Odense
05 August 2010

# MAS Course 2

**Yves Demazeau**
Yves.Demazeau@imag.fr



## SCHEDULE OF THE COURSE + EXAMINATION

| | | |
|---|---|---|
| MAS 01 | 04 Aug. | Introduction, Methodology, Agents |
| MAS 02 | 05 Aug. | Agents, Environments, Interactions |
| MAS 03 | 06 Aug. | |
| MAS 04 | 09 Aug. | |
| MAS 05 | 10 Aug. | |
| MAS 06 | 11 Aug. | |

attendance ; handouts ; individual work
[ Ferber 95 ] [ HERMES 01 ] [ OFTA 04 ]

MAS Examination 13 Aug. Written Control

## Classes of Reactive Agents [Erceau 91]

| | |
|---|---|
| organized agents | colonies |
| reproducing agents | reproduction mechanisms |
| cooperative agents | recruiting and agregating mechanisms |
| coordinated actions | activation / inhibition mechanisms |
| stimulus answer | finite state automata |



## Alternative Classification [Demazeau 91 -> 01]

**Elementary behaviours**

- Brooks, Steels — walking robots
- Steels, Maes — games

**Elementary situations**

- Demazeau 90 — walking robots
- Agree, Chapman — games

**Elementary interactions**

- Demazeau 93 — image analysis
- Ferber — games

**Elementary capabilities**

- Demazeau 96 — sociology
- Everybody — personal assistants

## Freddy Walker (academic project)

**A robot that learns to walk** LIFIA-CNRS (F), VUB (B)

**A legged robot which learns to coordinate the moves of its legs to acheive a go-forward gait**

### Tool

- ad-hoc metal structure, step-to-step motors
- global feedback sensor : forward, backward, no-move
- implementation of control and learning is not embedded

### Model

- legs = agents as finite state automata
- node : position of the leg (4)
- weight : probability of transition between states



## Freddy Walker : experiments

### Learning process

- choice of the next position : Bayes, Uniform Distr.
- updating of the weights : reinforcement learning
- satisfactory gait = 60 à 100 % forward
- experiments with(out) connection between graphs

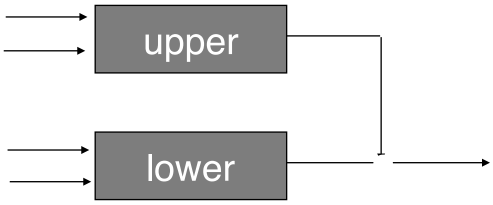
### A - E interactions alone

- autonomous legs : no interconnection between state graphs
- effective coordination in ± 300 steps

### A - E + A - A interactions

- semi-autonomous legs : fully interconnected state graphs
- extended learning process
- effective coordination in ± 100 steps

## Agent Architecture : Brooks



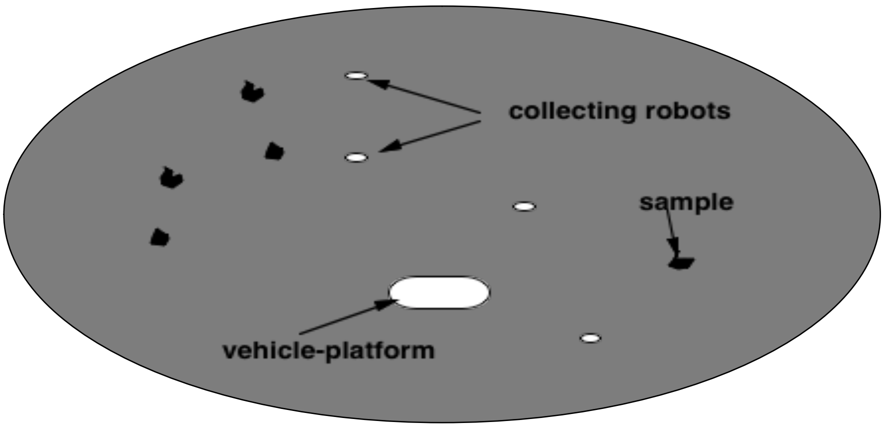

Each module is a finite state automaton
The message issued from the upper has priority in front of the message issued by the lower (subsumption architecture)

Realisation of a number of real robots, including a soda can collector, a walking robot, ...

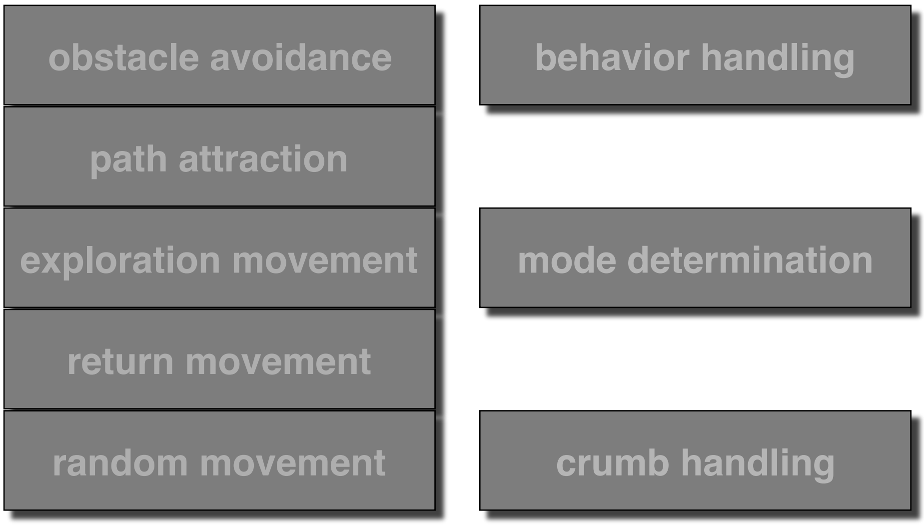


## Steels agents : the problem to solve

A set of robots have to collect samples and bring them to some vehicle-platform



## Steels agents : Agent Design






## Steels agents : Control Behaviors

### Behavior handling
- if I sense a sample and am not carrying one, I pick it up
- if I sense the vehicle-platform and am carrying a sample, I drop it

