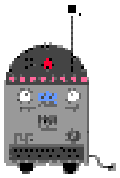


MultiAgentSystems

context: "Although there are situations where an agent can operate usefully by itself, the increasing interconnection and networking is making such situations rare"(Huhns). The idea is then to use the metaphor of agents living and *interacting* in a society to solve distributed AI problems.

motivation:

- when centralized approach fails
- decomposition of a complex problem
- when different expertise and functionalities are available
- concurrent resolution
- fault-tolerance: redundancy
- dynamic systems, open systems
- physically distributed systems
- interoperability
- the whole is bigger than the sum of its parts
- ...

However, simply decomposing a problem into simpler *modules*, which is good engineering practice, cannot always be considered a multi-agent approach. Agents are not just modules. Also, keep in mind that communication in distributed systems is costly.

The nature and type of *interaction* between agents can depend on many different factors:

- the situation: compatibility of goals, scarcity of resources, competence. Depending on those factors, interaction can range from indifference to cooperation, and even competition.
- the nature of agents: ReactiveAgents vs CognitiveAgents. This also determines the complexity of the AgentCommunicationLanguage.

A MultiAgentSystems designer should therefore analyze the above factors in order to identify the type of situations that are either likely to emerge or desirable:

- AgentCoordination
- AgentNegotiation
- AgentCooperation
- AgentOrganization

The designer will then provide adequate InteractionProtocols to deal with such situations.

back to SoftwareAgentCourse

(last edited March 16, 2004)
Find Page by browsing or searching

ReactiveAgents

definition (vague) Reactive agents are agents that do not contain any symbolic knowledge representation (ie: no state, no representation of the environment, no representation of the other agents, ...). Their behavior is simply defined by a set of perception-action rules.

examples

- Brooks (cf Multiagent Systems book by G. Weiss, in SoftwareAgentBooks) defines situated agents in his subsumption architecture. Perception-action rules are ordered into subsumption layers, where the lower layers take precedence. Here is an example of rock sample collecting robots, described by Steels, inspired by ants, and taken from the book:

```
if detect an obstacle then change direction
if carrying samples and at the base then drop samples
if carrying samples and not at the base then travel up gradient
if detect a sample then pick sample up
if true then move randomly
```

which can be augmented with the following indirect communicative behavior:

```
        if carrying samples and not at the base then drop 2 (radioactive) crumbs and
travel up gradient
        if sense crumbs then pick up 1 crumb and travel down gradient
```

- Ferber describes "eco-resolution", inspired by ethology. Collective behavior *emerges* from simple interactions between agents, which individual behavior is dictated by instinctive self-preservation (need for resources, fear of threat, ...). See [1] for an example.
- Artificial life (Droghoul)

forces

- simplicity
- elegance
- fault-tolerance
- inherent lack of methodology

applications

- map rearrangement (Baeijs and Demazeau)
- "hot-potato" routing (it actually pre-existed research on reactive agents)
- network management (White)

CognitiveAgents

definition CognitiveAgents are agents with an explicit knowledge representation of own capability, other agents, the environment...

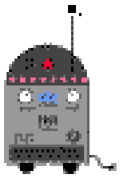
description here is an example of the behavior cycle of a cognitive agent (Boissier):

```
var:
 s: state
 eq: event queue


s:= initialise();
repeat
       options := option_generator(eq,s);
       selected := deliberate(options, s);
       s := update_state(selected, s);
       execute(s);
       eq := get_new_events();
forever
```

Usually CognitiveAgents use the mentalistic notions of BeliefDesireIntention.

(last edited November 2, 1999)
Find Page by browsing or searching



AgentCoordination

definition: "management of interdependencies between activities" [Malone 94]

context: "In an environment with limited resources, agents must coordinate their activities with each other to further their own interests or satisfy group goals. The actions of multiple agents need to be coordinated because they are dependencies between agents' actions, there is a need to meet global constraints, and no one agent has sufficient competence, resources or information to achieve system goals." (Huhns & Stephens, in Multiagent Systems book)

Therefore the key aspects are:

- the activities of the agents should solve **all** the components of the global problem
- the interactions between agents should be **coherent** and integrate in the global solution

given available **resources** (time, memory, ...).

Good coordination should:

- provide timely information and help to the agents
- avoid redundancies between agents actions
- make sure that the agents optimize their use of resources for their tasks instead of communication and coordination.

elements of methodology:

1. Task allocation: who should do what, how... Tasks could be allocated *a priori* using predetermined AgentOrganization, or dynamically using ContractNet.
2. Planning: how to find the right sequencing of subgoals and subtasks allocated to the different agents in order to achieve the global goal? Plans could be determined in a centralized way (using for instance STRIPS or NOAH [Sacedotti]) and then sent to the various agents [Corkill 79]. Or plans can be calculated locally, and the agents would then send their partial plans to each other [Durfee 87]
3. AgentNegotiation

Each step can be dealt with in a centralized or in a distributed way. The centralized way assures easy coherence maintenance, whereas the distributed way allows better scalability, dynamicity and fault-tolerance.

protocols:

