Asian Journal of Information Technology 4 (11): 1001-1005, 2005

Intelligent Agent Communication: Use of KQML

Md. Mosharrof Hossain Sarker
Department of Electrical and Electronic Engineering Islamic University of Technology, Bangladesh

Abstract: This study presents Knowledge Query and Manipulating Language (KQML), a language for knowledge and information exchange. KQML is considered for intelligent agent communication in the context of Aerospace design as a step forward from IDMC approach to IDMC system. Three types of message are considered for multi-agent communication. The communication mechanism is based on KQML structures. ZEUS agent toolkit is considered as platform facilitating communication.

Key words: Intelligent, agent, KQML

INTRODUCTION

Design of a complex product is concerned with several differential stages and heterogeneous skills of participants. To complete this type of product within a specific timeframe and with a reasonable cost, the task of engineering design is distributed by nature. Developing a virtual teamwork based on web technology where the participants can express their opinions without traveling could reduce this complexity. With the advent of the internet and the techniques available from Distributed Artificial Intelligence (DAI) electronic collaborative design has become possible[1].

EID (Environmental for Distributed Integrated Design) involves a consortium of several European partners from the European aerospace industry. Itsaimed at developing an intelligent system that would detect and resolve design mismatches that can occur during the integration phase of a distributed design. The work led to the development of the Intelligent Mismatch Control System (IMCS)[2]. One of the drawbacks of the IMCS stems from the fact that the knowledge is encapsulated into one single unit: the expert system. Taratoukhine[3] took the IMCS a step further and developed the

Intelligent Distributed Mismatch Control (IDMC). The IDMC approach used a multi-agent framework that enabled knowledge to be encapsulated into several agents. Hence, communication of agent has become important to deliver the information of mismatches among partners.

The aim of this study is to establish the use of KQML for communication between geographically distributed agents.

INTELLIGENT AGENT

There is no agreed definition of the term agent. 'Agent' has been, for some time at least, an over-used term. The following definition could be used more widely[4].

Definition 1: An agent is a computational system that inhabits in a complex, dynamic environment (e.g., computers and networks). The agent senses the changes in its environment and affects the environment synchronously, and has a set of goals or motivations that it tries to achieve through these actions.

Agent's characteristics: Intelligent Agents have several characteristics. Certain characteristics are discussed here[4]. Autonomy - An agent should be able to operate with some degree of autonomy without direct intervention of the human user. Re-activeness -Agents should be able to perceive their environment and respond in a timely fashion to the changes that happen in it. Pro-activeness - Agents can take the initiative and plan their course of action.

AGENTS' COMMUNICATION

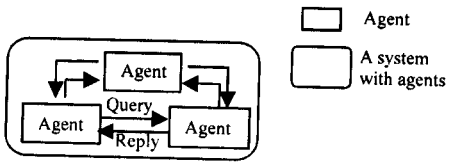
In the kingdom of agent, an agent communicates with another agent. In this communication two different abilities are followed: one is receiving of messages, another is sending of messages.

Agents' messaging: An agent receives input via its sensor. The input of software agent is message. Message is information, which consists of a tag and a body. There are three types of messages: broadcast / multicast message[1,5], internal message and external message.These types are discussed in detail below

All of these messages stated above helps intelligent agent to join in dialogue. These messages could be active or passive depending on the role of dialogue.Two types of capabilities are followed in agent in sending or in receiving of messages: one is assertion, another is query.

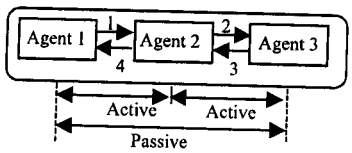
An agent receives a request from a source and replies to the source by making an assertion. Both assertion and query could be either active or passive. Using these artificial techniques, an agent potentially controls another agent for query or for information asserted.

Internal message: This is a message sent from an agent to another agent within a system.

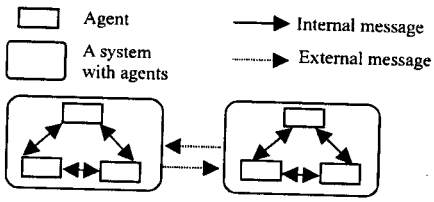


Active internal message: Agent 1 sends a query to agent 2 and agent 2 receives the query. Agent 2 sends assertion to agent 1 using its degree of intelligence. This type of assertion is called active assertion.