### Mode determination
- if I am in exploration mode and I sense no lower concentration than the concentration in the cell on which I am located, I put myself in return mode
- if I am in return mode and I am at the vehicle-platform, I put myself in exploration mode
- if I am holding a sample, I am in the return mode

### Crumb handling
- if I carry a sample, I drop 2 crumbs
- if I carry no sample and crumbs are detected, I pick up one crumb

## Steels agents : Movement Behaviors

### Obstacle avoidance
- if I sense an obstacle in front, I make a random turn

### Path attraction
- if I am not carrying a sample and I sense crumbs, I move towards the highest concentration of crumbs

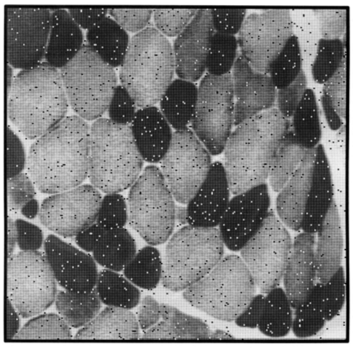
### Exploration movement
- in exploration mode I chose the direction with the lowest gradient

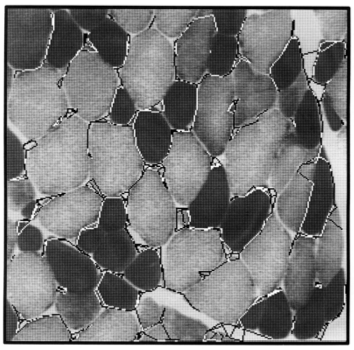
### Return movement
- in return mode I chose the direction of highest gradient

### Random movement
- choose randomly a direction to move and move in that direction



## EPs : Segmentation into Regions





## Agent Architecture : Demazeau (1) Principle

**PACO : A solution of a problem is an equilibrium of a set of independent programmable agents which locally interact one with each other as well as with the environment data by means of forces.**

**The PACO model is defined by the predetermined combination of the forces**

**Scopes : Each agent is characterized by a perception scope, a communication scope, an action scope, and by the forces exerted by the environment and the other agents.**

**A problem is solved by as soon as the set of agents reach a perceived equilibrium**



## Agent Architecture : Demazeau (2) Cycle

### Local Perception and Communication

- tune the sensitivity with the environment (PSi)
- perceive
- tune the sensitivity with other agents (CSi)
- communicate

### Local Processing

- compute the interactions with the environment (EF)
- compute the interactions with other agents (IF)
- combine of the forces exerted on the agent

### Local Action

- tune the sensitivity to act using action scope (ASi)
- act

## PACO : Environment and Agents

### Environment

- subset of an N-space, shared by the agents. Statical or dynamical values.

### Agents Xi

- the agent state denotes a part of the global solution.
- mass, position, speed, acceleration, scopes

### Agent's Scopes Si

- perception scope PSi : determines the subset of the environment that the agent can perceive at a given time.
- communication scope CSi : determines the subset of agents the agent can communicate with at a given time
- action scope DSi : determines the subset of actions that the agent can perform at a given time
- static or dynamic scopes, but controlled by the agent according to its state and to the global goal to be satisfied.
- analogy with the Fire-Fighting



## PACO : Interactions and Organisations

### Interactions with the environment

- Each agent interacts with each element of the environment which it can perceive at a given time
- Interactions are modelled by as many types of forces that the agent is able to distinguish types of entities in the environment (not necessarily physical) (EF)

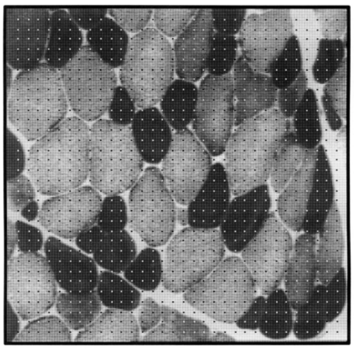
### Interactions with the other agents

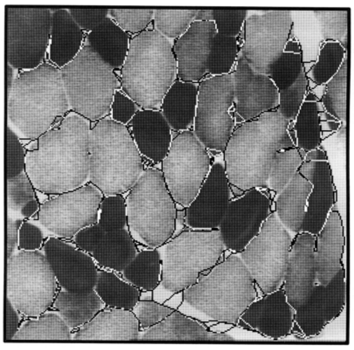
- Each agent interacts with the other agents with which it can communicate at a given time
- Interactions are modelled by forces (usually spring forces) that translate the granularity and the rigidity of the solution (IF)

### Organisations

- possible constraints to the agents by initial links between them a well as other kinds of constraints

## EPs : Segmentation into Regions (Cells)







## EPs : Segmentation into Regions

**Environment**
- {Yj} set of contrast points

**Agents Xi**
- PSi : infinite or fixed
- CSi : 1 ≤ Card(Xj/Xi perceives Xj according CSi) ≤ 3
- DSi : in coherence with the contrast

**Interactions with the environment**
- ∑j (PSi lXi - Yjl + 1)**(-k), k=1, 2
- ∑j exp -(PSi ln lXi-Yjl -f(PS))**2

**Interactions between agents**
- ∑j //CSi(j)// sign (lXi-Xjl - *μ*) [ß(lXi-Xjl - *μ*]**k, k=1, 3

**Getting the solution (by an external operator)**
- Visualising the links between agents which mutually perceive each other

## EPs : Intelligent Contour Detection

### Environment
- {Yj} set of contrast points

### Agents Xi
- PSi : infinite or fixed
- CSi : 1 ≤ Card(Xj/Xi perceives Xj according CSi) ≤ 2
- DSi : in coherence with the contrast

