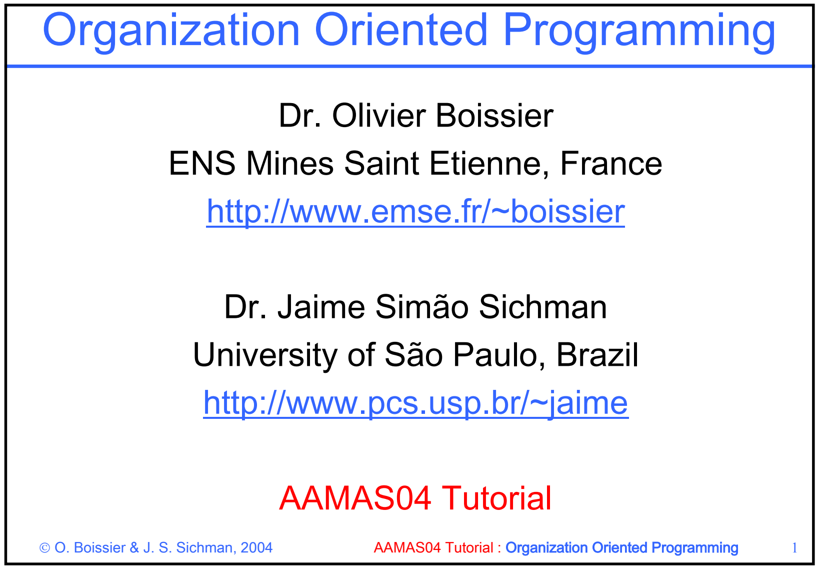
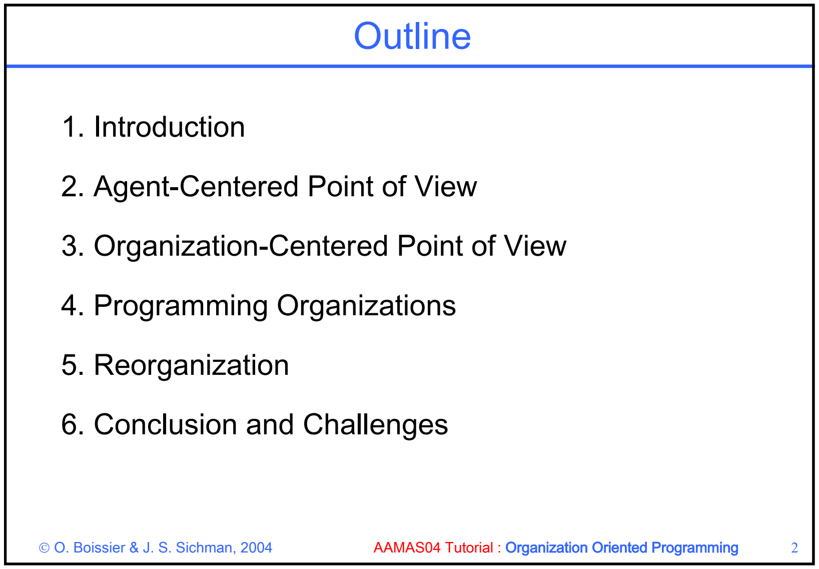


Organization Oriented Programming

Dr. Olivier Boissier
ENS Mines Saint Etienne, France
http://www.emse.fr/~boissier

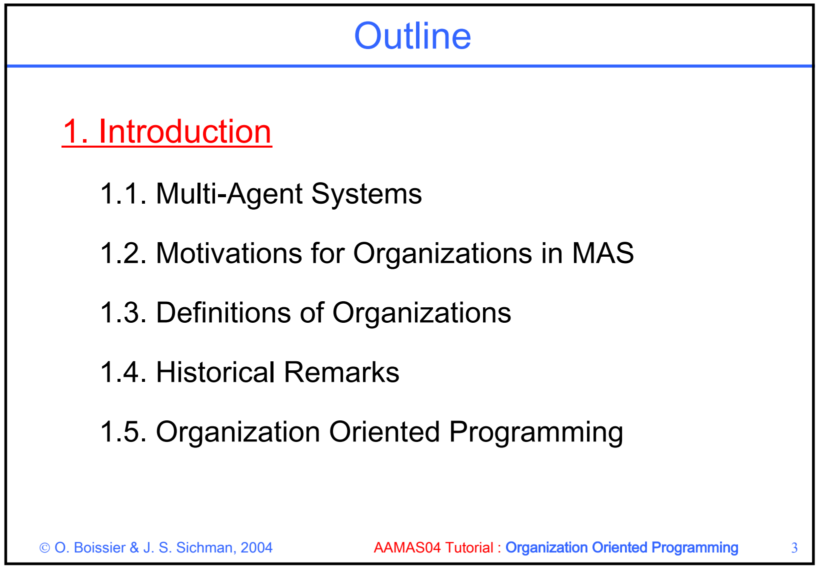
Dr. Jaime Simão Sichman
University of São Paulo, Brazil
http://www.pcs.usp.br/~jaime

AAMAS04 Tutorial



Outline

1. Introduction
2. Agent-Centered Point of View
3. Organization-Centered Point of View
4. Programming Organizations
5. Reorganization
6. Conclusion and Challenges



Outline

1. Introduction

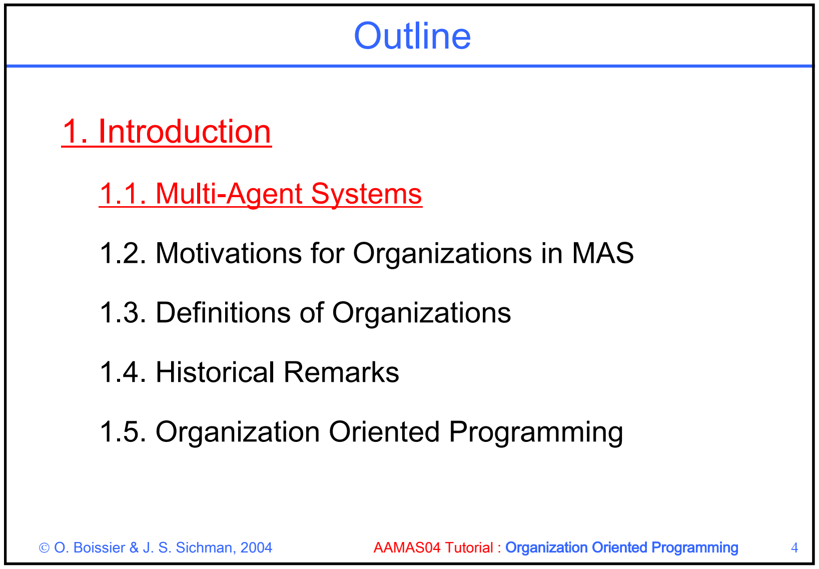
   1.1. Multi-Agent Systems

   1.2. Motivations for Organizations in MAS

   1.3. Definitions of Organizations

   1.4. Historical Remarks

   1.5. Organization Oriented Programming



Outline

1. Introduction

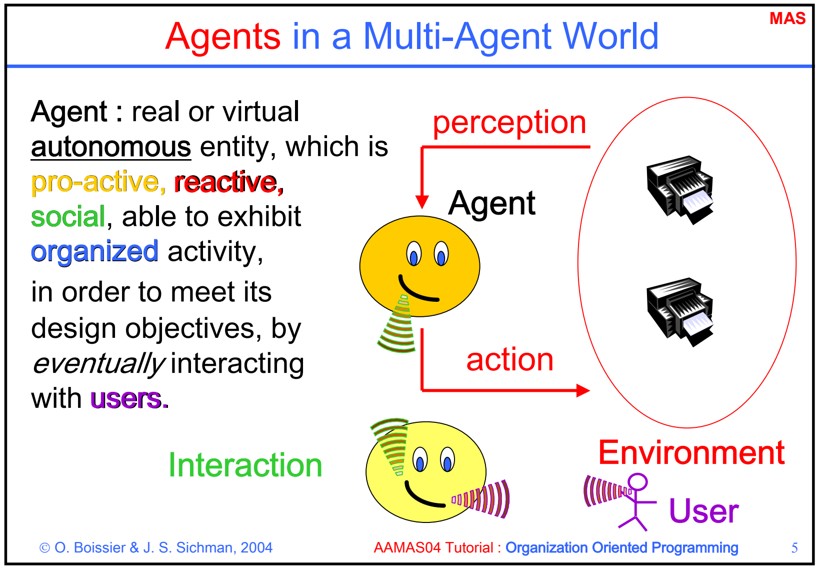
   1.1. Multi-Agent Systems

   1.2. Motivations for Organizations in MAS

   1.3. Definitions of Organizations

   1.4. Historical Remarks

   1.5. Organization Oriented Programming



MAS

Agents in a Multi-Agent World

Agent : real or virtual autonomous entity, which is pro-active, reactive, social, able to exhibit organized activity, in order to meet its design objectives, by *eventually* interacting with users.

perception

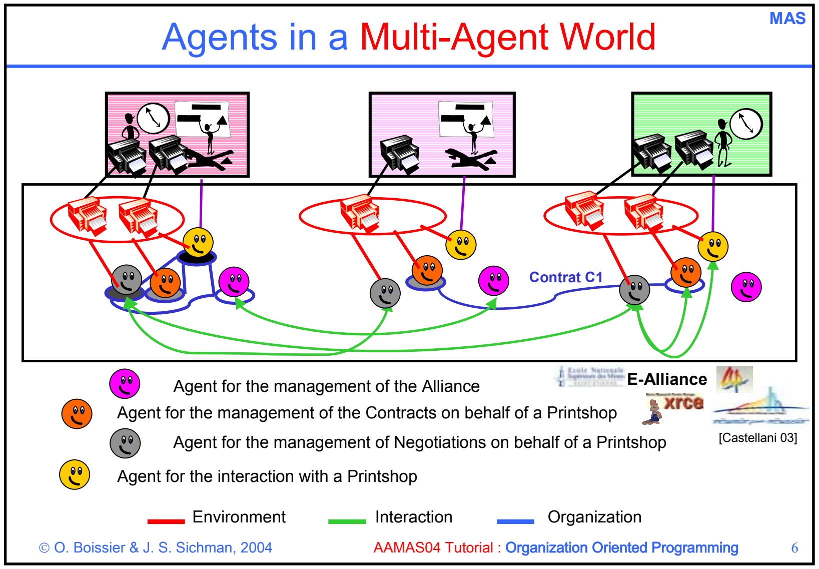
Agent

action

Interaction

Environment

User



MAS

Agents in a Multi-Agent World

Agent for the management of the Alliance

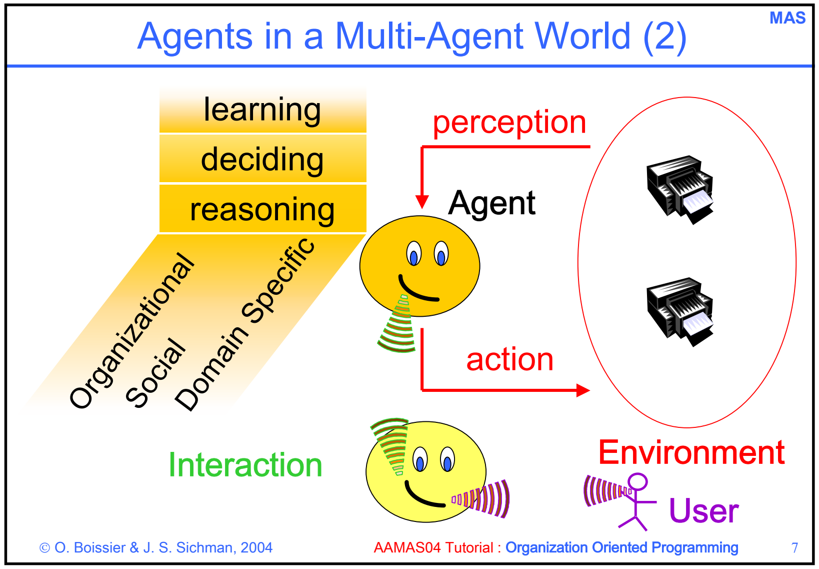
Agent for the management of the Contracts on behalf of a Printshop

Agent for the management of Negotiations on behalf of a Printshop

Agent for the interaction with a Printshop

[Castellani 03]

Environment Interaction Organization



MAS

Agents in a Multi-Agent World (2)

learning

deciding

reasoning

Organizational

Social

Domain Specific

perception

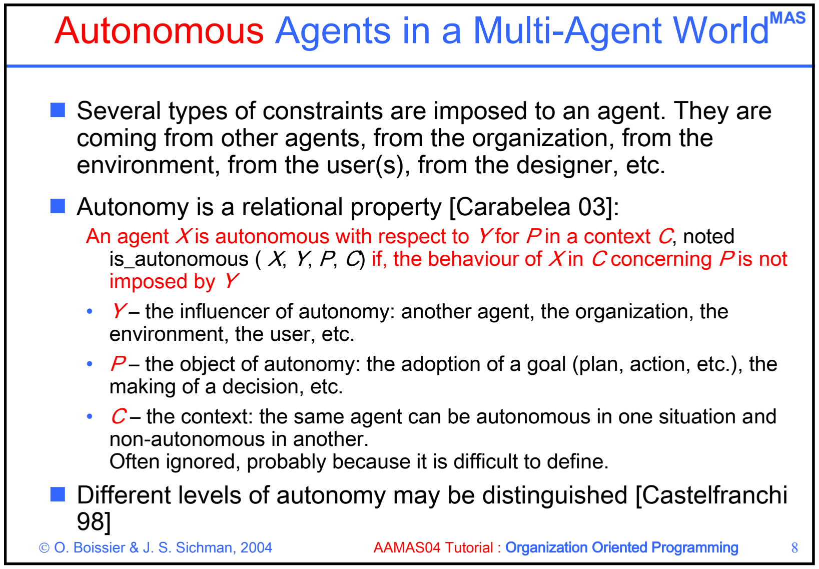
Agent

action

Interaction

Environment

User



Autonomous Agents in a Multi-Agent World

MAS

- Several types of constraints are imposed to an agent. They are coming from other agents, from the organization, from the environment, from the user(s), from the designer, etc.
- Autonomy is a relational property [Carabelea 03]:

  An agent *X* is autonomous with respect to *Y* for *P* in a context *C*, noted is_autonomous (*X*, *Y*, *P*, *C*) if, the behaviour of *X* in *C* concerning *P* is not imposed by *Y*

  - *Y* – the influencer of autonomy: another agent, the organization, the environment, the user, etc.
  - *P* – the object of autonomy: the adoption of a goal (plan, action, etc.), the making of a decision, etc.
  - *C* – the context: the same agent can be autonomous in one situation and non-autonomous in another. Often ignored, probably because it is difficult to define.
- Different levels of autonomy may be distinguished [Castelfranchi 98]

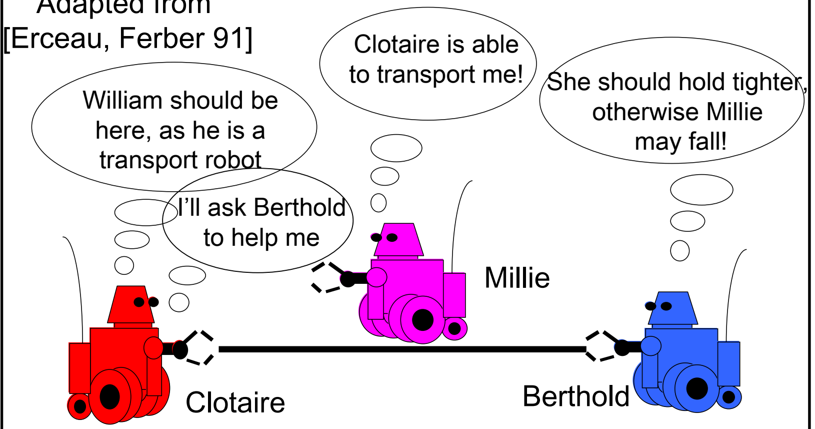
Outline

1. Introduction

- 1.1. Multi-Agent Systems
- 1.2. Motivations for Organizations in MAS
- 1.3. Definitions of Organizations
- 1.4. Historical Remarks
- 1.5. Organization Oriented Programming

From an agent point of view

Adapted from [Erceau, Ferber 91]


William should be here, as he is a transport robot
Clotaire is able to transport me!
She should hold tighter, otherwise Millie may fall!
I'll ask Berthold to help me
Millie
Clotaire
Berthold

From an agent point of view (2)

- Needs to insure a better integration of the agents in the system in order to better adapt themselves to eventual changes in the environment :
  - agents should explicitely represent and exploit (by using internal reasoning mechanisms) the other agents' capacities
- Delegation/Adoption of tasks/beliefs between the agents may produce coalitions,structures that need to be represented, exploited

Despite or Thanks to

- Multiple limitations
  - Cognitive, Physical, Temporal, Institutional,
- Autonomy of the agents
  - agents act autonomously according to their goals, skills,
- Organizations the agents take part in (they should explicitely represent and exploit them)

From a MAS point of view

- Needs to insure a global behavior at the MAS level
  - In terms of cooperation, collaboration, …
  - To be sure that the global goals of the system or collective instance are achieved

Despite or Thanks to

- Multiple limitations
  - Cognitive, Physical, Temporal, Institutional,
- Autonomy of the agents
  - agents act autonomously according to their goals, skills,
- Delegation/Adoption of tasks between the agents that need to be controled

From applications point of view

- Current applications show an increase in
  - Number of agents,
  - Duration and repetitiveness of agent activities,
  - Heterogeneousness of the agents, Number of designers of agents
  - Ability to act, to decide,
  - Action domains of agents, …
- More and more applications require the integration of human communities and technological communities (ubiquitous and pervasive computing), building connected communities (ICities) in which agents act on behalf of users
  - Trust, security, …, flexibility, adaptation

Outline

1. Introduction

1.1. Multi-Agent Systems

1.2. Motivations for Organizations in MAS

1.3. Definitions of Organizations

1.4. Historical Remarks

1.5. Organization Oriented Programming

Intuitive Notions of Organizations

- in everyday life, e.g. an office table, ...
- in ethology, e.g. an ant hill, ...
- in biology, e.g. a cell, ...
- in computer science, e.g. Software/hardware architecture, class diagram, design patterns, information system, ...
- in human society, e.g. a soccer team, a school, an enterprise, a government, ...

What is an Organization ?

a) Organizations are structured, patterned systems of activity, knowledge, culture, memory, history, and capabilities that are distinct from any single agent [Gasser 01]
→ Organizations are supra-individual phenomena

b) A decision and communication schema which is applied to a set of actors that together fulfill a set of tasks in order to satisfy goals while guarantying a global coherent state [Malone 87]
→ definition by the designer, or by actors

c) An organisation is characterized by : a division of tasks, a distribution of roles, authority systems, communication systems, contribution-retribution systems [Bernoux 85]
→ pattern of predefined cooperation

d) An arrangement of relationships between components, which results into an entity, a system, that has unknown skills at the level of the individuals [Morin 77]
→ pattern of emergent cooperation

What is an Organization ?

- Organization is a supra-agent pattern of emergent cooperation or predefined cooperation of the agents in the system, that could be defined by the designer or by the agents themselves.

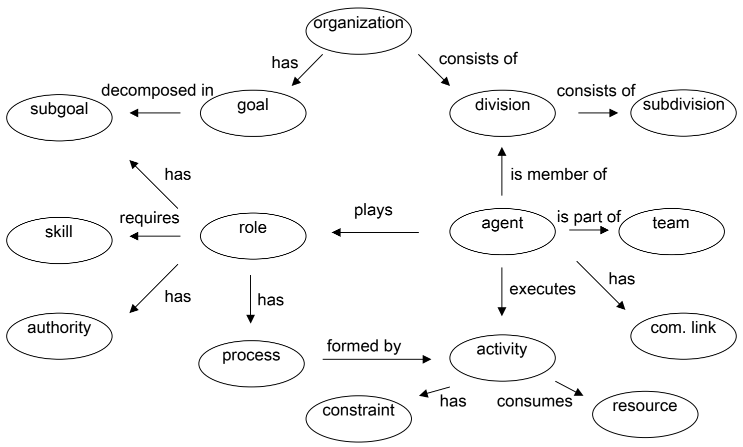
→ Pattern of emergent/potential cooperation

- Organizational entity, institution, social relations, commitments

→ Pattern of predefined cooperation

- Organizational structure, norms, …

Organizational Ontology [Fox et al. 98]



Organization typology [Baeijs 96]

- Centralised
  - » Simple hierarchies : centralized decision,
  - » multi-level hierarchies : decision on different levels
  - » recursive structures : …
- Decentralized
  - » multiple hierarchies :
  - » Market : contractual dimension
- Unstructured
  - » Groups : shared goal, task division, heterarchical decision, several information exchanges
  - » Teams : common environment in which agents interact,
  - » SIG : interest sharing
  - » Communities of practice : grouping of individuals in an independent manner of existing organizations

Outline

1. Introduction

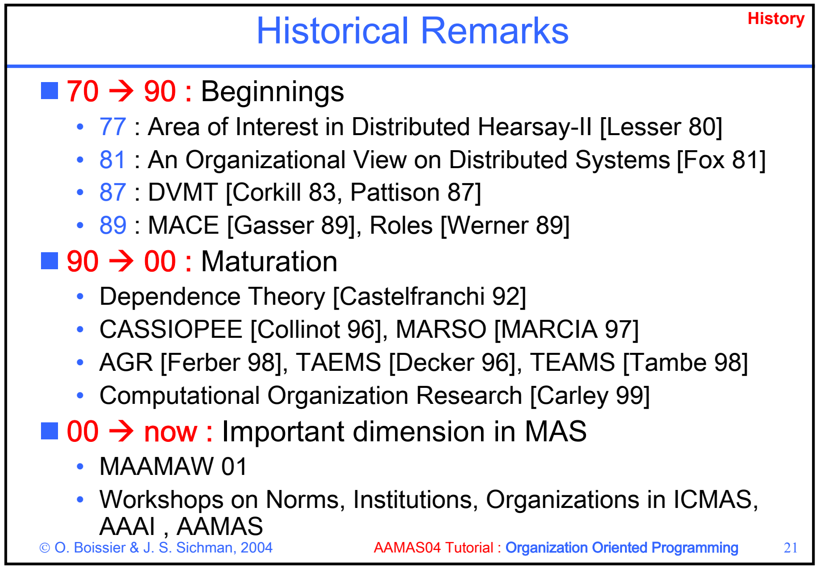
1.1. Multi-Agent Systems

1.2. Definitions of Organizations

1.3. Motivations for Organizations

1.4. Historical Remarks

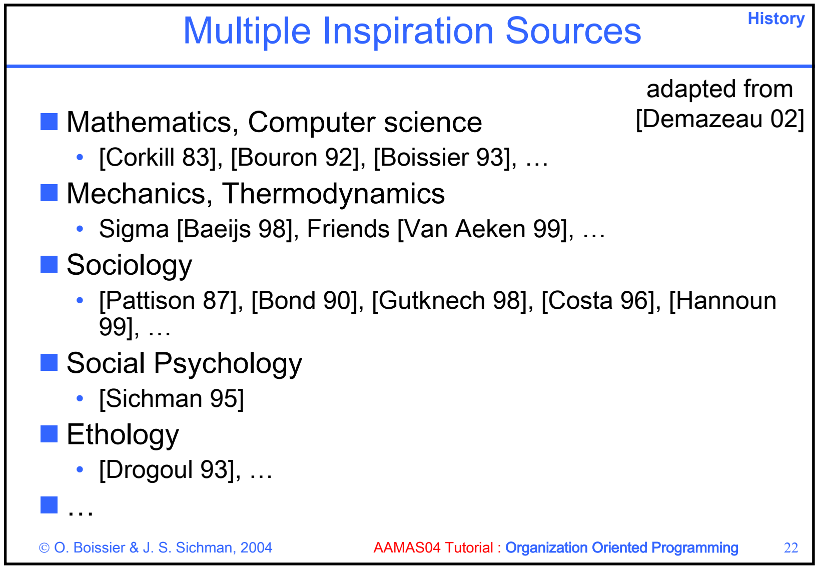
1.5. Organization Oriented Programming



Historical Remarks

History

- 70 → 90 : Beginnings
  - 77 : Area of Interest in Distributed Hearsay-II [Lesser 80]
  - 81 : An Organizational View on Distributed Systems [Fox 81]
  - 87 : DVMT [Corkill 83, Pattison 87]
  - 89 : MACE [Gasser 89], Roles [Werner 89]
- 90 → 00 : Maturation
  - Dependence Theory [Castelfranchi 92]
  - CASSIOPEE [Collinot 96], MARSO [MARCIA 97]
  - AGR [Ferber 98], TAEMS [Decker 96], TEAMS [Tambe 98]
  - Computational Organization Research [Carley 99]
- 00 → now : Important dimension in MAS
  - MAAMAW 01
  - Workshops on Norms, Institutions, Organizations in ICMAS, AAAI , AAMAS