- ContractNet for task allocation
- AgentNegotiation protocols

(last edited November 2, 2000)
Find Page by browsing or searching

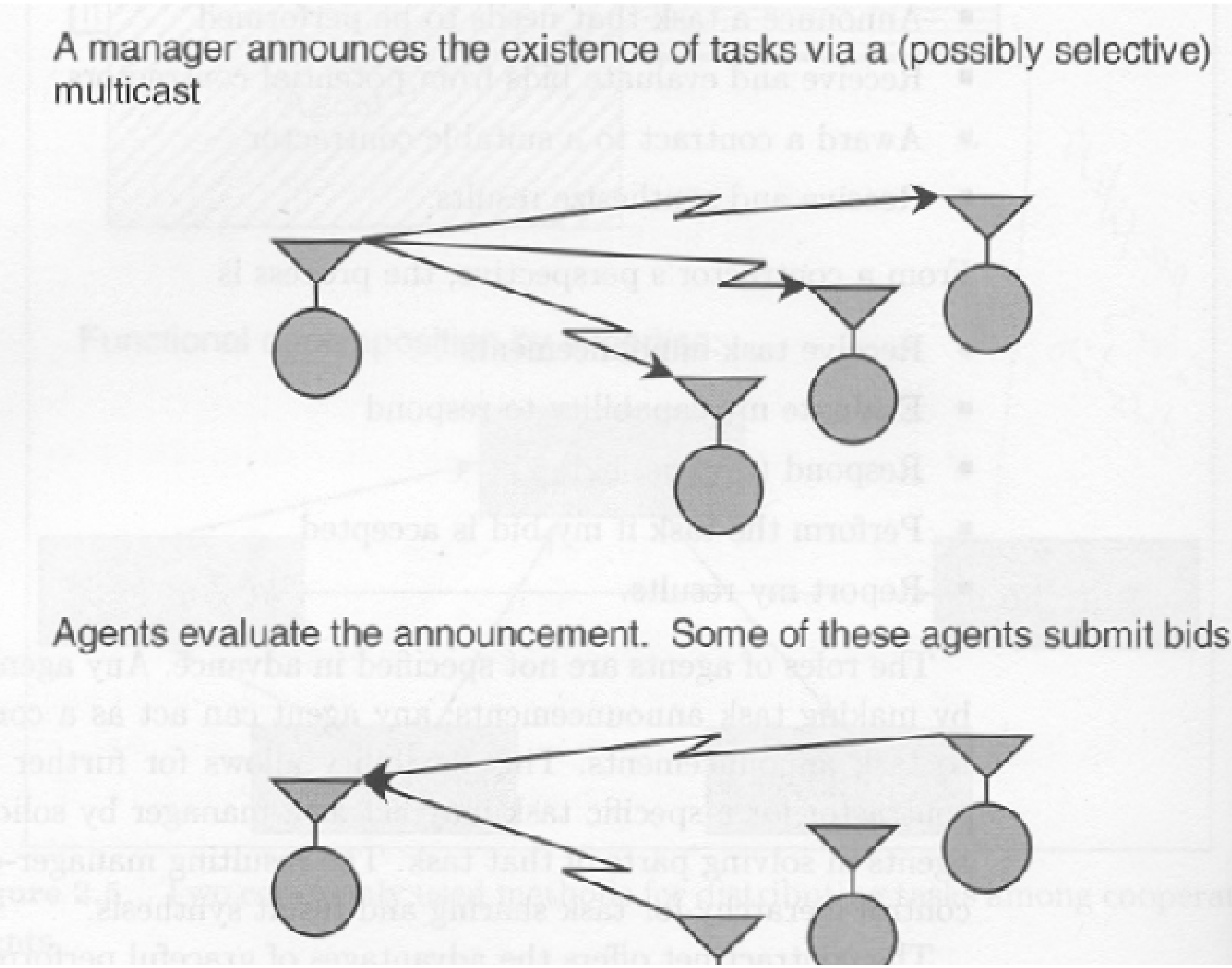
ContractNet

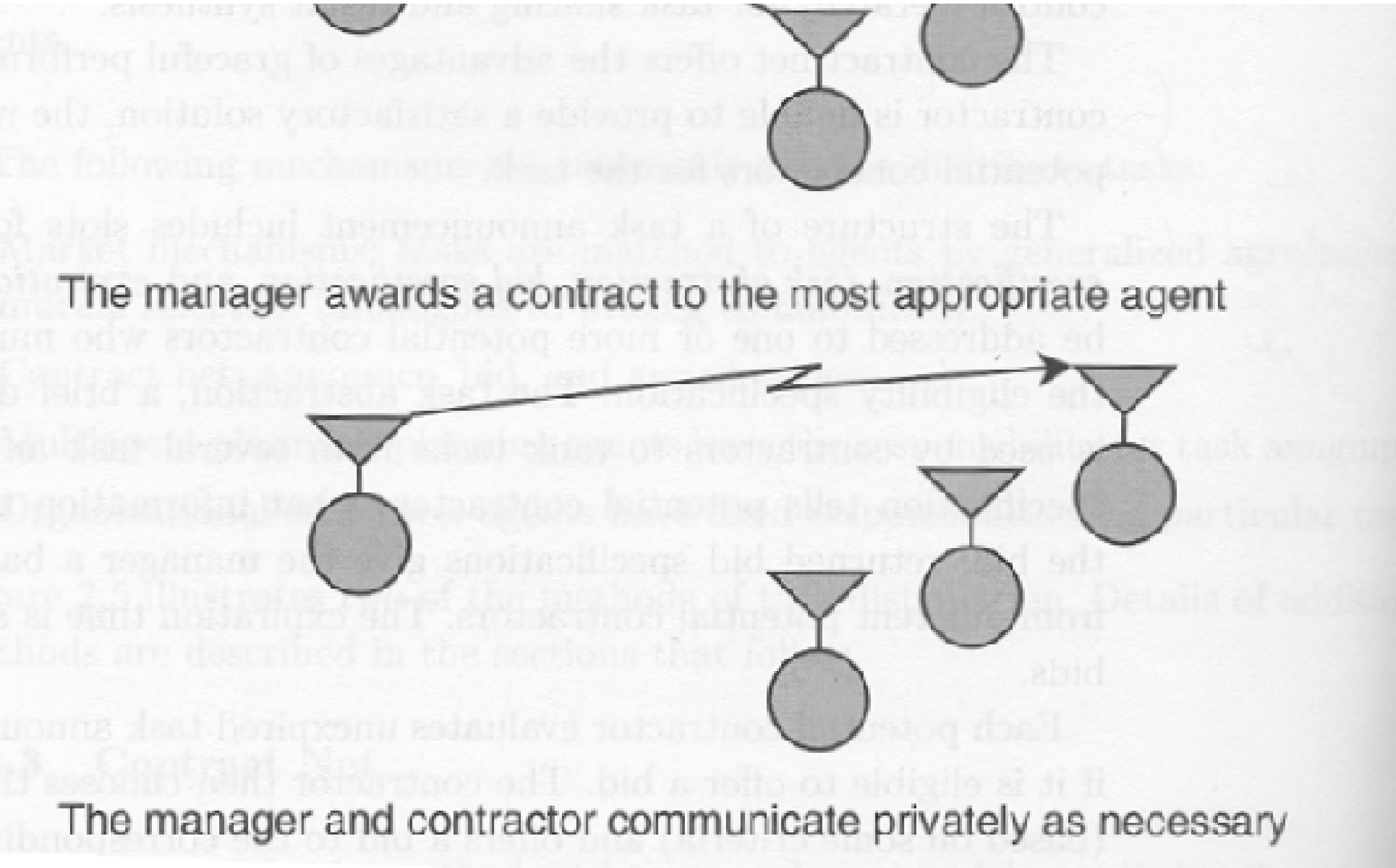
intent:

- for task allocation between different "sub-agents"
- to establish contracts and enforce commitments
- for bazars

see also AgentOrganization, AgentCoordination

principle:

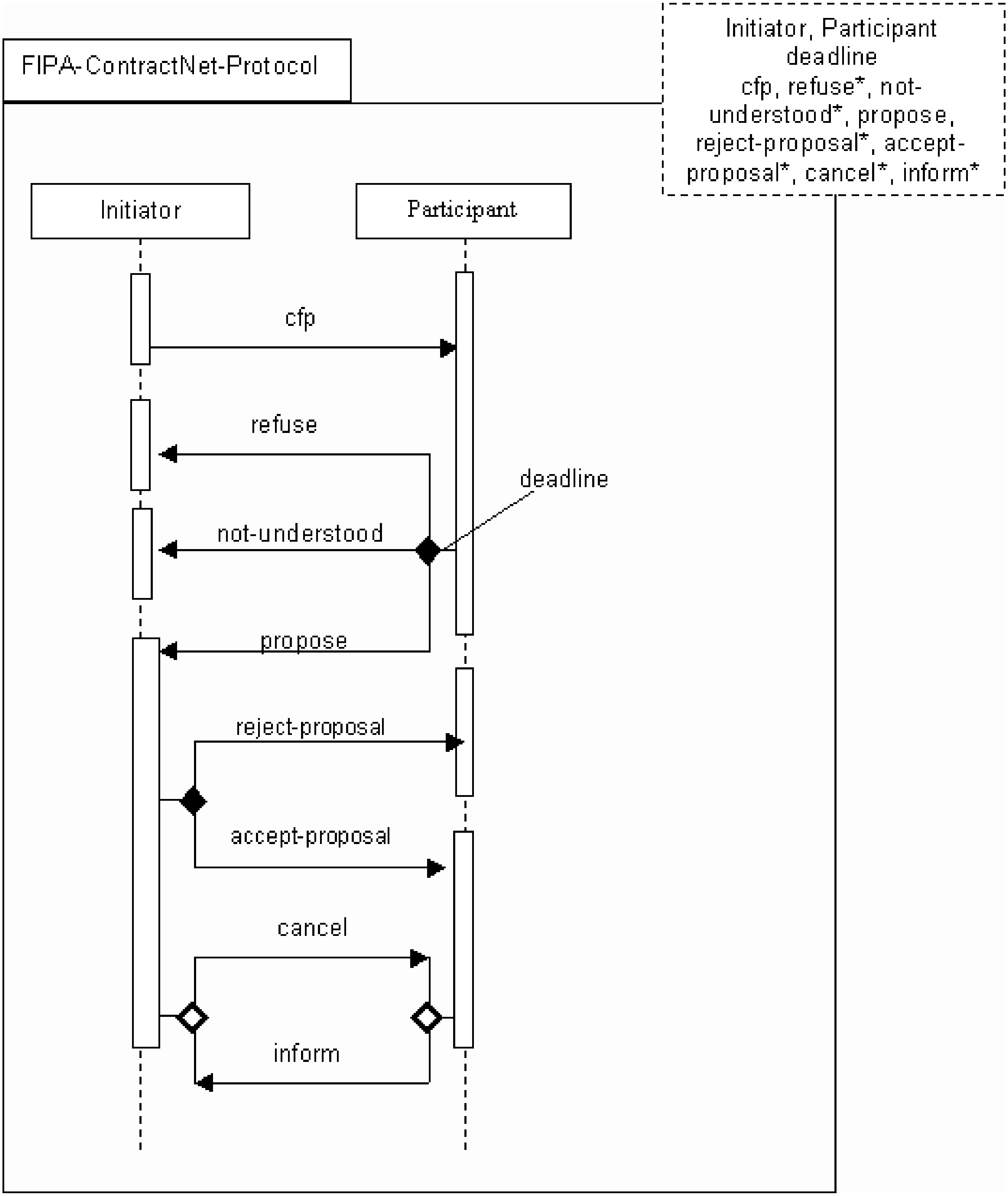




forces:

- dynamic: contracts established on the fly, new partipants may join
- degrades gracefully

protocol:



relevant aspects:

- choice of [AuctionProtocols]
- contract enforcement

- performance monitoring
- violation detection

more:

- use of caching to decrease the number of exchanged messages
- use of machine learning to discount bids based on quality of service [mitel patent]
- combining this with mobile agents to minimize message exchange?

known uses:

- resource allocation (Mitel)

(last edited November 9, 2004)
Find Page by browsing or searching

AgentNegotiation

context:

Negotiation arises from the need for agents to reach *mutually beneficial agreements*.

