第30卷 第4期 电子科技大学学报 Vol.30 No.4
2001年8月 Journal of UEST of China Aug. 2001

ACL Based Interoperability Addon Between Mobile Agent and Intelligent Agent*

Zhang Yunyong Liu Jinde Guo Weina

(College of Computer Science & Engineering, UEST of China Chengdu 610054)

Abstract Software agents are becoming an important design paradigm for inter-operating open, distributed systems. Mobile agents and intelligent agents are obviously different in origins, emphases, implementation and application platforms and interests of the supporting communities. But both mobile agents and intelligent agents paradigms are targeting adaptive and flexible co-operation, particularly for interoperability between or within distributed systems in a dynamically changing environment. In this paper, the similarities and differences, advantages and disadvantages of the OMG mobile agent and FIPA intelligent agent approaches in specific application context are analyzed. Based on the analysis, two ACL based possibilities are discussed for harmonizing and integrating the two standards and a MASIF-FIPA addon is also developed.

Key words mobile agent; intelligent agent; interoperability; agent communication language

1 Agent Mobility Versus Intelligence

Software agent technology is quickly becoming a key design paradigm for dynamic and heterogeneous open, distributed environments. At the moment, however, the software agent community is divided into two religions: one for mobile agents and another for usually static intelligent agents. The representatives of the two religions are OMG MASIF and FIPA standardization efforts[1,2].

OMG MASIF aims at enabling mobile agents to migrate between agent systems of the same profile (language, agent system type, authentication type and serialization methods) via standardized CORBA IDL interfaces. FIPA focuses on enabling the intelligent agents interoperability via standardized agent communication and content languages.

The mobile agent's religion focuses on mobility of program codes together with their states among network sites. Via higher granularity of mobile agents (i.e. higher number of operations encapsulated within a mobile agent) and on-demand migrations, the mobile agents approach can help to dynamically adapt interfaces and services of remote systems, reduce dependency on the constant availability of underlying network connectivity, achieve dynamic load balance and enable dynamic distribution of functions. With these features, mobile agents provide a robust, flexible and effective design paradigm for the distributed dynamic environments.

Intelligent agents refer to the classes of agents, most of which are static, that rely on high level, speech act agent communications (e.g. via KQML/KIF, FIPA-ACL/SL). Within an agent communication paradigm, co-operation is realized via the agent communication language (ACL) [3], the content language and the ontology which identifies the set of basic concepts (taxonomy) used in the message content for co-operations. An ontology here is similar to an API in the RPC context, which identifies a specific co-

Received April 5, 2001
* The project is supported by National Defense Research Foundation

operation interface of the intelligent agents[4]. Via integrating message and exchanging knowledge or concept definitions, we can achieve similar effects to those of mobile agents.

Both OMG MASIF and the FIPA standardization efforts focus on the dynamic interoperability between static software systems like agent systems or intelligent agents. The major difference between mobile agents and intelligent agents and the corresponding OMG MASIF, and FIPA specifications is that a mobile agent usually uses a low level programming language, while the intelligent agent typically has a speech act alike communication language and a predicate logic based content language.

2 MASIF Specification and MASIF Addon

Mobile agent has taken a very momentous influence on the fields such as collaborative system, workflow manage system since 1990 because of its unique technical advantages[5~7]. Until today, a large number of mobile agent platforms have been developed and MASIF is brought forward in order to solve the interoperability of the different agent platforms.

The idea behind the MASIF standard is to achieve a certain degree of interoperability between mobile agent platforms of different manufacturers without enforcing radical platform modifications. The following list comprises the mandatory requirements that were identified within MASIF:

- Marshalling and un-marshalling of agent programs
- Encoding of agent containers for transport
- Transport of agents from one agent facility (i.e. execution engine) to another
- Runtime registration and invocation of agent facilities
- Runtime query of a named agent facility by agents
- Runtime security of agents

