Intelligent Agent Communication by Using DAML to Build Agent Community Ontology

Cheng-Hsiung Hung, Hong-Jie Dai, and Jason Jen-Yen Chen

Abstract—This paper presents a new approach for intelligent agent communication based on ontology for agent community. DARPA agent markup language (DAML) is used to build the community ontology. This paper extends the agent management specification by the foundation for intelligent physical agents (FIPA) to develop an agent role called community facilitator (CF) that manages community directory and community ontology. CF helps build agent community. Precise description of agent service in this community can thus be achieved. This facilitates agent communication. Furthermore, through ontology update, agents with different ontology are capable of communicating with each other. An example of advanced traveler information system is included to illustrate practicality of this approach.

Keywords—Intelligent agent communication, DARPA agent markup language (DAML), Community ontology, Advanced Traveler Information System (ATIS).

I. INTRODUCTION

THE information on the Web has been increasing rapidly. It is mostly marked- up in Hypertext Markup Language (HTML) for human to read, but not for program to find or to interpret. Thus, World Wide Web Consortium (W3C) developed Extensible Markup Language (XML) [1] that allows user-defined tags to precisely mark-up the information for program to access.

However, XML is rather limited in its capability to describe the relationship among objects. To remedy that, DARPA Agent

C. H. Hung is with the information management center of Chung-Shan institute of science and technology, Taiwan. (e-mail: hongcx@ pchome.com).
H. J. Dai is with the Department of Computer Science and Information Engineering, National Central University, Jhongli, 32099, Taiwan. (e-mail: 92522045@cc.ncu.edu.tw).
J. J.-Y. Chen is with the Department of Computer Science and Information Engineering, National Central University, Jhongli, 32099, Taiwan. (phone: +886-3-4227151 ext. 35313; fax: +886-3-4222681; e-mail: jychen@csie.ncu.edu.tw).

Markup Language (DAML) [2] program was officially launched in August 2000 to develop the language and tools to facilitate the realization of Semantic Web. DAML extends XML and Resource Description Framework (RDF) [3]. Its new version DAML+OIL provides a rich set of constructs to create ontology, and to markup information for attaining machine readability and understandability.

Consequently, this paper uses DAML to build Community Directory and Community Ontology. In addition, this paper extends the Foundation for Intelligent Physical Agents (FIPA) Agent Management Specification [4] to develop an agent role called "Community Facilitator" (CF) to manage the Community Directory and the Community Ontology in order to build agent community.

II. RELATED WORK

This section describes, respectively, the work before DAML, namely, XML, RDF, and RDFS. Besides, the reasons for using DAML to build community directory and ontology will be depicted.

1) XML (eXtensible Markup Language) [1]:

XML is a subset of Standard Generalized Markup Language (SGML) [5]. It is simple with flexible textual format. It looks like HTML, but its user-defined tags remedy the shortcomings of HTML. It thus is more scalable than HTML.

1) RDF (Resource Description Framework) [6]:

RDF is developed by W3C. It is an infrastructure to encode, exchange, and reuse structured metadata, which are marked with XML tags. And, XML tag provides unambiguous expression of semantics. Thus, RDF is capable of expressing semantics of data and describing relationship among resources, such as ontology.

2) RDFS (Resource Description Framework Schema) [7]:

As just mentioned, RDF defines a simple model for describing interrelationships among resources in terms of named properties and values. However, it provides neither mechanisms for declaring these properties, nor mechanisms for defining the relationship between them and other resources. RDFS, an object-oriented system extended from RDF, fixes the above problems.

The reason why DAML, rather than XML, RDF, or RDFS, is

chosen to build the community directory and ontology is that, although XML facilitates the representation of task or domain specific data on the web, XML lacks the semantics [8]. On the other hand, RDF and RDFS provide the semantics, but they lack adequate expressive power in terms of uniqueness and inverse of properties, necessity and sufficiency of class membership, equivalence and disjoint-ness of classes, and so on. Furthermore, they cannot specify domain or range constraints on properties. In other words, the semantic remains under-specified this way. Apparently, DAML appears to be the best among the four in describing ontology. There seems no literature using XML, RDF, or RDFS to manage ontology.

III. DAML COMMUNITY FACILITATOR

This section describes the application of DAML, and introduces Community Facilitator (CF) which is an agent role extended from FIPA Agent Management Reference Model [4]. Finally, the issue of implementation will be discussed.

1) *DARPA Agent Markup Language (DAML)*

DAML builds upon XML, RDF and RDFS, as just mentioned. It contains not only the advantages of the others mentioned above, but also possesses machine readability, semantic checking, and relationship constraint. Besides, it has capability of inference so that membership or service of community can be precisely described. With the assistance of DAML-based ontology, members of agent community are able to communicate with each other.