Conflict resolution:

- conflict of access to resources
- different solutions to the same problem
- conflicts of interest or goal conflicts

parameters: (Sandholdm)

- guaranteed success
- maximizing social welfare
- pareto efficiency
- individual rationality
- stability (see NashEquilibrium)
- simplicity
- distribution

techniques:

- use of force, authority...
- use of a mediator agent, allowing some centralized and "impartial" way of conflict resolution
- negotiation

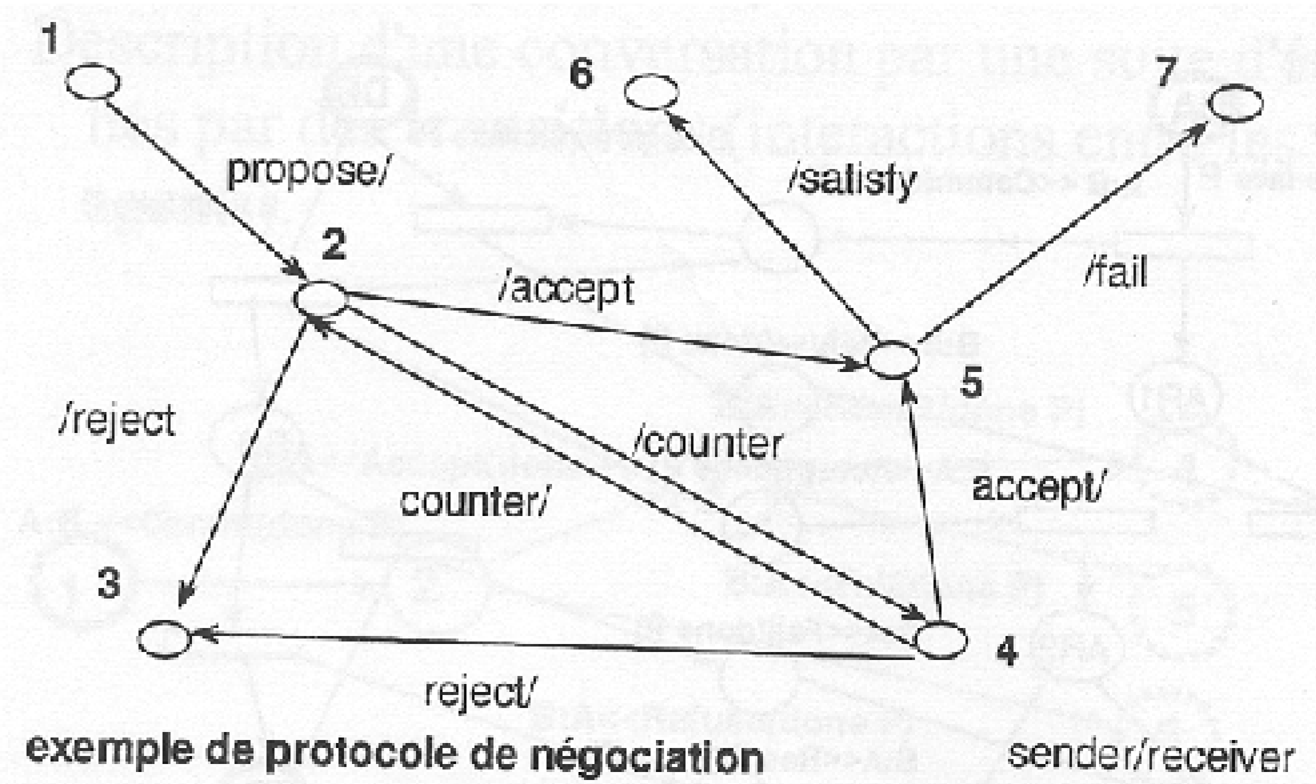
Negotiation can be used for:

- task allocation
- conflict resolution
- definition of organizational structures
- maintenance of the coherence of a system

protocols:

AuctionProtocols offer nice properties with respect to the parameters listed above. They can be used for task or good allocation purposes.

For more general needs to reach agreements, more complex protocols should be devised, such as:



(last edited March 16, 2004)
Find Page by browsing or searching

NashEquilibrium

(Wooldridge)

Two strategies s1 and s2 are in Nash Equilibrium if:

1. under the assumption that agent i plays s1, agent j can do no better than play s2
2. under the assumption that agent j plays s2, agent i can do no better than play s1

There is a NashEquilibrium for the prisonner's dilemma which consist in both agents defecting.

Other scenarios may contain 0 or more than one nash equilibrium.

(last edited March 16, 2004)
Find Page by browsing or searching

AgentCooperation

To cooperate or not to cooperate? Insight can be gained from GameTheory:

- Prisoner's dilemma [1][2][3]
- Tragedy of the commons [4]

Bouron claims in his Ph.D. thesis that Cooperation consists of communication and organization. Refer to AgentCommunicationLanguage and AgentOrganization for more on those topics.

(last edited March 16, 2004)
Find Page by browsing or searching

GameTheory

disclaimer: This is a very shallow introduction to the topic. For more info, check out Sandholm's chapter in Gerhard Weiss' Multiagent Systems.

general idea: the goals is for a rational agent to choose actions that maximize its utility function. Such a function evaluates an action based on the value of its outcome.

When two agents simultaneously choose an action, the outcome will depend on the combination of both.

The essence of game theory is deciding which action (aka strategy) to choose given the outcomes available and by reasoning on the action that will be chosen by the other agent.

One interesting property is that of NashEquilibrium.

(last edited March 16, 2004)
Find Page by browsing or searching

The Prisoner's Dilemma

The **prisoner's dilemma** is the story of two criminals who have been arrested for a heinous crime and are being interrogated separately. Each knows that if neither of them talks, the case against them is weak and they will be convicted and punished for lesser charges. If this happens, each will get one year in prison. If both confess, each will get 20 years in prison. If only one confesses and testifies against the other, the one who did not cooperate with the police will get a life sentence and the one who did cooperate will get parole. The table below illustrates the structure of payoffs.

