Appl Intell (2016) 44:1–16
DOI 10.1007/s10489-015-0690-x

Semantic web technology for agent interoperability: a proposed infrastructure

Fang-Ping Pai[2] · I-Ching Hsu[1] · Yeh-Ching Chung[2]

Published online: 9 July 2015

Abstract In recent studies, ontology related concepts have been introduced into FIPA ACL content language to convey information for agent communication. However, these works have only applied ontology-based knowledge representation in communication message and then demonstrated the advantage of this association. In fact, although ontology can represent semantic implications needed for decidable reasoning support, it has no mechanism for defining complex rule-based representation to support inference. The motivation of this study is to address this issue by developing a semantic-based infrastructure to integrate Semantic Web technologies into ACL message contents. This semantic-based infrastructure defines two different semantic frameworks: the three-tier knowledge representation framework for message content and the Multi-layer Ontology Architecture for content language. The former is developed based on Semantic Web stack to support ontology-based reasoning and rule-based inference. The latter is adopted to develop a Lightweight Ontology-based Content Language (LOCL) to describe agent communication messages in an unambiguous and computer-interpretable way Jena reasoner is used in an application scenario that exploits agent communication with LOCL as content language, OWL as ontology language, and SWRL as rule language to demonstrate the feasibility of the proposed infrastructure.

Keywords ACL · Content language · Semantic web · MAS · Ontology

✉ I-Ching Hsu
hsuic@nfu.edu.tw

1 Department of Computer Science and Information Engineering, National Formosa University, 64, Wenhua Rd., Huwei Township, Yunlin 632, Taiwan

2 Department of Computer Science, National Tsing Hua University, 101, Section 2, Kuang-Fu Road, Hsinchu 300, Taiwan

1 Introduction

Agent communication language (ACL) provides language primitives based on speech act theory to facilitate the communicative interoperability among agents. Knowledge Query and Manipulation Language (KQML) [39] and Foundations for Intelligent Physical Agents ACL (FIPAACL) [13] are the two most typical ACLs. ACL defines a set of performatives (also called Communicative Acts), their meanings and the standard message structure. ACL can further be viewed as a wrapper language independent of content language and ontology specification. The content language is used to describe the content embedded in ACL messages, which allows agents to share information from their knowledge base in order to interact autonomously and intelligently. For example, the agent communication message following FIPA-ACL specification includes parameters like communication act name, sender, receiver, content, language, and ontology. Propositions, actions and objects (identifying reference expression, IRE) would be embedded in the content field of an ACL message. Actually, the FIPA-ACL should be considered as a protocol suite rather than a single declarative agent communication language. Besides the protocol for communication acts and interaction process, FIPA-ACL also includes protocols for content language and content ontology.

The Semantic Web [4, 36] is introduced by Tim Berners-Lee as "an extension of the current web in which information is given well defined meaning, better enabling computers and people to work in cooperation". Many studies [20–22, 28, 32, 33] have adopted Semantic Web technologies to build intelligent applications in various domains. The Semantic Web technologies, based on XML standard, include ontology markup language (such as RDF [27], RDFS [7] , DAML and OWL [30]), and rule markup language (such as SWRL [19] and RuleML [34]). RDF and RDF Schema are languages for describing resources and their types, and thus are building blocks for the Semantic Web. OWL, built upon RDF by including a substantial fragment of RDF-Schema, is essentially an XML encoding of expressive Description Logic. Compared with RDF and RDFS, OWL has more facilities for expressing meaning and semantics such that OWL surpasses these languages in its ability to represent machine-readable content on the Web. Unfortunately, these ontology markup languages are insufficient to describe the conditions under which specific relations might hold, which requires the explicit representation of implications, as is provided by logic programs, such as rules. A well-accepted consensus has evolved in the Semantic Web community that the scope of the Semantic Web must include the mechanism to handle rules as well as ontology.

This work exploits Semantic Web technologies to address two issues relevant to the content language of FIPA ACL. The first one is about how to enhance the intelligence of content language. FIPA has proposed Semantic Language (SL) [16] to be used in conjunction with FIPA ACL as the language for the content field of a message. The main problem of FIPA SL is that it is defined based on a sub-grammar of the very general s-expression syntax which lays stress on syntax and format rather than knowledge representation. Hence, FIPA SL exhibits the advantage of communication message association, but it still lacks computer-interpretability to support intelligent inference. Semantic Web technologies can provide catalytic solutions to this problem In this work a novel Lightweight Ontology-based Content Language (LOCL) adopting Semantic Web technologies is proposed to provide computer-interpretability for the message content.

The second issue is about how to enhance the level of intelligence of message contents. The FIPA SL does not provide semantic expression mechanism for the terms and operators in message contents. Traditional ACL content representation is useful for information exchanging and organizing however, it also lacks the computer-interpretability needed to facilitate intelligent inference Ontology defines a common vocabulary and formal semantics for communication among agents, can provide semantics of concepts used not only in the content language, but also in the message content. Although ontology can represent semantic implications needed to support decidable reasoning, it still has no mechanism for defining arbitrary, multi-element antecedents. For instance, description logics of ontology cannot represent complex nonmonotonic rules. To resolve this problem, this work applies Semantic Web technologies to develop a three-tiers knowledge representation framework that supports ontology-based reasoning and rule-based inference to enhance interoperability among agents.

The primary goal of this study is to reinforce knowledge representation capability provided by conventional FIPA ACL and resolve the two issues described above. Two main contributions are made: First, Semantic Web technologies are integrated into ACL content language to provide a semantic-based infrastructure to support the interoperability of agents. Two different semantic frameworks are defined in this semantic-based infrastructure, i.e. Multi-layer Ontology Architecture and a three-tier knowledge representation framework. The proposed infrastructure attempts to integrate the three fast-developing research areas: Semantic Web Agent, and ACL content language Second, we propose the LOCL not only allows seamless integration with heterogeneous ontology policies for various application domains, it also combines SWRL to enhance the reasoning capability of ACL message content. An application scenario is presented to demonstrate the intelligent agent communication using the proposed architecture. The demonstration exploits LOCL as content language, OWL as ontology language, SWRL as rule language and Jena [23] as reasoner.

The remainder of this paper is organized as follows: The next section presents a survey of related works. The multi-layer ontology architecture is introduced in Section 3, and then the Lightweight Ontology-based Content Language (LOCL) based on this architecture is defined. A workflow-oriented model to describe the agent communication with LOCL is presented in Section 4. In Section 5, a concrete example to illustrate how Instance Base, Ontology Base and Rule Base can be developed based on the semantic. Web is presented in Section 6, a demonstration is presented to illustrate how LOCL can be adopted to describe content messages for agent communication. In the final section, summary and concluding remarks are included.

2 Related works

A comprehensive comparison of research works related to our approach is presented in this section The comparisons are classified into two parts according to the two FIPA-ACL issues mentioned previously (i.e., content language and content message).

Table 1 Comparison of existing content language and the proposed content language using ontological approach

Reference	Use RDF	Use RDF Schema	Use DAML	Use OWL	Use SWRL	Support complex nonmonotonic rules
[15]	V	V	X	X	X	X
[44]	V	V	X	V	X	X
[45]	V	X	V	X	X	X
[5]	V	V	X	V	X	X
[31]	V	V	X	V	X	X
[17]	V	V	X	V	X	X
This study(LOCL)	V	V	X	V	V	V

Although very expressive, the FIPA Semantic Language (FIPA SL) is still unnecessarily powerful for some agent communication tasks. There exists two major drawbacks: first, it would be a very complex task to develop a reasoner for the message contents expressed using FIPA-SL. Second, FIPA-SL has only limited internet interoperability since it is not based on Web-based technologies. Although the Knowledge Interchange Format (KIF) [14] can translate one content language to another and provides a common content language for two agents using different native content languages, KIF has the same major limitation as FIPA SL, that is , neither a generic inference engine nor a public API supports KIF existed.