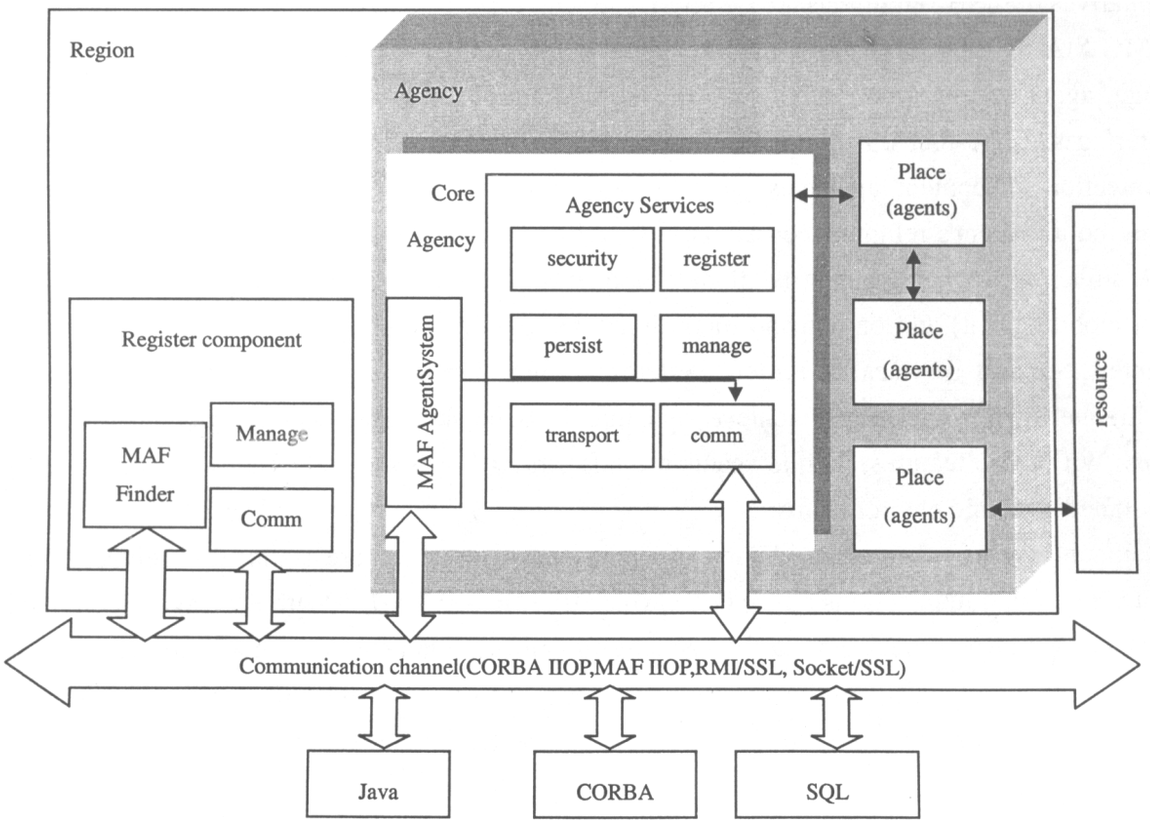

Fig.1 MASIF addon architecture

Additionally, several optional requirements were defined by MASIF, covering the identification and localization of agents, the starting, stopping, and suspending of an agent's execution, and the runtime monitoring of agent facilities and agents via their names or characteristics.

Under the specification, we have developed MASIF addon. As shown in Fig.1, the addon has adopted the concepts of agent systems (i.e. agencies), places and regions that are also used by several existing agent platforms.

A place groups the functionality within an agency, encapsulating certain capabilities and restrictions for hosted agents. Each agency comprises at least one place in which the hosted agents are running.

A region facilitates the platform management by grouping sets of agencies that belong to a single authority.

Two interfaces represent the core of the MASIF standard: the MAFAgentSystem interface is associated with every MASIF-compliant agency and provides operations for the management and transfer of agents. The MAFFinder interface is associated with a region, i.e. a set of agencies. It is a part of region registration component that supports the localization of agents, agencies, and places in the scope of a region.

3　Possible Integration Strategies

Both the mobile agent and intelligent agent technologies have their advantages and disadvantages in specific application contexts. It is expected that in the near future both technologies will be prevalent in the telecommunications applications. We can envisage in this context a variety of possibilities for harmonizing and integrating the OMG and FIPA frameworks for agent interoperability.