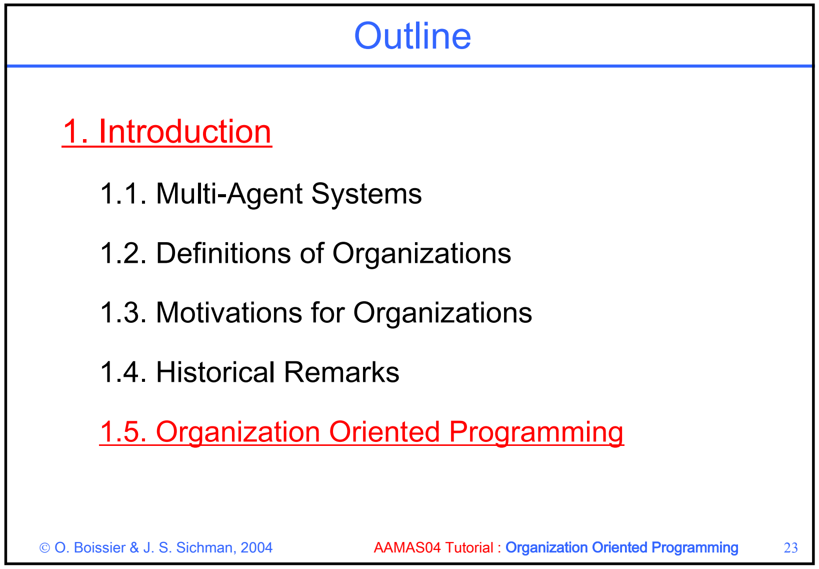


Multiple Inspiration Sources

History

adapted from
[Demazeau 02]

- Mathematics, Computer science
  - [Corkill 83], [Bouron 92], [Boissier 93], …
- Mechanics, Thermodynamics
  - Sigma [Baeijs 98], Friends [Van Aeken 99], …
- Sociology
  - [Pattison 87], [Bond 90], [Gutknech 98], [Costa 96], [Hannoun 99], …
- Social Psychology
  - [Sichman 95]
- Ethology
  - [Drogoul 93], …
- …



Outline

1. Introduction

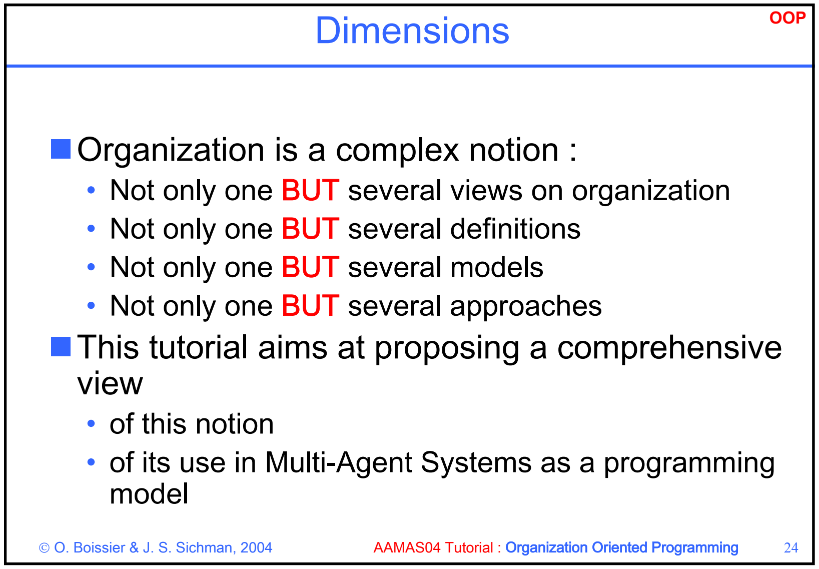
1.1. Multi-Agent Systems

1.2. Definitions of Organizations

1.3. Motivations for Organizations

1.4. Historical Remarks

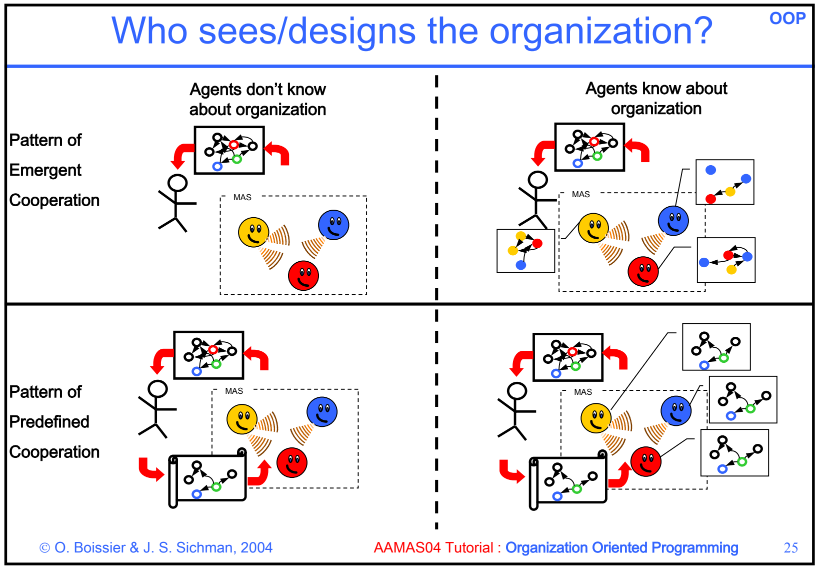
1.5. Organization Oriented Programming



Dimensions

OOP

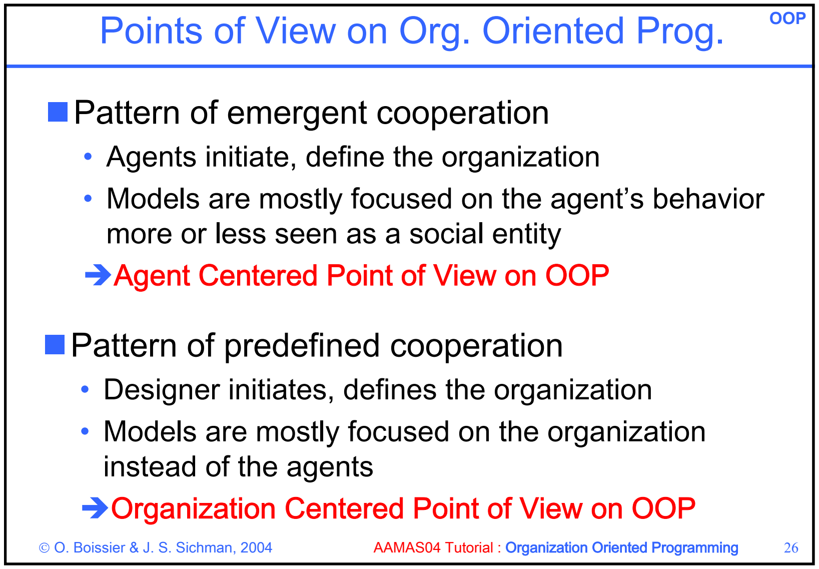
- Organization is a complex notion :
  - Not only one BUT several views on organization
  - Not only one BUT several definitions
  - Not only one BUT several models
  - Not only one BUT several approaches
- This tutorial aims at proposing a comprehensive view
  - of this notion
  - of its use in Multi-Agent Systems as a programming model



Who sees/designs the organization?

OOP

Agents don't know about organization
Agents know about organization
Pattern of Emergent Cooperation
Pattern of Predefined Cooperation
MAS
MAS
MAS
MAS



Points of View on Org. Oriented Prog.

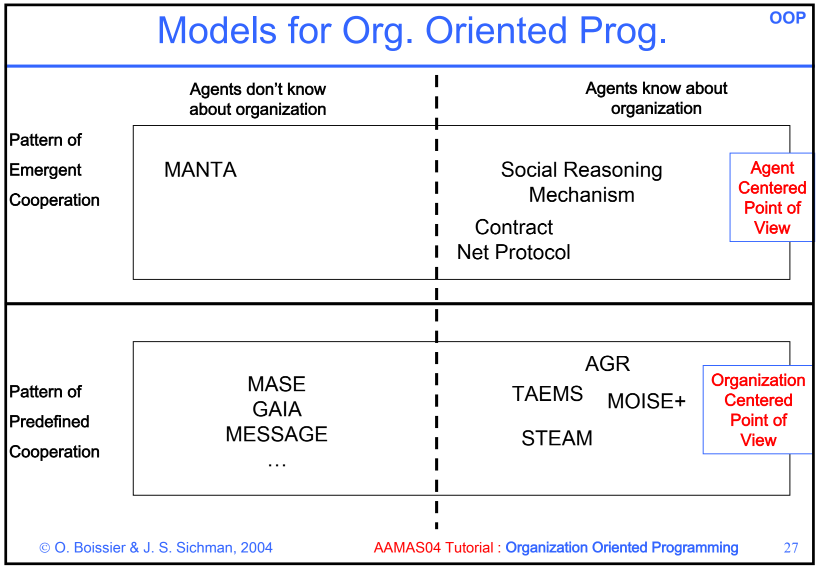
OOP

- Pattern of emergent cooperation
  - Agents initiate, define the organization
  - Models are mostly focused on the agent's behavior more or less seen as a social entity

  ➔Agent Centered Point of View on OOP

- Pattern of predefined cooperation
  - Designer initiates, defines the organization
  - Models are mostly focused on the organization instead of the agents

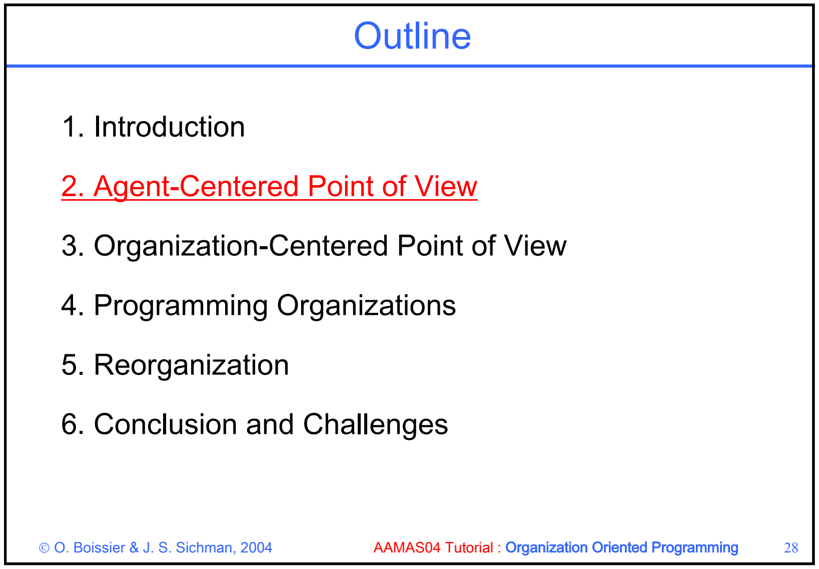
  ➔Organization Centered Point of View on OOP



Models for Org. Oriented Prog.

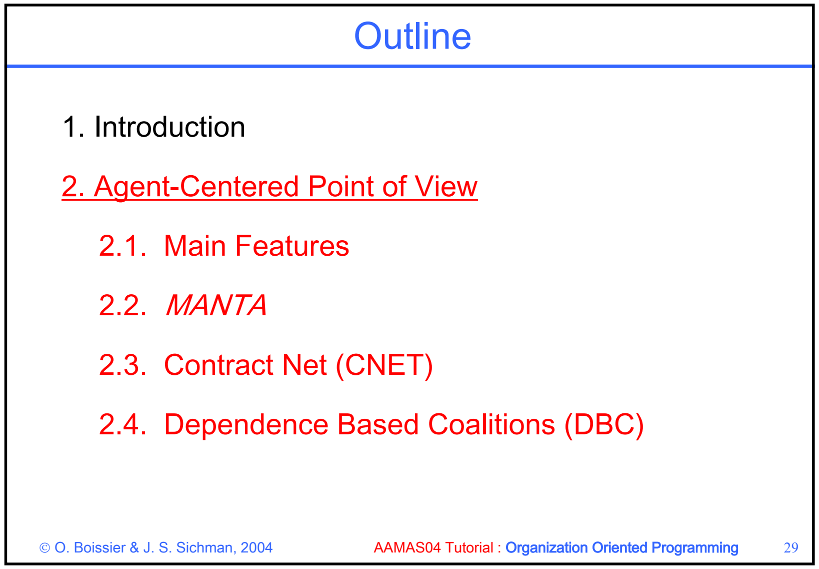
OOP

	Agents don't know about organization	Agents know about organization	
Pattern of Emergent Cooperation	MANTA	Social Reasoning Mechanism Contract Net Protocol	Agent Centered Point of View
Pattern of Predefined Cooperation	MASE GAIA MESSAGE …	AGR TAEMS MOISE+ STEAM	Organization Centered Point of View



Outline

1. Introduction
2. Agent-Centered Point of View
3. Organization-Centered Point of View
4. Programming Organizations
5. Reorganization
6. Conclusion and Challenges



Outline

1. Introduction

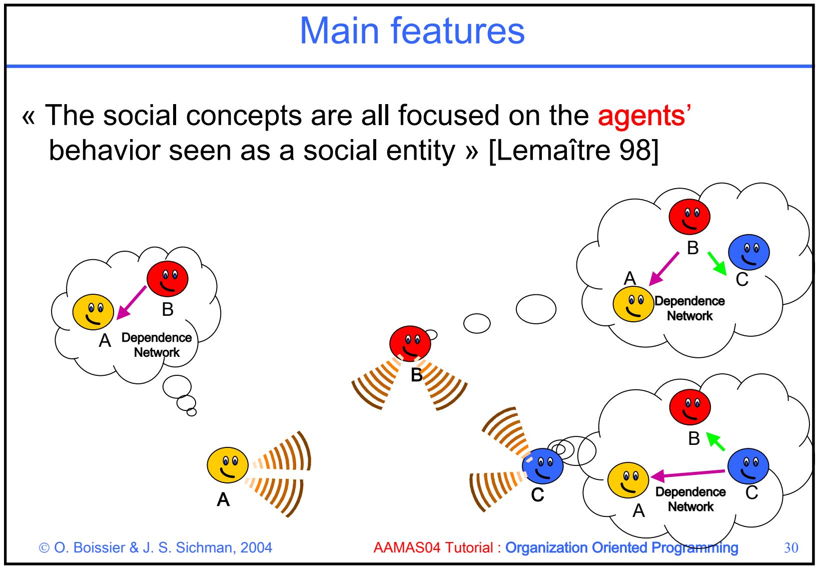
2. Agent-Centered Point of View

   2.1. Main Features

   2.2. *MANTA*

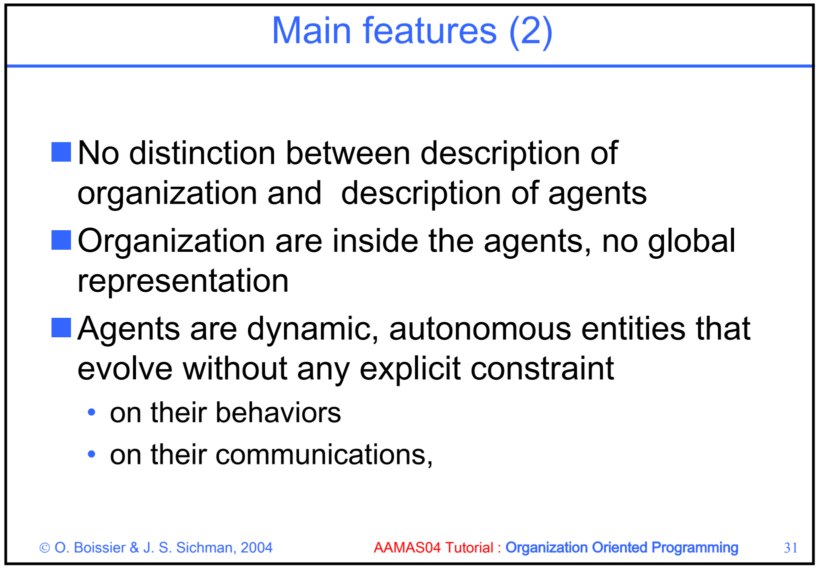
   2.3. Contract Net (CNET)

   2.4. Dependence Based Coalitions (DBC)



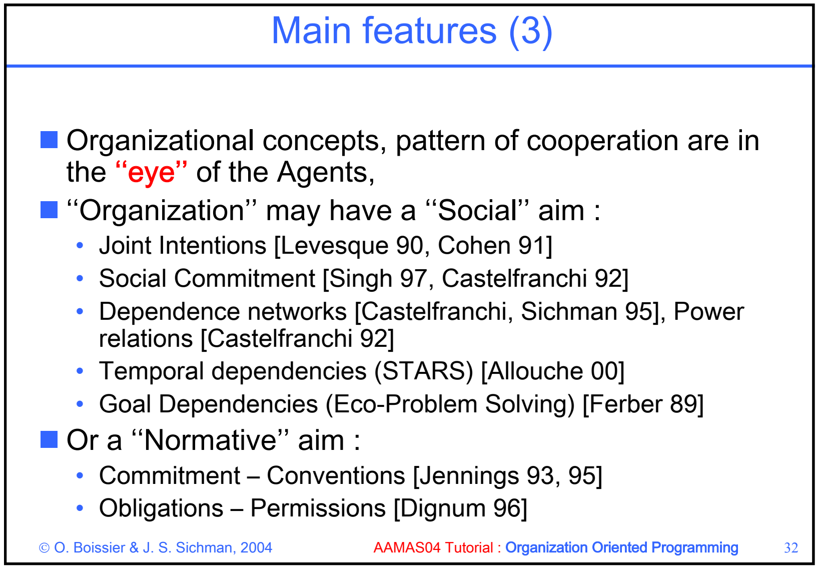
Main features

« The social concepts are all focused on the agents' behavior seen as a social entity » [Lemaître 98]



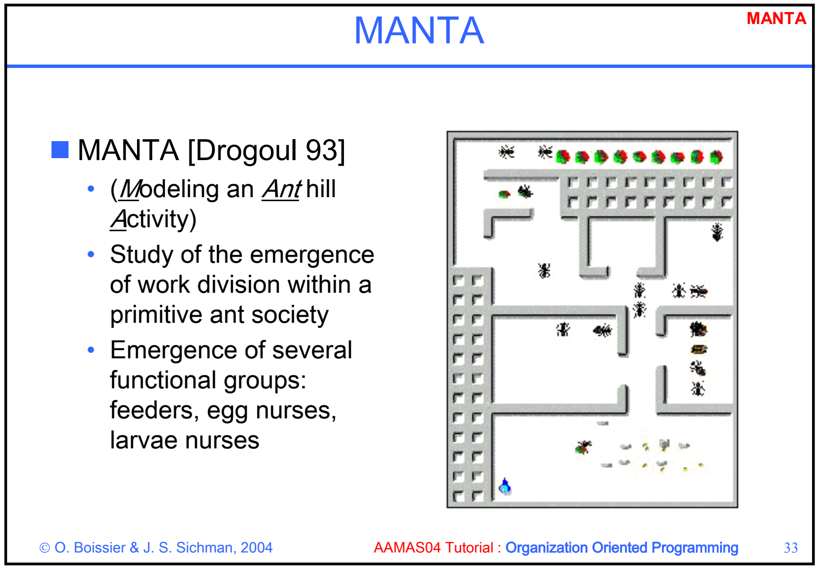
Main features (2)

- No distinction between description of organization and description of agents
- Organization are inside the agents, no global representation
- Agents are dynamic, autonomous entities that evolve without any explicit constraint
  - on their behaviors
  - on their communications,



Main features (3)

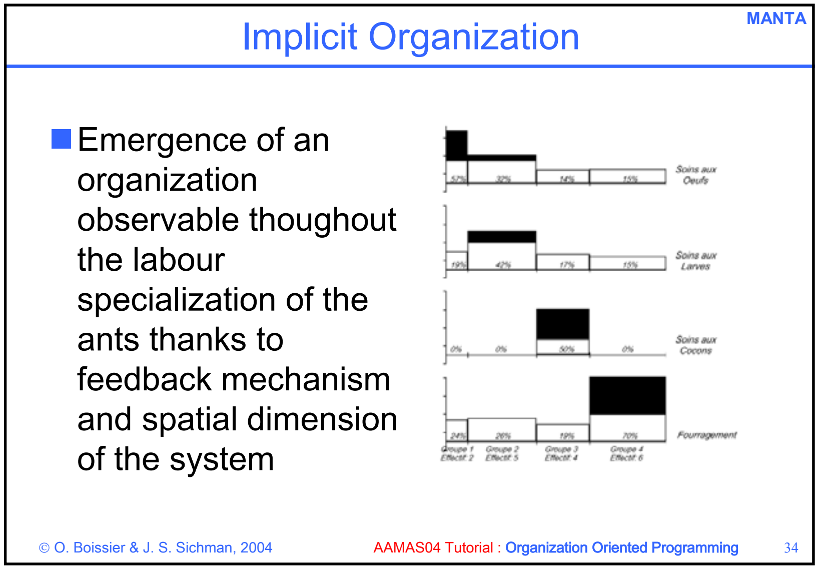
- Organizational concepts, pattern of cooperation are in the "eye" of the Agents,
- "Organization" may have a "Social" aim :
  - Joint Intentions [Levesque 90, Cohen 91]
  - Social Commitment [Singh 97, Castelfranchi 92]
  - Dependence networks [Castelfranchi, Sichman 95], Power relations [Castelfranchi 92]
  - Temporal dependencies (STARS) [Allouche 00]
  - Goal Dependencies (Eco-Problem Solving) [Ferber 89]
- Or a "Normative" aim :
  - Commitment – Conventions [Jennings 93, 95]
  - Obligations – Permissions [Dignum 96]