Fig.1 depicts partial community directory of an Advanced Traveler Information System (ATIS) that will be explained in the next section. Besides recording information of Community Facilitator (lines 1-9) and members of community (lines 20-22), the community directory also records meta-data about community, such as community name (line 12), community description (lines 13-15), ontology used in community (lines 16-18), and so on.

```
1  <community:CF rdf:ID="theCF">
2
<community:agentName>"CF"</community:agentName>
3  <community:agentDescription>
4    "ATIS Community Facilitator"
5  </community:agentDescription>
6  <community:locator>
7    "http://ape44.csie.ncu.edu.tw/ATIS/agent/CF"
8  </community:locator>
9  </community:CF>
10
11 <community:Community rdf:ID="ATISCommunity">
12
<community:communityName>"ATIS"</community:communityName>
13  <community:communityDescription>
14    "Advanced Traveler Information System"
15  </community:communityDescription>
16  <community:ontology>
17
"http://ape44.csie.ncu.edu.tw/ATIS/ontology/ATIS.daml"
18  </community:ontology>
19
20  <community:hasCF rdf:Resource="#theCF"/>
21  <community:consistOf rdf:Resource="#agent1"/>
22  <community:consistOf rdf:Resource="#agent2"/>
23 </community:Community>
```

Fig. 1 ATIS Community Directory

2) *Community Facilitator (CF)*

Community Facilitator (CF) must maintain a community directory for recording the information of its members, such as agent name, agent locator, service name, service type, and so on. When an agent joins or leaves the community, it must register or cancel registration through CF. Through querying CF, an agent can find out other members' locators and services. While an agent requests to join the community, CF is able to certify security to acquire trust. By doing so, community members hold higher authority than that of non-members.

Fig. 2 shows that CF maintains a set of domain ontology (called community ontology) and a community directory for agent communication. When an agent requests a service, it first asks CF to check the services provided by this agent community, and to acquire necessary ontology. Then, the agent can directly communicate with the agent that provides the service, without going through CF. Moreover, when a new agent joins in, it informs CF about the external ontology it provides.
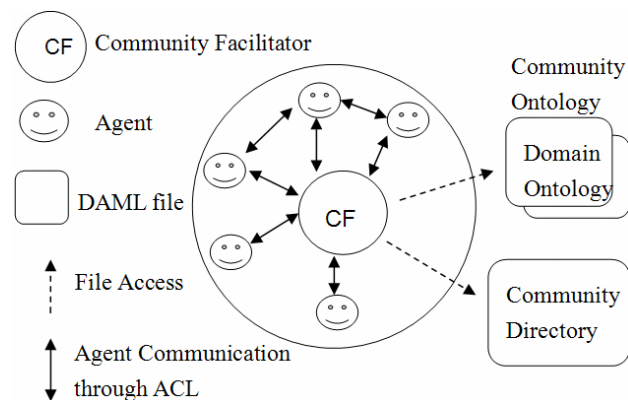

Fig. 2 Communication in Agent Community

Note that this is a centralized registry of ontologies in CF with some drawbacks like single point of failure and limited scalability. This can be improved by using distributed ontologies that are managed by various ontology management systems.

3) *Implementation*

Jena semantic web toolkit [9], a DAML API by HP Labs, is used to handle the community directory built using DAML, and to develop a Java class called "Directory". The main functions

of the community directory are as following:
1) Adding the information of an agent.
2) Removing the information of an agent.
3) Getting the list of agent names of all members.
4) Getting the information of individual agent by name.
5) Getting all ontology used by members in the community.
6) Adding external ontology which an agent provides.

IV. AN EXAMPLE

This section describes the Advanced Traveler Information System (ATIS) agent community.

After joining the ATIS community, a user agent can query CF to acquire service information of other agents. Because that the community directory is constructed using DAML, every member can easily and precisely find out other members (or their services) with CF's assistance in accessing the directory. Furthermore, all the members' domain knowledge (ontology) may not be the same. The user agent, for example, doesn't understand the knowledge that a parking service agent holds. In addition to the community basic ontology (the knowledge of this ATIS) provided by CF, members can also provide external ontology (the knowledge of parking, in this case). Therefore, through DAML-based ontology, members can communicate with each other to acquire requested services.

