

# MAS -- Exercise: Contract-Net Protocol

The purpose of this exercise is to learn about protocols for agent communication and distributed problem solving methods. The exercise is to develop a distributed application that makes use of the *Contract-Net Protocol* [Smith, 1980, Smith & Davis, 1983]. Carefully study Wooldridge 2009, pp 156 -- 158 and the original papers. The FIPA version [2002] is detailed, but uses a different terminology.

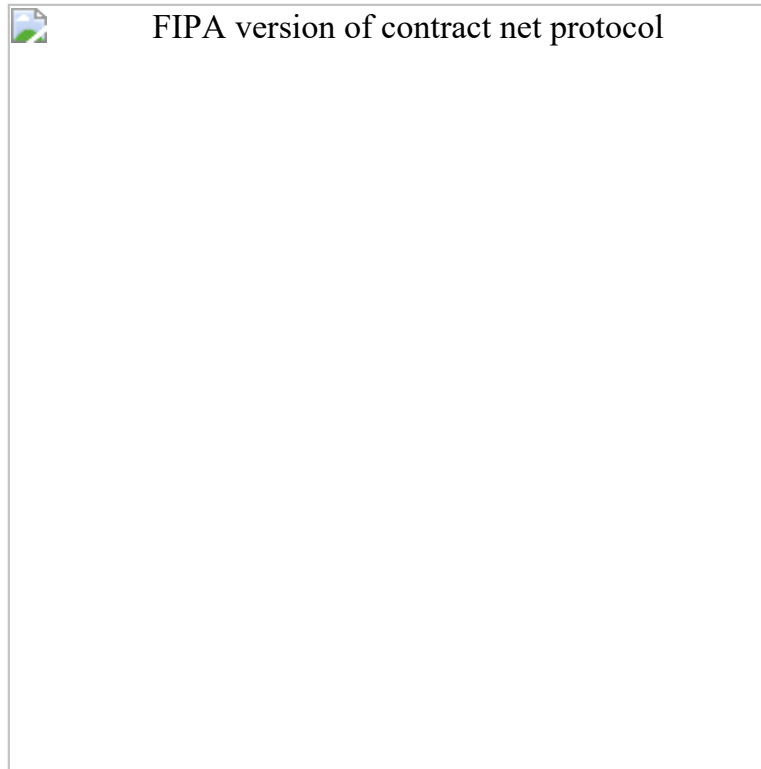


Figure 1: FIPA version of Contract Net Protocol

## Protocol

Agents (nodes in a network) use the protocol to make contracts about how they should allocate tasks. We distinguish a *manager* node and possible *contractor* nodes. The manager analyzes his problem, sends out a task announcement to all possible contractors on the network, receives bids from the contractors and after careful evaluation, awards the contract to one (or more) of the contractors.

For each bidding cycle, the communication involves the following message types:

- **Task announcement:** The manager announces a task by a broadcast to the network. A task announcement message includes abstract information about the task, expected capabilities of potential contractors, the information that a bid should contain, and a deadline for when bids should be received.
- **Bid:** Idle nodes receive and evaluate the announcement. Nodes with the appropriate resources, expertise, and information reply to the manager with bids that indicate their suitability to perform the task.
- **Award :** After a specific deadline has passed, the manager evaluates the bids and awards the task to the most suitable node or to several suitable nodes if redundancy is needed to ensure reliability. The other bidders are notified with some kind of cancel message.

- **Report messages:** after the contract has been awarded, the manager supplies task information and the contractor reports progress and the eventual result (completion or failure). This phase also contains requests for additional information, and subsequent informs.

## Application

In this exercise we will use the contract-net protocol for the following task and domain: *configuration of computer hardware*. Computers consist of components, such as a processor, a case, a motherboard and possibly a graphical processor. These components are typically made by different producers. A computer retail company like DELL needs to get the best deal for each component from different vendors. For this exercise, an environment .jar file is not needed, just implement the manager and contractor agents.

## Exercise

1. Give a detailed specification of one cycle of the contract-net protocol for computer hardware configuration. Specify the process, the message types and the possible content of messages. The content of messages depends on how computers and components are specified (see below for details). This can be written out and submitted in a simple .txt file.
2. Implement and test one bidding cycle of the contract-net protocol for a manager agent who needs a number of computers with specific specification, and 3 contractor agents, each with their own set of components at different prices.s

The manager agent can have a list of (needed) computers in its beliefbase. Each computer can be specified by means of a fact with the following structure.

**pc(number, case, motherboard, processor, graphics)**

where number is a natural number representing the number of needed computers, case, motherboard, processor and graphics are the name of the product (e.g., antex, ATI RADEON 7000, etc). You can use the following files for the three contractor agents:

[components1.pl](#)

[components2.pl](#)

[components3.pl](#)

Each contractor has a set of components at its disposal and can make bids by means of a set of components that the manager can use to assemble computers. Components can be specified by means a fact with the structure as indicated in the components prolog file. In the example of a PC case it is the following:

**pc\_case( Manufacturer, Type, Shape, ATX, Power (watt), Color, Suitable For, Price )**

You can avoid deadlines by requiring that each node always sends a response, either positively or negatively. The evaluation of bids by the manager agent should be such that the needed computers can be assembled with the lowest amount of money.

**Bonus:** Notice that in the structure of the PC components several other attributes like color and type are mentioned. For a bonus grade, expand your code so that the manager can ask for pc components by only specifying some attributes of the component. Furthermore, make sure that the contractor searches its beliefbase for the cheapest component that satisfies all requests from the manager.

## References

- FIPA, [FIPA contract Net Interaction Protocol Specification \(SC00029\)](#), Foundation for Intelligent Physical Agents, 2002.
- FIPA, [FIPA Request Interaction Protocol Specification \(SC00026\)](#), Foundation for Intelligent Physical Agents, 2002.
- J. McDermott. R1: A rule-based configurer of computer systems. Artificial Intelligence, 19:39-88, 1982.
- Reid G. Smith: The contract net protocol. High-level communication and control in a distributed problem solver. IEEE Transactions on Computers 29(12), 1104 - 1113. December 1980.
- Reid G. Smith, Randall Davis: Negotiation as a metaphor for distributed problem solving. Artificial Intelligence 20, 63 - 109, 1983.
- Mark Stefik, Introduction to Knowledge Systems, Morgan Kaufmann, 1995.
- Michael Wooldridge: An Introduction to MultiAgent Systems (Second Edition), Wiley, 2009.