MANTA

MANTA

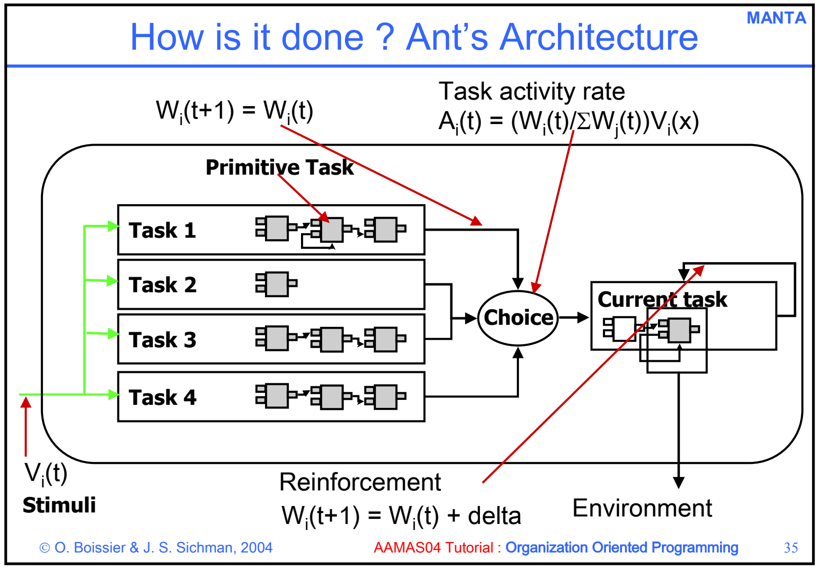
- MANTA [Drogoul 93]
  - (Modeling an Ant hill Activity)
  - Study of the emergence of work division within a primitive ant society
  - Emergence of several functional groups: feeders, egg nurses, larvae nurses



MANTA

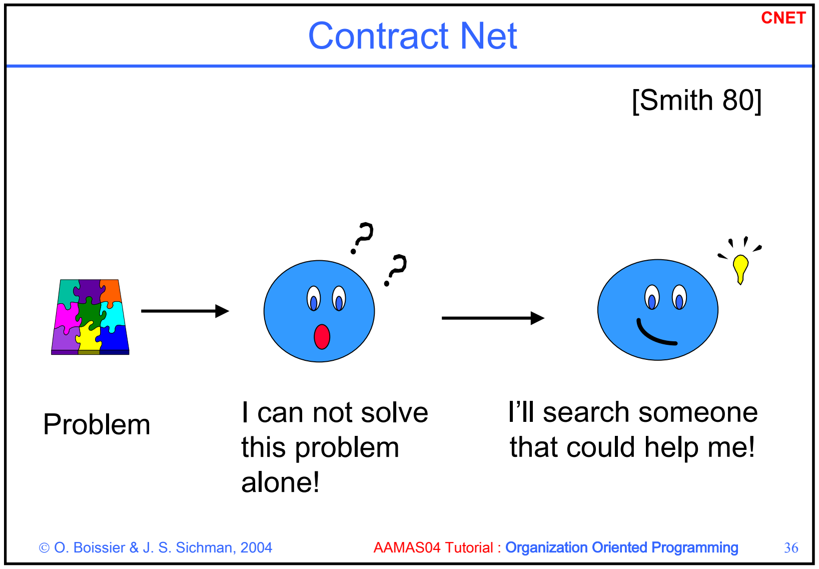
Implicit Organization

- Emergence of an organization observable thoughout the labour specialization of the ants thanks to feedback mechanism and spatial dimension of the system



MANTA

How is it done ? Ant's Architecture



CNET

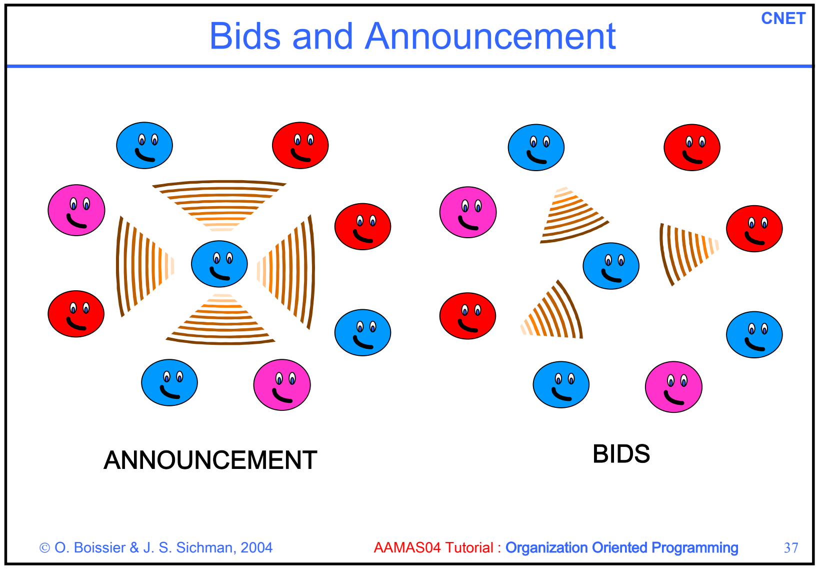
Contract Net

[Smith 80]

Problem

I can not solve this problem alone!

I'll search someone that could help me!



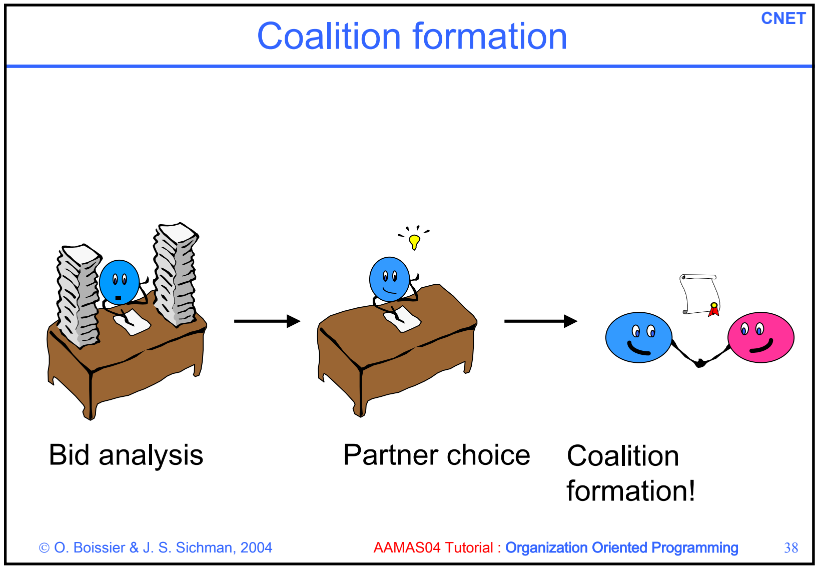
CNET

Bids and Announcement

ANNOUNCEMENT

BIDS

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 37

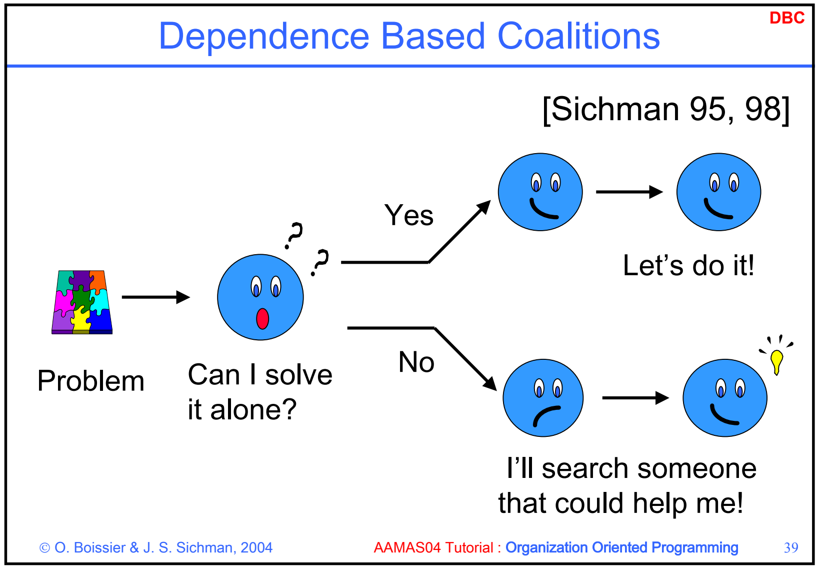


CNET

Coalition formation

Bid analysis → Partner choice → Coalition formation!

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 38



DBC

Dependence Based Coalitions

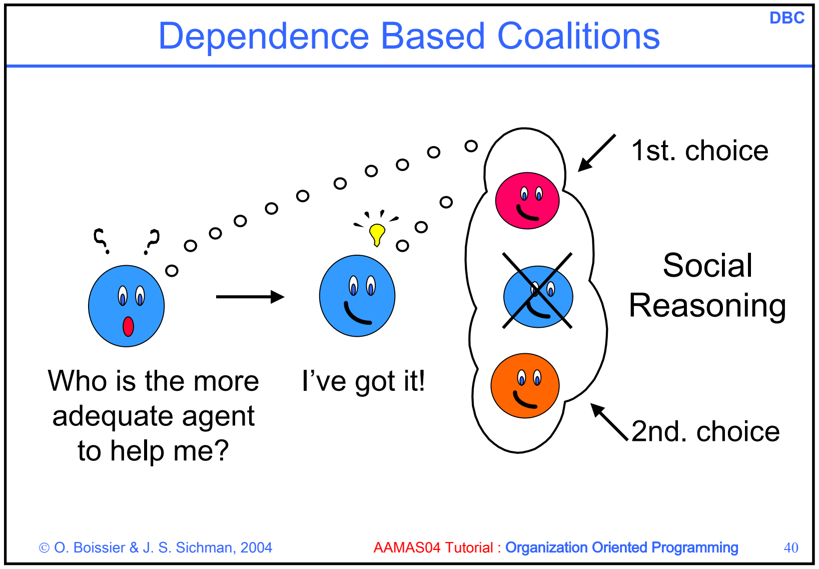
[Sichman 95, 98]

Problem → Can I solve it alone?

Yes → Let's do it!

No → I'll search someone that could help me!

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 39



DBC

Dependence Based Coalitions

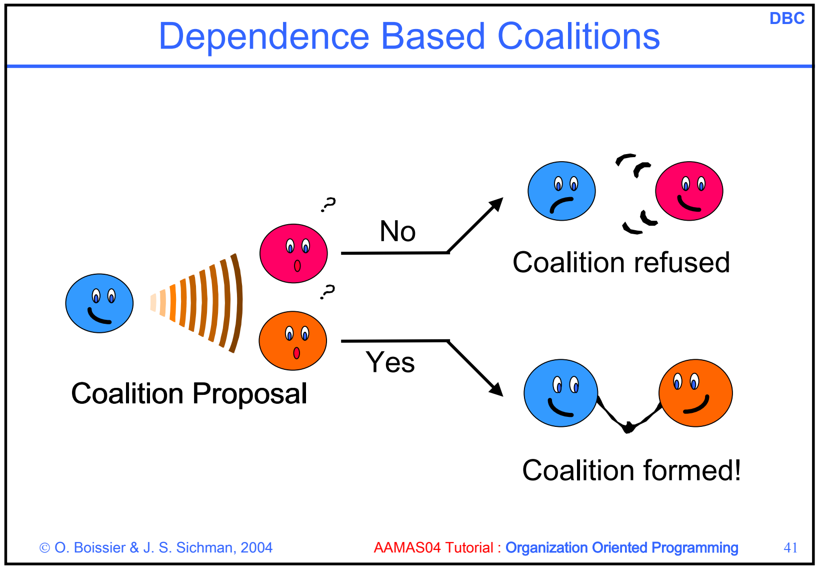
Who is the more adequate agent to help me? → I've got it!

1st. choice

Social Reasoning

2nd. choice

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 40



Dependence Based Coalitions

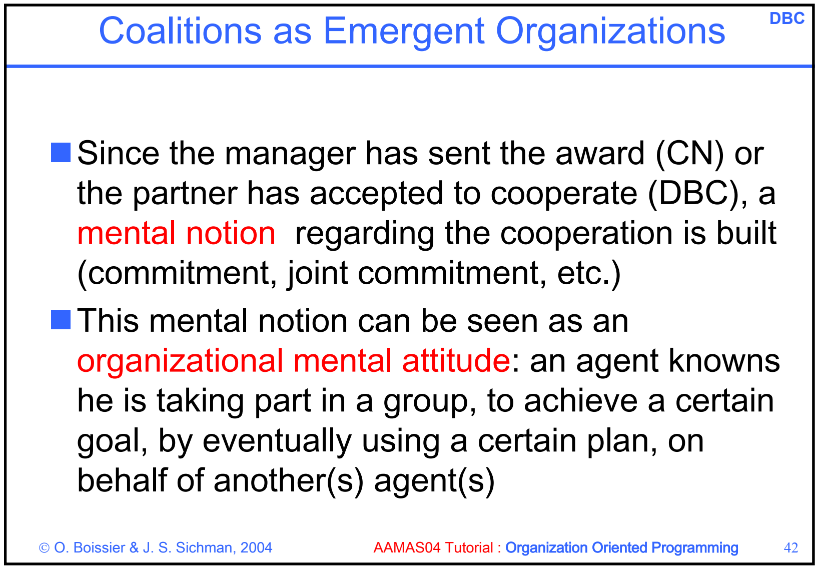
Coalition Proposal

No

Coalition refused

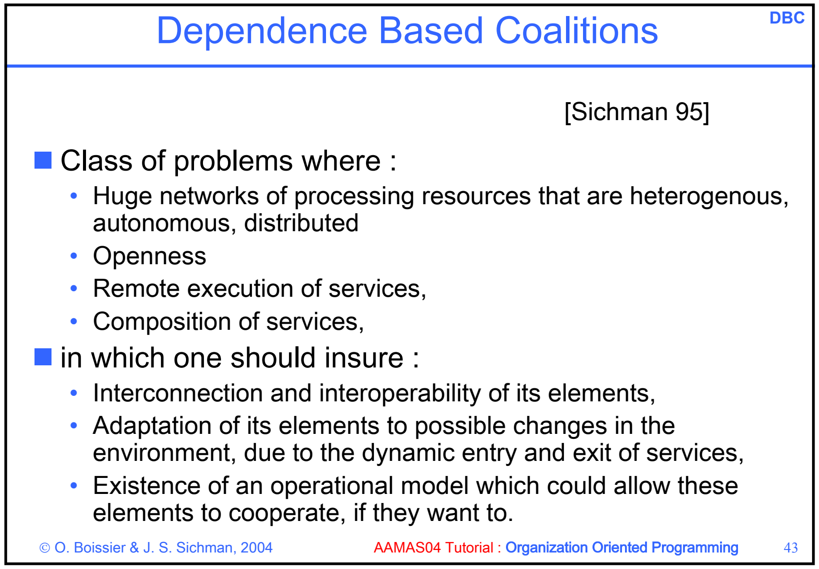
Yes

Coalition formed!



Coalitions as Emergent Organizations

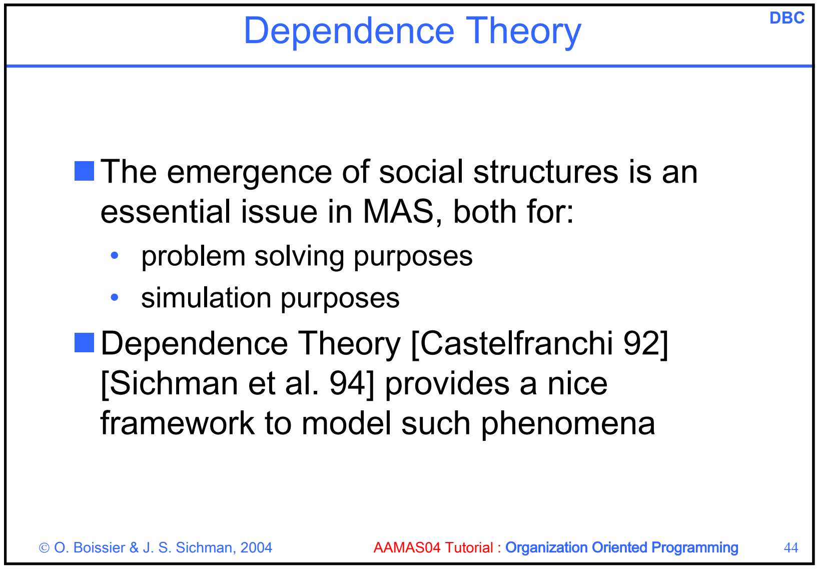
- Since the manager has sent the award (CN) or the partner has accepted to cooperate (DBC), a mental notion regarding the cooperation is built (commitment, joint commitment, etc.)
- This mental notion can be seen as an organizational mental attitude: an agent knowns he is taking part in a group, to achieve a certain goal, by eventually using a certain plan, on behalf of another(s) agent(s)



Dependence Based Coalitions

[Sichman 95]

- Class of problems where :
  - Huge networks of processing resources that are heterogenous, autonomous, distributed
  - Openness
  - Remote execution of services,
  - Composition of services,
- in which one should insure :
  - Interconnection and interoperability of its elements,
  - Adaptation of its elements to possible changes in the environment, due to the dynamic entry and exit of services,
  - Existence of an operational model which could allow these elements to cooperate, if they want to.



Dependence Theory

- The emergence of social structures is an essential issue in MAS, both for:
  - problem solving purposes
  - simulation purposes
- Dependence Theory [Castelfranchi 92] [Sichman et al. 94] provides a nice framework to model such phenomena

Dependence Theory

- Socially situated agents may depend on one another to achieve their *own* goals. In terms of the dependence theory, an agent ag_i depends on some other agent ag_j with regard to one of its goal g_k, when:
  1. ag_i is not autonomous with regard to g_k: it lacks at least one of the actions or resources necessary to achieve g_k, while
  2. ag_j has the missing action/resource

Dependence Theory

- An agent ag_i *depends* on another agent ag_j for a given goal g_k, according to a set of plans P_{qk} if she has g_k in her set of goals, she is not autonomous for g_k and there is a plan p_{qk} in P_{qk} that achieves g_k where at least one action used in this plan is in ag_j 's set of actions.
- An example of a basic dependence relation could be:

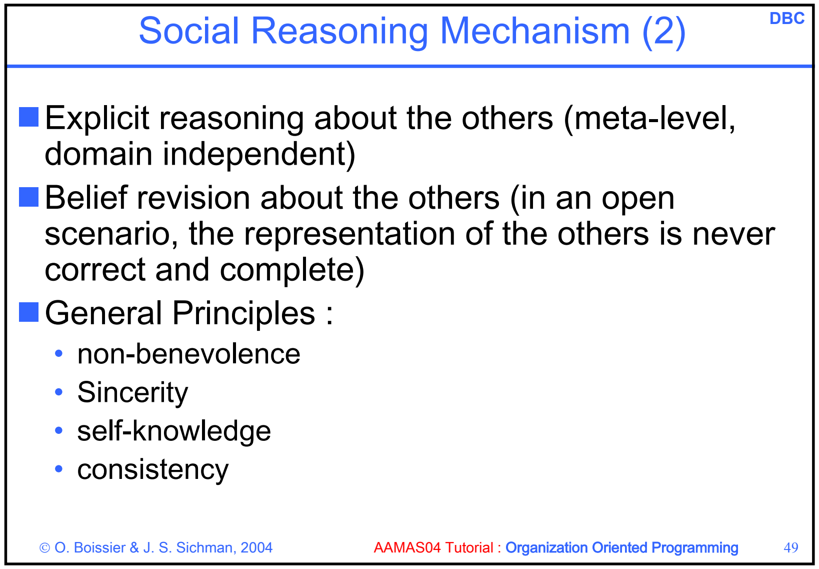
$basic\_dep(ag_1, ag_2, g_1, p_{111} = a_1(),a_2(),a_4(), a_2)$

Dependence Theory

- An agent ag_i *OR-depends* on a set of agents Ag_j when she holds a disjunction set of dependence relations upon any member ag_k of Ag_j. Any member of the set Ag_j is sufficient but unnecessary for ag_i's goal. *OR-dependence mitigates social dependence*.
- An agent ag_i *AND-depends* on a set of agents Ag_j when she holds a conjunction set of dependence relations upon all members of Ag_j. All members of the set Ag_j are necessary for ag_i's goal. *AND-dependence strengthens social dependence*.

Social Reasoning Mechanism (1)

- Based on Dependence Theory [Castelfranchi 92]
- Explains why social interactions occur, based on agents' complementarity
- Each agents represents in a private external description his information about the others
  - goals, plans, actions and ressources

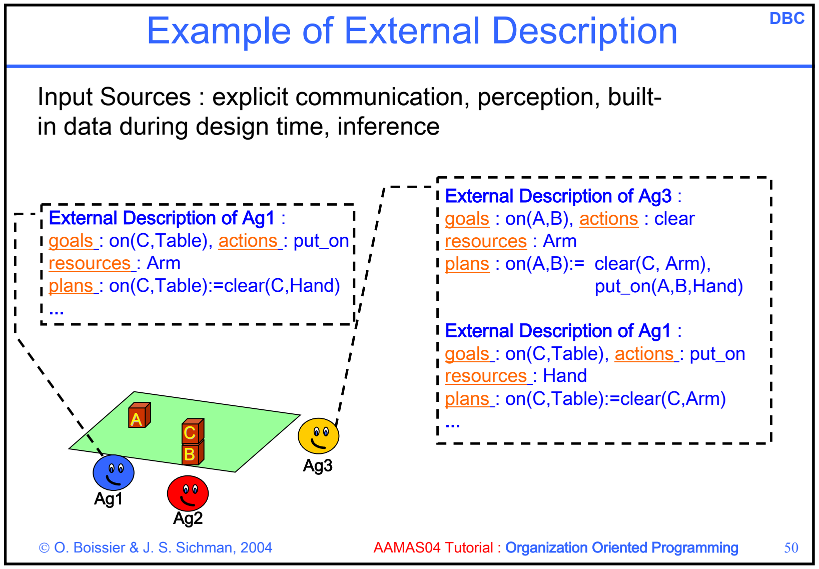


Social Reasoning Mechanism (2)

DBC

- Explicit reasoning about the others (meta-level, domain independent)
- Belief revision about the others (in an open scenario, the representation of the others is never correct and complete)
- General Principles :
  - non-benevolence
  - Sincerity
  - self-knowledge
  - consistency

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 49



Example of External Description

DBC

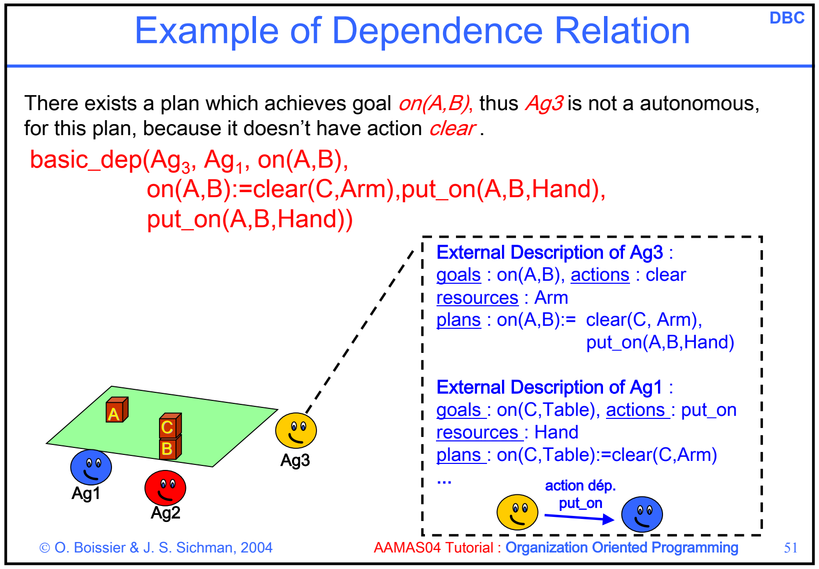
Input Sources : explicit communication, perception, built-in data during design time, inference

External Description of Ag1 :
goals : on(C,Table), actions : put_on
resources : Arm
plans : on(C,Table):=clear(C,Hand)
...