### Interactions with the environment
- ∑j (PSi lXi - Yjl + 1)**(-k), k=1, 2
- ∑j exp -(PSi ln lXi-Yjl -f(PS))**2

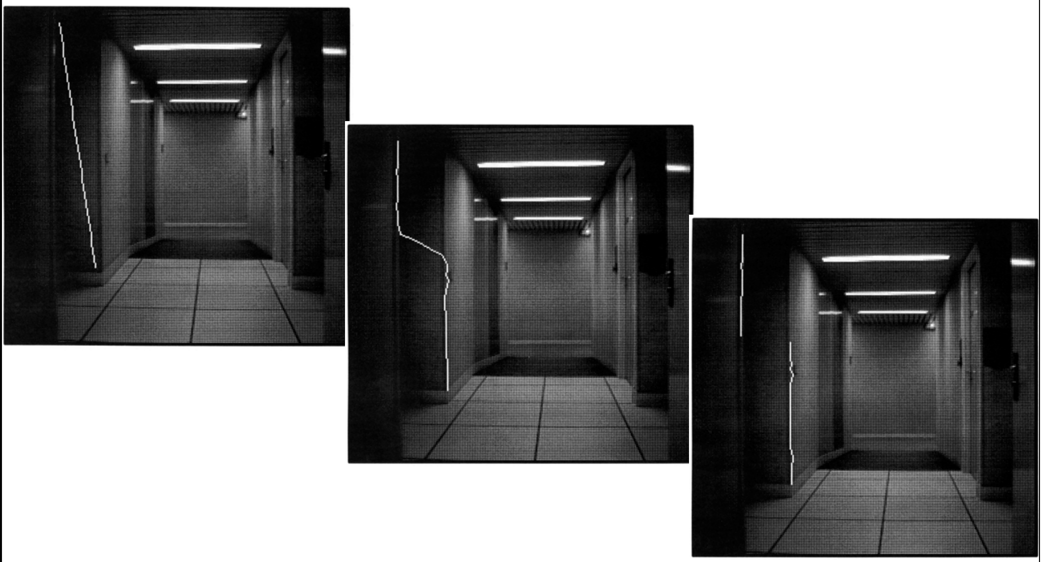
### Interactions between agents
- ∑j //CSi(j)// sign (lXi-Xjl - *μ*) [ß(lXi-Xjl - *μ*]**k, k=1, 3

### Getting the solution (by an external operator)
- Visualising the links between agents which mutually perceive each other



## EPs : Intelligent Contour Detection (Niçoise)

## EPs : Intelligent Contour Detection (Corridor)





## ENVIRONMENTS

## Global structure of a World modeller

| | |
|---|---|
| **LEVEL 3**<br>symbolic actions | Symbolic Knowledge |
| **LEVEL 2**<br>geometric actions | Physical Knowledge<br>about the universe |
| **LEVEL 1**<br>physical events | Physical Knowledge<br>about the agent |
| **LEVEL 0** | Temporal Updating<br>of the Real World |

Level 2 is a translator from symbolic to geometric actions, level 1 is a translator from geometric actions to events



## Agent, Envelop, Updator in the World modeller

| | | |
|---|---|---|
| **LEVEL 3**<br>symbolic actions | **agent**<br>**agent** | Symbolic Knowledge |
| **LEVEL 2**<br>geometric actions | **envelop**<br>**envelop** | Physical Knowledge<br>about the universe |
| **LEVEL 1**<br>physical events | **envelop**<br>**envelop** | Physical Knowledge<br>about the agent |
| **LEVEL 0** | **updator**<br>**updator** | Temporal Updating<br>of the Real World |

The envelop (levels 2 & 1) translates actions to be performed in the real world into the simulated world. The updator (level 0) is the sequencer and processor of the events produced in the simulated world.

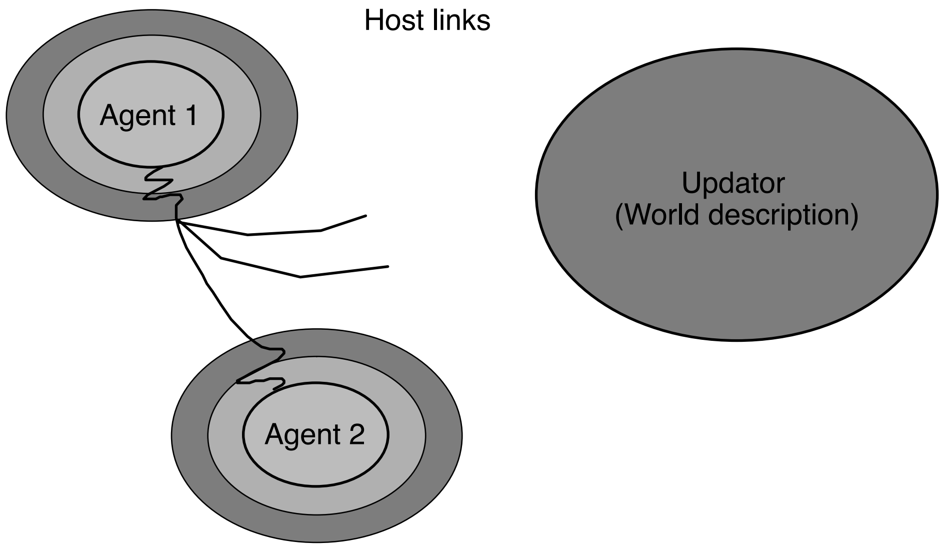
## Objective & subjective representations of the world

| | | |
|---|---|---|
| **LEVEL 3** | **subjective** | Symbolic Knowledge |
| symbolic actions | **subjective** | |
| **LEVEL 2** | **subjective** | Physical Knowledge |
| geometric actions | **subjective** | about the universe |
| **LEVEL 1** | **objective** | Physical Knowledge |
| physical events | **objective** | about the agent |
| **LEVEL 0** | **objective** | Temporal Updating |
| | **objective** | of the Real World |