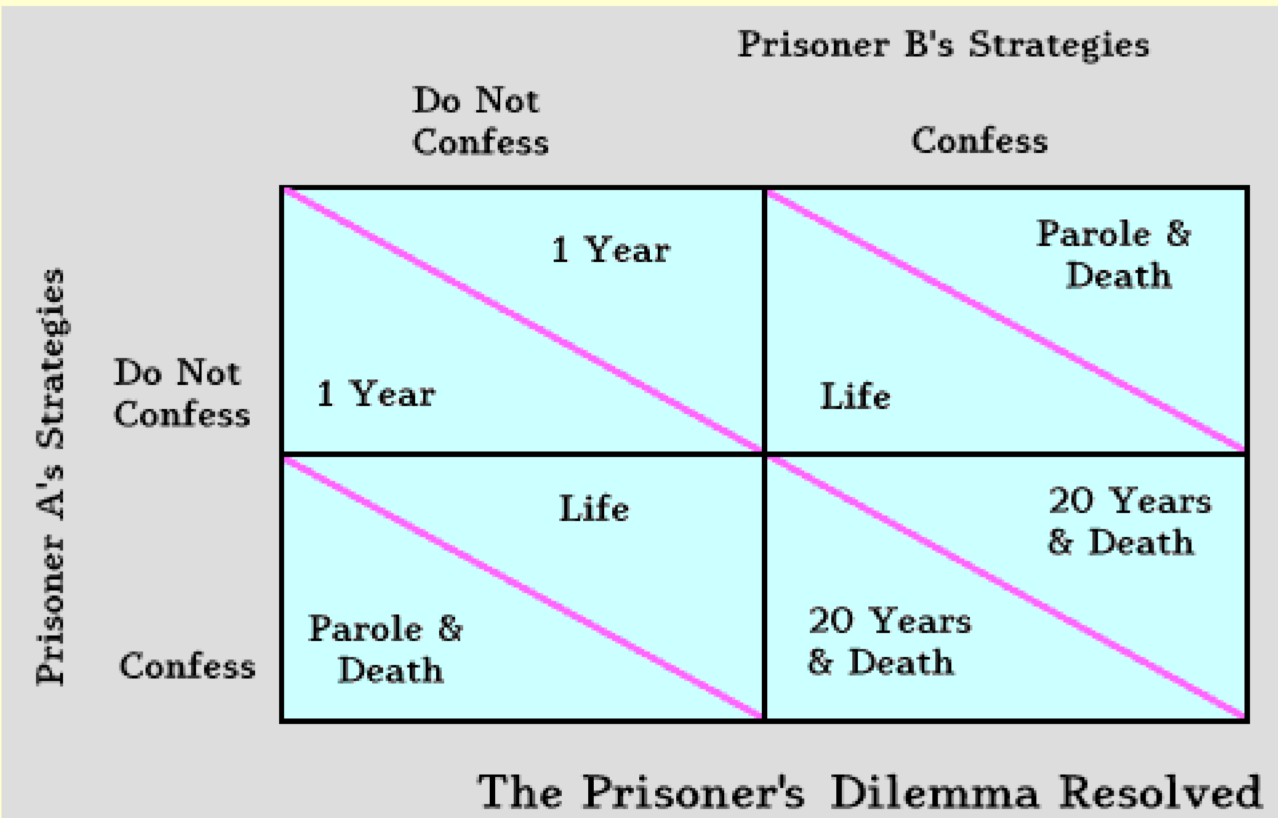
Prisoner A's Strategies	Prisoner B's Strategies: Do Not Confess	Prisoner B's Strategies: Confess
Do Not Confess	1 Year 1 Year	Parole Life
Confess	Life Parole	20 Years 20 Years

Given this set of payoffs, there is a strong tendency for each to confess, which you can be see by considering the choices and payoffs of either one. If prisoner **A** remains silent, prisoner **B** is better off confessing (because parole is better than a year in jail). However, **B** is also better off confessing if **A** confesses (because 20 years is better than life). Hence, **B** will tend to confess, regardless of what **A** will do; and by an identical argument, **A** will also tend to confess.

The prisoner's dilemma is a case in which actions determined by self-interest are not in the group's interest (where the group is defined to include only the criminals, not the larger society). It is a story that would have pleased Thomas Hobbes, a political theorist of the mid-17th century. Hobbes is a grandparent of economics because he introduced into intellectual discussion two assumptions: first, that the individual is the starting point of social analysis, and, second, that people are motivated by self-interest. He believed that the unrestrained pursuit of self-interest would result in chaos. The development of government, with its power to coerce people, was necessary to bring order out of chaos.

More Prisoner's Dilemma

The story of the prisoner's dilemma can be retold with a slight alteration. Assume that our two criminals belong to an organization of criminals that will kill any member who confesses to the police or testifies against other members of the organization. With this change, the outcome of the story will tend to be totally different: both will remain silent. Self-interest and group interest are identical. This happens not because the motives or goals of the criminals have changed, but because incentives or payoffs have changed. The table below shows the new payoffs. A paradox of this case is that imposition of this "death penalty" can actually improve the well-being of individuals making up the group. One would not normally think that making possible outcomes worse could improve results.



For Hobbes, the introduction of government could change payoffs so that cooperative behavior would replace the "war of all against all." But Adam Smith better illustrates the results of this second story. His *Wealth of Nations* argues that the well-being of the group can be consistent with the pursuit of self-interest if the right incentives, which Smith said were free trade and competition, existed. Nowhere does he explain his point better than when he talks about the butcher, baker, and brewer who provide quality products because they love themselves, not because they love humanity.

If the individuals in a group are motivated by self-interest, the group cannot survive and prosper unless it finds ways to make individual self-interest consistent with group interest. This is a problem for the largest of organized groups, a nation, as well as for many of the subgroups within it. There are two basic ways to deal with this problem. One is the Hobbesian way of regulation and planning from the top. The other is the Smithian way of private-property rights and the market.

Though we could jump to consider these two ways of coordination, it might be better to pause to defend the assumption of self-interest used in the prisoner's dilemma.

Overview ¿Review? Next >>

The Commons

Suppose that the table below shows the relationship between the number of cows using a large meadow and the amount of milk that can be produced.[1] If one person controlled access to this land, it would be foolish for him to allow more than 400 cows to graze there. But when the land is open to all, 500 cows may end up grazing if each of the many herders tries to do what is best for himself. Thus, if there are 100 herders, each with four cows, each cow will produce seven gallons of milk. If one herder adds an extra cow, the total production of milk does not increase, but the production from his herd will. He will have five cows, and each produces 6.98 gallons (after ignoring all digits beyond the second to the right of the decimal point). The fifth cow adds 6.98 gallons of milk. However, each of the four cows he already has will produce .02 gallons less, so the net gain to the herder will be 6.9 gallons. He will not take into consideration the drop in production of .02 gallons from each of the 396 cows that he does not own. Because each herder has the same set of incentives, it may easily happen that there will be 500 cows grazing on the land. Clearly there is an unintended consequence here, and it is of the by-product type. Each herder ignores the effects that his cows have on the productivity of cows he does not own.

Production on the Commons	
Number of Cows	Output of Milk (Gallons)
100	1000
200	2000
300	2600
399	2799
400	2800
401	2800
500	2700
600	2600

The story above is an example of the **problem of the commons**. The commons exist whenever there is a resource--such as grazing land, a hunting area, or fishing grounds--which is open to all. The problem of the commons refers to the absence of any automatic mechanism or incentive to prevent the overuse and depletion of the commonly held resource. To see why this overuse can happen, consider a common grazing area.

The logic of the problem of the commons is behind many ecological problems, from deforestation to depletion of the ozone layer. In a sense, the core of the problem is that behavior is unconstrained in any way. There are two basic methods that restrictions can be placed on people's actions in order to give better results.

AgentOrganization

definitions: (sociology)

"an arrangement of relationships between individuals or components. The resulting system is empowered with qualities unknown to its components." (translated from [Morin 77]).

"division of tasks, distribution of roles, authority systems, communication channels and contribution-awards." (translated from [Bernoux 85])

Therefore, the key elements are:

- tasks
- roles
- authorities
- commitments (see BeliefDesireIntention), contracts...

types:

a priori organization vs self-organisation

- centralized
- hierarchical
- bazar/market: self-organization based on the laws of the market and the use of currencies

elements of methodology:

- in *a priori* organization: specification of a graph of dependences and influences [Collinot 96], allocation of roles based on tasks, definition of relationships (authorities, communication channels), multi-level hierarchies in case of possible bottleneck
- in self-organization: use of ContractNet and auctions for contract allocation, use of mechanisms for contract enforcement (performance monitoring, violation detection and recovery)

protocols:

- FacilitatorsAndBrokers, for recursive group/agent view of multi-level hierarchies. ex:

Jurisdiction (cf Mitel): brokering for functionally organized agents

- AuctionProtocols: for bazars
- ContractNet:
  - for task allocation between different "sub-agents"
  - to establish contracts and enforce commitments
  - for bazars

(last edited November 5, 2002)
Find Page by browsing or searching

FacilitatorsAndBrokers

definition: a facilitator is an agent that performs various useful communication services, such as:

