A Survey
of agent interoperability
methodologies

Agent Academy: Deliverable 2.1

IST-2000-31050

Document Name:	Survey of agent interoperability methodologies
Document Authors:	UT-APS
Document Editors	All
Document Reference Number:	**AA_D_2.1_FIN**
Circulation:	Public

IST-2000-31050 Agent Academy

Project Start Date:	**1 November 2001**
Project End Date:	**1 November 2003**
	(Duration of the project: 2 years)

ABSTRACT

The following document is a document, which introduces a survey on the agent interoperability methodologies related to the implementation and design decisions of the AA consortium. The survey is based on the state of the art document presenting an overview of the agent communication languages and JADE development platform. Major implementations and working platforms designed for the selected architectures are also explained. The document is concluded by a section summarizing the implications of the AA choice of the development platform on interoperability.

History

Version	Date	Changes	From	Review
Final	22/2/2002	Final Version	UT-APS	All

TABLE OF CONTENTS

1 Introduction

1.1 Purpose

In this document the interoperability issues within Agent Academy (AA) platform are specified on the basis of a survey of existing methodologies. Chapter 1 is the introduction. In Chapter 2 the various issues of interoperability are presented. In Chapter 3, the AA requirements and context is applied for the framework in chapter 2. Next, Chapter 4 handles the requirements of the three test case scenarios. Finally, Chapter 5 summarises the design and implementation decisions for AA platform.

1.2 Interoperability in multi-agent systems

Agent-based systems have been designed to support more and more complex tasks (ref), which has led to the need of grouping agents into societies, where they interact to achieve their goals. This includes the development of agents that can negotiate and reason to reach agreements or persuade the others to pursue a course of action. Either way, this has brought about the need of agent interactions and the issue of interoperability has become one of great significance. Interoperability is concerned with allowing agents to exchange information and services with one another, as well as negotiating and reasoning against one another. There is an increasing effort to create standards for agent interoperability. Among these initiatives, three of them will be addressed in this document:

1. The ARPA Knowledge Sharing Effort (KSE) is a consortium to develop conventions facilitating sharing and reuse of knowledge bases and knowledge-based systems. (Mayfield, Labrou & Finin, 1996).
2. FIPA is a non-profit organisation aimed at producing standards for the interoperation of heterogeneous software agents. (Poslad, 2001)
3. The OMG Mobile Agent System Interoperability Facility (MASIF) is a standard developed by Crystaliz, General Magic, GMD FOKUS, IBM, and The Open Group.

1.3 Interoperability issues within the AA Project context

Interoperability is an important issue in Agent Academy (AA). First of all, the framework will be developed as a multi-agent system, this way, it is important to guarantee that the agents that will perform the different actions in AA can interact with each other to achieve their goals.

> After the untrained agent is created by the Agent Factory, "it enters the Agent Training (AT) module, where its world perception increases substantially during a virtual interactive session with an agent master (AM). (…) After training is completed, the new intelligent agent, armed with tools for reporting its behaviour to Agent Academy, is released to the world" (Agent Academy Technical Annex).

In this scenario, two cases of agent interoperability can be easily identified: the first one is the interaction between the *untrained agent and the agent master*, in order to train the newly born agent; the second one is the interaction between the *intelligent agent and AA*, for the purpose of reporting to the system about its experience in the world.

There is still one reason why interoperability should be a concern of Agent Academy. As a system designed specifically for training intelligent agents, it is important that the AA "pupils" will be able to communicate with one another and with other agent-based systems.

> *"Agent environments are becoming increasingly open, interconnected and heterogeneous. This suggests that future agents will need to be able to deal with multiple agents communication languages, multiple ways of expressing content and multiple ontology representations." (Steven, Constantinescu & Calisti, 2001).*

The main interaction cycle within AA is performed between the three components Data Mining, Agent Factory and Agent Training Modules. This requires that all of them are able to accept and interpret the messages from each of them.

An important issue for this interaction is related to the question how the rules in ATM are represented in FIPA-ACL messages by DMM, and in what form they sent to the agents to be trained by the ATM.

As proposed in the Requirements specification document a Semantic Language (SL) is used as the content language among the agents of Agent Academy, since JADE has built-in parsers and encoders for FIPA-SL-0. This way message templates expressed in SL are presented for the messages between ATM <-> DMM, ATM <-> AF and ATM <-> Untrained agents.

As proposed in the Requirements specification document JESS rules are used for the messaging format between the ATM and the agents to be trained. JESS rules are transported with FIPA-ACL messages.

Agent Training Module (ATM)

ATM inserts number of initial rules to the rule base of the agent in order to facilitate the agent with the ability to operate on its knowledge base and achieve its behaviour.

ATM interacts with the DMM and enhances agent's behaviour and use in an iterative and recursive manner.

ATM also trains the existing agents incrementally in order to improve their intelligence, by taking the new tendencies from the Data Mining Module.

ATM converts the data structures represented in SL into JESS rules.

Data Mining Module (DMM)

DMM refines the rules for better behaviour representations and sends them back to the ATM.

The rules in DMM are represented in FUPA-ACL messages

Agent Factory Module (AF)

The main responsibility of the module is to create untrained agents by defined by user agent type and attributes and then send them to the ATM to be loaded with initial abilities of that specific agent type.

Each of the agents created by the AF need to understand the syntax and also the semantics of the messages sent by the ATM.

1.4   References

1. Review of the state of the art – AA_IR_T1.1.doc
2. Agent Academy proposal – AA-proposal_final.doc
3. Requirements specifications – AA_IR_T1.2.doc