Several studies [5, 15, 17, 31, 44, 45] have focused on using ontology technologies, such as RDF, RDF Schema, DAML and OWL to develop a FIPA ACL content language. Other significant researches [1, 3, 8–12, 35, 37, 38, 40–43] try to use the ontology technologies in ACL message contents to provide specific domain knowledge and facilitate agent interoperability. Tables 1 and 2 present

Table 2 Comparison of existing message content and proposed message content using the ontological approach

Reference	Use Ontology Language	Use Rule Language	Support complex nonmonotonic rules	Domain Ontology name	Application Domain
[1]	OWL	X	X	QAS	question-answer system
[12]	OWL	X	X	process	cloud service
[37]	Formal Language	X	X	context	context-awareness
[42]	OWL	X	X	energy-saving taxonomy	energy-saving application
[43]	Formal Language	X	X	finance	financial application
[35]	OWL	X	X	power transformer	power transformer fault diagnosis
[10]	Formal Language	X	X	agent crawler	information retrieve
[41]	OWL	X	X	buyer, seller	virtual enterprise
[38]	OWL	X	X	rehabilitation	medical rehabilitation
[40]	RDF Schema	X	X	Animal	data mining
[8]	Formal Language	X	X	personal knowledge	search system
[3]	OWL-S	X	X	cloud service taxonomy	cloud service
[11]	OWL	X	X	e-learning	e-learning application
[9]	OWL	X	X	CoreGRID	grid computing
This study (Three-tier knowledge representation)	OWL	SWRL	V	allows the user to customize any domain ontology	any domain

comparisons about the technologies adopted in these studies with respect to the two mentioned issues. It should be noted that all these studies limited their scope to ontology-based knowledge representation only. How to exploit the capability of Ontology to represent semantic implications needed to support decidable reasoning and how to introduce a mechanism for defining arbitrary, multi-element antecedents to represent complex non-monotonic rules are never considered.

The FIPA-RDF specification [15] describes how the RDF can be used as content language in an ACL message and shows how RDF schemas can be defined to provide modular RDF extensions. Those extensions will be able to handle example rules, logic algebra constructs, and others. The major advantage of using RDF for FIPA ACL content language is that data exchange and schema reuse can be achieved in a simple way. The major disadvantage of RDF and RDF Schema is that they are limited in expressing meanings and semantics. In [5], OWL is adopted for FIPA ACL-compliant messages, which requires separation of content semantics from ACL semantics. One limitation of this approach is that only propositions or referential expressions can be used in the ACL message content. TAGA [44] also adopts RDF and OWL to specify the domain ontology. The agents use OWL as the content language for exchanging information and knowledge within ACL messages. Laclavik et al. [31] describe how semantic web technologies can be used in Multi-agent System (MAS) by applying OWL and SPARQL as content languages. Fornara et al. [17] propose an approach that uses the ontology language OWL 2 DL to represent ACL message content and to develop ACL content language.

This study exploits Semantic Web technologies to construct Multi-layer Ontology Architecture for the purpose of improving the level of intelligence of content language. The proposed Multi-layer Ontology Architecture covers Meta layer, Language layer, Ontology layer and Instance layer. This architecture will enable the proposed content language to provide the capabilities to describe complete semantics and complex nonmonotonic rules, and therefore can support ontology-based reasoning and rule-based inference in a consistent way.

In recent works that introduce ontologies to describe real-world domain knowledge and then to facilitate agent interoperability most of them such as [1, 9, 11, 12, 35, 38, 41, 42], use OWL as the semantic markup language for publishing and sharing ontologies in their frameworks or applications. In SACoSS(Semantic Agent Based System for Cloud Service) [3], semantic agents use a cloud service ontology, based on OWL-S [29], to extract the knowledge about the cloud service and produce a list of SaaS level and IaaS level cloud services as suggestion according to the consumer requirement intelligently.

In [8, 10, 37, 43], formal language was used to define an abstract ontology to describe domain knowledge. Summaries of these works are illustrated in Table 2, it reveals that OWL is the most popular ontology language used to develop domain knowledge, on the other hand, these studies still lacked to use rule-based language for knowledge representation. As mentioned earlier, this study applies Semantic Web technologies to generate a three-tiers knowledge representation framework for integrating OWL-based ontology with OWL-based rule (i.e. SWRL). The proposed three-tiers knowledge representation framework is built according to Semantic Web stack that includes XML layer, Ontology layer, and Rule layer to support ontology-based reasoning and rule-based inference.

3 Lightweight ontology-based content language

In this section, we'll first introduce the Multi-layer Ontology Architecture corresponding to the semantic layer (i.e., ontology layer) of traditional Semantic Web stack and then defines a Lightweight Ontology-based Content Language (LOCL) based on the Multi-layer Ontology Architecture as content language for ACL.

3.1 Multi-layer ontology architecture

An ontology is commonly defined as an explicit, formal specification of a shared conceptualization of a domain of interest. The compositions of an ontology include a set of concepts, a set of properties, and the relationships between the elements of these two sets. It has been widely accepted that ontology and metadata are the core elements for the Semantic Web. The Multi-layer Ontology Architecture is composed of four layers: Meta Layer, Language Layer, Ontology Layer, and Instance Layer, as shown in Fig. 1. The multi-layer ontology framework is corresponding to the semantic layer of traditional Semantic Web stack. The meta layer contains rdfs:Class only. The rdfs:Class is the root class in the RDF data model. Any other class is regarded as an instance of the rdfs:Class. RDF, RDFS, DAML+OIL, and OWL are the candidates in the language layer. These languages are used to create domain ontologies, among them OWL has more facilities for expressing meaning and semantics than the others and thus can represent machine-readable content on the web better than other languages. In this study we will specially focus on the ontology layer. We proposed LOCL based on FIPA RDF content language and SWRL to realize agent communication messages. Realistic message entities reside in the instance layer, each represents an instance of the Request or Response class of the LOCL, which contains the metadata needed to annotate the agent communication message contents.