- maintaining a registry of service names
- forwarding messages to named services
- routing messages based on content
- providing matchmaking between information providers and clients
- providing mediation and translation services

principle

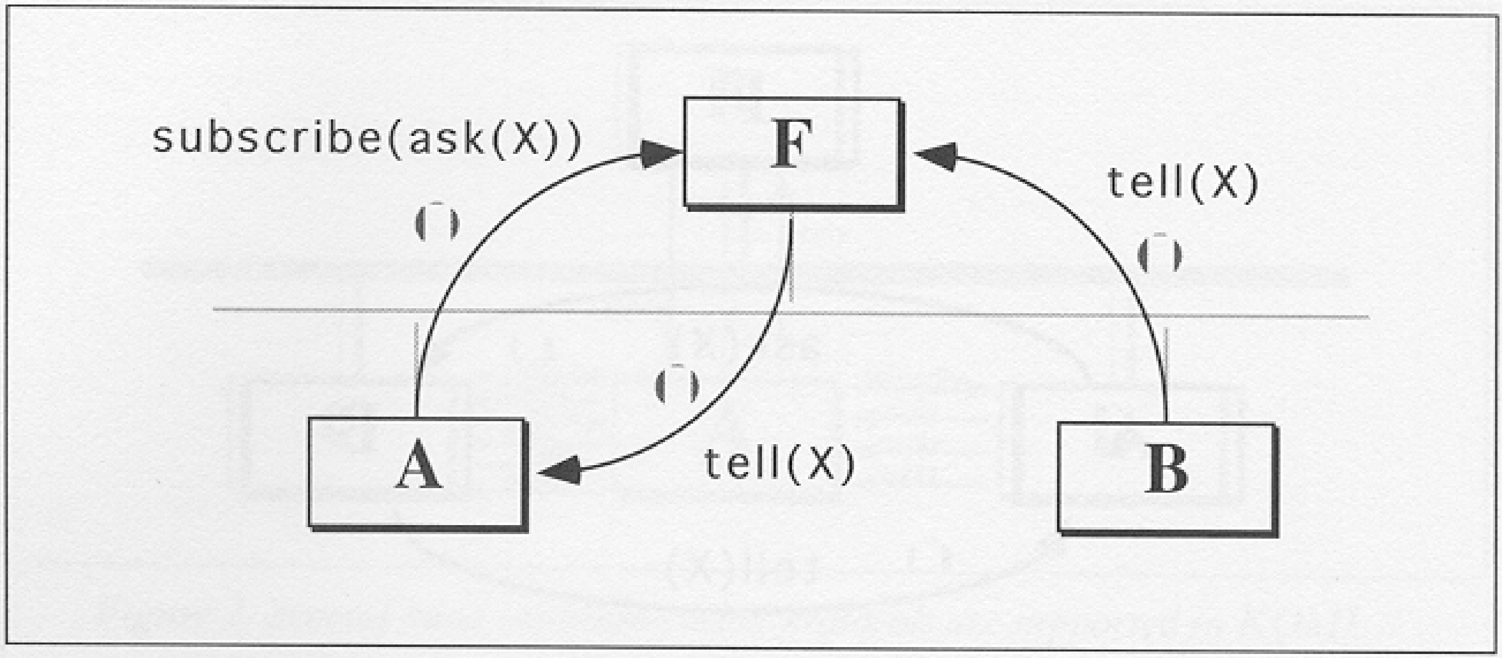
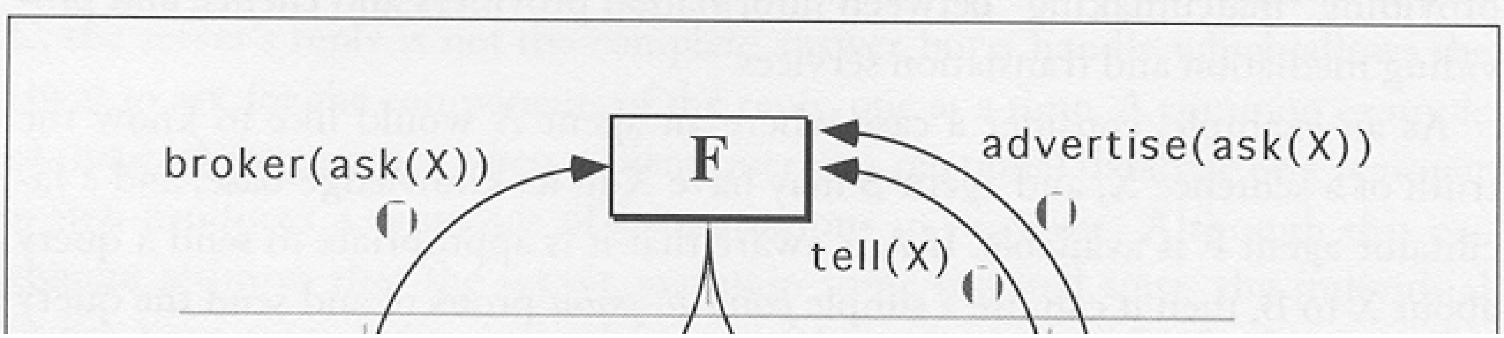


Figure 4. Agent A can ask facilitator agent F to monitor for changes in its knowledge-base. Facilitators are agents that deal in knowledge about the information services and requirements of other agents and offer such services as forwarding, brokering, recruiting, and content-based routing.



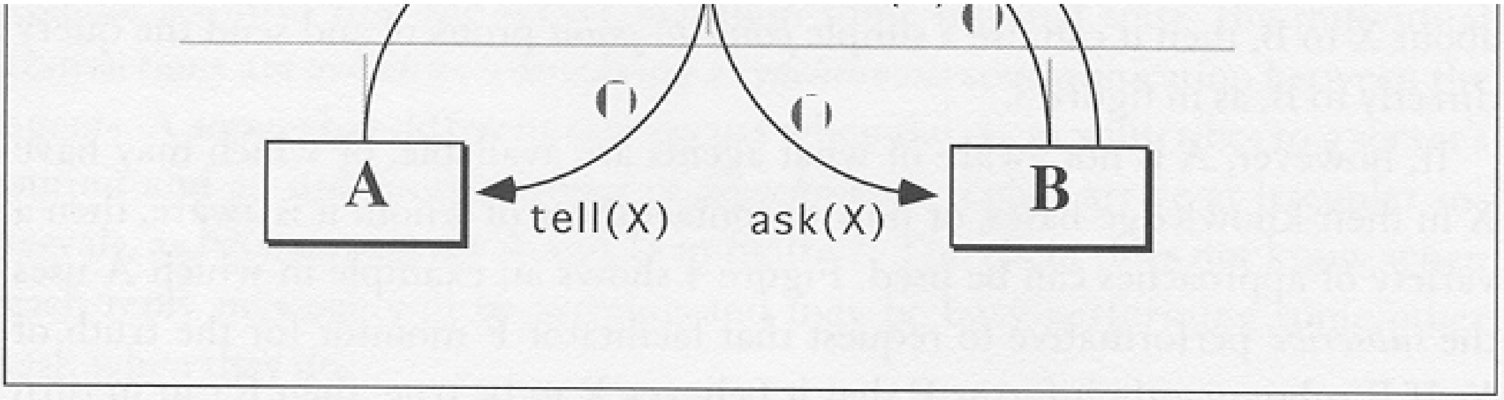


Figure 5. The broker performative is used to ask a facilitator agent to find another agent which can process a given performative and to receive and forward the reply.