As shown in Fig. 3, while joining the ATIS community, the parking service agent will inform CF about the parking ontology it provides (Fig. 3 (a)). The CF maintains not only the original ATIS ontology, but also the parking ontology (Fig. 3 (b)). Note that Fig. 3 (a) and Fig. 3 (b) show that the ontology can be updated. While the user agent wants to communicate with other agents, it first queries CF about all necessary ontology. CF will inform the user agent, if the ontology is acquired (Fig. 3 (c)). Then, the user agent can communicate with the parking service agent (Fig. 3 (d)).
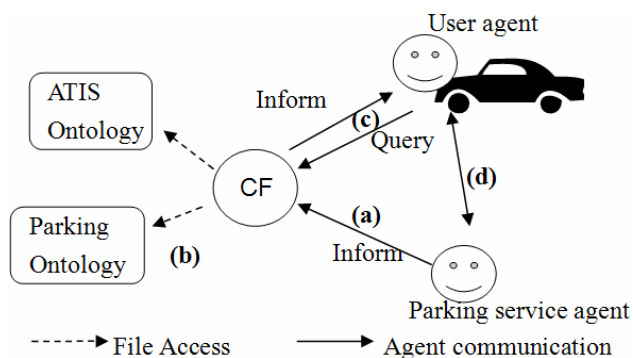

Fig. 3 Ontology update

V. CONCLUSIONS

This paper uses DAML to build agent community ontology to facilitate intelligent agent communication in the following ways:

1) Precise agent service description:
   Because that DAML is capable of machine readability, semantic checking, relationship constraint specification, and inference, the agent services described in DAML are rather precise that facilitates intelligent processing by agents.

2) Updated agent community ontology:
   The community ontology built by DAML is used to facilitate communication within agent community. As ontology held by each agent is often different, how to communicate among the agents is very important. When a new agent joins the community, it informs community facilitator (CF) about the new ontology with it. And, CF will maintain the community ontology to include the new ontology. As a result, all the agents can understand each other, and thus can communicate intelligently with each other.

REFERENCES

[1] World Wide Web Consortium. (1998, February). [Online]. Extensible Markup Language. Available: http://www.w3.org/XML/
[2] DARPA Agent Markup Language. (2000, August). [Online]. Available: http://www.daml.org/
[3] World Wide Web Consortium. (1999, February). [Online]. Resource Description Framework. Available: http://www.w3.org/RDF/
[4] Foundation for Intelligent Physical Agents. (2002, December). [Online]. FIPA Agent Management Specification. Available: http://www.fipa.org/specs/fipa00023/
[5] Information Organization for Standardization. (2001, August). [Online]. Standard Generalized Markup Language (SGML). Available: http://www.iso.ch/cate/d16387.html
[6] D-Lib Magazine. (1998, May). [Online]. An Introduction to the Resource Description Framework. Available: http://www.dlib.org/dlib/may98/miller/05miller.html
[7] World Wide Web Consortium. (2003, January). [Online]. Resource Description Framework Schema. Available: http://www.w3.org/TR/rdf-schema/
[8] S. A. Mclraith, T. C. Son, H. Zeng, "Mobilizing the Semantic Web with DAML-enabled Web Services," Presented at Second Int'l Workshop Semantic Web (SemWeb'2001), Hong Kong, May 1, 2001.
[9] HP Labs. (2003, January). [Online]. Jena Semantic Web Toolkit. Available: http://www.hpl.hp.com/semweb/jena.htm

Cheng-Hsiung Hung received his B.S. degree in Mathematics and his M.S. degree in Computer Science and Information Engineering from National Central University in Taiwan in 2001 and 2003, respectively. Since 2003, he has been an assistant researcher at the Information Management Center of Chung-Shan Institute of Science and Technology. His research interests include: agent-based software engineering and web service.
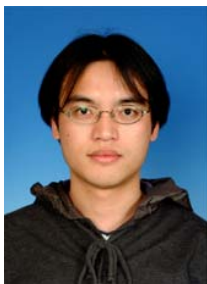
Hong-Jie Dai received his B.S. degree in Computer Science and Information Engineering from Tung Hai University and his M.S. degree in Computer Science and Information Engineering from National Central University in Taiwan in 2003 and 2005, respectively. Since 2005, he has been an assistant researcher at the Academia Sinica. His research interests include: agent-based software engineering and semantic web.

Jason Jen-Yen Chen received a BS degree in industrial engineering from Tung Hai University, Taiwan; an MS degree in industrial engineering, an MS degree and a PhD degree in computer science and engineering from the University of Texas at Arlington. He is currently a Software Engineering Professor at the Department of computer science and information engineering in National Central University, Taiwan.

His current research interests center on agent-based process environment. Dr. Chen won Top, Third, and Fifth Scholar in the world in System and Software Engineering in 1995, 1996, and 1997, respectively. The ranking is based on cumulative publication of six leading journals in that field. He is a senior member of the IEEE.