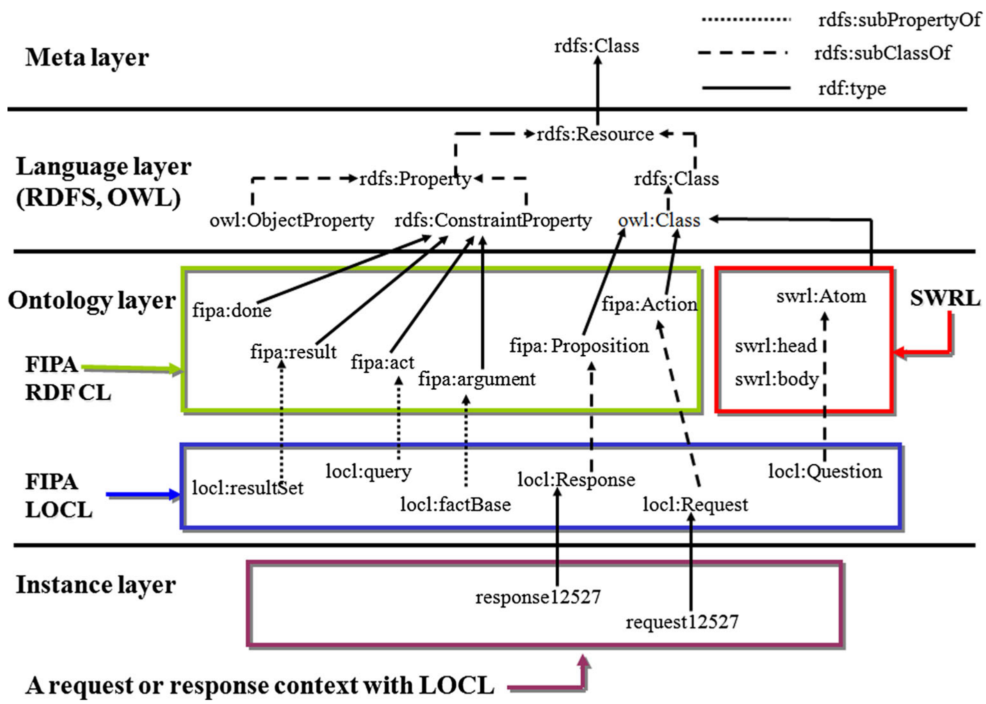


Fig. 1 Multi-layer Ontology Architecture

3.2 A lightweight content language based on FIPA RDF and SWRL

A key component of the ACL is the content language to annotate agent communication messages. LOCL is such an ontology-based content language based on FIPA RDF content language and SWRL that can realize agent communication messages unambiguously and in a computer-interpretable format. LOCL supports ACL to facilitate interoperability in agent communication through semantic descriptions. Figure 2 depicts a partial class semantic structure of LOCL.

In Fig. 2, locl, fipa, rdf, swrl, and owl are the namespaces for the LOCL, FIPA RDF content language, RDF, SWRL, and OWL, respectively. For example, locl:Request is subclass of fipa:Action, it inherit semantic knowledge from FIPA RDF content language to describe the messages of agent communication actions. An elaborated description of LOCL compositions is given in Table 3.

4 Three-tier semantic based knowledge representation

4.1 Three-tier knowledge representation framework

To address the content intelligence issue, a three-tiers knowledge representation framework to support semantic reasoning of agent communication is presented. Developed based on Semantic Web stack [24], this framework includes XML layer (OWL-based Instance Base), Ontology layer (OWL-based Domain Ontology Base), and Rule layer (SWRL-based Rule Base) as shown in Fig. 3. The XML layer is composed of Instance Bases those are XML-based documents for describing the real world information based on domain ontology. The Ontology layer provides OWL-based ontologies, which can describe a conceptualization of the specific domain. SWRL-based rules built on top of OWL-based ontologies that support more complex inferences than the Ontology layer are stored in the Rule layer. It should be emphasized that ontology is based on description logics to provide sound and decidable reasoning in contrast, the rule is a logic program which can complement ontology to support more complex rule-based inferences [24, 26].

4.2 Agents communication architecture

The core components of an intelligent system include the Agent, Instance Base, Rule Base, and Ontology Base. All agents denoted here exploit Jena inference engine as the OWL and Jena-based rule reasoner. The agents also have the ability to transform SWRL-based rules into Jena-based rules. The other components are defined as follows:

- **Ontology Base** is composed of OWL-based domain ontologies that represent conceptualization of specific domain for semantic reasoning. In agent communications, the domain ontology will be commonly

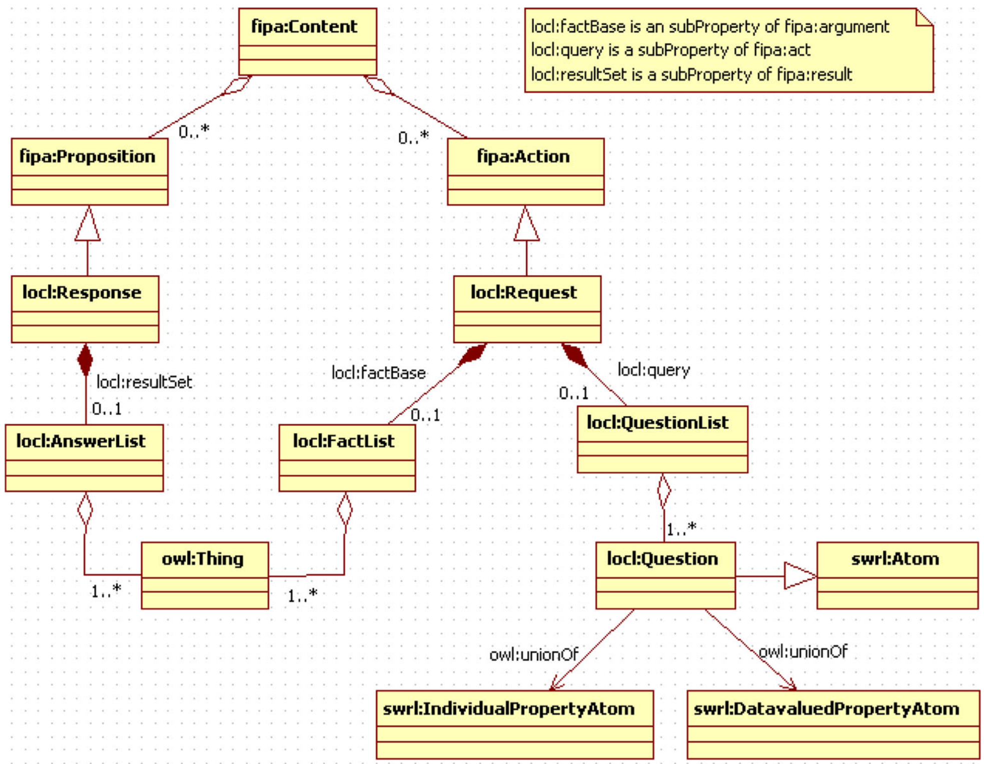


Fig. 2 Semantic structure of LOCL

understood by the agents to interpret the message expression. The Ontology Base is located at the Ontology layer of the three-tiers knowledge representation framework as shown in Fig. 3. Different intelligent systems will share the same ontology base.

- **Instance Base** is composed of class instances describing the real world information, which plays the same role as the fact base in a traditional expert system. An instance base is an RDF document containing a set of instances and relationship links. Furthermore, an instance base supports the principle of separating markup from contents, and allows flexible arrangement for inter-linked resources without having the user to edit. The Instance Base is located at the XML layer of the three-tiers knowledge representation framework, as shown in Fig. 3. Each intelligent system has its own dedicated instance base to record the facts that agent knows.
- **Rule Base** consists of SWRL-based rules to support a flexible and complex reasoning mechanism that is lacking in OWL-based ontologies. The Rule Base is located at the Rule layer of the three-tiers knowledge representation framework, as shown in Fig. 3. Each intelligent system has its own dedicated rule base to store the rules that agent accepts.

Figure 4 shows the flow-oriented agents communication architecture.

The information flow of the agent communication progresses as follows:

1. Agent A accesses the instance base to retrieve the relevant facts (class instances).
2. Agent A embeds these facts into the ACL message using LOCL to create a request, and then sends this request to invoke Agent B.
3. Agent B accesses the instance base based on the request to check instance consistency.
4. Agent B performs ontology-based reasoning with the following tasks:

   4.1 The class instances are extracted from the request.
   4.2 The OWL-based domain ontology is loaded into Jena reasoner.
   4.3 The Jena reasoner derives new facts from these class instances and domain ontology.

5. Agent B performs rule-based reasoning with the following tasks:

   5.1 The relevant ontology classes, mentioned in Step 4.2, are utilized to query the Rule Base to retrieve relevant SWRL-based rules.
   5.2 The SWRLTransform.xsl XSLT style sheet is employed to transform these SWRL-based rules into Jena-based rules.