For a more detailed analysis we need to dig deeper into the FIPA intelligent agent platform, i.e. the agent reference model. The directory facilitator (DF), agent management system (AMS) and agent communication channel (ACC) are specific types of agents that support agent management. The AMS and ACC support inter-agent communication. The ACC supports interoperability within and across different platforms. The internal platform message transport (IPMT) provides a message routing service for agents on a particular platform which must be reliable, orderly and adhere to the requirements specified in FIPA Part 2. The ACC, AMS, IPMT and DF form will be termed as the agent platform (AP). These are mandatory, normative components of the model.

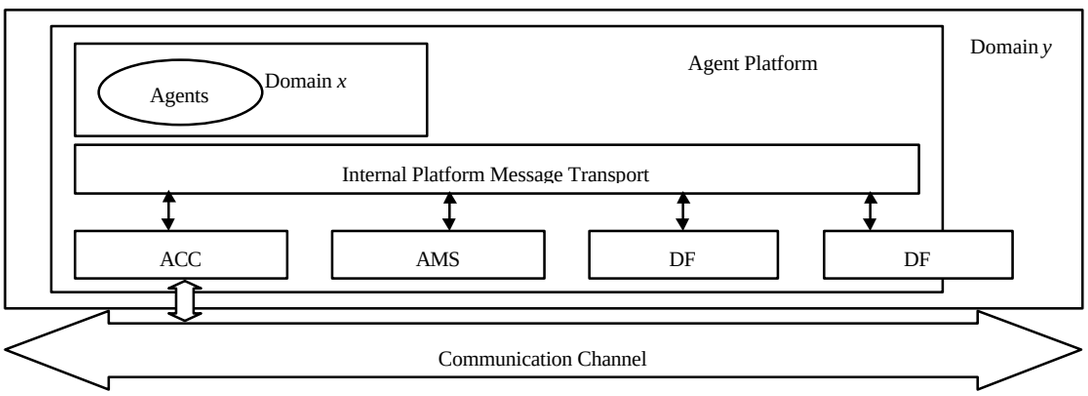

Fig.2　FIPA agent reference model

To implement the interoperability between mobile agent systems and intelligent agent systems, the embodiment of ACL in MASIF addon should be noticed since ACL is the core of ACC and ACC is the core of FIPA systems. We put forward two solutions: implementation of ACL module in MASIF addon to embody ACL directly or construction of IDL/ACL gateway in MASIF addon to embody ACL indirectly. The former is complete and the later is convenient while has low efficiency. Our MASIF-FIPA addon has both the ACL module and IDL/ACL gateway, we will discuss them in the following section.

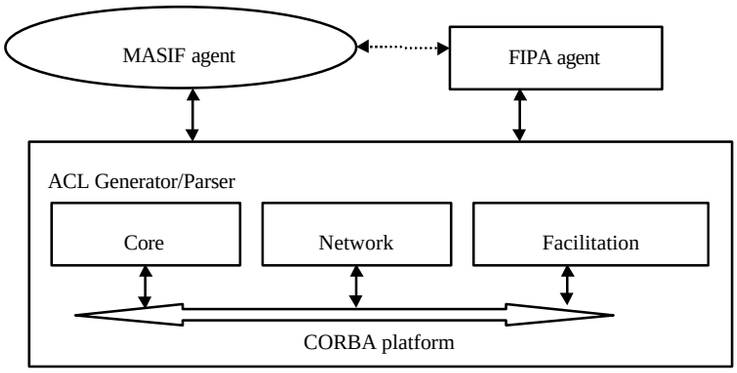

Fig.3 ACL module in the MASIF addon

3.1 Direct Approach

As shown in Fig.3, the ACL module in the MASIF addon consists three sub modules: common basic performative module, network performative module and facilitation performative module. Here, we only give the MASIFACL module′s IDL:

```
module MASIFACL{
module Core{//common basic performative
  interface CoreACL{
    void askIf(in string sender,//the sender
               in string receiver,//the reciver
               in string inReplyTo,//point out to which message the reply relates
               in string replyWith,//the reply
               in string language,//the language，such as prolog, KIF
               in string ontology,//the ontology，a expression of concepts among agents
               in string content//the transportation content);
```