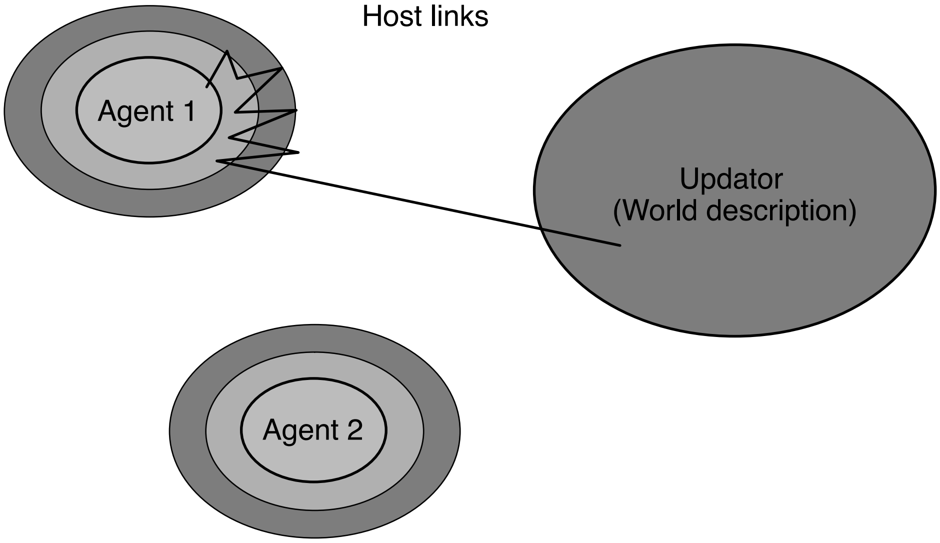
The subjective representation of the world by an agent is the one it perceives from the unique objective one that is encoded into the updator.



## Implementing a Communicating Action

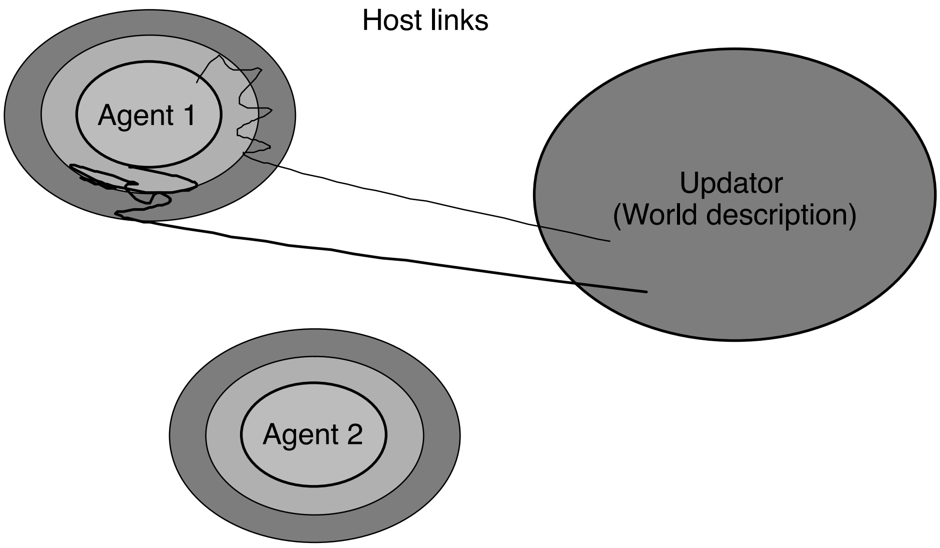


## Implementing an Acting Action






## Implementing a Perceiving Action



## INTERACTIONS



## Game Theoretic Interaction

**Agents are assumed to be high-level decision makers confronted with a static interaction**

- the interaction is modeled
- utilities are assigned to potential outcomes
- an analysis is made of one's opponent
- an action is selected.

**The object here is to define rationality axioms that constrain the agents' behavior in interesting ways.**

**Variations can include**

- communication and deal-making
- probabilistic assumptions regarding payoffs
- special assumptions regarding one's opponent
- conjunctive offers

## Iterated Case Analysis

| | | K | |
|---|---|---|---|
| | | c | d |
| J | a | 3<br>3 | 4<br>2 |
| | b | 1<br>1 | 2<br>4 |

J cannot, at first, rules out anything.
But reasoning about K's choosing d, J will choose b

| | | K | |
|---|---|---|---|
| | | c | d |
| J | a | 3<br>3 | 5<br>2 |
| | b | 2<br>5 | 1<br>1 |

(Game of Chicken)
Less stable situation.
How does the rational agent solve such an interaction ?



## More complex situations

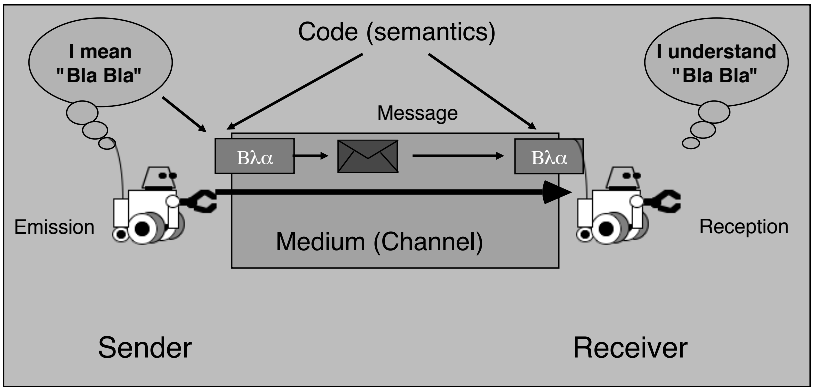
| | | K | |
|---|---|---|---|
| | | c | d |
| J | a | -1<br>-1 | 2<br>1 |
| | b | 1<br>2 | -1<br>-1 |

(Battle of the Sexes)
How about this basic interaction ?