Table 3 LOCL Components

Name	Type	Description	Source
Content	Class	Content is the root element in the FIPA RDF CL model. A content of ACL message may be an	RDF CL
action		or a proposition.	
Action	Class	Action may be a single action or a composite action built based on the sequencing and alternative operators. An Action is used as a content expression when the act is request and other communication acts derived from it.	RDF CL
Proposition	Class	Proposition may be assigned a truth value in a given context, which is a well-formed formula and used in the inform communicative act.	RDF CL
Request	Class	Request is a subclass of Action and is composed of FactList and QuestionList.	LOCL
FactList	Class	FactList is a subclass of rdf:List, which is an container and used to contain facts.	LOCL
factBase	Property	factBase is a subproperty of fipa:argument and has the attribute rdf:parseType="Collection" , which is used to keep the relation between Request and FactList.	LOCL
QuestionList	Class	QuestionList is a subclass of rdf:List, which is an container and used to contain Question.	LOCL
query	Property	query is a subproperty of fipa:act and has the attribute rdf:parseType="Collection", which is used to keep the relation between Request and QuestionList.	LOCL
Question	Class	Question is a subclass of swrl:Atom, which	LOCL
Atom	Class	Atom may refer to individual, data literal, individual variable or data variable.	SWRL
IndividualPropertyAtom	Class	IndividualPropertyAtom is a subclass of Atom, which is refer to a individual variable.	SWRL
DatavaluedPropertyAtom	Class	DatavaluedPropertyAtom is a subclass of Atom, which is refer to a data literal.	SWRL
Response	Class	Response is a subclass of Proposition, which is composed of AnswerList.	LOCL
AnswerList	Class	QuestionList is a subclass of rdf:List, which is an container and used to contain answers from service agents.	LOCL
resultSet	Property	resultSet is a subproperty of fipa:result and has the attribute rdf:parseType="Collection", which is used to keep the relation between Response and AnswerList.	LOCL

5.3 The rule-based new facts are inferred from these Jena-based rules.

6. Agent B combines the inference results generated from the ontology-based and rule-based approach to create a response with LOCL. The Agent B then sends the response to the Agent A.
7. Agent A extracts the class instances from the response and then executes ontology-based reasoning.

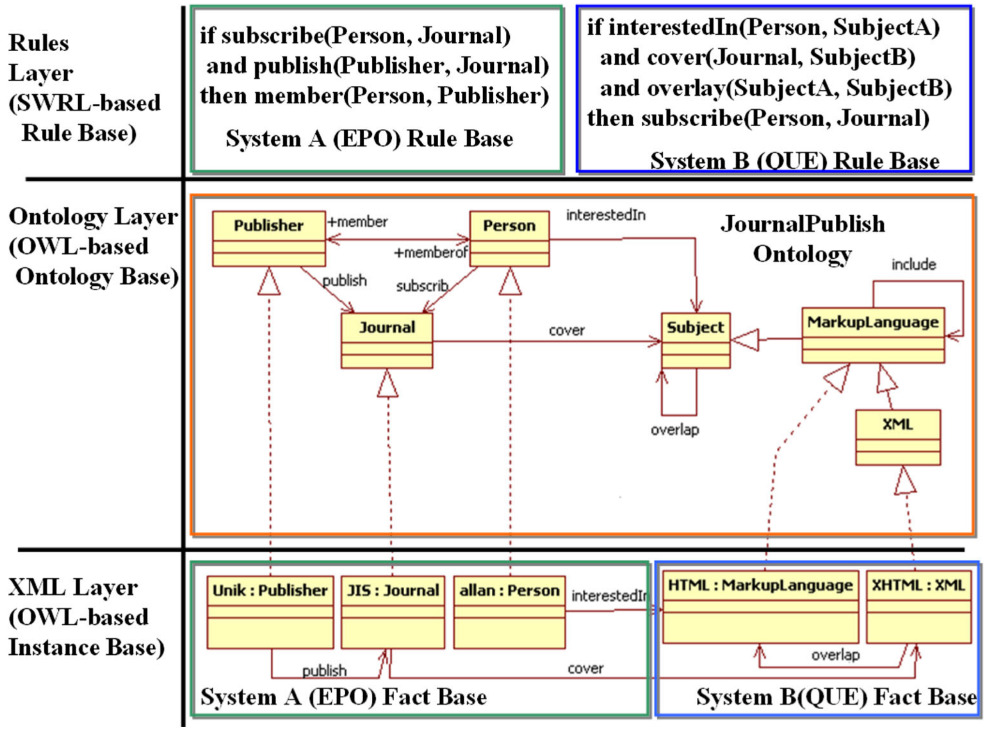


Fig. 3 Three-tier knowledge representation based on Semantic Web stack

8. Agent A invokes rule-based inference based on the existing facts to get the final inference results.

5 Domain knowledge development

Next, we will provide a concrete example about journal publishing domain to present how the three-tiers knowledge representation, including Instance Base, Ontology Base, and Rule Base, can be mapped into XML, ontology, and rule layer of Semantic Web stack, respectively. The example will also be used to illustrate the reasoning capabilities of agent communication.

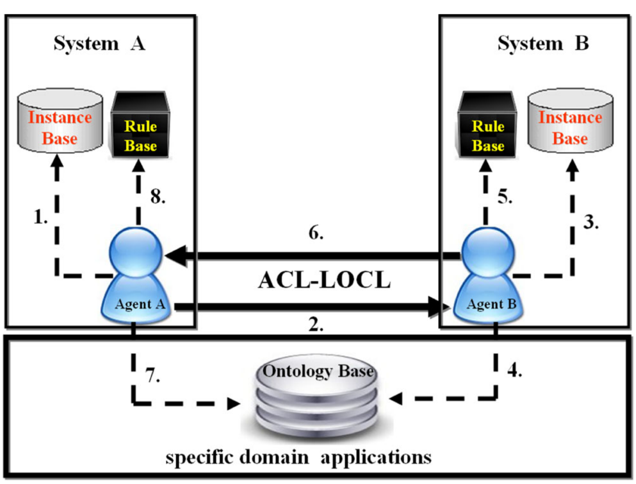


Fig. 4 The flow-oriented agent communication architecture with ACL LOCL

In the ontology layer, JournalPublish ontology is developed to provide semantic descriptions for the application domain. In the rule layer, we define some nonmonotonic rules encoded in SWRL format on top of the JournalPublish ontology These rules can be inferred to get new predications. Finally, we illustrate how the real world information in our application can be annotated using the JournalPublish ontology to construct an instance base.

5.1 Domain ontology

Ontology describes some application-relevant part of the world in a machine understandable way [6]. The core components of an OWL-based ontology comprise a set of classes, a set of properties and the relationships between the elements of these two sets. Classes are interpreted as sets of objects that represent the individuals in the domain of discourse. Properties are binary relations that link individuals, and are represented as sets of ordered pairs that are subsets of the cross product of the set of objects.

The JournalPublish ontology is used to capture the semantic-based knowledge in a generic way that provides a commonly agreed structure, such as Journal, Publisher, Person, Subject class, etc., involved in the application domain. The Ontology layer in Fig. 3 shows the semantic structure

Fig. 5 Partial code of SWRL-based rule in System B (QUE Rule Base)

```
<?xml version="1.0" encoding="UTF-8"?>
…..
<swrl:Variable rdf:ID="x1"/>
<swrl:Variable rdf:ID="x2"/>
<swrl:Variable rdf:ID="x3"/>
<ruleml:Imp rdf:ID="subscribe">
  <ruleml:body rdf:parseType="Collection">
    <swrl:IndividualPropertyAtom>
      <swrl:propertyPredicate rdf:resource="interestedIn"/>
      <swrl:argument1 rdf:resource="#x1" />
      <swrl:argument2 rdf:resource="#x2" />
    </swrl:IndividualPropertyAtom>
    <swrl:IndividualPropertyAtom>
      <swrl:propertyPredicate rdf:resource="cover"/>
      <swrl:argument1 rdf:resource="#x3" />
      <swrl:argument2 rdf:resource="#x4" />
    </swrl:IndividualPropertyAtom>
    <swrl:IndividualPropertyAtom>
      <swrl:propertyPredicate rdf:resource="overlap"/>
      <swrl:argument1 rdf:resource="#x2" />
      <swrl:argument2 rdf:resource="#x4" />
    </swrl:IndividualPropertyAtom>
  </ruleml:body>
  <ruleml:head rdf:parseType="Collection">
    <swrl:IndividualPropertyAtom>
      <swrl:propertyPredicate rdf:resource="subscribe"/>
      <swrl:argument1 rdf:resource="#x1" />
      <swrl:argument2 rdf:resource="#x3" />
    </swrl:IndividualPropertyAtom>
  </ruleml:head>
</ruleml:Imp>
…..
```

of JournalPublish ontology as UML class diagram. This class diagram provides a graphical overview of the domain concepts and their relationships

The case scenario in this study involves System A (EPO) a journal publisher company and System B (QUE), a journal survey company. The EPO and QUE companies are in the same journal publishing domain and share the common JournalPublish ontology. However, because they have different business properties, the EPO and QUE require a dedicated rule base and a dedicated instance base, respectively.