Other performative，such as askAll, askOne, streamAll, eos, tell, untell, deny, insert, uninsert, deleteOne, deleteAll, undelete, achieve, unachieve, advertise, unadvertise, subscribe, error, sorry, standby, ready, next, rest, discard, forward, broadcast are silimar to askIf.

```
};
};
module Network{//network performative
interface NetworkACL{
void register();
void unregister();
void transportAddress();
                    };
};
```

```
module Facilitation{//facilitation and service performative
interface FacilitationACL{
brokerOne、brokerAll、recommendOne、recommendAll、recruitOne、recruitAll performative
};
};
};
```

3.2 Indirect Approach

The function of IDL/ACL gateway is mapping the services MASIF addon supports to the FIPA specification′s format. The architecture of the IDL/ACL gateway is shown in Fig.4.

The following mappings of MASIF and FIPA concepts must be identified:

- FIPA domain corresponds to MASIF region
- FIPA agent platform corresponds to MASIF agent system
- FIPA directory facilitator corresponds to MASIF MAF finder
- FIPA agent management system corresponds to MASIF MAF agent system
- FIPA agent wrapper corresponds to mobile agent resource interfaces

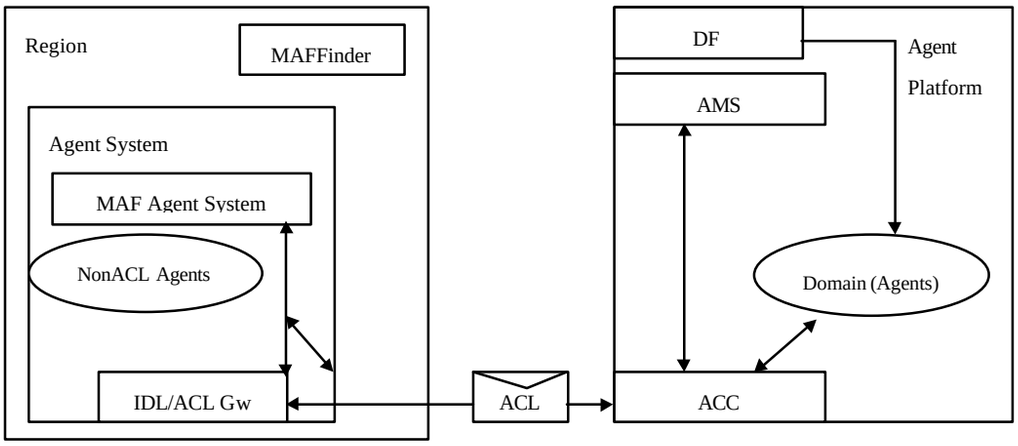

Fig.4　IDL/ACL gateway

4 Experiment and Summary

We develop the MASIF-FIPA addon by extending the MASIF addon and test it, and the test model is shown in Fig.5. The test platform are Solaris(Orbix Web3.0[8]，KOBALT，MASIF-FIPA addon) and NT(Orbix Web3.0，FIPA compliance Jade). The experiment steps is that first running orbix Web CORBA services, then running agent daemon and MASIF-FIPA addon, and running the test program in the end.

This model can be looked as a chain stores model, suppose agent1 is the depot director of chengdu branch and agent2 is the depot director of beijing branch. Now, the chengdu store lacks of some goods and wants to query that whether beijing store has these goods. First, agent1 send askIf performative to agent2 to query whether agent2 has these goods. If agent2 has these goods, agent2 will send reply performative to agent1 and agent1 will create a new agent to agent2 to look for goods(agent2 will authenticate agent1 for the security reason and if authentication fails, agent2 will send deny performative to agent1). Finally send the goods information to agent1, If agent2 does not have these goods, it will agent2 will send sorry performative to agent1, and agent1 can contact id2 city using the “reply-with id2” field which is in the

sorry performative. We focus on the difference of the two system, and validate the validity of MASIF-FIPA addon by the experiment.

