A FORMAL METHOD FOR THE DEVELOPMENT OF AGENT-BASED SYSTEMS

P. Kefalas[1], M. Holcombe[2], G.Eleftherakis[1], M.Gheorghe[2]

[1] Dept. of Computer Science, CITY College,
13 Tsimiski Street, Thessaloniki 546 24, Greece
Tel. +30310-275575, Fax. +30310-287564
{kefalas, eleftherakis}@city.academic.gr

[2] Dept. of Computer Science, University of Sheffield,
Regent Court, 211 Portobello Street, Sheffield S1 4DP, UK
{m.holcombe, m.gheorghe}@dcs.shef.ac.uk

A FORMAL METHOD FOR THE DEVELOPMENT OF AGENT-BASED SYSTEMS

ABSTRACT

Recent advances in both the testing and verification of software based on formal specifications of the system to be built have reached a point where the ideas can be applied in a powerful way in the design of agent-based systems. The software engineering research has highlighted a number of important issues: the importance of the type of modelling technique used; the careful design of the model to enable powerful testing techniques to be used; the automated verification of the behavioural properties of the system; the need to provide a mechanism for translating the formal models into executable software in a simple and transparent way.

This chapter presents a detailed and comprehensive account of the ways in which some modern software engineering research can be applied to the construction of effective, and reliable agent-based software systems. More specifically, we intend to show how simple agents motivated from biology can be modelled as X-machines. Such modelling will facilitate both verification and testing of an agent model, since appropriate strategies for model checking and testing are already developed around the X-machine method. In addition, modular construction of agent models is feasible since X-machines are provided with communicating features, which allow simple models to interact.

INTRODUCTION

An *agent* is an encapsulated computer system that is situated in some environment and that is capable of flexible, autonomous action in that environment in order to meet its design objectives (Jennings, 2000). There are two fundamental concepts associated with any dynamic or reactive system, such as an agent, that is situated in and reacting with some environment (Holcombe & Ipate, 1998):

- the environment itself, which could be precisely or ill-specified or even completely unknown, but nevertheless involves identifying the important aspects of the environment and the way in which they may change in accordance with the activities of the agent,
- the agent will be responding to environmental changes by changing its basic parameters and possibly affecting the environment as well. Thus, there are two ways in which the agent reacts, i.e. it undergoes internal changes and it produces outputs that affect the environment.

Agents, as highly dynamic systems, are concerned with three essential factors:

- a set of appropriate environmental stimuli or inputs,
- a set of internal states of the agent, and
- a rule that relates the two above and determines what the agent state will change to if a particular input arrives while the agent is in a particular state.

One of the challenges that emerge in intelligent agent engineering is to develop agent models and agent implementations that are "*correct*". According to Holcombe & Ipate (1998), the criteria for "correctness" are:

- the initial agent model should match with the requirements,
- the agent model should satisfy any necessary properties in order to meet its design objectives, and