The following four constraints present some partial code of the JournalPublish ontology to illustrate OWL-based description logics including subclass, symmetric property, transitive property, and inverse property, respectively.

Constraints 1. Subclass

OWL expression :

<owl:Class rdf:ID=”XML”>
<rdfs:subClassOf rdf:resource=”#MarkupLanguage” />
</owl:Class>

Semantic meaning : The XML is a subclass of the MarkupLanguage.

Rule expression :

if XML(x) then MarkupLanguage(x)

Table 4 The facts in EPO Instance Base

Type	Instance ID	Tag Name	From	To
Class	allan	Person		
Class	JIS	Journal		
Class	Unik	Publisher		
Property		publish	Unik	JIS
Property		interestedIn	allan	http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#HTML,
Property		cover	JIS	http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#XHTML

Table 5 The facts in QUE Instance Base

Type	Instance ID	Tag Name	From	To
Class	HTML	MarkupLanguage		
Class	XHTML	XML		
Property	overlap		XHTML	HTML

Constraints 2. Symmetric property
OWL expression :
<owl:SymmetricProperty rdf:ID=”overlap”>
<rdfs:domain rdf:resource=”#Subject”/>
<rdfs:range rdf:resource=”#Subject”/>
</owl:SymmetricProperty>
Semantic meaning :
The overlap relation is symmetric.
Rule expression :
if overlap(x,y) then overlap(y,x)

Constraints 3. Transitive property
OWL expression :
<owl:TransitiveProperty rdf:ID=”include” />
<rdfs:domain rdf:resource=”#MarkupLanguage” />
<rdfs:range rdf:resource=”#MarkupLanguage” />
</owl:TransitiveProperty>
Semantic meaning :
if x include y, and y include z then x include z.
Rule expression :
if include(x, y) and include(y, z) then include(x, z)

Constraints 4. Inverse property
OWL expression :
<owl:ObjectProperty rdf:ID=”member”>
<owl:inverseOf rdf:resource=”#memberOf”/>
<rdfs:domain rdf:resource=”#Publisher”/>
<rdfs:range rdf:resource=”#Person”/>
</owl:ObjectProperty>
Semantic meaning :
There is an inverse relation between member and memberOf.
Rule expression :
if member(x,y) then memberOf(y,x)

5.2 Semantic web rule base

The following practices have been performed to integrate the OWL-based ontology with SWRL-based rules SWRL model is used to represent Horn clauses rules. An OWL class is treated as a unary predicate and OWL property is treated as a binary predicate. Assertions about instances in a class are treated as rule atoms (e.g., facts) in which the class predicate appears. Assertions about property links between class instances are treated as rule atoms in which the property predicate appears. The SWRL allows a predicate symbol to be an URI; this capability is used significantly herein, since the names of OWL classes are URIs.

Fig. 6 The request message using FIPA ACL combined with LOCL content

```
(QUERY-REF
:sender EPO
:receiver QUE
:content (
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:sp="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/dosp.owl#"
         xmlns:swr="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/doswr.owl#"
         xmlns:locl="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/locl.owl#"
         xmlns="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/dosp.owl#">
<locl:Request rdf:ID="request12527">
 <locl:factbase rdf:parseType="Collection">
 <Person rdf:ID="allan">
  <interestedIn
   rdf:resource="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#HTML"/>
 </Person>
 <Journal rdf:ID="JIS" >
  <cover
   rdf:resource="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#XHTML" />
 </Journal>
 </locl:factbase>
 <locl:Query rdf:parseType="Collection">
  <locl:question
   rdf:resource="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BRuleBase.rdf#subscribe" />
  </locl:Query>
  </locl:Request>
</rdf:RDF>)
:language LOCL
:ontology http://sparc.nfu.edu.tw/~hsuic/sw/ontology/dosp.owl
…..
```

The QUE concerns about the subscription and subject issues, while the EPO concerns about the membership issues. The following two inference rules, expressed in non-monotonic format, are defined by EPO and QUE to represent their domain knowledge respectively.

Rule-1 : If a person subscribes a certain journal that is published by a certain publisher, then the person is a member of the publisher.
if subscribe(Person, Journal) and publish(Publisher, Journal)then member(Person, Publisher)

Rule-2 : If a person is interested in a certain subject that is covered by a certain journal, then the person subscribes the journal.
if interestedIn(Person, Subject A) and cover(Journal, Subject B) and overlap(Subject A, Subject B) then subscribe(Person, Journal)

Figure 5 depicts a SWRL-based rule for the non-monotonic Rule-2 mentioned above. This rule requires the representation of complex implications, a capability beyond the semantic implications ability supported by OWL. In this example, it is demonstrated that SWRL can not only provide general implication in the form of Horn clauses but that its XML encoding form makes it the ideal choice for correlated use with OWL. All the classes and properties of the previously defined JournalPublish ontology can be used as elements in the SWRL-based rules.

5.3 Instance base

In the XML layer, EPO Instance Base and QUE Instance Base are created to contain real world information based on the JournalPublish ontology depicted in Fig. 3. The facts in the EPO Instance Base are summarized in Table 4. Each row represents a concept associated with a semantic clue that implies class inheritance or property relationship. Similarly, the facts in QUE Instance Base are summarized in Table 5.

6 Experimental case of ACL with LOCL

The following experimental scenario explicitly demonstrates how LOCL can be adopted to describe the content of agent communication messages produced in the application domain given in Section 5. In this scenario, EPO is a journal publisher company and QUE is a journal survey company. EPO concerns about the membership issues defined according to Rule-1 in Section 5.1 whereas the QUE concerns the subscription issue defined according to Rule-2 in Section 5.1. The agent communication process follows

Fig. 7 The response message using FIPA ACL combined with LOCL content

```
(INFORM
:sender QUE
:receiver EPO
:content (
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:sp="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/dosp.owl#"
         xmlns:swr="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/doswr.owl#"
         xmlns:locl="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/locl.owl#"
         xmlns="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/dosp.owl#">
  <locl:Response rdf:ID="response12527">
   <locl:resultSet rdf:parseType="Collection">
   <XML rdf:about="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#XHTML" />
   <MarkupLanguage rdf:about="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#XHTML" />
   <Subject rdf:about="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#XHTML" />
   <MarkupLanguage rdf:about="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#HTML" />
   <Subject rdf:about="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/BFactBase.rdf#HTML" />
   <Person rdf:about="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/AFactBase.rdf#allan">
      <subscribe rdf:resource="http://sparc.nfu.edu.tw/~hsuic/sw/ontology/AFactBase.rdf#JIS"/>
  </Person>
  <locl:resultSet>
  </locl:Response>
</rdf:RDF>)
:language LOCL
:ontology http://sparc.nfu.edu.tw/~hsuic/sw/ontology/dosp.owl
.....
```