Passive internal message: Agent 1 sends a query to agent 2. Agent 2 forwards this query to agent 3 because it is unable to satisfy. Agent 3 accepts the query and sends assertion to agent 1 via agent 2 This type of assertion is called passive assertion.

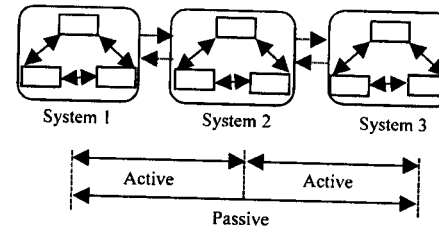


External message: This message is sent between a system with agents and another system with agents.



Active external message: System 1 sends a query to system 2 and system 2 receives the query. System 2 sends assertion to system 1 using its degree of intelligence. This type of assertion is called active assertion.

Passive external message: System 1 sends a query to system 2. System 2 forwards the query of system 1 to system 3 because it is unable to satisfy. System 3 accepts the query and sends assertion to system 1 via system 2. This type of assertion is called passive assertion.



Broadcast / Multicast message: This is a message to search an entity (e.g.; atomic object) in the cloud of agent when the address is lost, or to find out the source of incomplete information. Each agent knows its own address contents, and the address of the environment where it lives. All agents, who are logged on, will receive the message. Agent 2 has recognized the message and replies positively Yes . The facilitator of agent 2 sends the result to named facilitator.

Speech acts: Speech act theory follows the natural conversation mechanism what we normally expose as a human beings. Speech Act has three distinct actions; 1 locution. It is involved with the physical utterance; 2) Illocution - it is involved to bear speaker intention to listener and 3) Perlocution - it is involved as a resultant effect of illocution. Electronic communication does not actually follow human speech mechanism; rather it tries to imitate human speech mechanism from the viewpoint of software engineering as well as linguistics description. The claim of speech act could be; 1) to constitute equilibrium in thoughts. Are my thoughts your thoughts? 2) What to say and when to say? and (3) completely reluctant to deliver thoughts[5,6]. For example, I am hot could mean either a request to put off clothes or ademand to decrease temperature.

The term performative is being used inSpeech Act theory to identify illocutionary force. There are several categories in illocutionary force assertives, directives, commissives and so forth[5,7]. GO word can be used in different forms in English sentence where each English sentence means a separate illocutionary force as shown below.

Command	Go
Request	Would you please go
Offer	I would go, if you were there.

The purpose of discussion about speech act is KQML is based on speech act theory.

KQML FOR AGENT COMMUNICATION

KQML is part of a broad research effort. KQML could be considered as a solid basic protocol for communication of agents. KQML structure, which is given below has two parts: one part is semantics of the message; another part is parameters of the message.

(KQML–performative
: sender < > } Semantics of the message
:receiver< >
: language < >
: ontology :< > } Parameters of the message
: content < >

KQML – performative - ask, tell, and so forth, language; the way to express the message, ontology- the vocabulary of the message, content –the actual message, sender ; the name of agent / facilitator who has sent a query, receiver ; the name of agent/facilitator who has replied the query[5,7].

Facilitator: Facilitator (F) is a piece of agent called facilitating agent, which is concern with communication services among agents. It deals with network application as it is a network software agent and its activity varies from one application to another. Certain activities are: 1) Routing messages to appropriate receiver 2) Maintaining each individual system access control and 3) Translating received messages. Facilitator (F) itself has a router, which handles the traffic function of incoming and outgoing messages[7]. Critical architectural decisions (e.g.; Facilitator) allowed us to experiment more widely.

All agents under a system / facilitator is aware of all other agents under that system / facilitator. The default value of timing in ZEUS has been considered for sending or receiving messages.