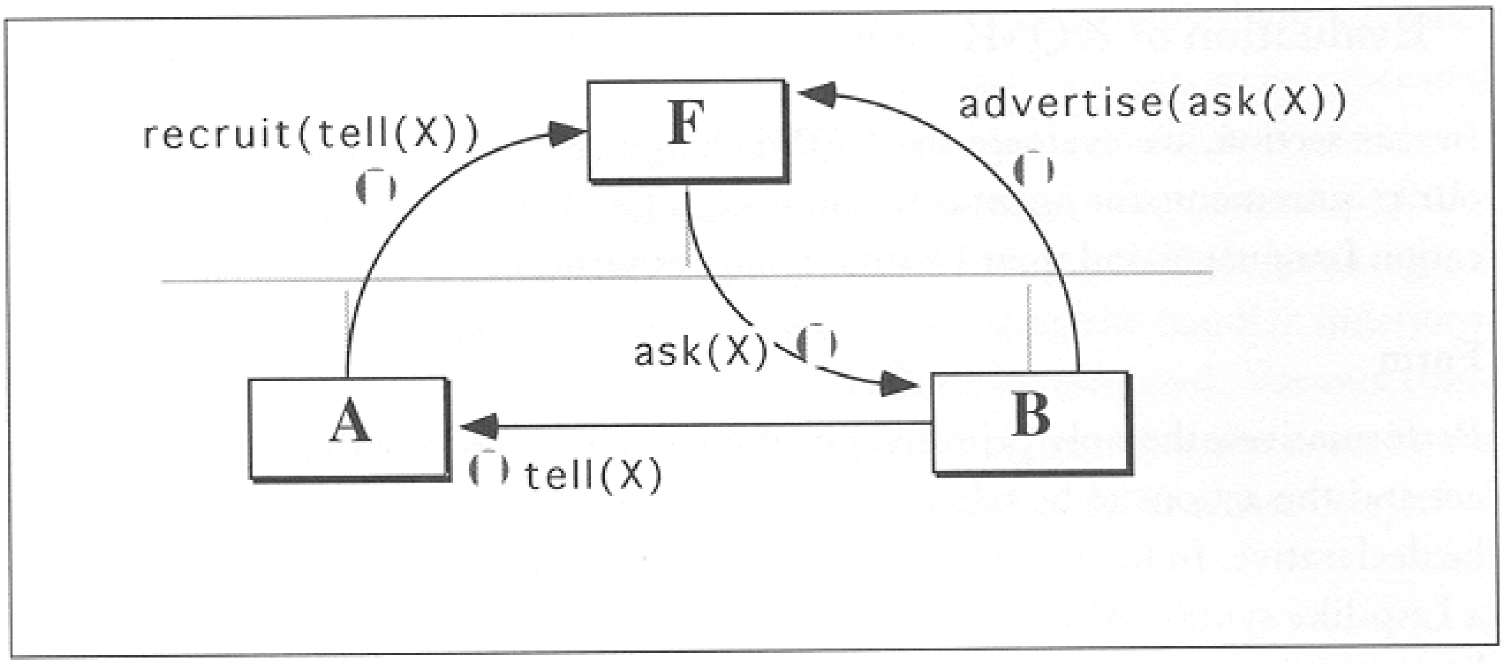
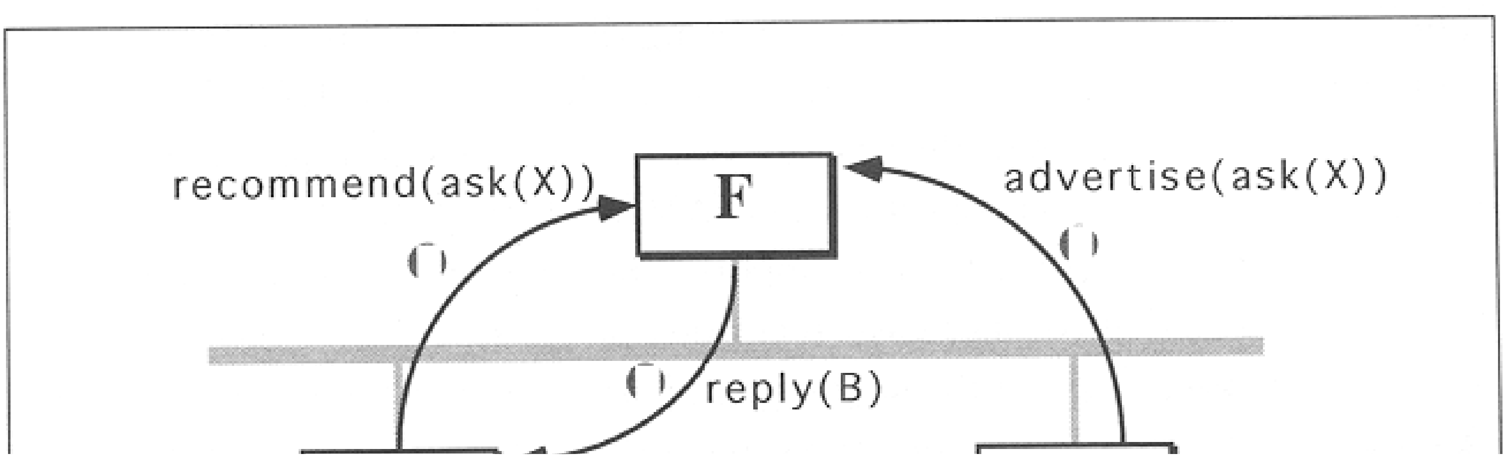


Figure 6. The recruit performative is used to ask a facilitator agent to find an appropriate agent to which an embedded performative can be forwarded. Any reply is returned directly to the original agent.



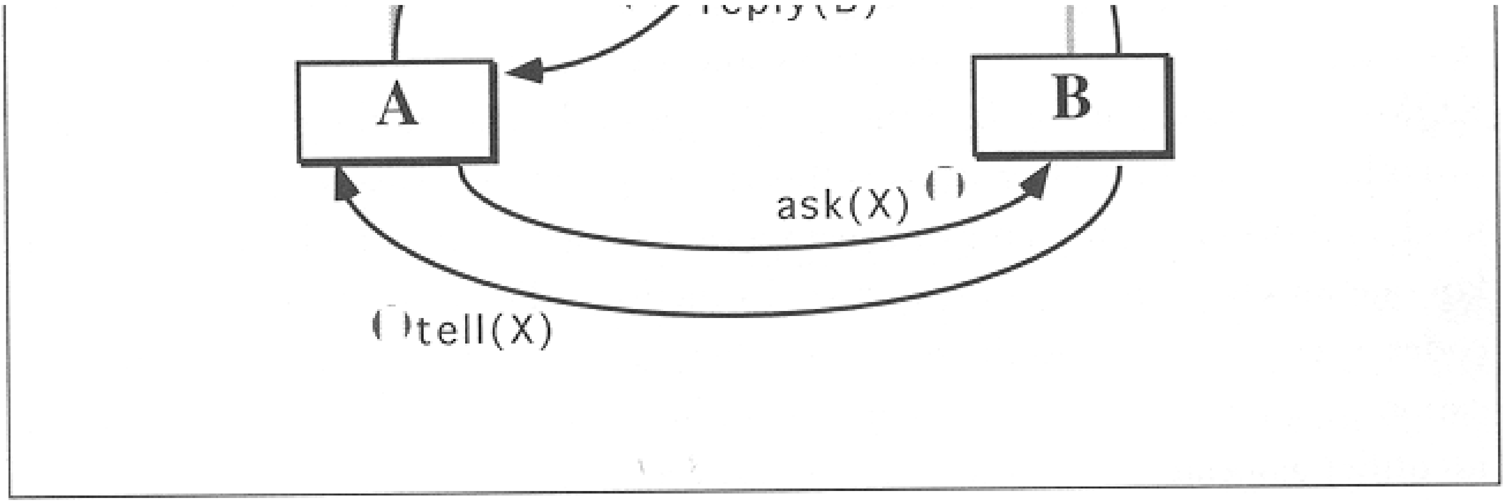
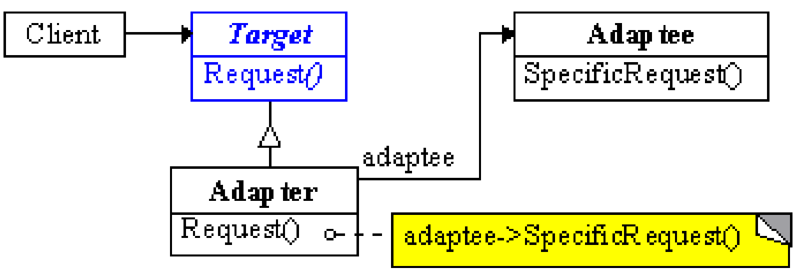


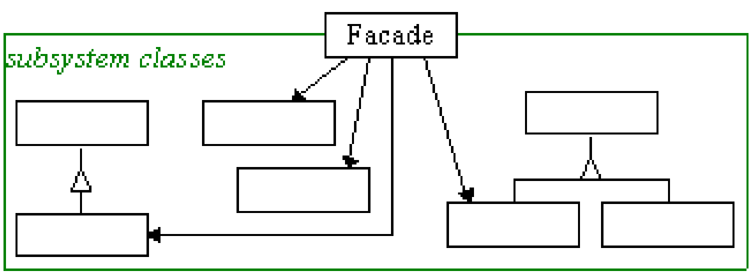
Figure 7. The recommend performative is used to ask a facilitator agent to respond with the "name" of another agent which is appropriate for sending a particular performative.

related patterns:

- adapter



- facade



forces:

- this is a centralized approach: what if the broker goes down? possibilities of distributed brokerage, mirroring...

known uses:

- CORBA, RMI for name services
- DNS

AuctionProtocols

The designer of an auction should determine parameters such as:

- whether the good to be auctioned off has a public or private value;
- how to determine the winner and how much should the winner pay
- whether to make the bids public or not
- the bidding mechanism: one shot, ascending, descending

By combining the above parameters, we can obtain many different auction protocols, including notably:

- English Auctions
- Dutch Auctions
- first-price, sealed-bid auctions
- Vickrey Auctions

The choice of the auction will be based on the auctioneers interest, the need to avoid collusions, as well as practical aspects.