External Description of Ag3 :
goals : on(A,B), actions : clear
resources : Arm
plans : on(A,B):= clear(C, Arm),
put_on(A,B,Hand)

External Description of Ag1 :
goals : on(C,Table), actions : put_on
resources : Hand
plans : on(C,Table):=clear(C,Arm)
...

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 50



Example of Dependence Relation

DBC

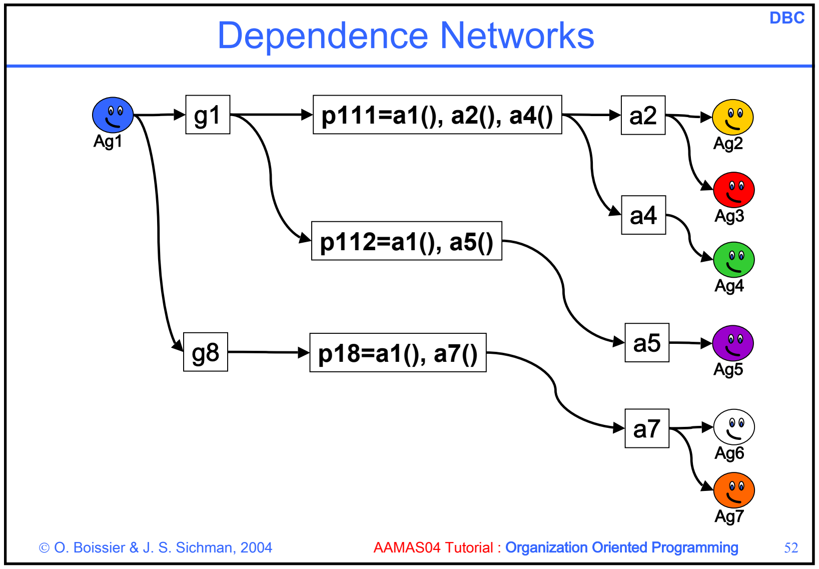
There exists a plan which achieves goal *on(A,B)*, thus *Ag3* is not a autonomous, for this plan, because it doesn't have action *clear* .

basic_dep(Ag_3, Ag_1, on(A,B),
on(A,B):=clear(C,Arm),put_on(A,B,Hand),
put_on(A,B,Hand))

External Description of Ag3 :
goals : on(A,B), actions : clear
resources : Arm
plans : on(A,B):= clear(C, Arm),
put_on(A,B,Hand)

External Description of Ag1 :
goals : on(C,Table), actions : put_on
resources : Hand
plans : on(C,Table):=clear(C,Arm)
...

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 51



Dependence Networks

DBC

© O. Boissier & J. S. Sichman, 2004 AAMAS04 Tutorial : Organization Oriented Programming 52

Social Reasoning

Given two agents *i* and *j* , the following situations may hold:

- **Independence**
- **Unilateral** Dependence (agent *i* depends on agent *j* for one of its goals *g*)
- **Bilateral** Dependence (agents *i* and *j* depend on each other for their goals g_1 and g_2)
  - Mutual Dependence MD: $g_1 = g_2$
  - Reciprocal Dependence RD : $g_1 \neq g_2$

Social Reasoning : Goal Situations

- A goal situation relates an agent to a goal :
  - NG(*i*,*g*) : the agent *i* does not have the goal *g*
  - NP(*i*,*g*) : the agent *i* has the goal *g* but it does not have any plans to achieve it
  - AUT(*i*,*g*) : the agent *i* has the goal *g*, and at least a plan *p* makes it action-autonomous to achieve *g*
  - DEP(*i*,*g*) : the agent *i* has the goal *g*, and every plan *p* to achieve *g* makes it action-dependent to achieve *g*

→This notion is taken into account for goal, plan and partner (acceptance) choice.

Social Reasoning : Dependence Situations

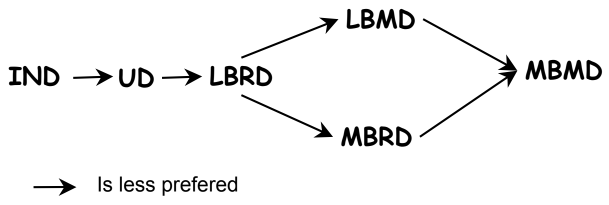
A dependence situation relates 2 agents and a goal :

- $IND(i,j,g) \equiv DEP(i,g) \wedge \neg\, dep\_on_a(i,j,g,i)$
- $LBMD(i,j,g) \equiv MD(i,j,g,i) \wedge \neg\, MD(i,j,g,j)$
- $MBMD(i,j,g) \equiv MD(i,j,g,i) \wedge MD(i,j,g,j)$
- $LBRD(i,j,g,g') \equiv RD(i,j,g,g',i) \wedge \neg\, RD(i,j,g,g',j)$
- $MBRD(i,j,g,g') \equiv RD(i,j,g,g',i) \wedge RD(i,j,g,g',j)$
- $UD(i,j,g) \equiv dep\_on_a(i,j,g,i) \wedge \neg \exists\, g'(\, is_g(j,g') \wedge dep\_on_a(j,i,g',i))$

→ This notion is taken into account for partner (proposal) choice

Social Reasoning : Dependence Situations

- Possible ordering of the dependence situations to choose a partner :



Social Reasoning : Goals and Plans

- A certain goal is achievable for an agent *i* if there is a plan whose all actions can be executed by at least one agent in the agency
- A certain plan is feasible for an agent *i* if all its actions can be executed by at least one agent in the agency
  - a goal is achievable if there is at least one feasible plan for it

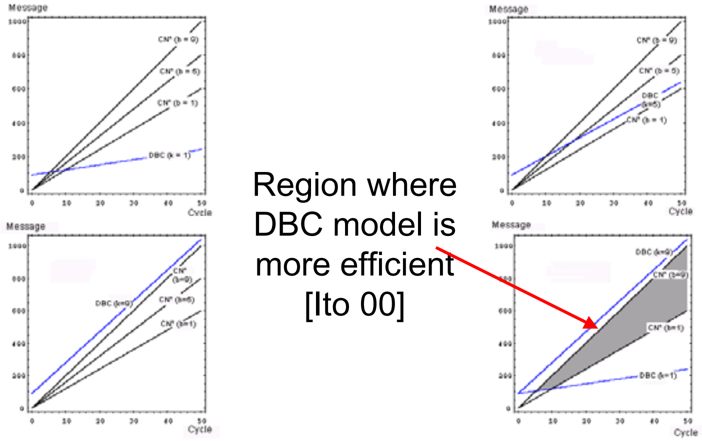
Dependence Based Coalitions (1)

- An agent may use his dependence networks and other associated notions (goal and dependence situations) to try to form organizations when he can not achieve his goals by himself
- Whenever the agents reasons socially well, this technique is useful in the long term

Dependence Based Coalitions (2)

- An agent first chooses a goal to achieve
  - its most important achievable goal
- Then, it chooses a plan to execute
  - Its less costly feasible plan for this goal
- According to its goal situation:
  - if he is AUT, he executes the plan alone
  - If he is DEP, he uses the dependence situations to choose a partner

Comparison between Contract Net and DBC



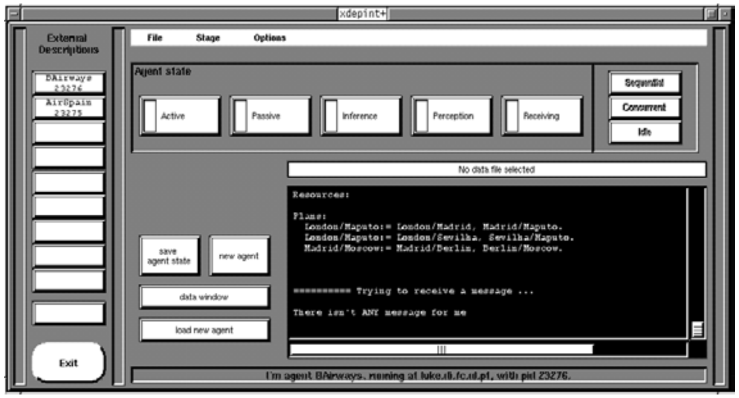
Region where DBC model is more efficient [Ito 00]

DEPNET

- Based on Social Reasoning Model
- External Description Editor
- Construction of dependence networks
- Computation of Goal Situations
- Computation of Dependence Situations
- Computation of Plans and Goals
- Simulations

[Conte 95]

DEPINT+

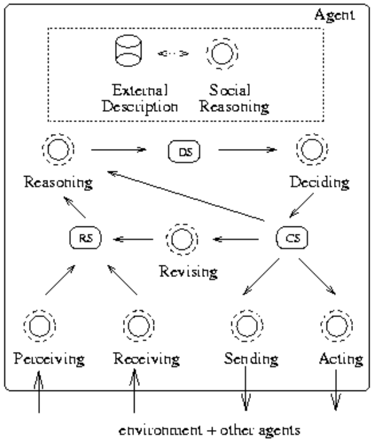


```
Resources:

Plans:
  London/Maputo:= London/Madrid, Madrid/Maputo.
  London/Maputo:= London/Sevilha, Sevilha/Maputo.
  Madrid/Moscow:= Madrid/Berlin, Berlin/Moscow.


========== Trying to receive a message ...

There isn't ANY message for me
```

I'm agent BAirways, running at luke.di.fc.ul.pt, with pid 23276.

[David 98]

DEPINT+



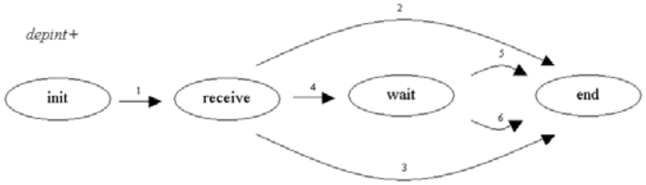
ASIC Agent Model
[Boissier 93]

Three layers dedicated to :

1. Management of goals
2. Management of plans
3. Management of actions

[Sichman 98]

DEPINT+



1.(all)(request,bargaining,dec,depintproposition,receive)(proposition,data)
2.(if error)(you)(answer,warning,obs,depintproposition,end)(revision,data)
3.(if (!error and !best_option))(you)(answer,informing,dec,depintproposition,end)(refusal)
4.(if (!error and best_option))(you)(answer,informing,dec,depintproposition,wait)(acceptance)
5.(if (suc_coalition)(you)(answer,confirming,dec,depintproposition,end)(confirmation)
6.(if (!suc_coaltion)(you)(answer,cancelling,dec,depintproposition,end)(cancelation)

Outline

1. Introduction

2. Agent-Centered Point of View

3. Organization-Centered Point of View

4. Programming Organizations

5. Reorganization

6. Conclusion and Challenges

Outline

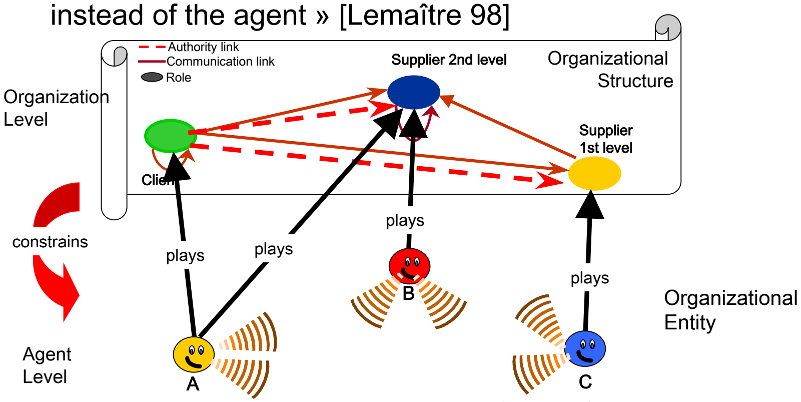
1. Introduction

2. Agent-Centered Point of View

3. Organization-Centered Point of View
   - 3.1. Main Features
   - 3.2. GAIA
   - 3.3. TAEMS
   - 3.4. AGR
   - 3.5. STEAM
   - 3.6. MOISE+

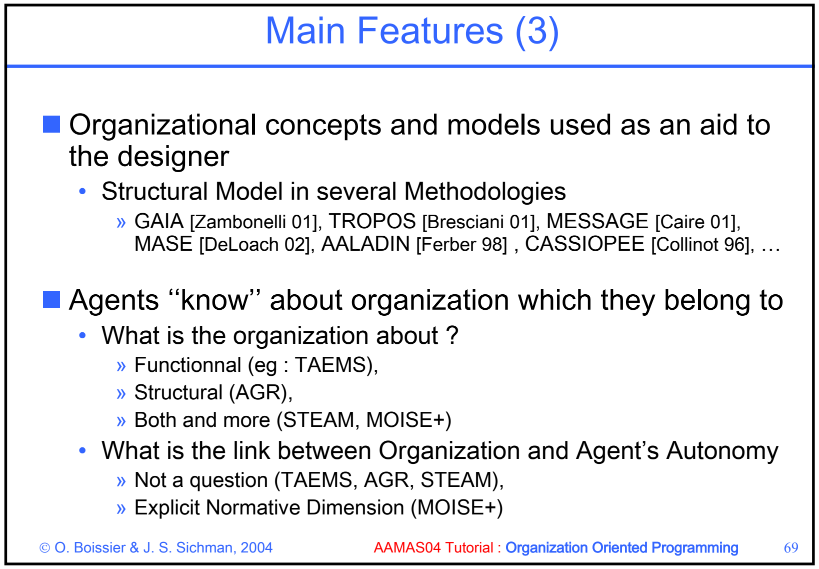
Main features (1)

« The leading concept is the group or the organization instead of the agent » [Lemaître 98]



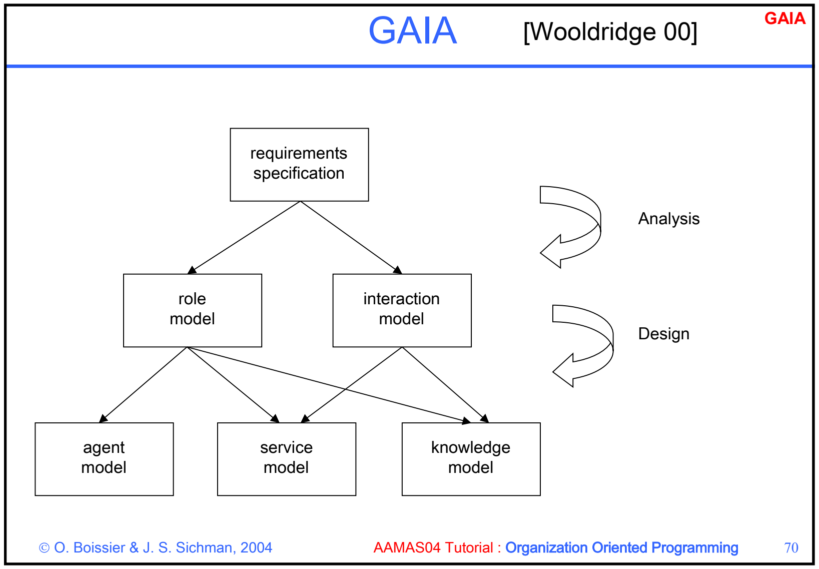
Main features (2)

- Make a clear distinction between description of organization and description of agents
  - Two levels : organization and agent
- Agents are dynamic, autonomous entities that evolve within organizations
  - Organizations constrain the behaviors of the agents
  - Organizations may be the result of the activities of agents



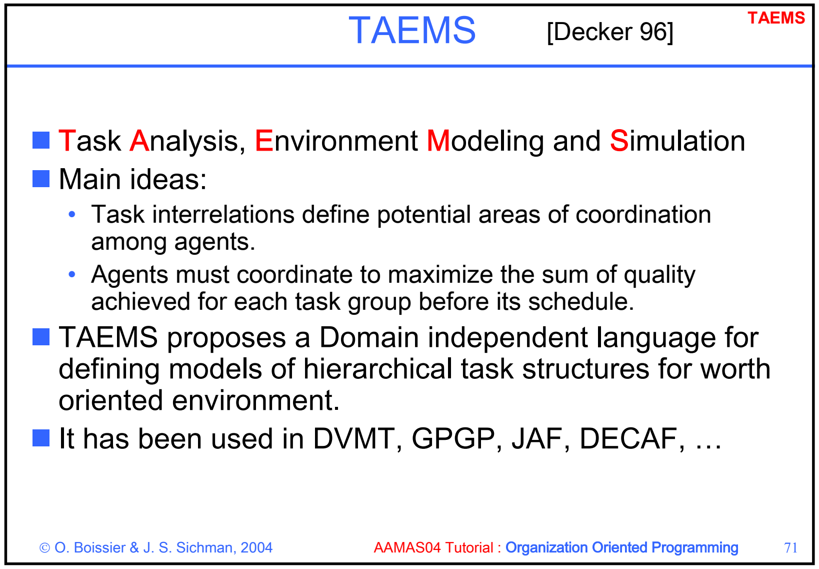
Main Features (3)

- Organizational concepts and models used as an aid to the designer
  - Structural Model in several Methodologies
    - » GAIA [Zambonelli 01], TROPOS [Bresciani 01], MESSAGE [Caire 01], MASE [DeLoach 02], AALADIN [Ferber 98] , CASSIOPEE [Collinot 96], …
- Agents "know" about organization which they belong to
  - What is the organization about ?
    - » Functionnal (eg : TAEMS),
    - » Structural (AGR),
    - » Both and more (STEAM, MOISE+)
  - What is the link between Organization and Agent's Autonomy
    - » Not a question (TAEMS, AGR, STEAM),
    - » Explicit Normative Dimension (MOISE+)



GAIA [Wooldridge 00]

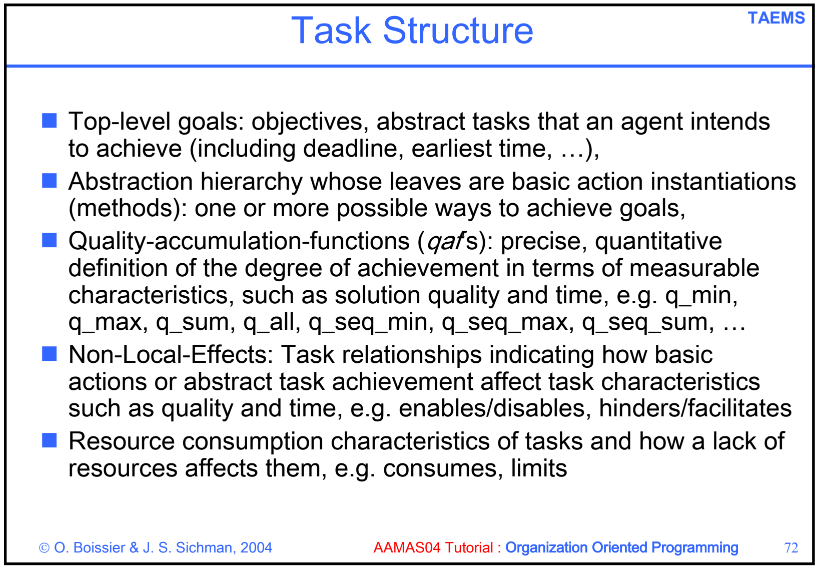
GAIA



TAEMS [Decker 96]

TAEMS

- Task Analysis, Environment Modeling and Simulation
- Main ideas:
  - Task interrelations define potential areas of coordination among agents.
  - Agents must coordinate to maximize the sum of quality achieved for each task group before its schedule.
- TAEMS proposes a Domain independent language for defining models of hierarchical task structures for worth oriented environment.
- It has been used in DVMT, GPGP, JAF, DECAF, …



Task Structure

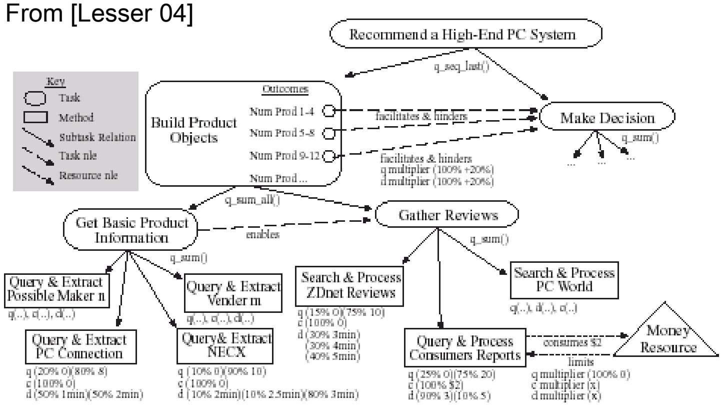
TAEMS

- Top-level goals: objectives, abstract tasks that an agent intends to achieve (including deadline, earliest time, …),
- Abstraction hierarchy whose leaves are basic action instantiations (methods): one or more possible ways to achieve goals,
- Quality-accumulation-functions (*qaf*s): precise, quantitative definition of the degree of achievement in terms of measurable characteristics, such as solution quality and time, e.g. q_min, q_max, q_sum, q_all, q_seq_min, q_seq_max, q_seq_sum, …
- Non-Local-Effects: Task relationships indicating how basic actions or abstract task achievement affect task characteristics such as quality and time, e.g. enables/disables, hinders/facilitates
- Resource consumption characteristics of tasks and how a lack of resources affects them, e.g. consumes, limits

Example

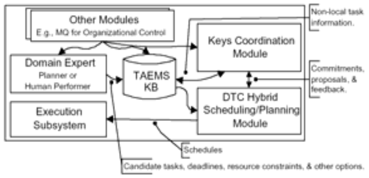
TAEMS

From [Lesser 04]



Agents using TAEMS

Programming TAEMS


Other Modules
E.g., MQ for Organizational Control
Domain Expert
Planner or
Human Performer
TAEMS KB
Keys Coordination Module
Execution Subsystem
DTC Hybrid Scheduling/Planning Module
Non-local task information.
Commitments, proposals, & feedback.
Schedules
Candidate tasks, deadlines, resource constraints, & other options.

- Agents using TAEMS:
  - Belief database
  - local scheduler
  - Coordination module
- The scheduler uses information in the database to schedule execution of methods, in a way to maximize quality.
- The coordination module handles communication with other agents and makes/breaks commitments with them in order to complete tasks.