| | | K | |
|---|---|---|---|
| | | c | d |
| J | a | 3<br>3 | 5<br>0 |
| | b | 0<br>5 | 1<br>1 |

(Prisoner's Dilemna)
In a case like this, binding deals can help to provide solutions

## Classical model of communication






## Sender / Receiver relationship

### Point to point

- (M1) A : B, Hello *A knows its receiver*

### Broadcast

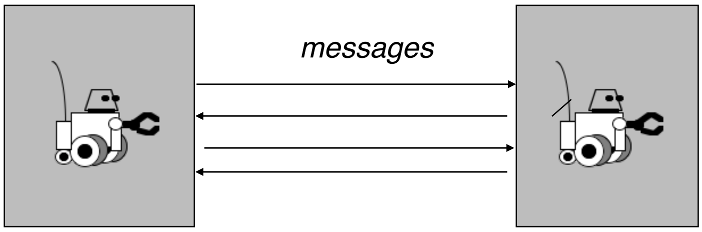
- (M2) A : All, Hello *A does not know its receiver*
- (M2') A : {x I dist(A,X) < d}, Hello

### Broadcast communications can be reduced to point to point communications using a "broker" as an intermediate agent

- (M3) A : {x I P(x)}, M *A does not know its receiver but an intermediate agent*
- (M3') A : C, broadcast M
- (M3") for all x that C knows C : x, M

## Message passing transmission

Communications are delivered directly to the receiver using a specialized service (e.g. surface mail, electronic mail,..)

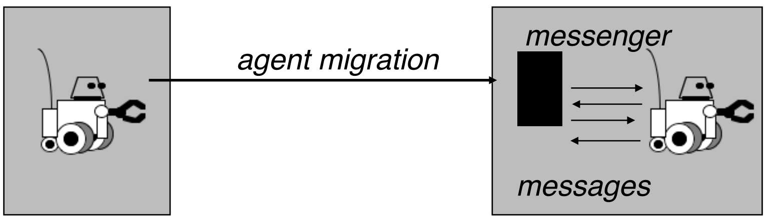



The most classical tranmission type in network communications
Complex protocols impose heavy traffic on the network



## Messenger transmission

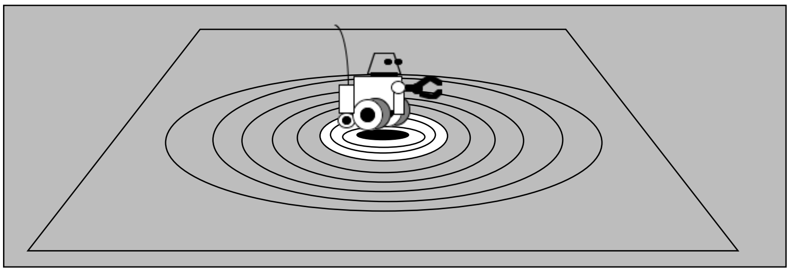
Communications are conveyed by an agent which migrates from places to places (Telescript)




Messengers are sent to distant places (sites)
A messenger contains a high-level message plus a behavior to implement low level conversations
High level conversations are easily distinguished from low level communications

## Propagation transmission

**The emitter sends a signal that propagates in the environment. The signal intensity decreases as a function of distance (and time)**



**V(x) = V(x0) / dist(x,x0)**



## Message meaning

**Fixed signification**
**=> intentional communications**
**Semantics of the communication is shared by the sender and the receiver**

- Supposes a language of communication common to all
- Pbs of standards definition

**The sender intends the meaning of the message**

**Meaning depends on the receiver**
**=> incident communications**
**The receiver gives a meaning to the communication**
**There is no "intentional" meaning of the sender**
**Communications are "signals"**

## Classification along mode / transmission / coding

| | Types of communications | Communication mode | Transmission | Coding and interpretation |
|---|---|---|---|---|
| Cognitive | Message passing communications | point to point and broadcast (using a broker) | direct | Intentional |
| | Messenger communications | point to point and broadcast (using a broker) | by messenger | Intentional (but depends on the messenger) |
| Reactive | Stimuli/signals | broadcast | propagation in the environment | Incident |



## Classification in the Information Society

**News : Communication by sharing information**
the agents are delivering and reading information using a shared resource

**Email : Communications by message passing**
- point to point communication
- broadcast communication

**Web : Communication through the environment**
the agents are leaving signals of their presence at some date, the signals may be perceived later by other agents

## Rationale of Communication Acts

Main theoretical support : **Speech Acts Theory** (Austin, Searle) where communications are regular actions

Agents communicate using **Interaction Languages** possibly associated with **Interaction Protocols** used to control the flow and sequencing of communication

Interaction protocols are modelled by **Automata** or **Petri nets**

Well-known Interaction Languages include **KQML** (without protocols, developped by the Knowledge Sharing Effort) and **ACL** (with protocols, developped by the Foundation for Intelligent Physical Agents)



## Introduction to Speech Act Theory (1)

**Concept developped initially in the context of the philosophy of language (Austin, Searle, Vanderveken, ..)**

**Communicating is acting**

- Sentences are not only true of false, they perform speech actions
- Communications are viewed as a regular actions which have to be generated and processed like every other kind of action

**Communication is pragmatic**

- it usually explains what is performed, not to what it refers (more direct communication rather than indirect one);
- requesting to do something is a way to achieve some goal

## Introduction to Speech Act Theory (2)

**Categorising communication types :** e.g. inform, ask ask-to-do, ask-for-info, request, answer, propose, warn, promise ...