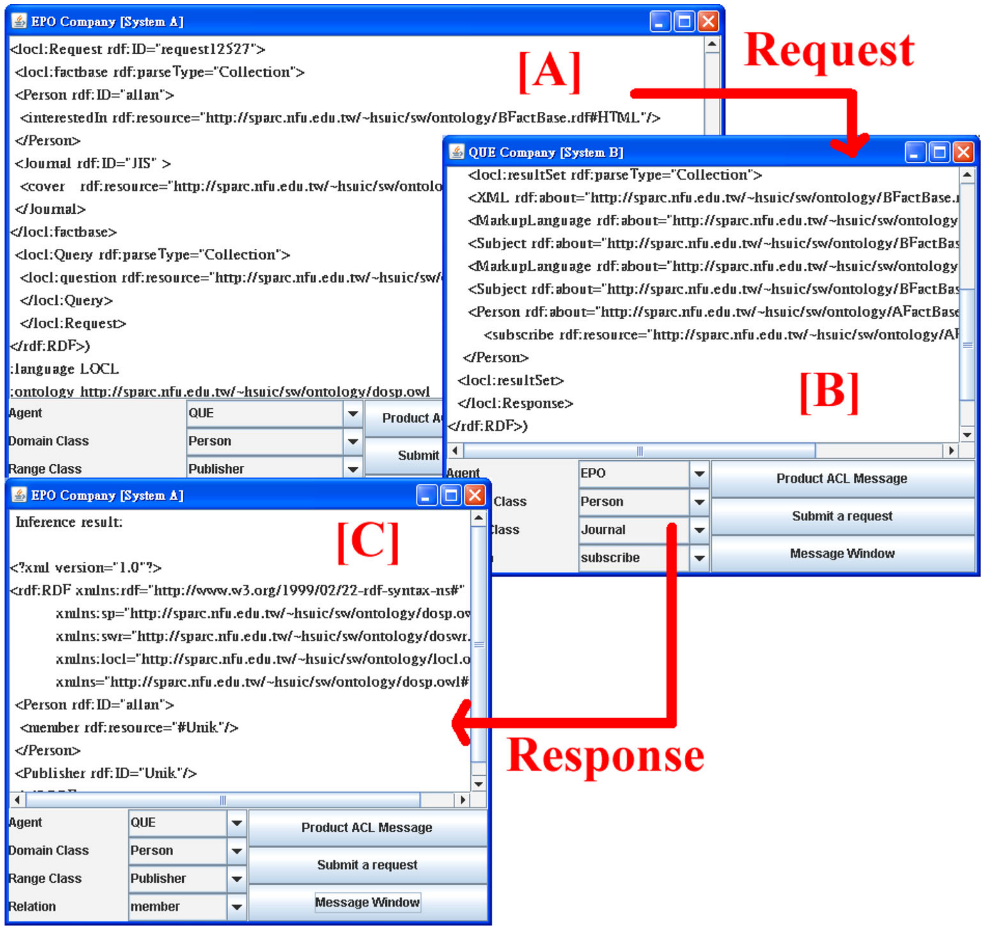


Fig. 8 Example of the communications between EPO agent and QUE agent

the workflow described in Section 4.2. To predicate whether a person is a member of a publisher, EPO must first send all its known facts to QUE with a QUERY-REF message so that QUE can determine the subscription status with these facts. QUE first performs ontology-based reasoning and then rule-based inference. It then sends the result about subscription to EPO with an INFORM message.

6.1 QUERY-REF message using LOCL

In case the EPO wants to predict whether person "allan" is a member of publisher "Unik" it will first access the instance base to retrieve the relevant facts about "allan" (see row 1, 2, 5 and 6 of Table 4)and then according to the FIPA ACL format, integrates these facts into the content after lcol:factbase

Table 6 Rules list

Rule number	Type	Rule expression
Rule-1	ontology(subclass)	if XML(x) then MarkupLanguage (x)
Rule-2	ontology(subclass)	if MarkupLanguage(x) then Subject(x)
Rule-3	ontology(symmetric)	if overlap(x, y) then overlap(y, x)
Rule-4	ontology(transitive)	if include(x, y) and include(y, z) then include(x, z)
Rule-5	ontology(inverse)	if member(x, y) then memberOf(y, x)
Rule-6	rule	if subscribe(Person, Journal) and publish(Publisher, Journal) then member(Person, Publisher)
Rule-7	rule	if interestedIn(Person, Subject A) and cover(Journal, Subject B) and overlap(Subject A, Subject B) then subscribe(Person, Journal)

Table 7 Test results

	Instance base query	Ontology-based reasoning	Rule-based inference
numbers	120 instances	262 relationships	2 policy rules
Times (ms)	101	446	248

tag. According to Rule-1, EPO needs the subscription information to make the membership predicate, so EPO should add the query about subscription status into the content after locl:Query tag. The QUERY-REF message is composed of locl:factbase and locl:Query (see Fig. 6).

6.2 INFORM message using LOCL

QUE executes the ontology-based reasoning and rule-based inference in order. Fig. 3 shows how the JournalPublish ontology can be referred by instances within the EPO and QUE instance base. The QUE first extracts the content of locl:factbase to retrieve the facts, and then performs the following ontology-based reasoning dependent upon the semantics of the JournalPublish ontology.

1. The HTML is an instance of the MarkupLanguage class (see row 1 of Table 5).
2. MarkupLanguage class is a subclass of Subject class.
3. Based on the above semantics, QUE infers that HTML is also an instance of Subject class and the interestedIn property from “allan” to “HTML” is validated.
4. The XHTML is an instance of the XML class (see row 2 of Table 5).XML is a subclass of MarkupLanguage class and MarkupLanguage is a subclass of Subject class.
5. Based on the above semantics, QUE infers that XHTML is also an instance of Subject class and the cover property from “JIS” to “XHTML” is validated.

QUE based on the above facts and Rule-2 (see Section 5.2) to infer the new fact: person “allan” subscribe the journal “JIS”. Finally, QUE combines the results from the ontology-based reasoning and rule-based inference into a INFORM message, shown as Fig. 7. QUE then reply to EPO with this INFORM message

Figure 8 shows the communications between EPO agent and QUE agent. The EPO agent accesses the fact base to product an ACL LOCL message (A), which is then sent to the QUE to invoke an inference request. The inference results from the QUE agent (B) are coded in LOCL and sent to the EPO agent. Finally, the EPO infers *member(allan, Unik)* according to the original fact, *publish(Unik, JIS)*, and the new fact *subscribe(allan, JIS)*, and Rule-1(see Section 5.2). The EPO agent inference results are shown in (C).

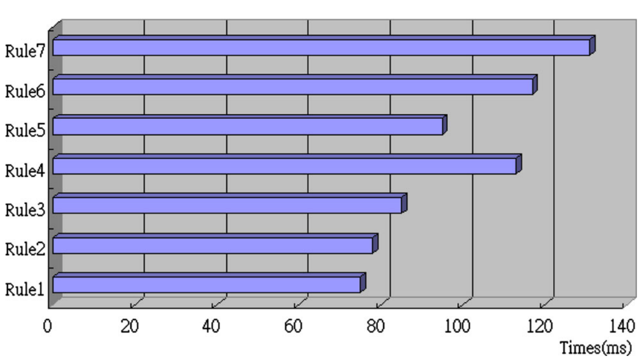


Fig. 9 The execution times of each rule

6.3 Inference performance evaluation

This section presents a preliminary experimental study to evaluate the inference performance of LOCL for agent communication. The test dataset contains 120 instances distributed in different classes of the JournalPublish ontology. The test database included 262 relationships annotated among those instances. Seven rules were used to infer for the relevant instances. The complete list of rules can be found in Table 6. The first five rules are ontology-based reasoning, and the first two rules do not directly support to produce instances but can be referred by other rules. The last two rules are rule-based inference.