AGR [Ferber, Gütknecht 98]

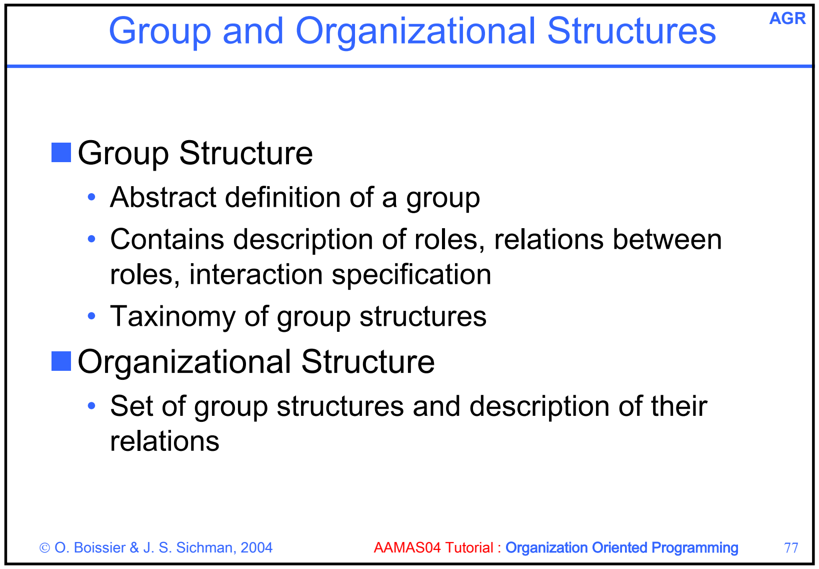
AGR

- Agent Group Role
- Previously known as AALAADIN
- Used within the MadKit platform
- Agent
  - Active entity that plays roles within groups. An agent may have several roles and may belong to several groups.
- Group
  - Set of agents sharing common characteristics, i.e. context for a set of activities.
  - Two agents can't communicate with each other if they don't belong to the same group

Role

AGR

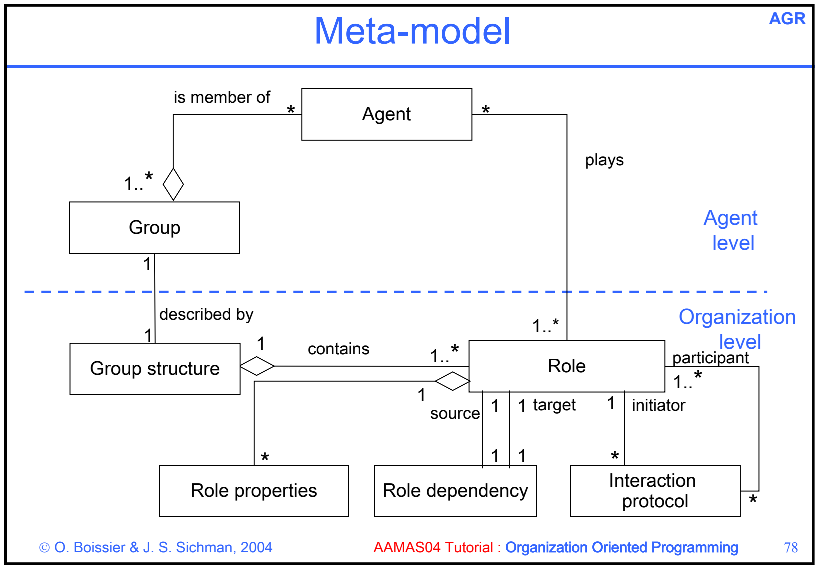
- Abstract representation of the status, position, function of an agent within a group.
- Roles are local to group
- Several agents can play the same role.
- A role is a description of an expected behavior of an agent
- A role describes constraints that agents playing that role should satisf
- Roles are interrelated through interaction description and relation/dependencies between roles



Group and Organizational Structures

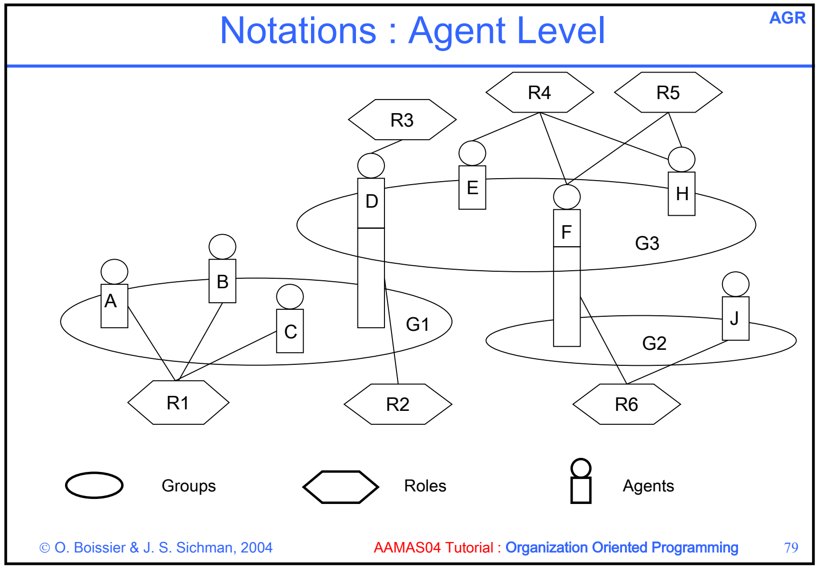
AGR

- Group Structure
  - Abstract definition of a group
  - Contains description of roles, relations between roles, interaction specification
  - Taxinomy of group structures
- Organizational Structure
  - Set of group structures and description of their relations



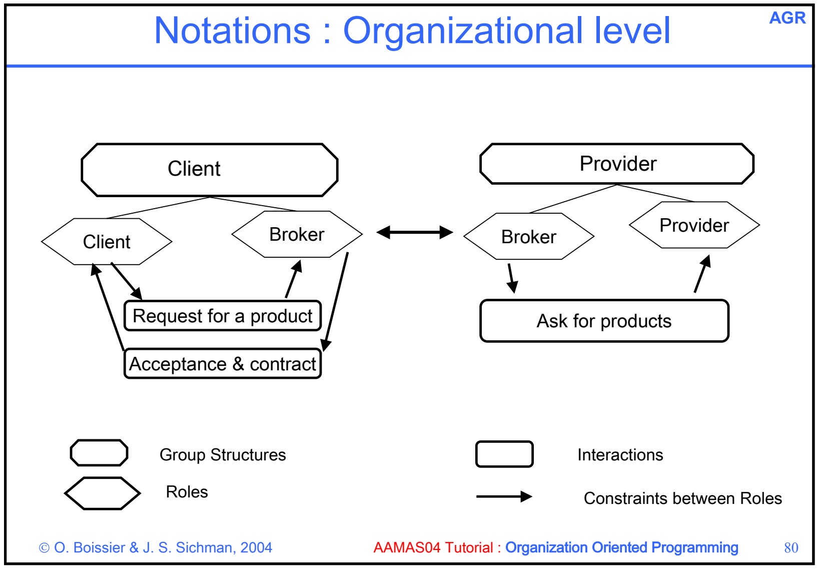
Meta-model

AGR



Notations : Agent Level

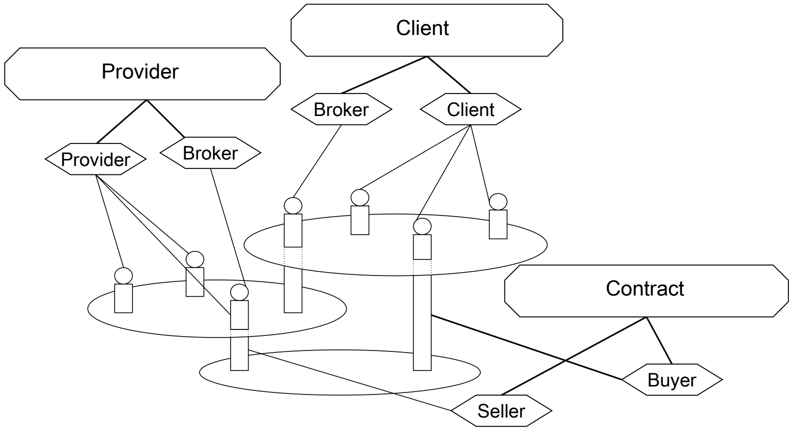
AGR



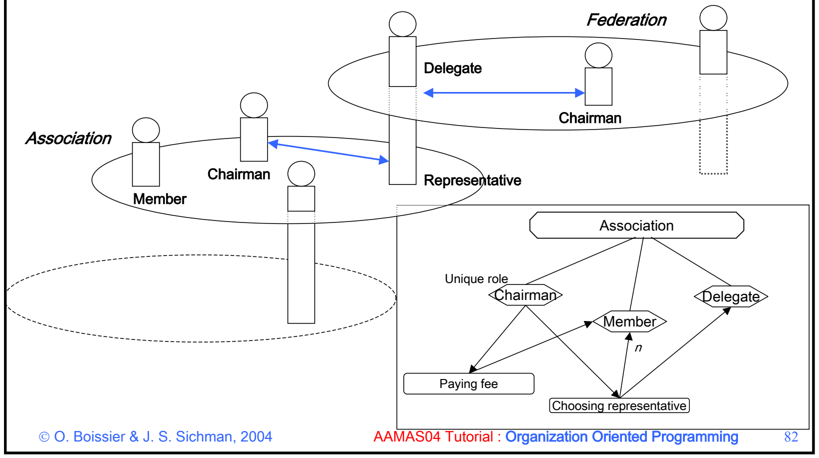
Notations : Organizational level

AGR

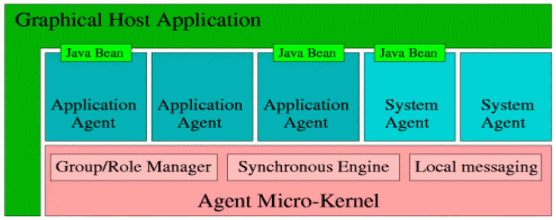
Example



Hierarchies representation

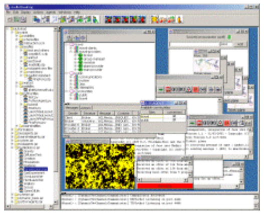


MADKIT



Multi-Agent Development Kit

www.madkit.org



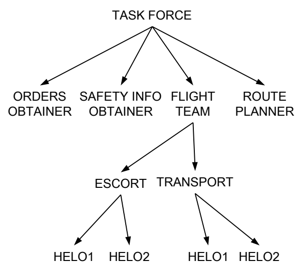
STEAM [Tambe 98]

- Shell for TEAMwork
- General framework to enable agents to participate in teamwork.
  - Different applications: Attack, Transport, Robocup soccer
- Based on an enhanced SOAR architecture and 300 domain independent SOAR rules
- Principles :
  - Team synchronization
    - Establish joint intentions, Monitor team progress and repair, Individual may fail or succeed in own role
  - Reorganize if there is a critical role failure
  - Reassign critical roles based on joint intentions
  - Decision theoretic communication

Main Components

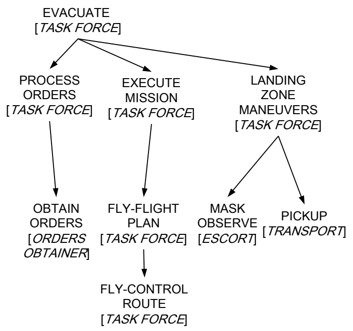
- Based on: [Pynadath 99]
  - Joint intentions theory as building block for a team’s mental attitude enabling flexible reasoning about coordination activities [Levesque 90, Jennings 95],
  - Shared Plans Theory: Hierarchical structure of joint intentions and individual intentions [Grosz 96, Rich 97]
- Teamwork knowledge consists of:
  - Coherence preserving rules requiring communication between team members to ensure coherent initiation and termination of team plans
  - Role-monitoring and repairing rules ensuring substitution of roles between team members
  - Decision-theoretic techniques to weigh communication costs and benefits to avoid excessive communication in the team.

TEAM SPECIFICATION



Organization: hierarchy of roles that may be filled by agents or groups of agents.

From [Tambe 00]


EVACUATE [*TASK FORCE*]
PROCESS ORDERS [*TASK FORCE*]
EXECUTE MISSION [*TASK FORCE*]
LANDING ZONE MANEUVERS [*TASK FORCE*]
OBTAIN ORDERS [*ORDERS OBTAINER*]
FLY-FLIGHT PLAN [*TASK FORCE*]
MASK OBSERVE [*ESCORT*]
PICKUP [*TRANSPORT*]
FLY-CONTROL ROUTE [*TASK FORCE*]

Team Plan: initial conditions, termination conditions, team-level actions.

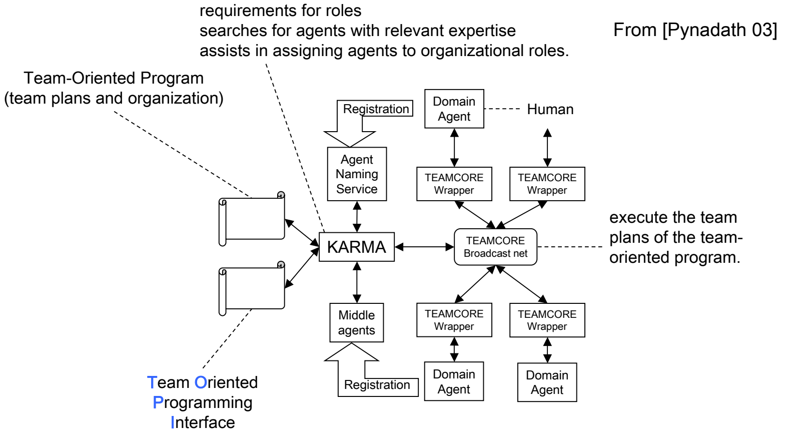
TEAMCORE

[Pynadath 03]

- Core Team Reasoning (TEAMCORE) focuses on enabling software developers to build large-scale agent organizations
  - Specification and monitoring of the agent organization
    - → Knowledgeable Agent Resources Manager Assistant (KARMA)
  - Robuts teamwork among agents
    - → TEAMCORE Wrappers based on STEAM Teamwork model

TEAMCORE

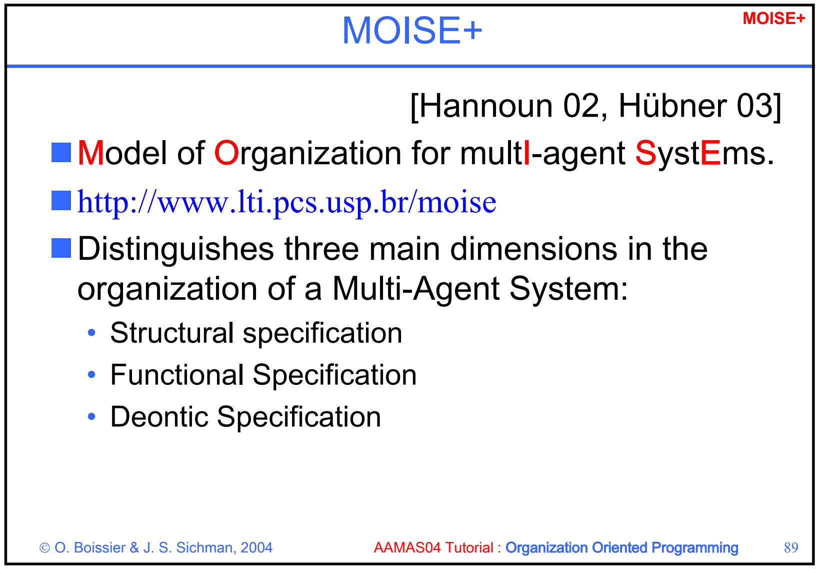
From [Pynadath 03]


requirements for roles
searches for agents with relevant expertise
assists in assigning agents to organizational roles.

Team-Oriented Program
(team plans and organization)

execute the team plans of the team-oriented program.

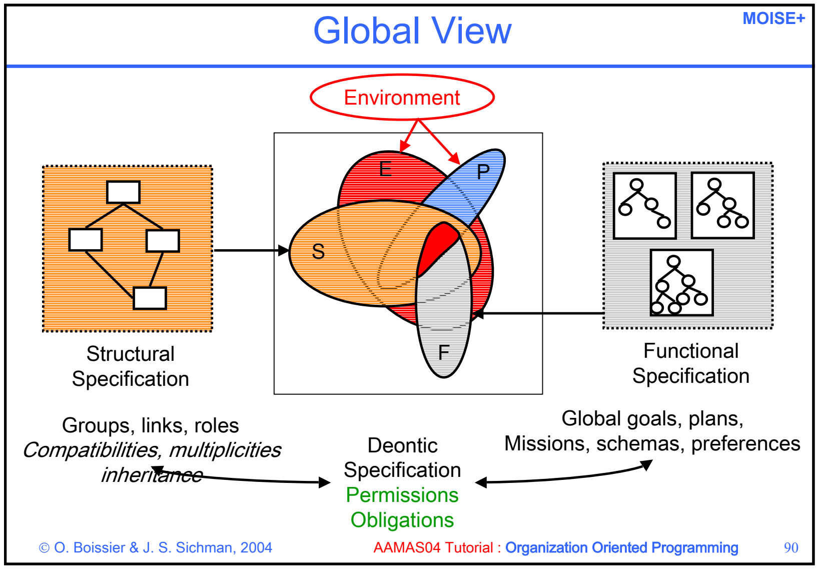
Team Oriented
Programming
Interface



MOISE+

[Hannoun 02, Hübner 03]

- Model of Organization for multI-agent SystEms.
- http://www.lti.pcs.usp.br/moise
- Distinguishes three main dimensions in the organization of a Multi-Agent System:
  - Structural specification
  - Functional Specification
  - Deontic Specification



Global View

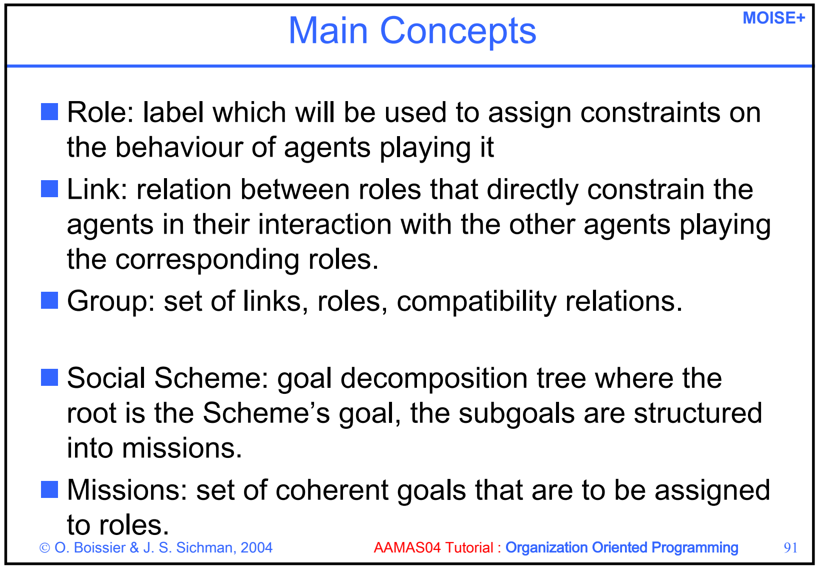
Structural Specification

Functional Specification

Groups, links, roles
Compatibilities, multiplicities
inheritance

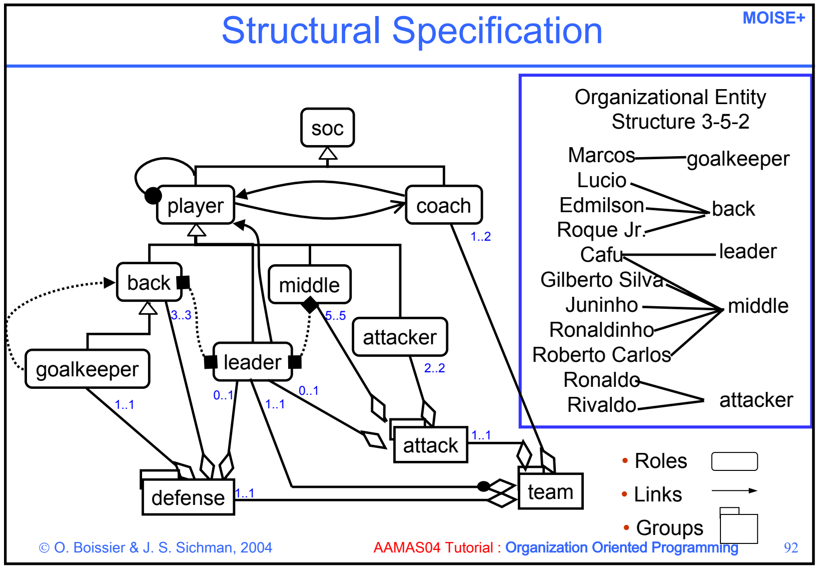
Deontic Specification
Permissions
Obligations

Global goals, plans,
Missions, schemas, preferences



Main Concepts

- Role: label which will be used to assign constraints on the behaviour of agents playing it
- Link: relation between roles that directly constrain the agents in their interaction with the other agents playing the corresponding roles.
- Group: set of links, roles, compatibility relations.
- Social Scheme: goal decomposition tree where the root is the Scheme's goal, the subgoals are structured into missions.
- Missions: set of coherent goals that are to be assigned to roles.