**Decompose a sentence into its performative and its content F(P)**

- Ask(the light is on) *is the light on?*
- Inform(the light is on) *the light is on!*
- Request(the light is on) *switch on the light, please*

Performative content

**Each communication type is associated with the set of its consequences** – definition of protocols associated with each type of communication



## Introduction to Speech Act Theory (3)

**A speech act contains 3 acts:**

- **Locutionary Act :** uttering of words and sentences with meaning
- **Illocutionary Act :** type of action, *intent* of the utterance
- **Perlocutionary Act :** expected *(desired) result* of the utterance

**Searle classification of illocutionary acts into assertives, directives, commissives, declaratives, and expressives**

**Brennenstuhl grouping into models, ordering of the categories according to their temporal relationship and degree of strength, appropriate frameworks to structure dialogue protocols**

## Classification of Speech Acts [Searle]

**Assertive : gives an information about the world by asserting something**

**Directive : gives directives for the interlocutor**

**Promissive : engages the locutor to accomplish certain acts in the future**

**Declarative : accomplish an act with the very pronouncement of the statement**

**Expressive : gives the interlocutor indications about the mental state of the locutor**



## Semantics of Speech Acts

**A communication, using speech act theory can be specified in terms of the mental states of both the sender and the receiver**

**There are necessary and sufficient conditions for performing speech acts**

**Request(s, r, p)** ***=def***
**Pre: Goal(s,$\phi$) $\wedge$**
**Bel(s,[p$\rightarrow\phi\wedge$Helpful(r,s)**
**$\wedge\neg$CanDo(s,p)$\wedge$CanDo(r,p) ])**
**Post: Bel(s, Intend(r,p))**

**Where p$\rightarrow$f means: if p happen then f will be true**

## KQML [Finin 92]

**One of the most important implementation of speech acts so far:**

- Describe a set of performative like Request, Achieve, Deny, Ask-all, Subscribe, evaluate, delete, ...

**Has been implemented as a set of communication primitives:**

- Java agents

**Semantics drawbacks**

- Mixture of performative of different categories: lack of structure such as ISO standards for communications.
- Some performative are lacking (e.g. there are no promissive)
- Full of incoherencies (cannot be used as such)
- Weak semantics (work on this subject follows the work of Cohen & Levesque)

**Does not consider messenger-style transmissions**



## KQML [Finin 92] : Basics

**Produced by ARPA- KSE (Knowledge Sharing Effort).**

- Based on Speech Act theory where a message is a performative indicating what the receiver is expected to do with the message
- Offers a variety of message types, represented as performatives, that express an attitude regarding the content of the exchange.
- Message content: KIF formalism (Knowledge Inter-exchange Format)
- Provides a message format and message handling protocol supporting run-time knowledge sharing and interaction among agents.
- LISP, 41 performatives, 1st order predicate logic
- Informal semantics, no protocols, no commitment performatives

## KQML [Finin 92] : Syntax

**The syntax of KQML message is based on a balanced parenthesis list**

- the initial element of the list is the performative ; the remaining elements are the performatives's arguments as keyword / value pairs

**( ask-one : receiver *weather-station***
**: sender *forecaster***
**: content *rain (today, X)***
**: language *prolog***
**: reply-with *day10* )**



## KQML [Finin 92] : Examples of performatives

**ask-one S wants one of R's instantiations of the *:content* that is true of R**

**ask-all S wants all of R's instantiations of *:content* that are true of R**

**stream-all multiple-response version of ask-all**

**tell the sentence is in S's Virtual Knowledge Base**

**achieve S wants R to do make something true of its physical environment**

**broker-one S wants R to find one response to a performative**

## Interaction Protocols

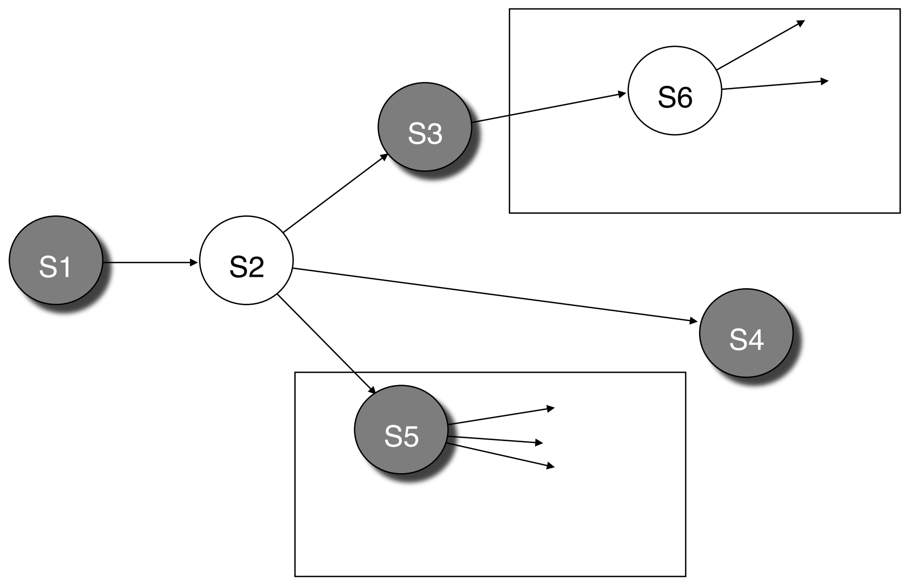
### A framework to define and structure
- with which and why, when, what and how communicate
- what the sender should expect after sending a message, and how to react to a message?

### A common frame of reference for the agents
- the rules that must be followed in order to interact,
- general rules that determine how agents should behave in various situations

### Representation as transition networks
- defining the set of possible transitions which link a set of states that the agents may alternately occupy according to the effective exchanged interaction acts
- a transition may constraint the filling of some fields in the interaction act.
- it also may be labelled by a condition that an agent has to satisfy before using the transition.