The experimental conditions are described as follows:

1. Experiments were performed on a 2.26GHz Intel Xeon (Quad-Core) PC with 4G of RAM, running Windows 2008 Server.
2. All instances were processed in random and one by one.
3. After an instance has been processed, the new inference facts could be kept in the memory and the running times could be added up for each rule that is triggered in the inference.

Total run time was 0.8 seconds. The search agent executed only one search of the instance base before extracting the relevant information to get the facts, which was completed in only 0.1 seconds. Additionally, the remaining run time was spent in inferring to get new facts, including ontology-based reasoning and rule-based inference. The summary of test results is shown in Table 7.

Figure 9 shows the execution time for the inference for each rule. The experimental results show that the execution

Table 8 The execution times depend on the number of instances

Instance Number	Relationship Number	Instance Base Query Time(ms)	Ontology Time(ms)	Rule Time(ms)	Total Time(ms)
500	1200	121	721	456	1298
1000	2550	128	793	593	1513
1500	4200	137	920	830	1886
2000	5700	149	1104	1315	2605
2500	7250	164	1546	2314	4023
3000	8850	181	2628	4235	7042

time grows with the complexity of the rule. The inference time for rule 4 increases as the result of the addition of a transitive property. The inference agent must execute a complicated recursive function to derive the transitive result. Compared to unary predicates, the binary predicates such as rule3, rule 4, rule 5, rule 6, and rule 7 have longer inference times. The last two rules have longer inference times due to numerous clauses and binary predicates.

To evaluate how the inference performance of agent communication varies with different dataset scale, this study extends the JournalPublish ontology to cover more datasets. Six datasets were used to process 500 to 3000 instances in increments of 500. The number of policy rule is increased to 30 to process each dataset. However, the number of semantic relationships in the six datasets from 500 to 3000 was 1200, 2550, 4200, 5700, 7250, and 8850, respectively. Table 8 summarizes the experimental results. Figure 10 shows the average execution time of agent communication for each dataset. Note that the threshold value for instance number was about 2500. When instance number was lower than the threshold value, the execute time is significantly in a linear trend grown with the scale of dataset. Conversely, when the instance number increases beyond this threshold number, execution time increased very rapidly because both instances and rule-based inference performance increased substantially. The instance base query time is slightly increased. This is mainly due to the fact that query does not invoke ontology-based reasoning or rule-based inference.

7 Concluding remarks

How to enhance the level of intelligence for the interacting agents in Multi-agent Systems (MAS) is the major purpose of this study. In this article we propose two approaches exploiting semantic web technologies in the agent communication content language and message content to reinforce the Agent Communication Language (ACL). For the agent communication content language, this article proposed Lightweight Ontology-based Content Language (LOCL) based on FIPA RDF content language and integrating SWRL to facilitate interoperability for agent communications in the distributive environment. For the massage content, we present a threetiers knowledge representation framework developed based on Semantic Web stack, including XML layer (Instance Base), Ontology layer (OWL based Ontology Base), and Rule layer (SWRL based Rule Base), to support ontology-based reasoning and rule-based inference. Hence, not only the agent communication language, but also the message content itself embedded in the content language got formal semantic expressions and can be tackled and interpreted autonomously by the agents. A demo scenario that agent communicate using LOCL as

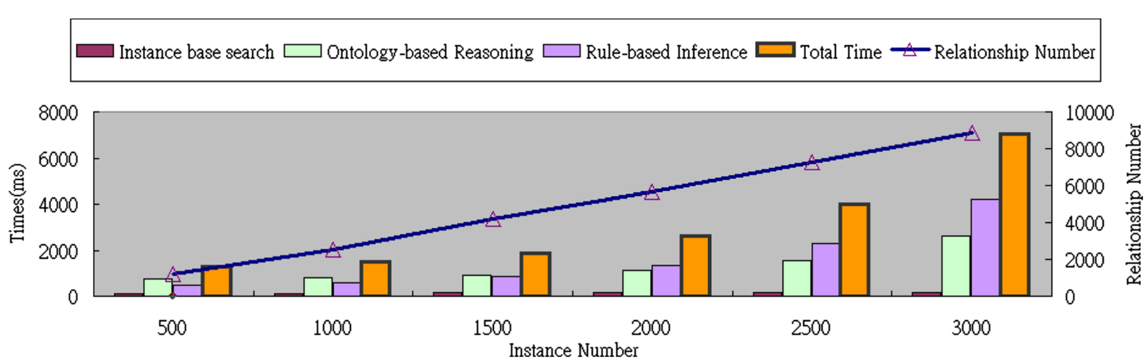


Fig. 10 The various execution times and relationship number depend on the instance number

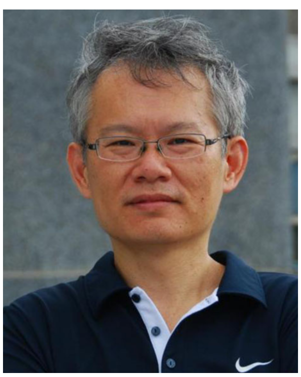
content language, OWL as ontology language, and SWRL as rule language is examined and evaluated to demonstrate the feasibility of our proposed architecture.

Further theoretical revision for heterogeneous multi-agent systems [18] using FIPA ACL with SWRL as content language for knowledge representation to facilitate computer-interpretability will be necessary after this work Moreover, since Cloud computing [2, 25] has evolved as the most important and long-term trend in computing over the Internet, another topic for future work will be extending the capability of LOCL to the cloud computing environment such that agents can intelligently retrieve and compose services in a multi-cloud environments.