Structural Specification

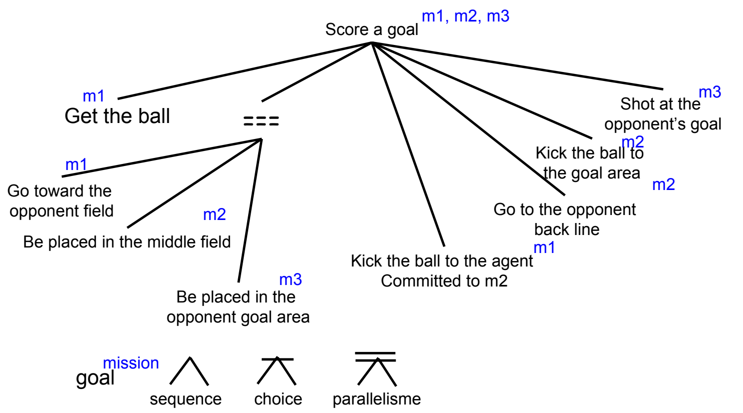
Organizational Entity
Structure 3-5-2

Marcos	goalkeeper
Lucio Edmilson Roque Jr.	back
Cafu	leader
Gilberto Silva Juninho Ronaldinho Roberto Carlos	middle
Ronaldo Rivaldo	attacker

- Roles ▭
- Links →
- Groups 🗀

Functional Specification

MOISE+


Score a goal
m1, m2, m3
m1
Get the ball
===
m1
Go toward the opponent field
m2
Be placed in the middle field
m3
Be placed in the opponent goal area
Kick the ball to the agent Committed to m2
m1
Go to the opponent back line
m2
Kick the ball to the goal area
m2
Shot at the opponent's goal
m3
goal
mission
sequence
choice
parallelisme

Deontic Specification

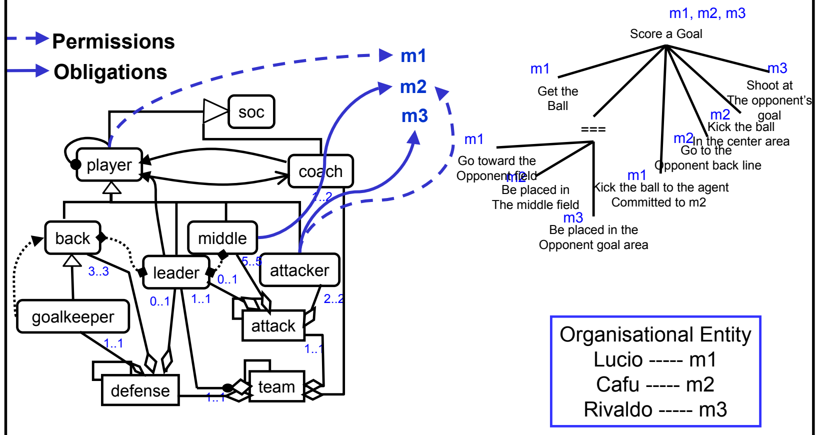
MOISE+

- Explicit relation between the functional and structural specifications
  - Permissions and obligations to commit to missions in the context of a role
  - To make explicit the normative dimension of a role

Role	Deontic Relation	Mission	Temporal Constraint (cf. [carron 01])
Back	Permission	m1	In [0 30]
Middle	Obligation	m2	during [Attacker]
Attacker	Obligation	m3	Any

Deontic Specification

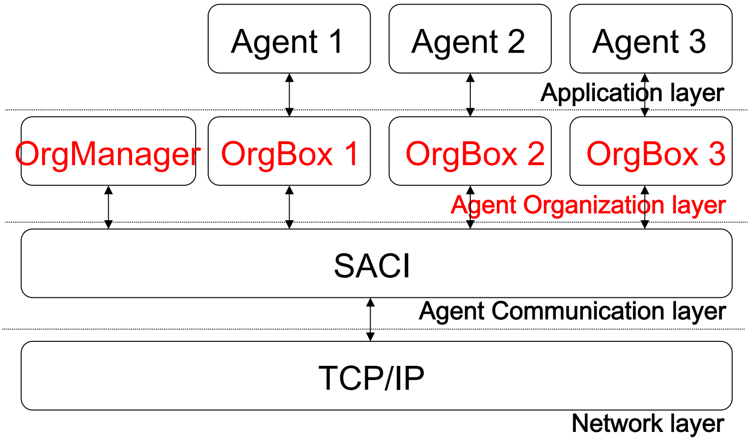
MOISE+

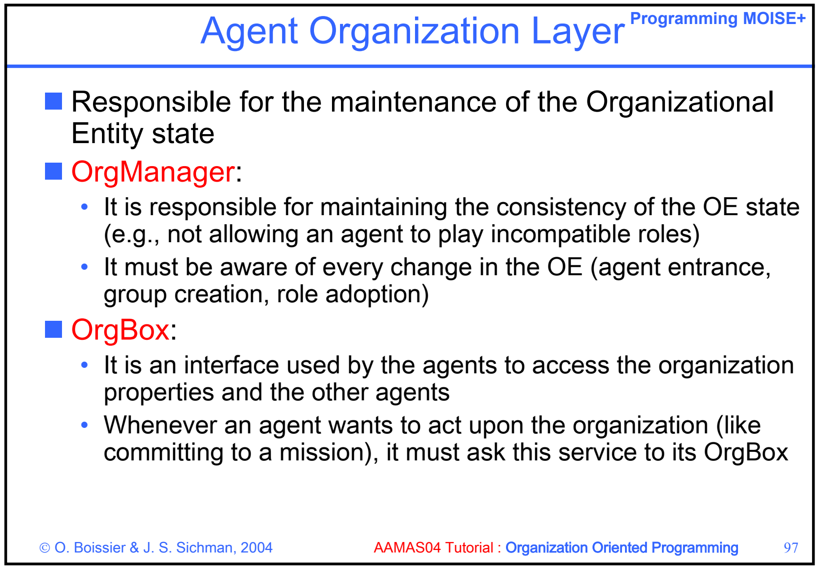

Permissions
Obligations
m1
m2
m3
soc
player
coach
back
middle
leader
attacker
goalkeeper
attack
defense
team
3..3
0..1
1..1
0..1
5..5
1..2
2..2
1..1
1..1
1..1
Score a Goal
m1, m2, m3
m1
Get the Ball
===
m1
Go toward the Opponent field
m2
Be placed in The middle field
m3
Be placed in the Opponent goal area
m1
Kick the ball to the agent Committed to m2
m2
Go to the Opponent back line
m2
Kick the ball In the center area
m3
Shoot at The opponent's goal

Organisational Entity
Lucio ----- m1
Cafu ----- m2
Rivaldo ----- m3

Layered Architecture

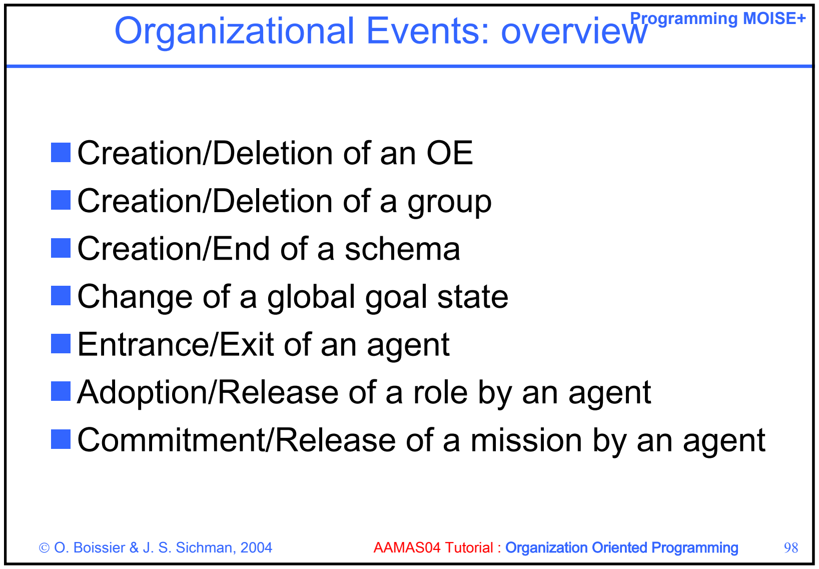
Programming MOISE+


Agent 1
Agent 2
Agent 3
Application layer
OrgManager
OrgBox 1
OrgBox 2
OrgBox 3
Agent Organization layer
SACI
Agent Communication layer
TCP/IP
Network layer



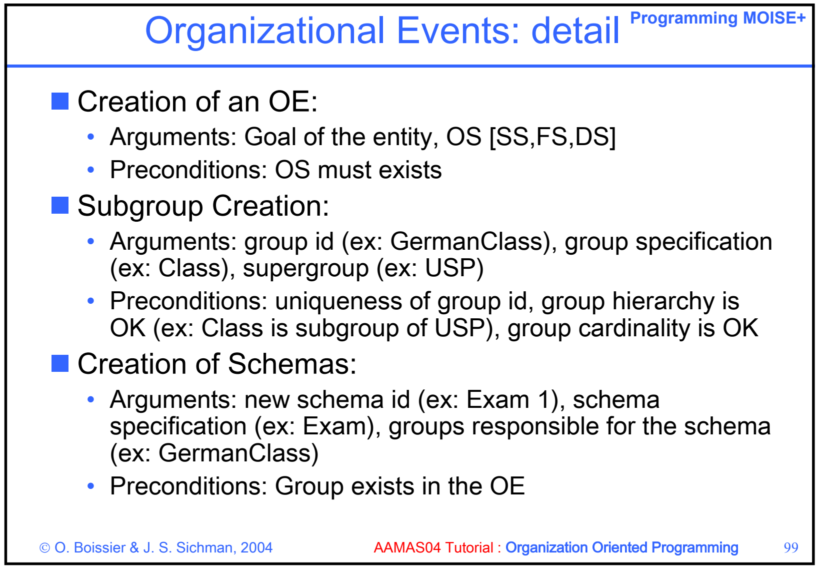
Agent Organization Layer

- Responsible for the maintenance of the Organizational Entity state
- OrgManager:
  - It is responsible for maintaining the consistency of the OE state (e.g., not allowing an agent to play incompatible roles)
  - It must be aware of every change in the OE (agent entrance, group creation, role adoption)
- OrgBox:
  - It is an interface used by the agents to access the organization properties and the other agents
  - Whenever an agent wants to act upon the organization (like committing to a mission), it must ask this service to its OrgBox



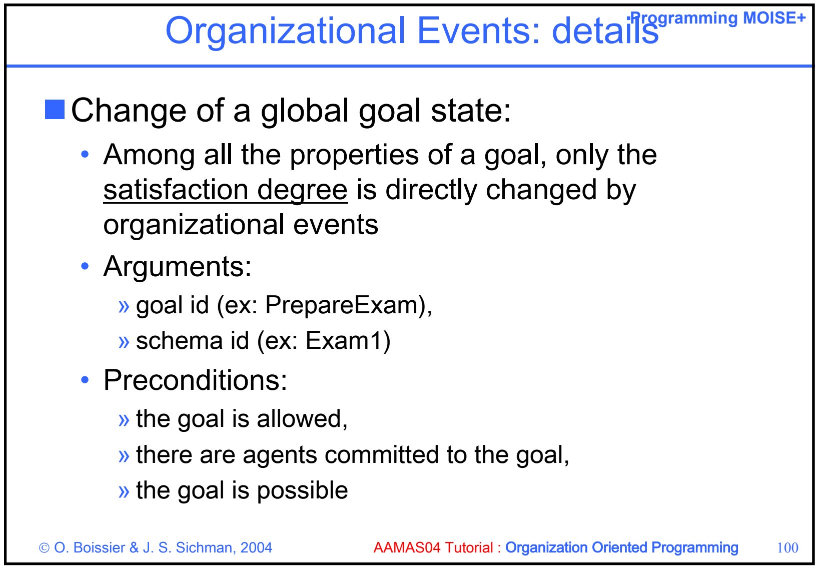
Organizational Events: overview

- Creation/Deletion of an OE
- Creation/Deletion of a group
- Creation/End of a schema
- Change of a global goal state
- Entrance/Exit of an agent
- Adoption/Release of a role by an agent
- Commitment/Release of a mission by an agent



Organizational Events: detail

- Creation of an OE:
  - Arguments: Goal of the entity, OS [SS,FS,DS]
  - Preconditions: OS must exists
- Subgroup Creation:
  - Arguments: group id (ex: GermanClass), group specification (ex: Class), supergroup (ex: USP)
  - Preconditions: uniqueness of group id, group hierarchy is OK (ex: Class is subgroup of USP), group cardinality is OK
- Creation of Schemas:
  - Arguments: new schema id (ex: Exam 1), schema specification (ex: Exam), groups responsible for the schema (ex: GermanClass)
  - Preconditions: Group exists in the OE



Organizational Events: details

- Change of a global goal state:
  - Among all the properties of a goal, only the satisfaction degree is directly changed by organizational events
  - Arguments:
    - » goal id (ex: PrepareExam),
    - » schema id (ex: Exam1)
  - Preconditions:
    - » the goal is allowed,
    - » there are agents committed to the goal,
    - » the goal is possible

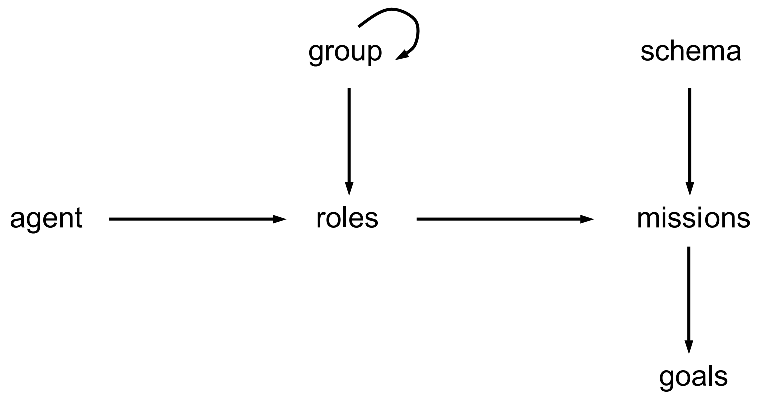
Global Goal Activation Degree

```
Global Goal Activation Degree:
  function isPermitted (schema $si$, goal $g$)
  if $g$ is root of $si$ then return true
  else if $g$ belongs to a plan $g0 = ... gi , g ...$ then
          if goal $gi$ is satisfied then return true
          else return false
    else return isPermitted($si$, $g0$)
Global Goal Commitment Degree
  function isCommitted (schema $si$, goal $g$)
  if there is at least one single agent commited to $g$ then
      return true
  else
      return false
```

Organizational Events: details

- Role Adoption:
  - Arguments: agent id (ex: Gustavo), role id (ex: Student), group id (ex: GermanClass)
  - Preconditions: agent belongs to the system, role exists within the group, role cardinality is not exceeded in the group, agent roles are compatible with the new role
- Commitment to a Mission by an Agent
  - Arguments: agent id (ex: Gustavo), mission id (ex: m42 – prepare exam), schema id (ex: exam 1)
  - Preconditions: cardinality of mission is not violated, schema is still active, the roles played by the agent in the groups responsible for the schema allow him to commit to the mission

Dependencies for a Group Deletion



OrgBox Services

- Communication:
  - sending/receiving KQML messages to/from other agents
  - verification of communication links
- Generation of organizational events:
  - agents may enter the system, commit to a role, create a group
- Informing obligations:
  - OrgBox keeps the agent informed of the missions he is obliged to commit
- Informing possible goals:
  - OrgBox keeps the agent informed of the possible goals he can choose to achieve

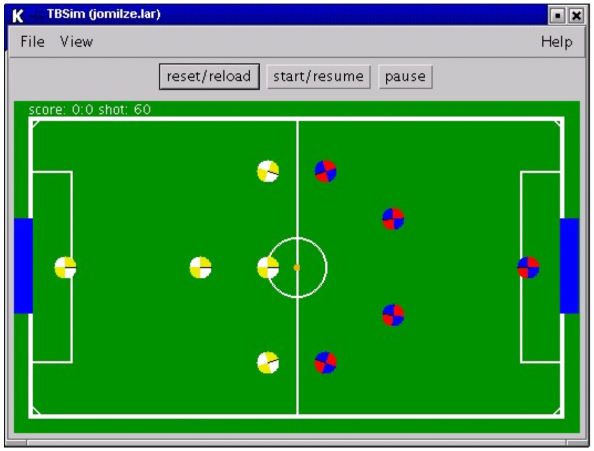
Getting the Obligations of an Agent

```
function getObligatedMissions (agent a)
all = {}
for all role r that a has committed to do
   gr = group where r has been assumed
   for all scheme si that gr is responsible for do
         if sheme has not finished yet then
                  for all mission m in si do
                           if r is obliged to m then
                               if cardinality of m is not exceeded then
                                    all = all ∪ {m}
sort all according to mission preference
return all
```

Getting the Possible Goal for an Agent

```
function getPossibleGoals (agent a)
all = {}
for all mission m that a has committed to do
   sch = scheme where m has been assumed
   for all goal g that belongs to m do
        if ¬IsSatisfied(g)∧IsPossible(g)∧IsPermitted(g) then
                for all gs that is a supergoal sch do
                        if ¬IsSatisfied(gs)∧ IsPossible(gs) then
                            if cardinality of m is not exceeded then
                                 all = all ∪ {g}
sort all according to mission preference
return all
```

Application Example: JOJTEAM

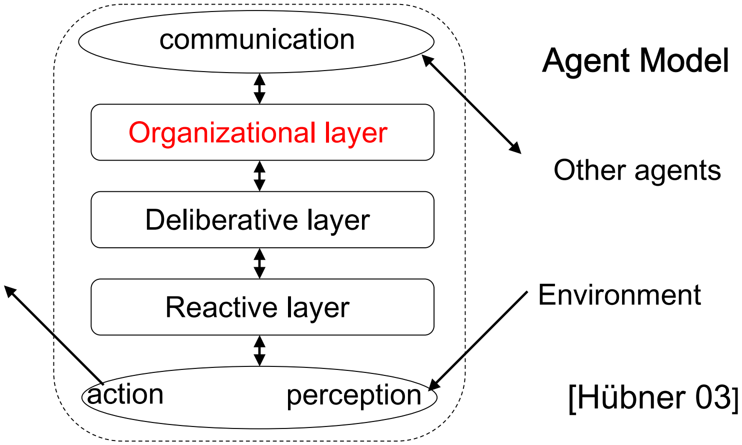


MOISE$^+$ Model

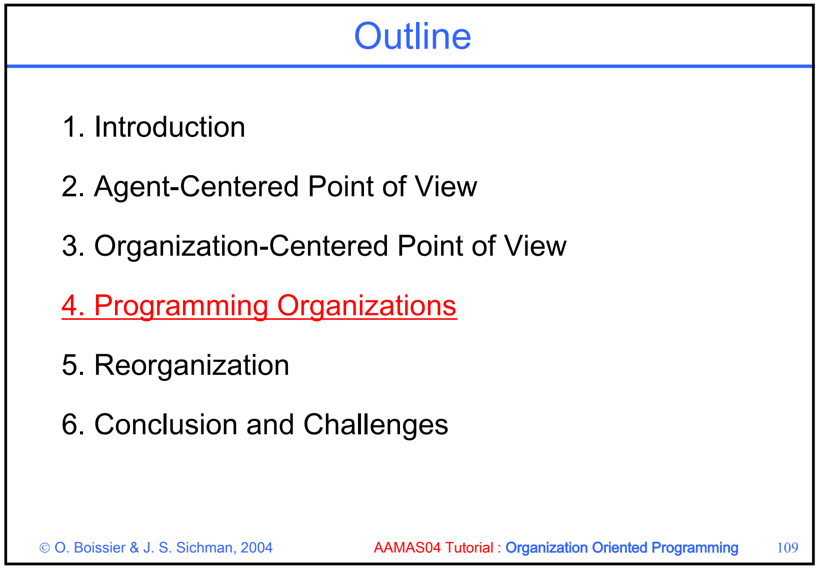
Teambots Simulator

[Hübner 03]

Agent Architecture: JOJTEAM

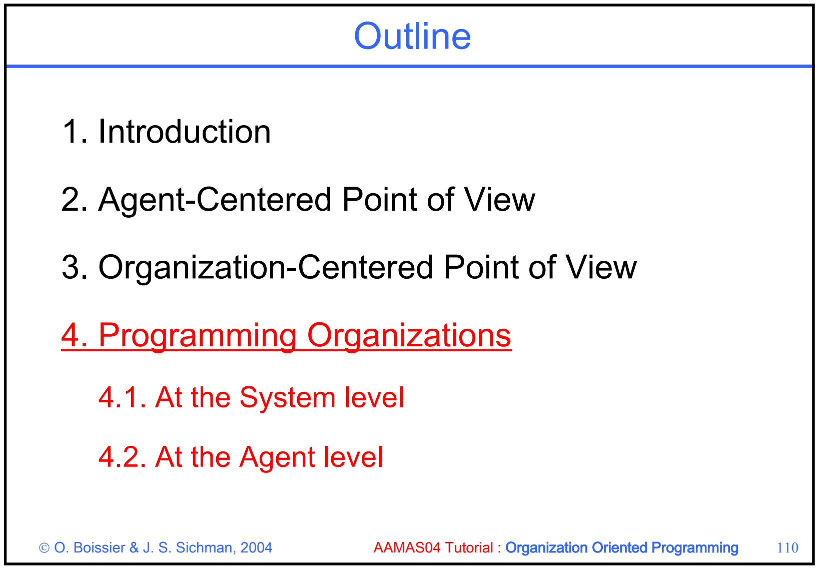


[Hübner 03]



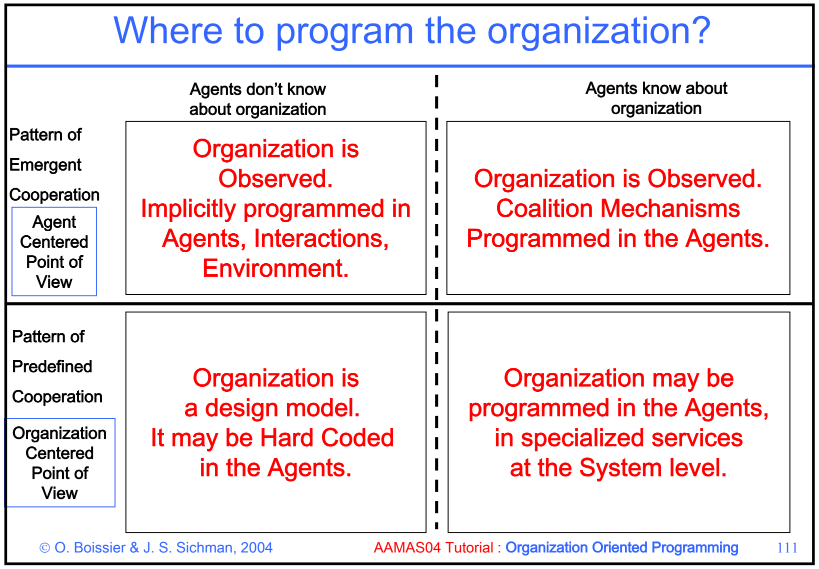
Outline

1. Introduction
2. Agent-Centered Point of View
3. Organization-Centered Point of View
4. Programming Organizations
5. Reorganization
6. Conclusion and Challenges



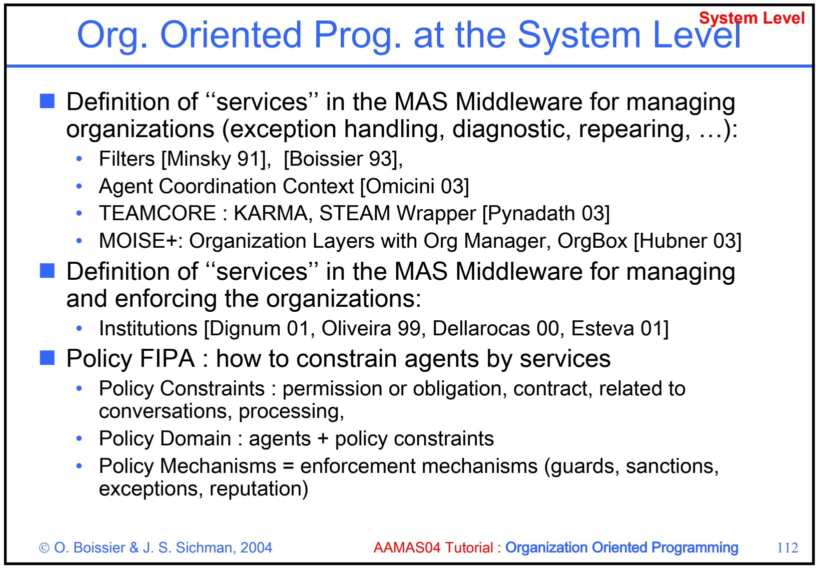
Outline

1. Introduction
2. Agent-Centered Point of View
3. Organization-Centered Point of View
4. Programming Organizations
   - 4.1. At the System level
   - 4.2. At the Agent level



Where to program the organization?

	Agents don't know about organization	Agents know about organization
Pattern of Emergent Cooperation Agent Centered Point of View	Organization is Observed. Implicitly programmed in Agents, Interactions, Environment.	Organization is Observed. Coalition Mechanisms Programmed in the Agents.
Pattern of Predefined Cooperation Organization Centered Point of View	Organization is a design model. It may be Hard Coded in the Agents.	Organization may be programmed in the Agents, in specialized services at the System level.