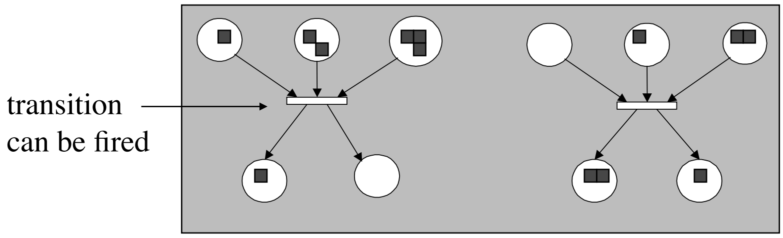


## A typical Request protocol



## Modelling Conversations as Petri Nets (1)

**Petri nets have been created to express concurrent processes as a generalization of automata**
**They are made of places, transitions and tokens (marks), with accompanying rules of transition firing**



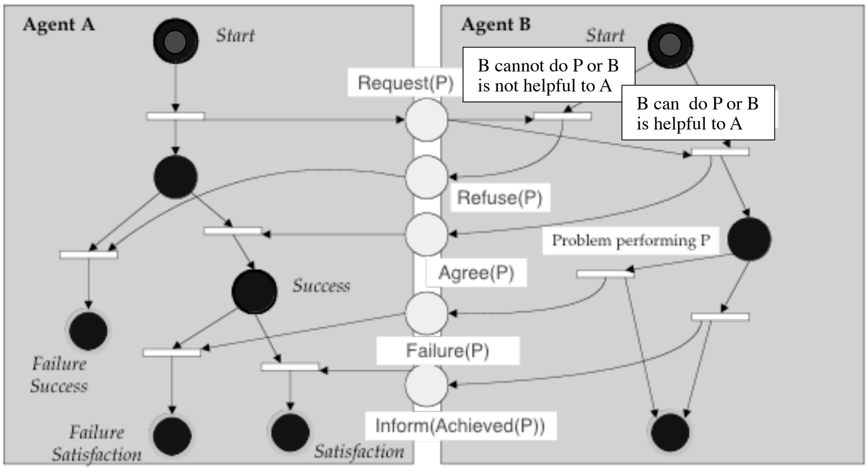

**Very well adapted to describe protocols**
- Already used broadly to describe network protocols

**A wide range of Petri net models**
- Basic, Coloured, Time, etc...



## Modelling Conversations as Petri Nets (2)

**Example: requesting to do something**



## IL [Demazeau 95]

**< IL message > ::=**

**< communication message >**
- translates the message from a pure distributed systems point of view

**< multi-agent message >**
- refering the multi-agent domain knowledge

**< application message >**
- e.g. application language for computer vision

**The IL message is physically supported by the communication message**

**The IL message is ontologically supported by the application message**



## IL : the Communication message

**The Communication physically supports the IL**

**< communication> ::=**

**< from >**
- refering the sender

**< to >**
- refering the receiver (agent entity or broadcast)

**< id >**
- identity of the message

**< via >**
- channel (direct message passing, BB, HBWC)

**< mode >**
- mode (synchronous, asynchronous)

## IL : the Multi-Agent message

**The Multi-Agent message determines the intention of the sender and its expected results in addition to a interaction protocol for reference**

**< multi-agent message > ::=**

**< type >**

- either : present, request, answer, or inform

**< strength >**

- prioritizing the message from sender's point of view

**< nature >**

- reflecting the expected control layer of the receiver

**<protocol>**

**<position>**

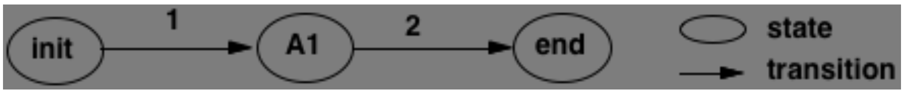


## IL <strength> (from [D'Inverno 90])

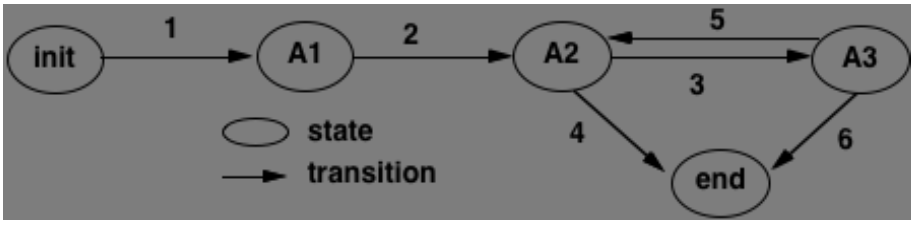
1. Action requesting
2. Information seeking
3. Information probing
4. Information checking
5. Instructing
6. Informing
7. Understanding event
8. Warning
9. Advising
10. Persuading
11. Promising
12. Bargaining
13. Impressing
14. Intimidating
15. Threatening
16. Commanding
17. Encouraging
18. Expressing
19. Offending
20. Misleading
21. Amusing

## IL "Request" Protocols [Boissier 93]

### The "Simplest Request-Answer" Protocol




### The "Request until Satisfaction" Protocol






## ACL [FIPA 96]

### Foundation for Intelligent Physical Agents

- International (but very european)

### ACL : Agent Communication Language

- A communication language based on speech acts similar to KQML, but with a clear semantics based also on modal logic
- It adopts the notion of interaction protocols
- In the process of standardization