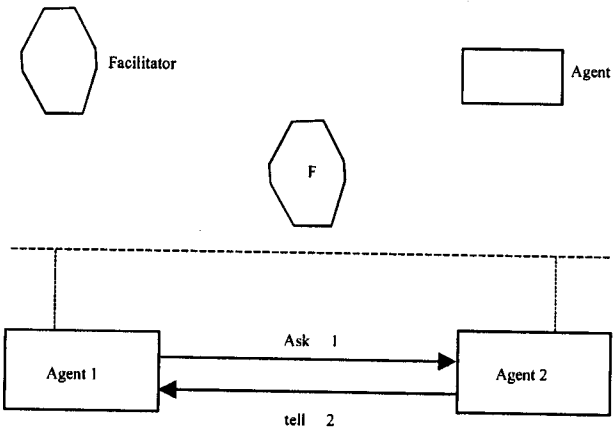
Implementation of KQML: All agent under a system/facilitators is aware of all otehr agent under that system / facilitators. The defult value of timing in ZEUS has been considered for sending or receiving messages.

KQML for Internal message

Active internal message: Communication of Agent 1 and agent 2 can be shown as follows:

Step 1. Incident way 1:
(ask
: sender Agent 1
: receiver Agent 2
:language KIF
:ontology : Design Consistency Checking
:content <expression>)

Step 2. Reflection way 2:
(tell
: sender Agent 2
: receiver Agent 1
:language KIF
:ontology : Design Consistency Checking
:content <expression>)

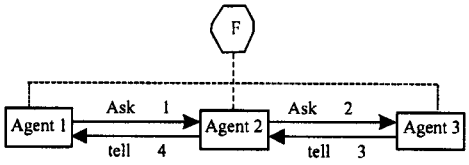


Passive internal message: Agent 1 sends a request to agent 2, but agent 2 forwards the query to agent 3 because agent 2 is unable to fulfill the query.

Step 1. Incident way, 1 and 2
(forward
: from Agent 1
: to Agent 2
:sender Agent 1
:receiver Agent 3
:language KQML
:ontology: kqml-ontology
:content (ask
:sender Agent 1
:receiver Agent 2
:language KIF
:ontology: Design Consistency Checking
:content <expression>))

Step 2. Reflection way, **3** and **4**
(forward
: from Agent 3
: to Agent 2
:sender Agent 3

:receiver	Agent 1
:language	KQML
:ontology:	kqml-ontology
:content	(tell
:sender	Agent 3
:receiver	Agent 2
:language	KIF
:ontology:	Design Consistency Checking
:content	<expression>))



KQML for external message

Active external message: Communication of facilitator 1 and facilitator 2 can be shown as follows:

Step 1. Incident way , 1

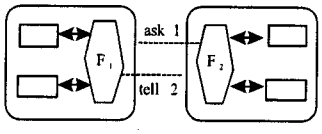
(ask

: sender	Facilitator 1
: receiver	Facilitator 2
:language	KIF
:ontology :	Design Consistency Checking
:content	<expression>)

Step 2. Reflection way, 2

(tell

: sender	Facilitator 2
: receiver	Facilitator 1
:language	KIF
:ontology :	Design Consistency Checking
:content	<expression>)



Passive internal message: Facilitator 1 sends a request to facilitator 2, but facilitator 2 forwards the query to facilitator 3 because agent 2 is unable to fulfill the query.

Step 1. Incident way, 1 and 2

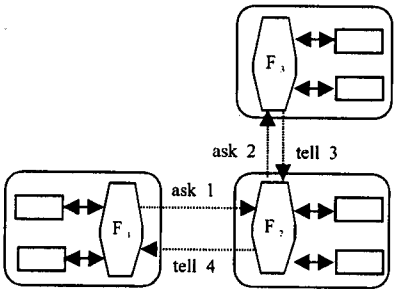
(forward

: from	facilitator 1
: to	facilitator 2
:sender	facilitator 1
:receiver	facilitator 3
:language	KQML
:ontology:	kqml-ontology
:content	(ask
:sender	facilitator 1
:receiver	facilitator 2
:language	KIF
:ontology :	Design Consistency Checking
:content	<expression>))

Step 2. Reflection way, 3 and 4

(forward

: from	facilitator 3
: to	facilitator 2
:sender	facilitator 3
:receiver	facilitator 1
:language	KQML
:ontology :	kqml-ontology
:content	(tell
:sender	facilitator 3
:receiver	facilitator 2
:language	KIF
:ontology :	Design Consistency Checking
:content	<expression>))



KQML for Broadcast / Multicast Message

Broadcast/Multicast message: Facilitators are considered to be responsible for multicast message here. For example, Facilitator 1 has sent message to facilitator 2 and has got No reply. Facilitator 1 has sent message to facilitator 3. One of the agents whoare associated with facilitator 3 replies,Yes. Therefore, facilitator 3 repliesYes to facilitator 1.