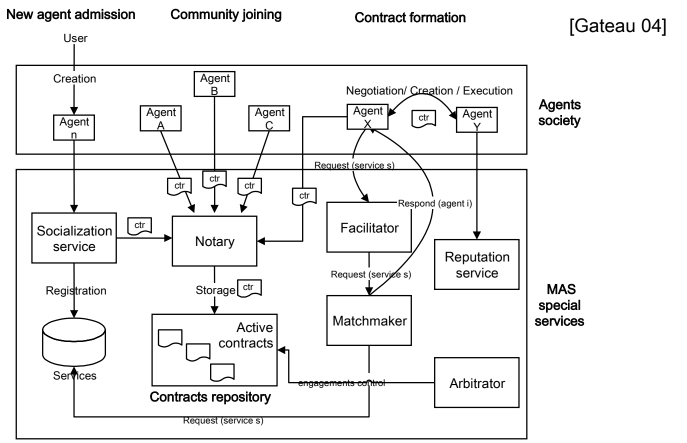
System Level

Org. Oriented Prog. at the System Level

- Definition of "services" in the MAS Middleware for managing organizations (exception handling, diagnostic, repearing, …):
  - Filters [Minsky 91], [Boissier 93],
  - Agent Coordination Context [Omicini 03]
  - TEAMCORE : KARMA, STEAM Wrapper [Pynadath 03]
  - MOISE+: Organization Layers with Org Manager, OrgBox [Hubner 03]
- Definition of "services" in the MAS Middleware for managing and enforcing the organizations:
  - Institutions [Dignum 01, Oliveira 99, Dellarocas 00, Esteva 01]
- Policy FIPA : how to constrain agents by services
  - Policy Constraints : permission or obligation, contract, related to conversations, processing,
  - Policy Domain : agents + policy constraints
  - Policy Mechanisms = enforcement mechanisms (guards, sanctions, exceptions, reputation)

Contractual Agent Society

System Level



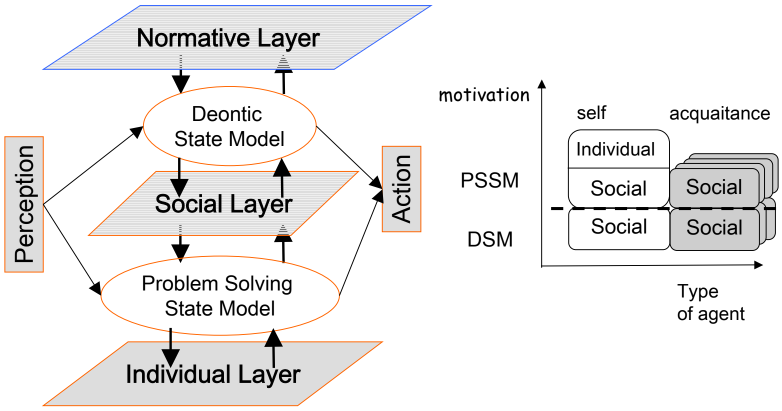
OOP at the Agent Level

Agent Level

- Agents Reasoning on organizations
  - ADEPT [Jennings 96]
  - TAEMS Agents [Decker 96]
  - MOISE+ Agents [Hannoun 02], [Hübner 03]
- Agents Reasoning on Norms within organizations [Boela 00, Castelfranchi 99, Ossowski 99]
  - Representation of norms, of the organization, …
  - Deliberation on respect/violation of norms, of the organization, …
  - Reaction to violation of norms, organizations by other agents

Agent reasoning on Norms ProsA$_2$ [Ossowski 99]

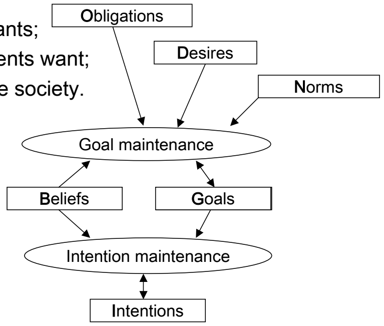
Agent Level

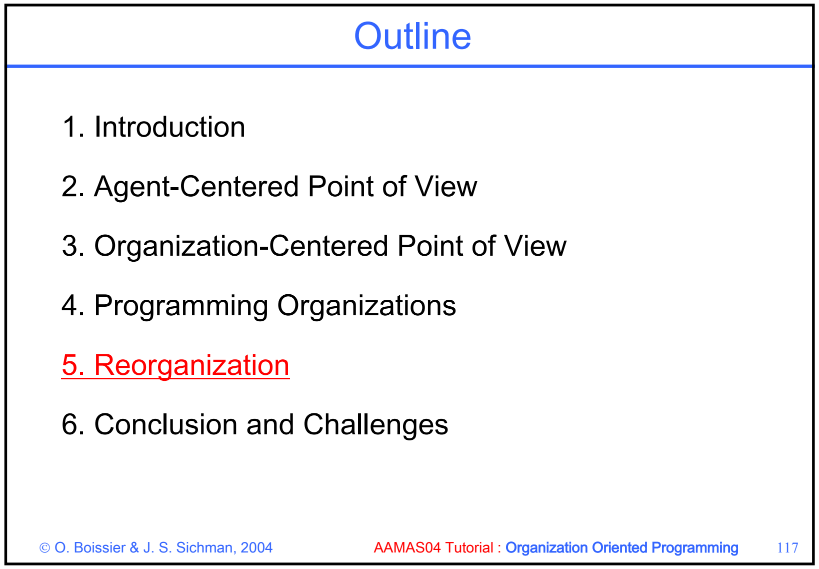


B-DOING

Agent Level

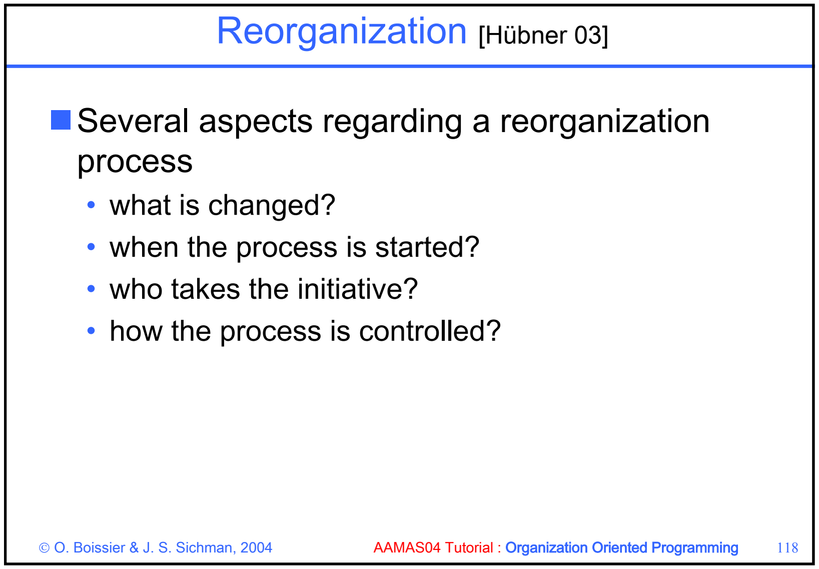
- Extends the BDI model. [Dignum 01]
- The agent's intentions are generated based on its current beliefs and a set of possibly conflicting goals.
- The goals are generated from:
  - a set of desires: what the agent wants;
  - a set of obligations: what other agents want;
  - a set of norms: what is good for the society.
- B-DOING logic: an extention of BDI-logic.





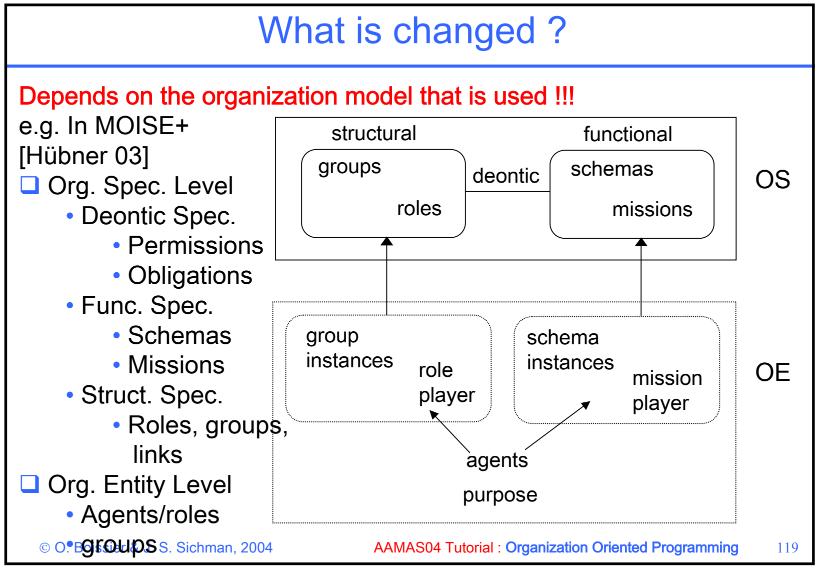
Outline

1. Introduction
2. Agent-Centered Point of View
3. Organization-Centered Point of View
4. Programming Organizations
5. Reorganization
6. Conclusion and Challenges



Reorganization [Hübner 03]

- Several aspects regarding a reorganization process
  - what is changed?
  - when the process is started?
  - who takes the initiative?
  - how the process is controlled?



What is changed ?

Depends on the organization model that is used !!!

e.g. In MOISE+ [Hübner 03]

- ❑ Org. Spec. Level
  - Deontic Spec.
    - Permissions
    - Obligations
  - Func. Spec.
    - Schemas
    - Missions
  - Struct. Spec.
    - Roles, groups, links
- ❑ Org. Entity Level
  - Agents/roles
  - groups

structural

groups

roles

deontic

functional

schemas

missions

OS

group instances

role player

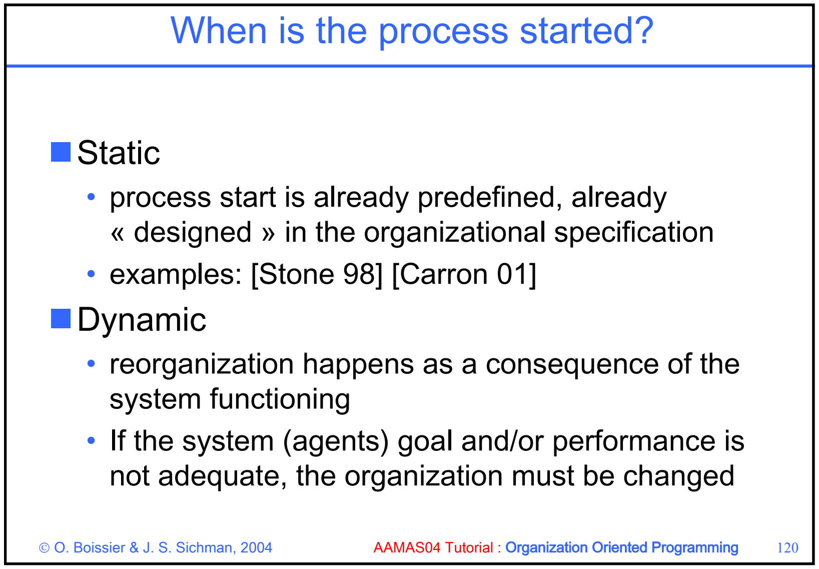
schema instances

mission player

agents

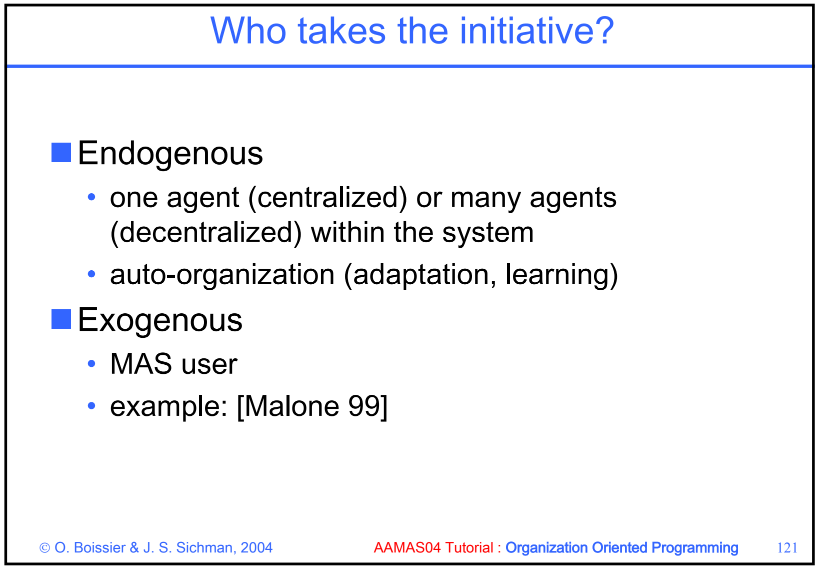
purpose

OE



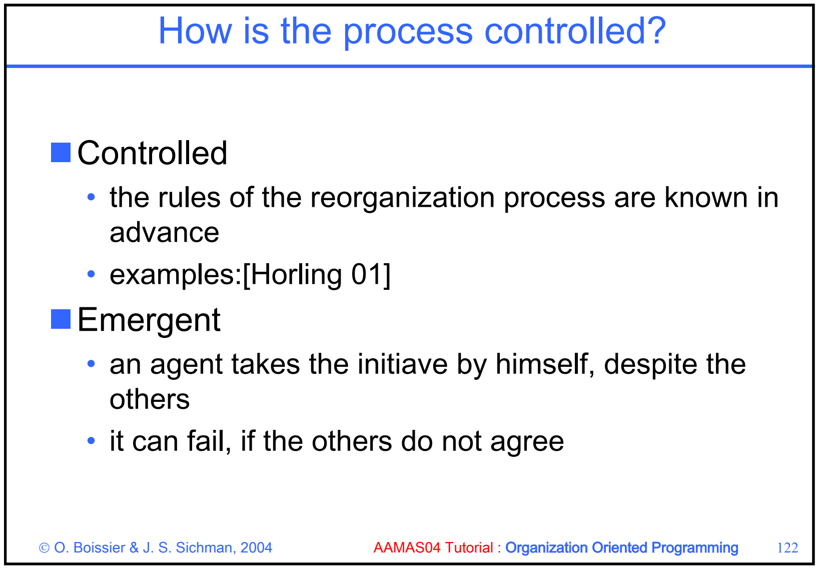
When is the process started?

- Static
  - process start is already predefined, already « designed » in the organizational specification
  - examples: [Stone 98] [Carron 01]
- Dynamic
  - reorganization happens as a consequence of the system functioning
  - If the system (agents) goal and/or performance is not adequate, the organization must be changed



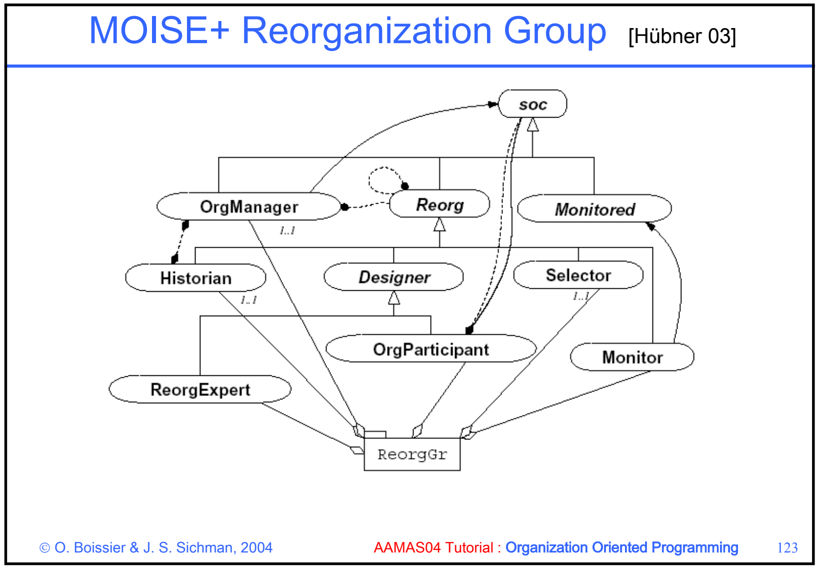
Who takes the initiative?

- Endogenous
  - one agent (centralized) or many agents (decentralized) within the system
  - auto-organization (adaptation, learning)
- Exogenous
  - MAS user
  - example: [Malone 99]

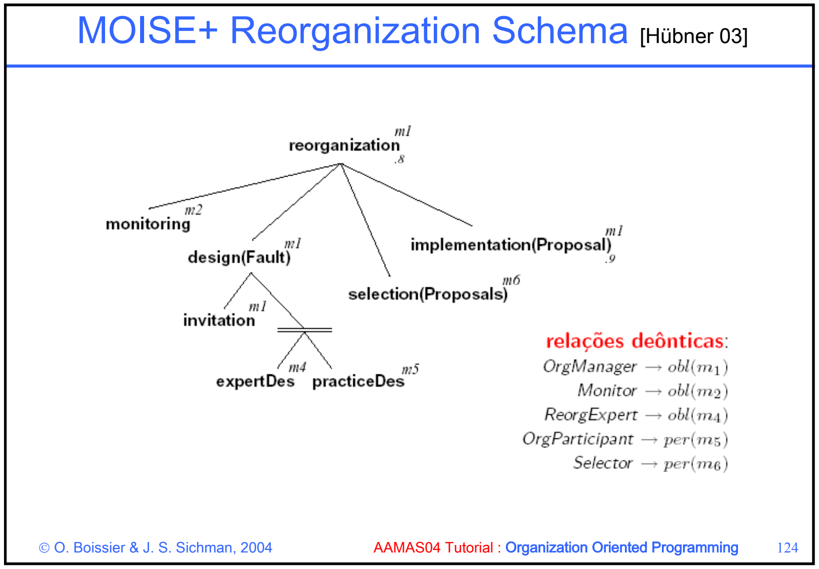


How is the process controlled?

- Controlled
  - the rules of the reorganization process are known in advance
  - examples:[Horling 01]
- Emergent
  - an agent takes the initiave by himself, despite the others
  - it can fail, if the others do not agree



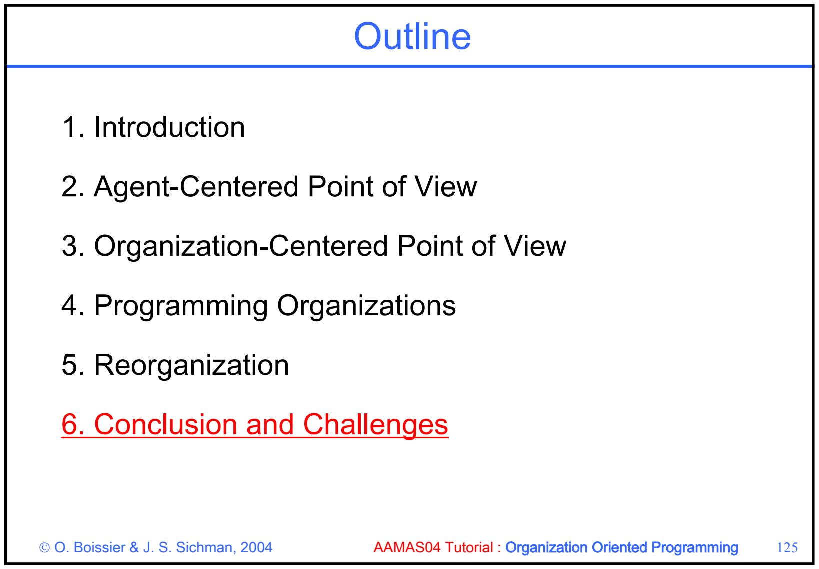
MOISE+ Reorganization Group [Hübner 03]



MOISE+ Reorganization Schema [Hübner 03]

relações deônticas:

$\mathit{OrgManager} \rightarrow obl(m_1)$
$\mathit{Monitor} \rightarrow obl(m_2)$
$\mathit{ReorgExpert} \rightarrow obl(m_4)$
$\mathit{OrgParticipant} \rightarrow per(m_5)$
$\mathit{Selector} \rightarrow per(m_6)$



Outline

1. Introduction

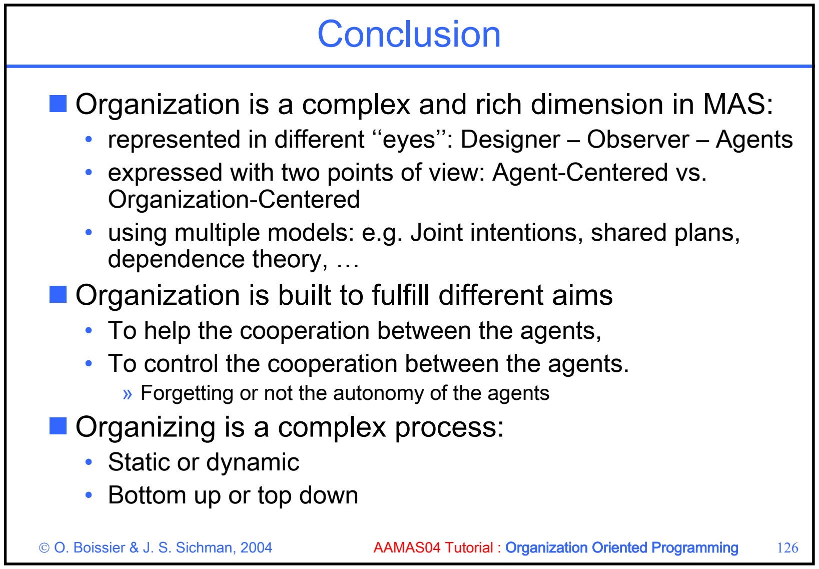
2. Agent-Centered Point of View

3. Organization-Centered Point of View

4. Programming Organizations

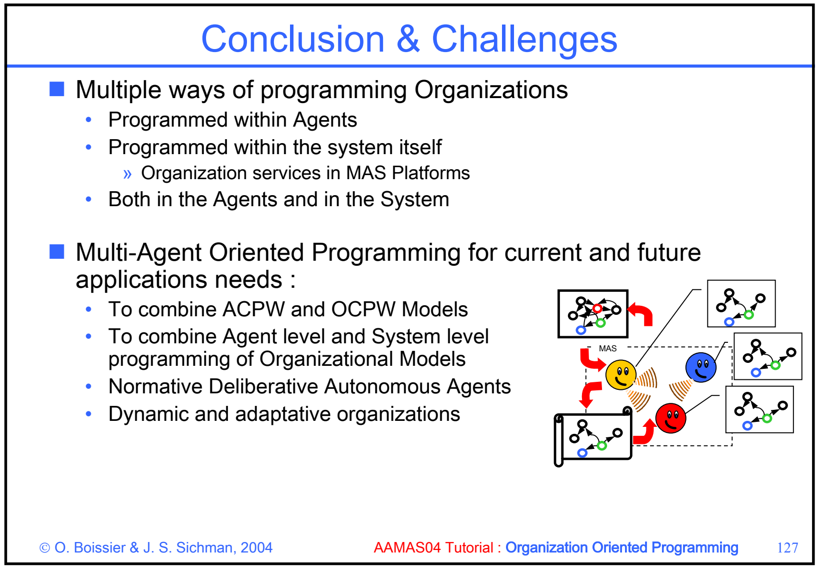
5. Reorganization

6. Conclusion and Challenges



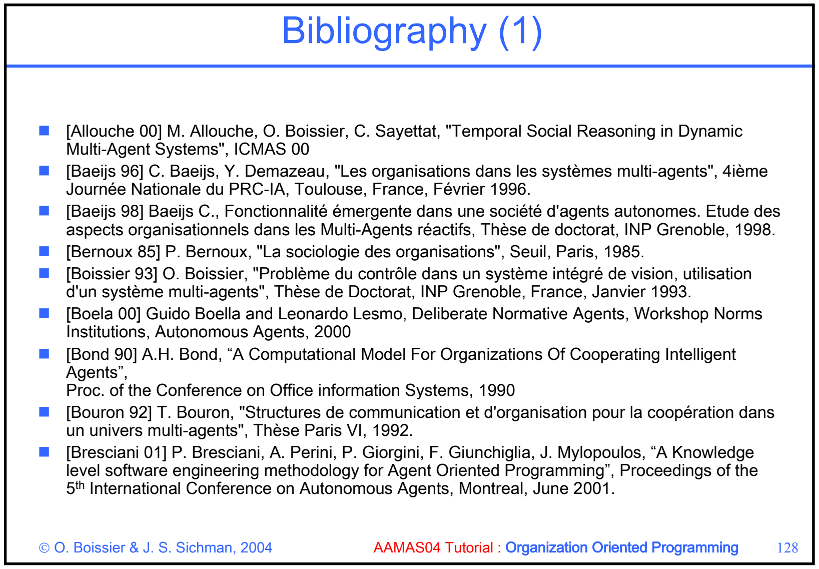
Conclusion

- Organization is a complex and rich dimension in MAS:
  - represented in different "eyes": Designer – Observer – Agents
  - expressed with two points of view: Agent-Centered vs. Organization-Centered
  - using multiple models: e.g. Joint intentions, shared plans, dependence theory, …
- Organization is built to fulfill different aims
  - To help the cooperation between the agents,
  - To control the cooperation between the agents.
    - » Forgetting or not the autonomy of the agents
- Organizing is a complex process:
  - Static or dynamic
  - Bottom up or top down