For a short and gentle introduction to Auction Protocols, check out Wooldridge's book "Introduction to Multiagent Systems".

(last edited March 16, 2004)
Find Page by browsing or searching

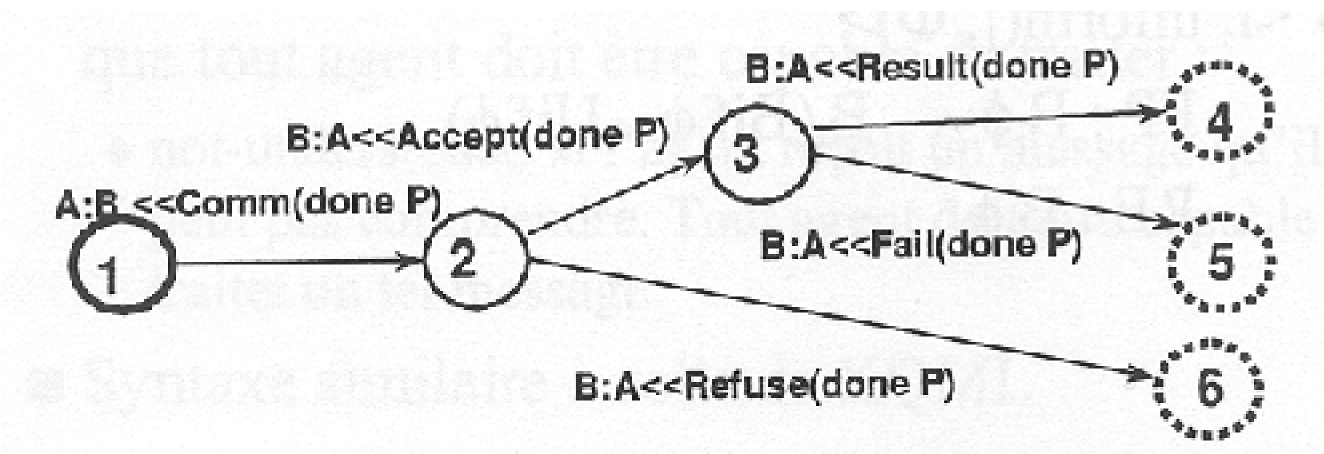
InteractionProtocols

definition: "Interaction is the dynamic establishment of a relationship between two or more agents through a set of reciprocal actions" (translated from Boissier). You can tell there is interaction when the behavior of an agent is modified/disturbed by the influence of other agents.

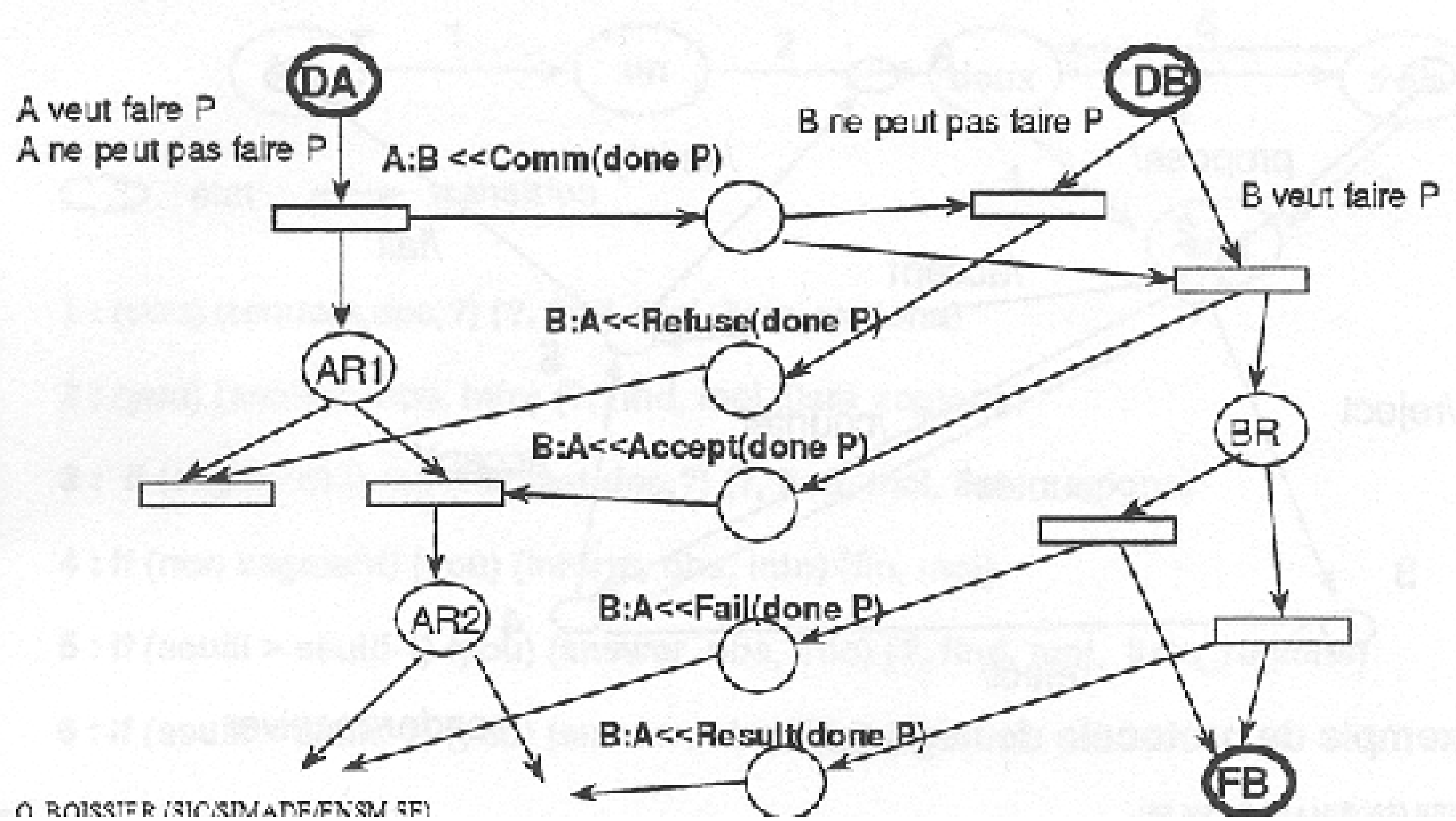
"An Interaction Protocol is a common scheme of conversation used to execute a task. It is a high-level strategy that controls the interactions between agents to facilitate their dialog." (still Boissier)

representation:

