Chapter 7

IMPLEMENTATION, NUMERICAL EXAMPLES AND TESTS

Implementing and evaluating real added services in manufacturing e-marketplace

Giovanni Perrone and Paolo Renna
Dipartimento di Ingegneria e Fisica dell' Ambiente, University of Basilicata, Viale dell' Ateneo Lucano 10, 85100 Potenza – Italy

Abstract: In order to test the functionality of the proposed agent based distributed architecture for manufacturing e-marketplaces, a proper simulation environment has been developed. Such an environment has been used to test the functionality of all the features discussed in the previous chapters of this book, but also to understand the real value that some added value services, such as re-planning and negotiation, can bring to the participants in a virtual district. This chapter discusses the above issues; in particular, the first part is dedicated to show all of the implementation phases; in the second part of the chapter a numerical example able to show the full functionality of all the applications, described in the previous chapters, will be analyzed; finally, in the third part, a set of experimental results will be discussed in order to understand the economic value of real added services in manufacturing e-marketplaces.

Key words: Discrete event-simulation, added value services, numerical tests

1. THE AGENT BASED ARCHITECTURE IMPLEMENTATION

The agent based architecture proposed in the research project [Perrone et al, 2003] has been implemented by developing a test environment consisting of a discrete event simulation environment able to execute all of the functionalities discussed in the previous chapters. The environment has been
developed by using open software technology entirely realized by using Java Development Kit.

The implementation has been particularly complex since several IT problems had to be solved. All of the implementation issues related to the software development phases are not of interest for the purpose of this book and therefore they will be neglected in the following.

In what follows, indeed, it will be clarified how the test environment (*The System*) interacts with external actors, how it has been formalized by using object-oriented methods, and how the simulation engine workflow works out.

The UML use case diagram, reported in Figure 7-1, clarifies how The System interacts with external actors.

![Figure 7-1. System Use Cases](image)

As the reader can notice the following external actors interact with the system:
- *The Network Customer*: it is a generic registered customer of the e-marketplace, who interacts with the system by inputting the order (use case “*input the order*”), by providing the Customer System with the negotiation knowledge (use case “*provide negotiation knowledge*”) and by eventually approving the finally agreement reached after the negotiation (use case “*final deal approval*”);
- *The Network Supplier*: it is a generic registered supplier of the e-marketplace who interacts with the system by providing the Supplier...