## Coconstructing the meaning of an utterance

The meaning of an utterance is co-constructed in the course of the dialogue

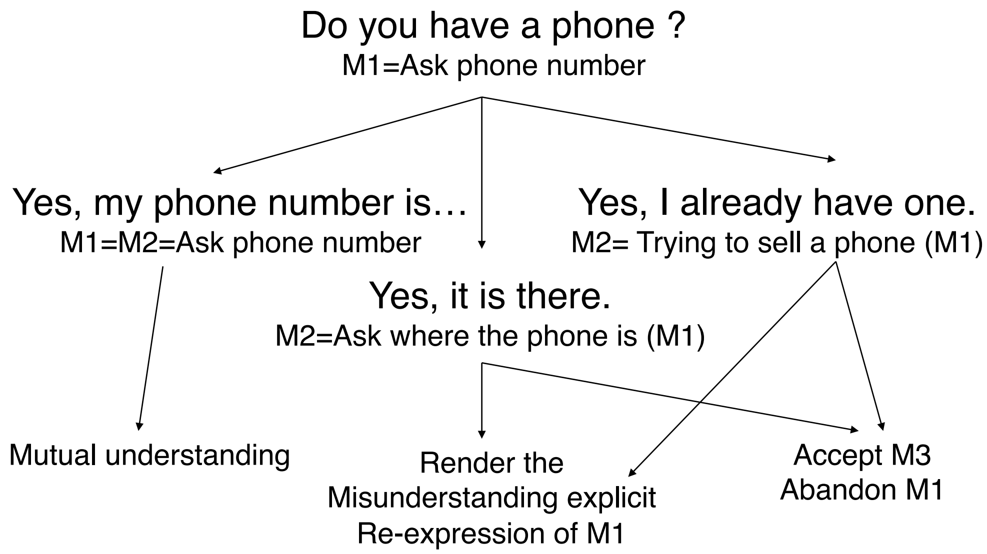
An utterance does not carry one meaning but a potential set of different meanings, including the intended meaning of the sender

The receiver interprets the utterance it gets into a meaning that may be different from the one expected by the sender, and reacts accordingly

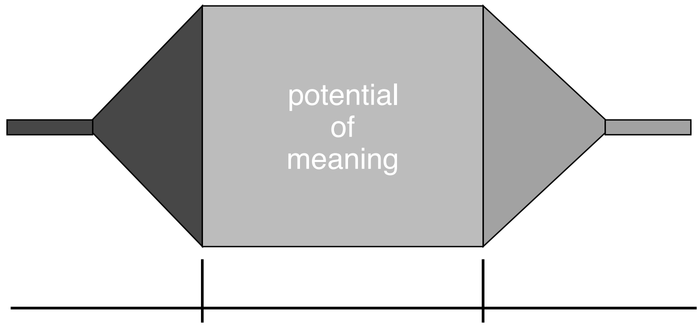
The receiver's reaction may be different from the one expected by the sender which then adapts its behaviour for further exchange



## Example of tracking and misunderstanding



## Potential of meaning & Conversation tracking



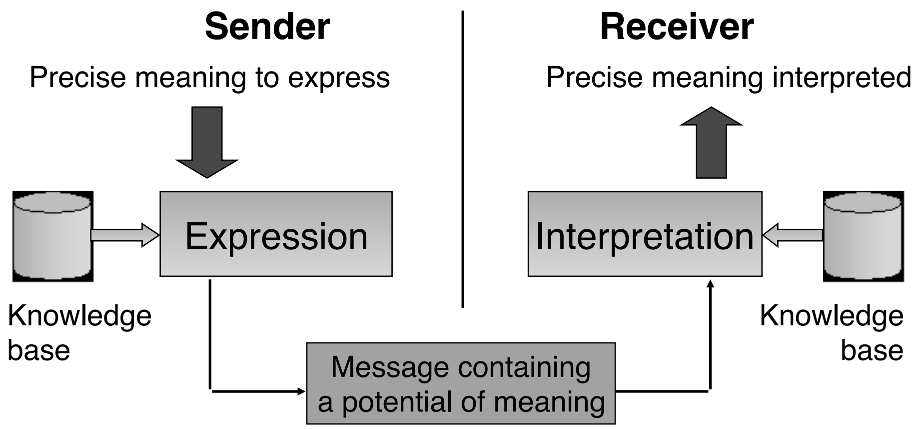

**The three meanings**

- the one the sender wants to give to its message
- the one the receiver interprets from the message
- the one the sender thinks the receiver has interpreted

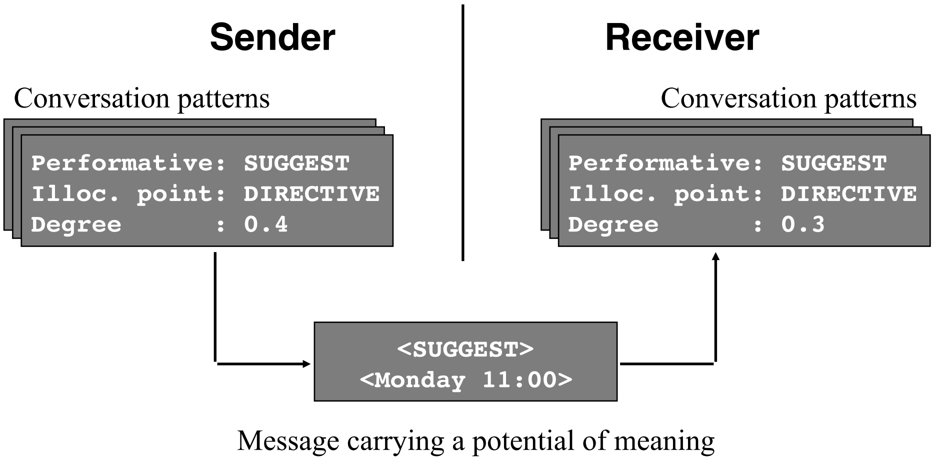
**are checked during the conversation to maintain mutual understanding in spite of possible shifts**



## Implementation model of the potential of meaning

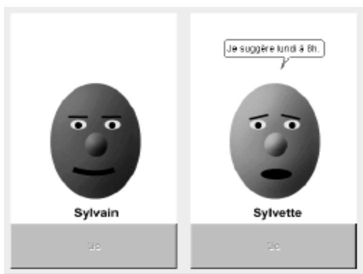


## Conversation patterns & sign / meaning matching






## Example of dialog synthesis




```
                               [SUGGEST] [Monday 09:00]
[SUSTAIN] [No]                 [ASK] [Monday 08:00]
[ASK] [Monday 14:00]           [HYPOTHESISE] [Monday 10:00]
[SUGGEST] [Monday 15:00]       [INSIST] [No]
[ASSERT]  [Monday 13:00]       [ASSERT] [Yes]
```