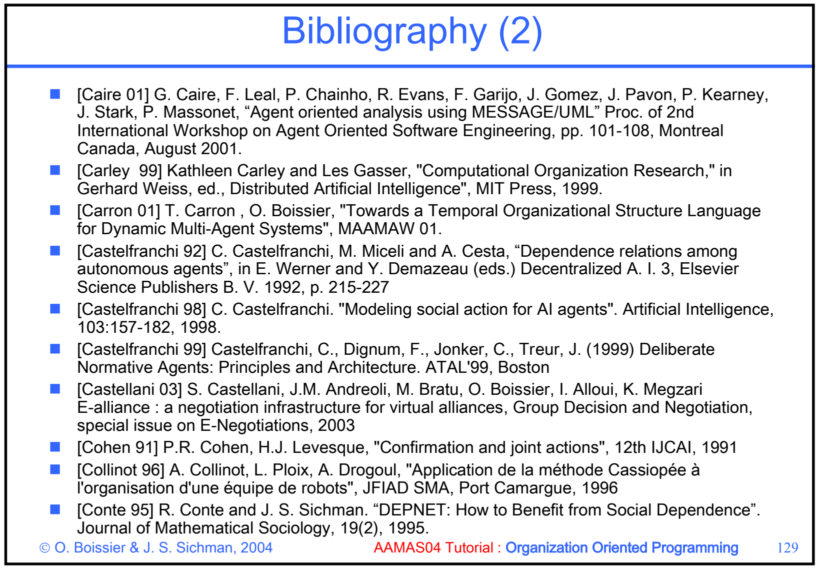
Conclusion & Challenges

- Multiple ways of programming Organizations
  - Programmed within Agents
  - Programmed within the system itself
    - » Organization services in MAS Platforms
  - Both in the Agents and in the System
- Multi-Agent Oriented Programming for current and future applications needs :
  - To combine ACPW and OCPW Models
  - To combine Agent level and System level programming of Organizational Models
  - Normative Deliberative Autonomous Agents
  - Dynamic and adaptative organizations



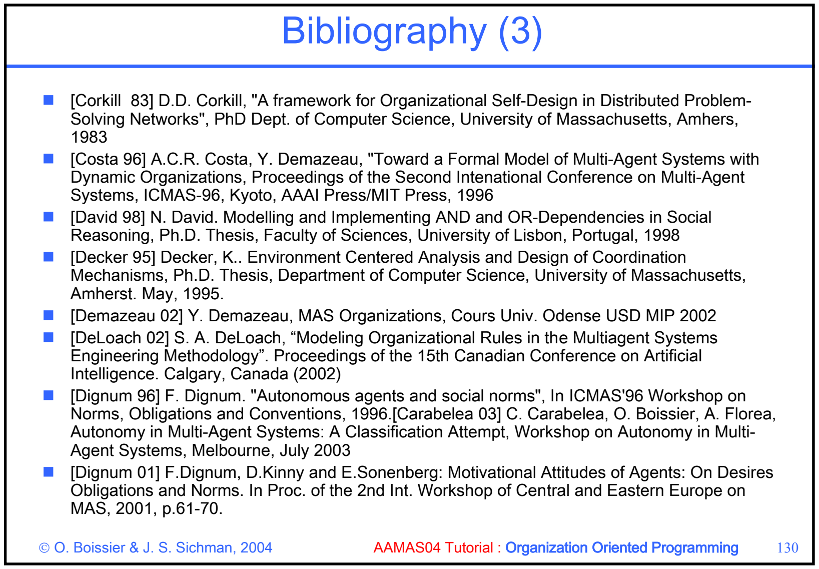
Bibliography (1)

- [Allouche 00] M. Allouche, O. Boissier, C. Sayettat, "Temporal Social Reasoning in Dynamic Multi-Agent Systems", ICMAS 00
- [Baeijs 96] C. Baeijs, Y. Demazeau, "Les organisations dans les systèmes multi-agents", 4ième Journée Nationale du PRC-IA, Toulouse, France, Février 1996.
- [Baeijs 98] Baeijs C., Fonctionnalité émergente dans une société d'agents autonomes. Etude des aspects organisationnels dans les Multi-Agents réactifs, Thèse de doctorat, INP Grenoble, 1998.
- [Bernoux 85] P. Bernoux, "La sociologie des organisations", Seuil, Paris, 1985.
- [Boissier 93] O. Boissier, "Problème du contrôle dans un système intégré de vision, utilisation d'un système multi-agents", Thèse de Doctorat, INP Grenoble, France, Janvier 1993.
- [Boela 00] Guido Boella and Leonardo Lesmo, Deliberate Normative Agents, Workshop Norms Institutions, Autonomous Agents, 2000
- [Bond 90] A.H. Bond, "A Computational Model For Organizations Of Cooperating Intelligent Agents",
  Proc. of the Conference on Office information Systems, 1990
- [Bouron 92] T. Bouron, "Structures de communication et d'organisation pour la coopération dans un univers multi-agents", Thèse Paris VI, 1992.
- [Bresciani 01] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, J. Mylopoulos, "A Knowledge level software engineering methodology for Agent Oriented Programming", Proceedings of the 5th International Conference on Autonomous Agents, Montreal, June 2001.



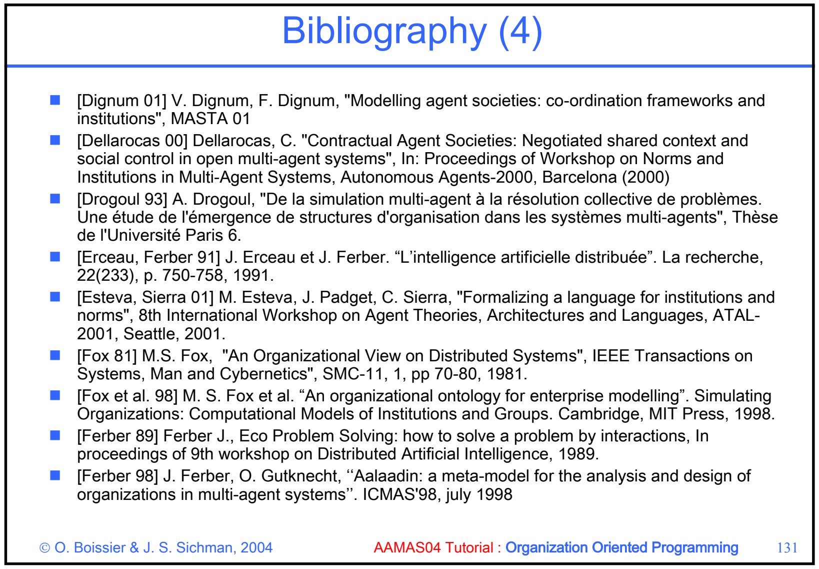
Bibliography (2)

- [Caire 01] G. Caire, F. Leal, P. Chainho, R. Evans, F. Garijo, J. Gomez, J. Pavon, P. Kearney, J. Stark, P. Massonet, “Agent oriented analysis using MESSAGE/UML” Proc. of 2nd International Workshop on Agent Oriented Software Engineering, pp. 101-108, Montreal Canada, August 2001.
- [Carley 99] Kathleen Carley and Les Gasser, "Computational Organization Research," in Gerhard Weiss, ed., Distributed Artificial Intelligence", MIT Press, 1999.
- [Carron 01] T. Carron , O. Boissier, "Towards a Temporal Organizational Structure Language for Dynamic Multi-Agent Systems", MAAMAW 01.
- [Castelfranchi 92] C. Castelfranchi, M. Miceli and A. Cesta, “Dependence relations among autonomous agents”, in E. Werner and Y. Demazeau (eds.) Decentralized A. I. 3, Elsevier Science Publishers B. V. 1992, p. 215-227
- [Castelfranchi 98] C. Castelfranchi. "Modeling social action for AI agents". Artificial Intelligence, 103:157-182, 1998.
- [Castelfranchi 99] Castelfranchi, C., Dignum, F., Jonker, C., Treur, J. (1999) Deliberate Normative Agents: Principles and Architecture. ATAL'99, Boston
- [Castellani 03] S. Castellani, J.M. Andreoli, M. Bratu, O. Boissier, I. Alloui, K. Megzari E-alliance : a negotiation infrastructure for virtual alliances, Group Decision and Negotiation, special issue on E-Negotiations, 2003
- [Cohen 91] P.R. Cohen, H.J. Levesque, "Confirmation and joint actions", 12th IJCAI, 1991
- [Collinot 96] A. Collinot, L. Ploix, A. Drogoul, "Application de la méthode Cassiopée à l'organisation d'une équipe de robots", JFIAD SMA, Port Camargue, 1996
- [Conte 95] R. Conte and J. S. Sichman. “DEPNET: How to Benefit from Social Dependence”. Journal of Mathematical Sociology, 19(2), 1995.



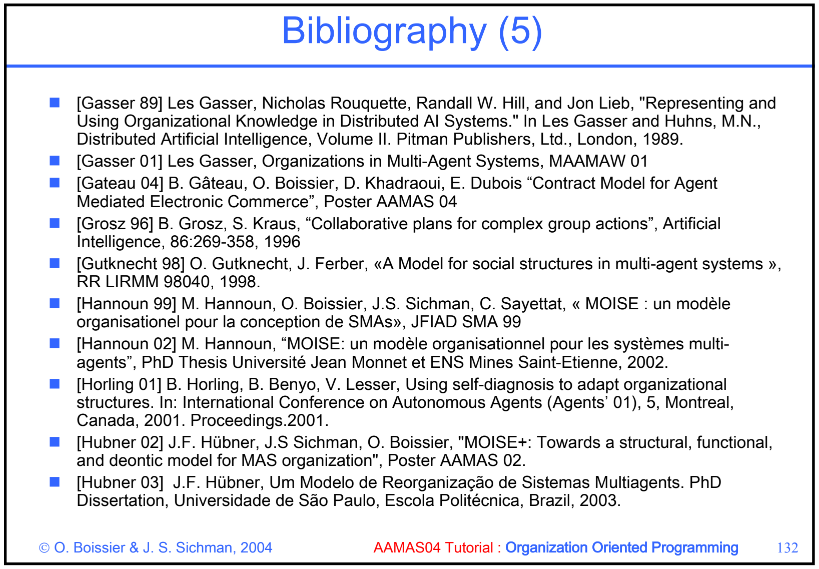
Bibliography (3)

- [Corkill 83] D.D. Corkill, "A framework for Organizational Self-Design in Distributed Problem-Solving Networks", PhD Dept. of Computer Science, University of Massachusetts, Amhers, 1983
- [Costa 96] A.C.R. Costa, Y. Demazeau, "Toward a Formal Model of Multi-Agent Systems with Dynamic Organizations, Proceedings of the Second Intenational Conference on Multi-Agent Systems, ICMAS-96, Kyoto, AAAI Press/MIT Press, 1996
- [David 98] N. David. Modelling and Implementing AND and OR-Dependencies in Social Reasoning, Ph.D. Thesis, Faculty of Sciences, University of Lisbon, Portugal, 1998
- [Decker 95] Decker, K.. Environment Centered Analysis and Design of Coordination Mechanisms, Ph.D. Thesis, Department of Computer Science, University of Massachusetts, Amherst. May, 1995.
- [Demazeau 02] Y. Demazeau, MAS Organizations, Cours Univ. Odense USD MIP 2002
- [DeLoach 02] S. A. DeLoach, “Modeling Organizational Rules in the Multiagent Systems Engineering Methodology”. Proceedings of the 15th Canadian Conference on Artificial Intelligence. Calgary, Canada (2002)
- [Dignum 96] F. Dignum. "Autonomous agents and social norms", In ICMAS'96 Workshop on Norms, Obligations and Conventions, 1996.[Carabelea 03] C. Carabelea, O. Boissier, A. Florea, Autonomy in Multi-Agent Systems: A Classification Attempt, Workshop on Autonomy in Multi-Agent Systems, Melbourne, July 2003
- [Dignum 01] F.Dignum, D.Kinny and E.Sonenberg: Motivational Attitudes of Agents: On Desires Obligations and Norms. In Proc. of the 2nd Int. Workshop of Central and Eastern Europe on MAS, 2001, p.61-70.



Bibliography (4)

- [Dignum 01] V. Dignum, F. Dignum, "Modelling agent societies: co-ordination frameworks and institutions", MASTA 01
- [Dellarocas 00] Dellarocas, C. "Contractual Agent Societies: Negotiated shared context and social control in open multi-agent systems", In: Proceedings of Workshop on Norms and Institutions in Multi-Agent Systems, Autonomous Agents-2000, Barcelona (2000)
- [Drogoul 93] A. Drogoul, "De la simulation multi-agent à la résolution collective de problèmes. Une étude de l'émergence de structures d'organisation dans les systèmes multi-agents", Thèse de l'Université Paris 6.
- [Erceau, Ferber 91] J. Erceau et J. Ferber. “L’intelligence artificielle distribuée”. La recherche, 22(233), p. 750-758, 1991.
- [Esteva, Sierra 01] M. Esteva, J. Padget, C. Sierra, "Formalizing a language for institutions and norms", 8th International Workshop on Agent Theories, Architectures and Languages, ATAL-2001, Seattle, 2001.
- [Fox 81] M.S. Fox, "An Organizational View on Distributed Systems", IEEE Transactions on Systems, Man and Cybernetics", SMC-11, 1, pp 70-80, 1981.
- [Fox et al. 98] M. S. Fox et al. “An organizational ontology for enterprise modelling”. Simulating Organizations: Computational Models of Institutions and Groups. Cambridge, MIT Press, 1998.
- [Ferber 89] Ferber J., Eco Problem Solving: how to solve a problem by interactions, In proceedings of 9th workshop on Distributed Artificial Intelligence, 1989.
- [Ferber 98] J. Ferber, O. Gutknecht, ‘‘Aalaadin: a meta-model for the analysis and design of organizations in multi-agent systems’’. ICMAS'98, july 1998



Bibliography (5)

- [Gasser 89] Les Gasser, Nicholas Rouquette, Randall W. Hill, and Jon Lieb, "Representing and Using Organizational Knowledge in Distributed AI Systems." In Les Gasser and Huhns, M.N., Distributed Artificial Intelligence, Volume II. Pitman Publishers, Ltd., London, 1989.
- [Gasser 01] Les Gasser, Organizations in Multi-Agent Systems, MAAMAW 01
- [Gateau 04] B. Gâteau, O. Boissier, D. Khadraoui, E. Dubois “Contract Model for Agent Mediated Electronic Commerce”, Poster AAMAS 04
- [Grosz 96] B. Grosz, S. Kraus, “Collaborative plans for complex group actions”, Artificial Intelligence, 86:269-358, 1996
- [Gutknecht 98] O. Gutknecht, J. Ferber, «A Model for social structures in multi-agent systems », RR LIRMM 98040, 1998.
- [Hannoun 99] M. Hannoun, O. Boissier, J.S. Sichman, C. Sayettat, « MOISE : un modèle organisationel pour la conception de SMAs», JFIAD SMA 99
- [Hannoun 02] M. Hannoun, “MOISE: un modèle organisationnel pour les systèmes multi-agents”, PhD Thesis Université Jean Monnet et ENS Mines Saint-Etienne, 2002.
- [Horling 01] B. Horling, B. Benyo, V. Lesser, Using self-diagnosis to adapt organizational structures. In: International Conference on Autonomous Agents (Agents’ 01), 5, Montreal, Canada, 2001. Proceedings.2001.
- [Hubner 02] J.F. Hübner, J.S Sichman, O. Boissier, "MOISE+: Towards a structural, functional, and deontic model for MAS organization", Poster AAMAS 02.
- [Hubner 03] J.F. Hübner, Um Modelo de Reorganização de Sistemas Multiagents. PhD Dissertation, Universidade de São Paulo, Escola Politécnica, Brazil, 2003.

Bibliography (6)

- [Ito 00] M. Ito. Uma análise comparativa do fluxo de mensagens entre os modelos da rede contratual (RC*) e coalisões baseadas em dependência (CBD), Ph.D. Thesis, Polytechnic School, University of São Paulo, Brazil, 2000.
- [Jennings 93] N.R. Jennings, « Commitments and Conventions : the foundation of coordination in multi-agent systems », Know. Eng. Rev. 8(3), 223-250
- [Jennings 95] N.R. Jennings, “Controlling Cooperative Problem Solving in Industrial Multi-Agent Systems using Joint Intentions”, Artificial Intelligence, 75, 1995.
- [Jennings 96] N. R. Jennings, P. Faratin, M. J. Johnson, P. O'Brien, M. E. Wiegand:Using Intelligent Agents to Manage Business Processes, Proceedings of First International Conference on The Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM96), London, UK, 345-360
- [Lemaître 98] C. Lemaître, C.B. Excelente, "Multi-Agent Organization Approach", In Francisco J. Garijo and Christian Lemaître, editors, Proceedings of II Iberoamerican Workshop on DAI and MAS, Toledo, Spain, 1998
- [Lesser 80] Lesser (V.R.) et Erman (L.D.), "Distributed interpretation : A model and experiment", IEEE Transactions on Computers, vol. C-2, n° 12, 1980, pp. 1144-1163.
- [Lesser 04] Lesser, V.; Decker, K.; Wagner, T.; Carver, N.; Garvey, A.; Horling, B.; Neiman, D.; Podorozhny, R.; NagendraPrasad, M.; Raja, A.; Vincent, R.; Xuan, P.; Zhang, X.Q.. Evolution of the GPGP/TAEMS Domain-Independent Coordination Framework, In Autonomous Agents and Multi-Agent Systems, Volume 9, Number 1, pp. 87-143. July, 2004. In Press
- [Levesque 90] H.J. Levesque, P.R. Cohen, J. Nunes, “On acting together”, Proc. Of the AAAI, 1990.

Bibliography (7)

- [Malone 99] MALONE, T. W. Tools for inventing organizations: Toward a handbook of organizational process. Management Science, v. 45, n. 3, p. 425–443, March 1999.
- [MARCIA 97] Groupe MARCIA, "L'auto-organisation comme objet d'étude dans les systèmes multi-agents", PRCIA 97[Malone 87] T.W. Malone, "Modeling coordination in organizations and markets", Management Science, 33(10):1317-1332, 1987.
- [Minsky 91] Minsky (N.H.). "The imposition of Protocols over Open Distributed Systems". Rapport technique n° LCSR-TR-154, Laboratory for Computer Science Research, Rutgers University, February 1991.
- [Morin 77] E. Morin, "La méthode (1) La nature de la nature", Seuil, 1977.
- [Oliveira 99] Rocha, A.P., Oliveira, E.: An Electronic Market Architecture for the formation of Virtual Enterprises.Proceedings of PRO-VE'99 IFIP/PRODNET Conference on Infrastructures for Industrial Virtual Enterprises, Porto, October (1999)
- [Omicini 03] Omicini, A.; Ricci, A.; and Viroli, M. 2003. Formal specification and enactment of security policies through Agent Coordination Contexts. In Focardi, R., and Zavattaro, G., eds., Security Issues in Coordination Models, Languages and Systems, volume 85(3) of Electronic Notes in Theoretical Computer Science. Elsevier Science B. V.
- [Ossowski 99] Ossowski, S. 1999. "Co-ordination in Artificial Agent Societies", LNAI 1535, Springer Verlag .
- [Pattison 87] H.E. Pattison, D.D. Corkill, V.R. Lesser, "Instantiating Description of Organizational Structures", DAI, Research Notes in AI, M.N. Huhns Ed., Morgan Kaufman Pitman Publishers, pp 59-96, 1987.

Bibliography (8)

- [Pynadath 03] D.V.Pynadath and M.Tambe. Automated Teamwork among heterogeneous software agents and humans, Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS). 7:71--100, 2003
- [Pynadath 99] D.V. Pynadath, M. Tambe, N. Chauvat, L. Cavedon, ‘‘Toward Team-Oriented Programming’’, ATAL 99
- [Rich 97] C. Rich, C. Sidner, “COLLAGEN: When agents collaborate with people”, Agents’ 97, 1997.
- [Sichman et al. 94] J.S. Sichman, R. Conte, Y. Demazeau and C. Castelfranchi. "A social reasoning mechanism based on dependence networks" . In: Proc. 12th European Conference on Artificial Intelligence (ECAI'94) Amsterdam, The Netherlands, August 1994.
- [Sichman 95] J.S. Sichman. "Du raisonnement social chez les agents: une approche fondée sur la théorie de la dépendance“, Phd thesis, Institut National Polytechnique de Grenoble, France, 1995.
- [Sichman 95] J.S. Sichman and Y. Demazeau. "Exploiting social reasoning to deal with agency level inconsistency" . In: Proc. 1st International Conference on Multi-Agent Systems (ICMAS'95) San Francisco, USA, June 1995.
- [Sichman 98] J.S. Sichman, "DEPINT: Dependence-Based Coalition Formation in an Open Multi-Agent Scenario" . In: Journal of Artificial Societies and Social Simulation, 1(2), 1998. Available at http://www.soc.surrey.ac.uk/JASSS/1/2/3.html
- [Singh 97] M.P. Singh, "Commitments Amont autonomous Agents in Information Rich Environment" LNAI 1237, Multi-Agent Rationality 8th MAAMAW M. Boman, W. Van de velde eds, Springer-Verlag, 1997

Bibliography (9)

- [Smith 80] R.G. Smith, "The contract net protocol : High Level Communication and Control in a Distributed Problem Solver", IEEE Trans. on Computers, 29(12), 1104-1113, 1980.
- [Stone 98] Stone, P., Veloso M., "Task Decomposition and Dynamic Role Assignment for Real-Time Strategic Teamwork", in ATAL'98, Paris, France, 1998.
- [Tambe 98] Tambe, M. and Zhang, W. 1998 Towards flexible teamwork in persistent teams, Proceedings of the International conference on multi-agent systems (ICMAS)
- [Tambe 00] Tambe, M., Pynadath, D. and Chauvat, N. 2000 Building dynamic agent organizations in cyberspace, IEEE Internet Computing, Vol 4, number 2
- [Tidhar 96] G. Tidhar, A. S. Rao, and E. A. Sonenberg. Guided-Team Selection, In Proceedings of the Second International Conference on Multi-Agent Systems, Kyoto, Japan, December 1996. AAAI Press.
- [Van Aeken 99] F. Van Aeken, Les systèmes multi-agents minimaux, PhD Thesis, Leibniz IMAG INPGrenoble, 1999,
- [Werner 89] E. Werner, "Cooperating Agents : A unified theory of communication and social structure", In Les Gasser and Huhns, M.N., Distributed Artificial Intelligence, Volume II. Pitman Publishers, Ltd., London, 1989.
- [Wooldridge 00] M. Wooldridge, N.R. Jennings, D. Kinny, “The GAIA Methodology for Agent oriented analysis and design”, Journal AAMAS, 3(3):285-312, 2000
- [Zambonelli 01] F. Zambonelli, N. Jennings, M. Wooldridge, “Organizational Rules as an Abstraction for the Analysis and Design of MultiAgent Systems”, Journal of Software and Knowledge Engineering, 11(3), 2001.