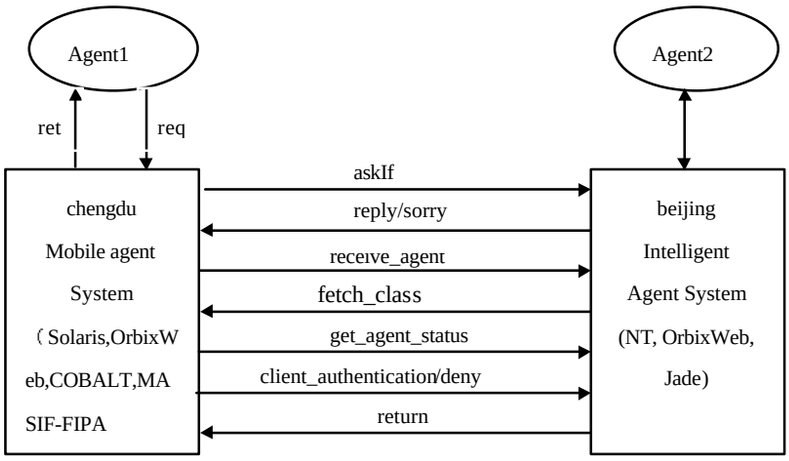

Fig.5 The chain stores model

(askIf

: content {Have goods? x} //the content level
: sender chengdu
: receiver beijing //the communication level
: reply-with id1
: language KIF//language, also can be Prolog
: ontology NYSE-TICKS)//ontology //the message level

the askIf performative for querying goods sending from chengdu to beijing

(sorry

: sender beijing
: receiver chengdu
: in-reply-to id1//the reply to id2
: reply-with id2//the future information providing to id1)

the sorry performative sending from Beijing to chengdu because of lacking of goods

The development of MASIF-FIPA addon strides a very important step to solve the interoperability between mobile agent and intelligent agent. We think that solving the interoperability efficiently will accelerate the further development of agent systems and let agent be more useful in the new application fields such as mobile computing and information service.

References

1 Object Management Group. Mobile agent standard. America, 1999

2 FIPA. FIPA98 draft Specification: Part11: agent management support for mobility. FIPA8415, Version 0.3, America, 1998

3 Sebastian Vieira. FIPA-ACL. http://www.cs.helsinki.fi/u/kraatika/Courses/Agents/fipa-acl.html

4 Ma Zheng. Intelligent Network——the key technology of network of intelligentization . Journal of University of Electronic Science and Technology of China, 2000, 29(1): 1~4[马争，魏险峰、智能网——网络智能化的关键技术、电子科技大学学报，2000, 29(1): 1~4]

5 Zhang Yunyong. Agent oriented software engineering: [Ph.d Dissertation]. Chengdu: College of Computer Science & Engineering, UEST of China, 2000 [张云勇、面向agent的软件工程:[博士论文]. 成都:电子科技大学，成都，2000]

6 Li Jinhou. Unification and separation of agent′s EGO. Journal of University of Electronic Science and Technology of China, 2001, 30(1): 77~81[李金厚、智能Agent的自我的统一与分离、电子科技大学学报，2001, 30(1):77~81]

7 Su Sen, Tang Xuefei, Liu Jinde. Object oriented interoperability technology. Journal of University of Electronic Science and Technology of China, 1998, 27(1): 90~94 [苏　森，唐雪飞，刘锦德、面向对象德互操作技术、电子科技大学学报，1998, 27(1): 90~94]

8 Inoa. OrbixWeb3.0 beta programmer's guide. Ireland, 1997, http://www.iona.com/

基于ACL的移动agent/智能agent互操作插件*

张云勇** 　　**刘锦德** 　　**郭维娜**

（电子科技大学计算机学院　成都　610054）

【摘要】软件agent正成为互操作的、开放的分布系统重要的设计方法。移动agent和智能agent的起源、实现、应用平台及其支持团体都不同，但是移动agent和智能agent方法的目的都是达到适应的、灵活的协作，尤其是动态变化环境中的分布系统内或分布系统间的互操作性。文中分析了OMG的移动agent与FIPA的智能agent的异同和在特定应用环境中的优劣；讨论了基于ACL的协调、集成MASIF与FIPA标准的方法，并开发了MASIF-FIPA插件。

关　键　词　移动agent;　智能agent;　互操作性;　Agent通信语言

中图分类号　TP316

2001年4月5日收稿

* 国防科工委预研基金资助项目

** 男　25岁 博士生