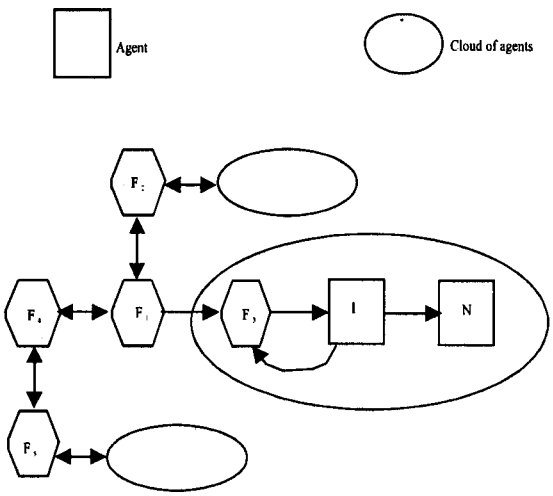
Step 1. Incident way

(ask-all

:sender	Facilitator 1
:receiver	Facilitator 2
:in – reply – to	logic 1
:reply – with	logic 2
:language	KIF
:ontology :	Design Consistency Checking
:content	<expression>)

Step 2. Incident way
(forward

:from	Agent 2
:to	facilitator 3
:sender	Agent 2
:receiver	facilitator 1
:language	KQML
:ontology :	kqml-ontology
:content	(tell
:sender	Agent 2
:receiver	facilitator 3
:in–reply–to	logic 1
:language	KIF
:ontology :	Design Consistency Checking
:content	<expression>))



DISCUSSION

Intelligent Distributed Mismatch Control (IDMC) is an advanced development for design consistency checking in distributed environment[3] after Intelligent Mismatch Control System (IMCS)[2]. Communication becomes important in IDMC approach for sharing knowledge and for delivering the information of mismatches among partners. Communication existed in IDMC approach, but not in a structured manner. Therefore, KQML and FIPA ACL are proposed byTara Toukhine [3] for Agent CommunicationLanguage (ACL).

KQML is considered for intelligent agent communication in the context of Aerospace design as a step forward from IDMC approach to IDMC system. Three types of message are considered for multi-agent communication. ZEUS agent toolkit is considered as a platform facilitating communication among agents.

CONCLUSIONS

This study introduced the use of KQML for communication of agents on behalf of IDMC approach. Communication is important to ensure cooperation among agents. The communication mechanism is based on KQML structures. Intelligent agent co-ordination to help communication to step forward IDMC approach to IDMC system should be investigated in future.

ACKNOWLEDGEMENT

I would like to thank *Dr. Kamal Bechkoum* for his helpful discussions.

REFERENCES

1. Fujita, K. and S. Akagi, 1999. Agent-based distributed design system architecture for basic ship design. Concurrent engineering: Research and Applications, 7: 83-93.
2. Bechkoum, K., 1997. Intelligent electronic mock-up for concurrent design Expert Systems with Applications J., 12: 21-36.
3. Taratoukhine, V., 2002b. A multi-agent approach for design consistency checking, Ph.D Thesis, de Montfort University, UK.
4. Bechkoum, K., 2004. Advance computer science, M.sc. Lecture Note, Sem 2, SCIT, University of Wolverhampton, UK.
5. Weiss, G., 1999. Multiagent Systems: A Modern Approach to distributed artificial intelligence, Mit Press. London .
6. Covington, A.M., 1998. Speech acts, electronic commerce and KQML. Elsevier Science, 22: 203-211.
7. Finin, T., R. Fritzson, D. Mckay and R. Mcentire, 1994. KQML as an agent communication language. ACM Press.