References

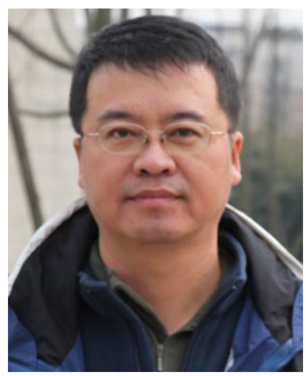
1. Amorim MT, Cury D, Menezes CSd (2014) On the use of ontologies to guide agents answering questions. Advances in Intelligent Systems and Computing 275 AISC(1):525-533
2. Armbrust M et al (2010) A view of cloud computing. Commun ACM 53(4):50–58
3. Balaji BS, Prabhu TN, Karthikeyan NK Dr (2012) SACoSS-semantic agent based system for cloud service suggestion using cloud service ontology. J Comput Inf Syst 8(21):8763–8773
4. Berners-Lee T, Hendler J, Lassila O (2001) The semantic web - a new form of web content that is meaningful to computers will unleash a revolution of new possibilities. Sci Am 284(5):34–43
5. Bernhard Schiemann US (2006) OWL DL as a FIPA ACL content language. In: Proceedings of workshop on formal ontologies for communicating agents, ESSLLI 2006. Malaga
6. Brewster C et al (2004) Knowledge representation with ontologies: the present and future. IEEE Intell Syst 19(1):72–81
7. Brickley D, Guha RV (2014) RDF vocabulary description language 1.0: RDF Schema. Available from: http://www.w3.org/TR/rdf-schema. Accessed 29 June 2015
8. Chu Z, Guo X, Wang X (2012) Research on search system based on agent and personal knowledge ontology. In: International conference on green communications and networks, GCN 2011. Springer, Chongqing, pp 1271-1278
9. Drozdowicz M et al (2010) Utilization of modified coreGRID ontology in an agent-based grid resource management system. In: 25th International conference on computers and their applications 2010, CATA 2010. International Society for Computers and their Applications, Honolulu, pp 240-245
10. Du Y, Wang Y, Chen S (2012) The understanding between agent crawlers based on domain ontology. Neural Network World 22(4):311–324
11. Dung PQ, Florea AM (2011) An architecture and a domain ontology for personalized multi-agent e-learning systems. In: 2011 3rd International conference on knowledge and systems engineering, KSE 2011. IEEE Computer Society, Hanoi, pp 181-185
12. Fathalipour M, Selamat A, Jung JJ (2014) Ontology-based, process-oriented, and society-independent agent system for cloud computing. Int J Distrib Sens Netw 2014
13. FIPA (2014) FIPA ACL Message Structure Specification 2002/12/03. Available from: http://www.fipa.org/specs/fipa00061/SC00061G.pdf. Accessed 29 June 2015
14. FIPA (2014) FIPA KIF Content Language Specification. 2003/01/29, Available from: http://www.fipa.org/specs/fipa00010/XC00010C.pdf. Accessed 29 June 2015
15. FIPA (2014) FIPA RDF Content Language Specification 2001/08/15. Available from: http://www.fipa.org/specs/fipa00011/XC00011A.pdf. Accessed 29 June 2015
16. FIPA (2014) FIPA SL Content Language Specification 2002/12/06. Available from: http://www.fipa.org/specs/fipa00008/SC00008I.pdf. Accessed 29 June 2015
17. Fornara N, Okouya D, Colombetti M (2012) Using OWL 2 DL for expressing ACL Content and Semantics, vol 7541 LNAI. Springer, pp 97–113
18. Gifford CM, Agah A (2009) Sharing in teams of heterogeneous, collaborative learning agents. Int J Intell Syst 24(2):173–200
19. Horrocks I et al (2012) SWRL: A Semantic Web Rule Language Combining OWL and RuleML 2003. Available from: http://www.daml.org/2003/11/swrl/. Accessed 29 June 2015
20. Hsu I-C (2013) Extensible access control markup language integrated with semantic web technologies. Inf Sci 238:33–51
21. Hsu I-C (2013) Integrating ontology technology with folksonomies for personalized social tag recommendation. Appl Soft Comput 13(8):3745–3750
22. Hsu I-C (2012) Intelligent discovery for learning objects using semantic web technologies. Educational Technology& Society 15(1):298-312
23. Hsu I-C (2013) Personalized web feeds based on ontology technologies. Inf Syst Front 15(3):465-479
24. Hsu I-C (2009) SXRS: An XLink-based recommender system using semantic web technologies. Expert Systems with Applications 36(2 PART 2): 3795-3804
25. Ivanova M, Ivanov G (2010) Cloud computing for authoring process automation. Procedia Social and Behavioral Sciences 2(2):3646-3651
26. Kao S-J, Hsu I-C (2007) Semantic web approach to smart link generation for web navigations. Software: Practice and Experience 37(8):857-879
27. Klyne G, Carroll JJ (2012) Resource Description Framework (RDF):Concepts and Abstract Syntax 2004. Available from: http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/. Accessed 29 June 2015
28. Maio CD et al (2014) Formal and relational concept analysis for fuzzy-based automatic semantic annotation. Appl Intell 40(1): 154–177
29. Martin D et al (2014) OWL-S: Semantic Markup for Web Services 2004. Available from:http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/. Accessed 29 June 2015
30. McGuinness DL, Harmelen Fv (2012) OWL Web Ontology Language Overview 2004. Available from: http://web5.w3.org/TR/owl-features/. Accessed 29 June 2015
31. Michal Laclavik ZB, Babik M, Ladislav H (2006) AgentOWL: semantic knowledge model and agent architecture. Comput Artif Intell 25(5):419–437
32. Nebot V, Llavori RB (2014) Exploiting semantic annotations for open information extraction: an experience in the biomedical domain. Knowl Inf Syst 38(2):365–389
33. Razzaq A et al (2014) Semantic security against web application attacks. Inf Sci 254:19–38
34. RuleML (2011) Rule Markup Language (RuleML) 2002. Available from: http://www.ruleml.org/. Accessed 29 June 2015
35. Samirmi FD, Tang W, Wu H (2013) Power transformer condition monitoring and fault diagnosis with multi-agent system based on ontology reasoning. In: Asia-Pacific Power and Energy Engineering Conference, APPEEC. IEEE Computer Society, Kowloon
36. Shadbolt N, Berners-Lee T, Hall W (2006) The semantic web revisited. IEEE Intell Syst 21(3):96–101
37. Souza JPd et al (2013) An Ontology-Based Agent For Context Aware Software Process Development. In: 2013 IEEE 17th

International Conference on Computer Supported Cooperative Work in Design, CSCWD 2013. Whistler, BC, pp 287-292
38. Su C-J, Peng CW (2012) Multi-agent ontology-based Web 2.0 platform for medical rehabilitation. Expert Systems with Applications 39(12):10311-10323
39. UMBC (2010) UMBC AgentWeb. Available from: http://www.cs.umbc.edu/kqml/. Accessed 29 June 2015
40. Wang H, Tang X (2012) Multi-Agent Semantic Mining System Based on Ontology. In: 2012 4th International Conference on Multimedia and Security, MINES 2012, Nanjing, pp 655-657
41. Wang X, Wong TN, Wang G (2012) Service-oriented architecture for ontologies supporting multi-agent system negotiations in virtual enterprise. J Int Manag 23(4):1331–1349
42. Yang S-Y (2013) Developing an energy-saving and case-based reasoning information agent with Web service and ontology techniques. Expert Systems with Applications 40(9): 3351–3369
43. Ying W, Ray P, Lewis L (2013) A methodology for creating ontology-based multi-agent systems with an experiment in financial application development. In: 46th Annual Hawaii International Conference on System Sciences, HICSS 2013. IEEE Computer Society, Wailea, pp 3397-3406
44. Youyong Zou TF, Li D, Harry C, Rong P (2003) Using Semantic web technology in Multi-Agent systems: a case study in the TAGA Trading agent environment. In: Proceeding of the 5th International Conference on Electronic Commerce
45. Zou Y et al (2003) Agent Communication in DAML World. In: Truszkowski W, Rouff C, HincheycM M (eds) First InternationalWorkshop on Radical Agent Concepts WRAC 2002. Springer, McLean, pp 347-354



Fang-Ping Pai received his MS degree in Aerospace Engineering from National Chung-Kung University in 1989. He is now a PhD student in the Department of Computer Science at National Tsing-Hu a University. His research in terests in clude High Level Architecture (HLA), Military modeling and simulation and Semantic web technologies.

I-Ching Hsu received a Ph.D. degree in the computer science department at the National Chung-Hsing University of Taiwan in 2007. Dr. Hsu has worked at Chung-Shan Institute of Science & Technology in the area of information engineering since 1991. He is currently an associate professor of Computer Science and Information Engineering at National Formosa University, Taiwan. He has participated and directed projects in the area of cloud computing, Semantic Web and MAS applications development. His current research interests are in Cloud Computing, Semantic Web, IoT and Software Engineering.



Yeh-Ching Chung received a B.S. degree in Information Engineering from Chung Yuan Christian University in 1983, and the M.S. and Ph.D. degrees in Computer and Information Science from Syracuse University in 1988 and 1992, respectively. He joined the Department of Information Engineering at Feng Chia University as an associate professor in 1992 and became a full professor in 1999. From 1998 to 2001, he was the chairman of the department. In 2002, he joined the Department of Computer Science at National Tsing Hua University as a full professor. His research interests include parallel and distributed processing, cloud computing, and embedded systems. He is a senior member of the IEEE computer society.