] G. Muller and L. Vercouter, Decentralized Monitoring of Agent Communication with a Reputation Model, Trusting Agents for trusting Electronic Societies, LNCS 3577, 2005.
- [Muller 08] G. Muller, L. Vercouter, Trust and Reputation, cours EASSS 08
- [Nodine 99] M. Nodine and J. Fowler. An overview of active information gathering in Infosleuth. Proc. Intern. Conference on Autonomous Agents, USA, 1999.
- [Parsons 03] S. Parsons et P. McBurney, « Argumentation-based Communication between Agents », Communication in Multiagent Systems, M.-P. Huget (ed.), LNCS 2650, 2003.
- [Pynadath 03] David V. Pynadath, Milind Tambe: An Automated Teamwork Infrastructure for Heterogeneous Software Agents and Humans. Autonomous Agents and Multi-Agent Systems 7(1-2): 71-100 (2003)
- [Rao 95] A. S. Rao and M. P. Georgeff, BDI-agents: from theory to practice, Proceedings of the First Intl. Conference on Multiagent Systems, 1995
- [Ricci 07] Alessandro Ricci, Mirko Viroli, Andrea Omicini. "Give Agents their Artifacts": The A&A Approach for Engineering Working Environments in MAS. 6th International Joint Conference "Autonomous Agents & Multi-Agent Systems" (AAMAS 2007)
- [Russel 91] Stuart Russell and Eric Wefald. Do The Right Thing. The MIT Press, Cambridge, Massachusetts, 1991
- [Shardanand 95] U. Shardanand, P. Maes (1995): Social Information Filtering: Algorithms for Automating ``Word of Mouth'', In: Proceedings of the CHI '95

Bibliography

- [Sichman 94] Jaime Simão Sichman, Rosaria Conte, Cristiano Castelfranchi, Yves Demazeau: A Social Reasoning Mechanism Based On Dependence Networks. ECAI 1994: 188-192
- [Smith 80] : R.G. Smith, « *The contract net protocol: High-level communication and control in a distributed problem solver* », IEEE Transactions on Computers, C29 (12), 1980.
- [Sycara 00] K. Sycara, S. Widoff, M. Klusch, J. Lu: LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in Cyberspace. Journal on Autonomous Agents and Multi- Agent Systems, Kluwer, 2000
- [Sycara 99] K. Sycara: In-context information management through adaptive collaboration of intelligent agents. In M. Klusch (ed.), Intelligent Information Agents, Springer, 1999.
- [Tambe 98] Milind Tambe: Implementing Agent Teams in Dynamic Multiagent Environments. Applied Artificial Intelligence 12(2-3): 189-210 (1998)
- [Terziyan 07] Vagan Terziyan, MIT Department, University of Jyvaskyla, AI Department, Kharkov National University of Radioelectronics http://www.cs.jyu.fi/ai/vagan/index.html
- [Van Elst 01] Ludger van Elst and Andreas Abecker: Ontology-Related Services in Agent-Based Distributed Information Infrastructures. In: Proceedings of the Thirteenth International Conference on Software Engineering & Knowledge Engineering, June 13-15, 2001, Buenos Aires, Argentina, pp. 79-85.
- [Walton 07] Christopher Walton. Agency and the Semantic Web. Oxford University Press, 2007.
- [Weyns 06] Danny Weyns, Tom Holvoet: A Reference Architecture for Situated Multiagent Systems. E4MAS 2006: 1-40

Bibliography

- [Wurman 98] : P. Wurman, M. Wellman et W. Walsh, « The Michigan Internet AuctionBot: a configurable auction server for human and software agents », Autonomous Agents 1998.
- [Zeng 97] Zeng, D. and Sycara, K. (1997). Benefits of Learning in Negotiation, In: Proceedings of AAAI-97
- E-Game : http://csres43.essex.ac.uk:8080/elearn/eg/