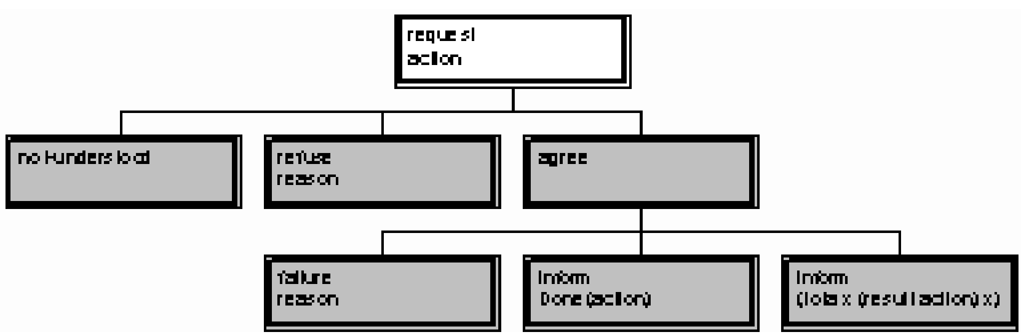
- FiniteStateMachines

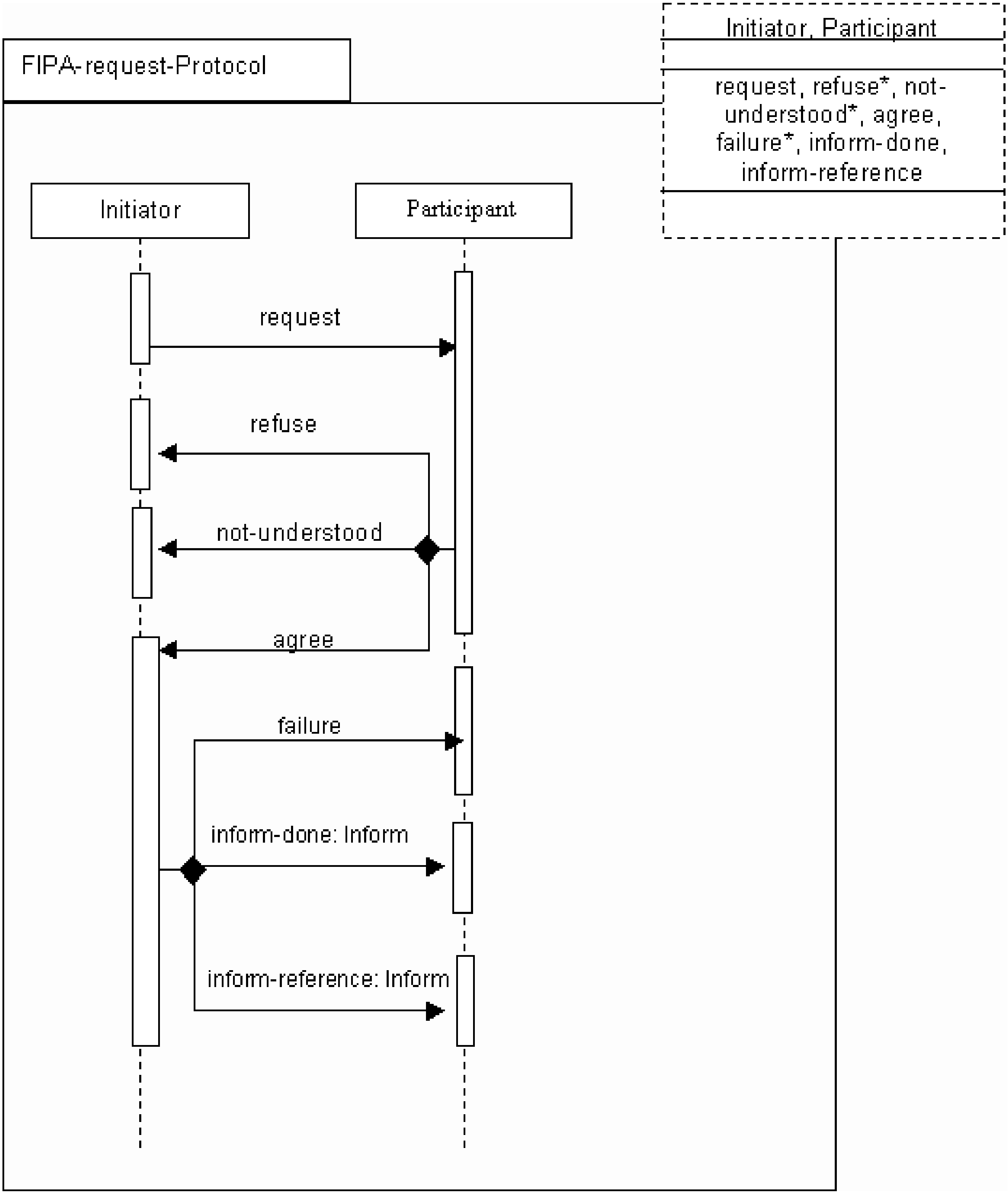


- PetriNets



- Message Sequence Charts (MSC), Sequence diagrams, UAML (FIPA extension to UML)[3]





protocols:

- query (Demazeau)
- SubscriptionProtocol (Java event model, Observer pattern, SNMP traps, CMIP event reports)

- tour de table
- FacilitatorsAndBrokers (KQML, Jini, Adapter pattern, Facade pattern, Bargain finder)
- blackboard (Hayes-Roth)
- ContractNet (Smiths, FIPA)
- AuctionProtocols: ascending (English), descending (Dutch)
- voting

refs

- Chapter of a FIPA document which deals with InteractionProtocols [6]
- tutorial by Huget and Koning [7]

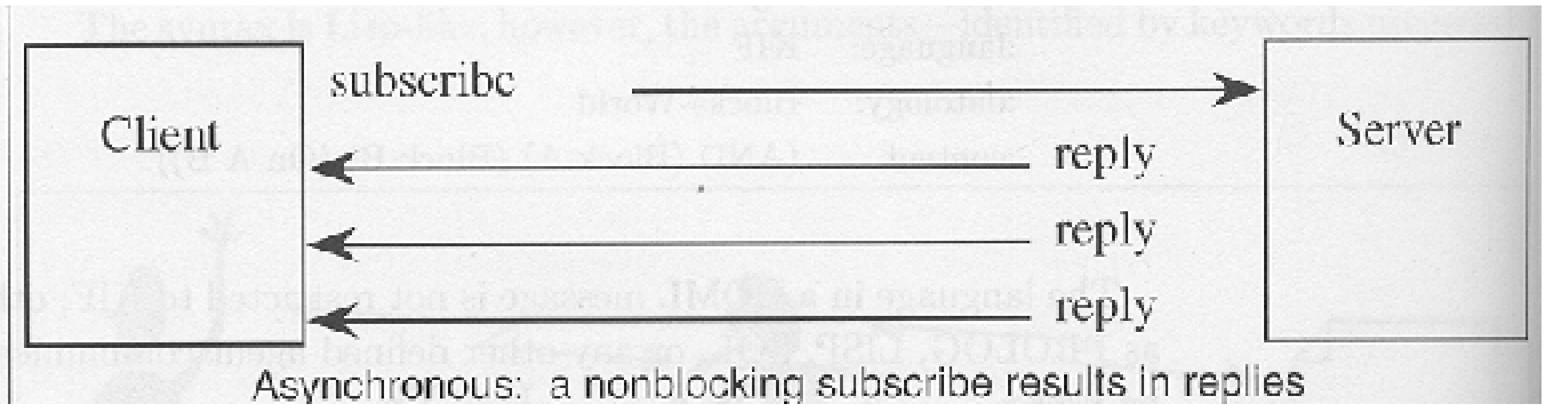
(last edited May 12, 2004)
Find Page by browsing or searching

SubscriptionProtocol

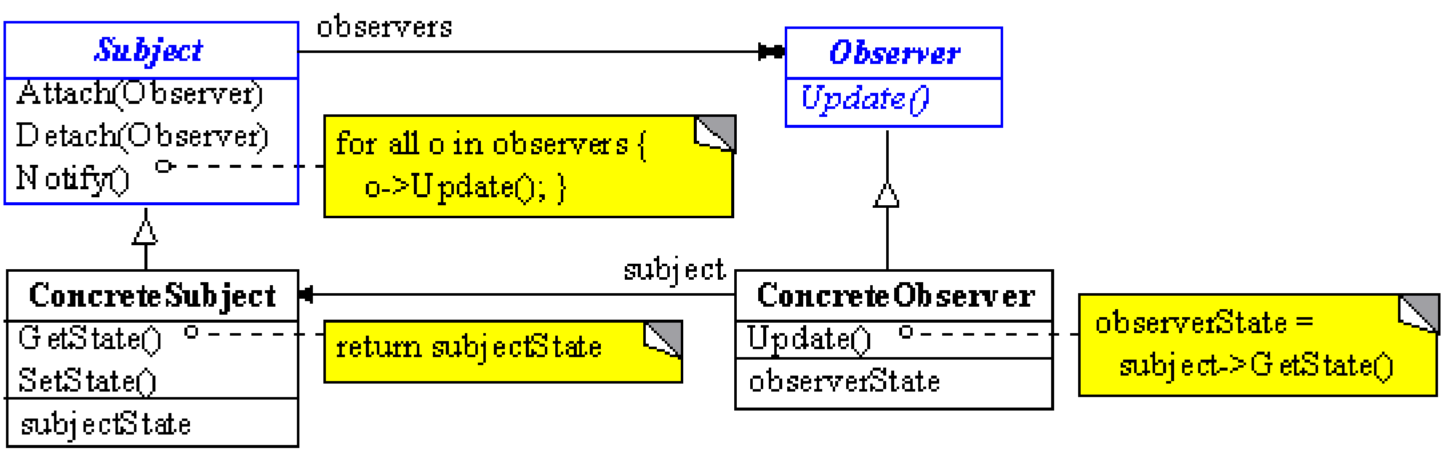
intent:

- register to be notified of changes

principle:



It is some kind of distributed version of the Observer pattern:



forces:

- might generate too much traffic, and sometimes useless notifications: need for *filtering*

known uses: