An Event Ontology Description Framework Based on SKOS

Yujia Zhang, Wei Liu, Ning Ding, Xu Wang, and Yue Tan
School of Computer Engineering and Science, Shanghai University
Shanghai 200444, China
yujia_zhang @shu.edu.cn

***Abstract*—Event ontology is a new paradigm for describing event-based knowledge in web, including action, time, place and objects in event. But it lacks a unified language for description of event classes or event individuals. Simple Knowledge Organization System (SKOS) is a common data model for sharing and linking knowledge organization systems via the Web. This paper proposes an event ontology description framework based on SKOS. In this framework, SKOS is used to link current ontologies or vocabularies for describing four key event elements. As well, the description framework increases the reusability of the event ontology.**

Keywords—Event Ontology, SKOS, OWL

I. Introduction

Event ontology is a shared, formal and explicit specification of an event class system model that exists objectively. Event ontology model could capture the dynamic aspects of application domain. In event ontology model, event classes provide a natural way to express complicated relations between people, places, actions and objects. Event relationships provide more sophisticated description and reasoning of event-centered concepts.

Event-based knowledge sharing is a key feature of event ontology. It means we can share knowledge from other ontologies while building an event ontology. On the other hand, while building a new event ontology, its reusability should be considered. An event ontology consists of event (classes) and event relations among the event (classes), an event consists of object, action, time, place and status. Now, there is a lack of unified ontology language for description of event ontology. But we found that some existent concept-centered ontologies or vocabularies are semantically related to some elements of event. In our opinion, it's reasonable to describes event classes and different types of event elements by using extended OWL 2 and integrating some terms from external ontology languages or vocabularies that based on W3C's RDF technology, fox example, reusing Agent class from FOAF [1] to describe actors of event, using terms from Time Ontology [2] to describe time of event, thus to improve the reusability of event element classes.

Simple Knowledge Organization System (SKOS) is a common data model for sharing and linking knowledge organization systems via the Web. In this paper, we propose an event ontology description framework based on SKOS. The SKOS data model provides a standard, low-cost migration path for porting existing knowledge organization systems to the Semantic Web. SKOS also provides a lightweight, intuitive language for developing and sharing new knowledge organization systems. It may be used on its own, or in combination with formal knowledge representation languages such as the Web Ontology language (OWL).In this framework, SKOS is used to link current ontologies or vocabularies for describing four key event elements.

The remainder of this paper is organized as follows. In Section 2, event model with 6 event elements and their semantic will be introduced. In Section 3, the method will be illustrated with representing the introductions of existent ontologies. In Section 4, there is a schema showed. In Section 5, conclusion and future work will be mentioned.

II. Event Model and Semantic

In this paper, event model is defined as a 6-tuple. The relations between event and elements, coupled with semantic descriptions of elements are introduced as follows.

A. Definition of Event and Event Class

Definition1 (Event) We defined event as a thing happening in a certain time and place, which is involved in some actors, objectives and action features with statuses changing. Event e is defined as a 6-tuple formally:

$$Event::=<A, O, T, P, S, L>$$

where, elements is named as event elements, A means an action happen in an event, O means actors and objects involved in an event, T means instant and interval time of an event, P means the place that an event happens in, S means statuses of object O before and after an event, L indicates language expressions of text-based event.

Definition2 (Event Class) Event Class is an abstract event that represents a set of events with some common characteristics, denoted as EC:

$$EC=(E, C_A, C_O, C_T, C_P, C_S, C_L)$$
$$C_i=\{c_{i1}, c_{i2}, \ldots, c_{im}, \ldots\}\ (i \in \{A, O, T, P, S, L\}, m\geq 0)$$

where E means an event set, called extension of the event class, C_i denotes the common characteristics set of certain event element (element i), called intension of the event class and event element class, c_{im} denotes one of the common characteristics of event factor i.

B. Semantic Descriptions of Event Element Classes

Due to traditional ontologies describing static concepts, we provide more specific definitions for event class and event element classes in order to map more easily.

According to the features of different event elements, event element classes is supposed to be classified into two groups: Dynamic Event Element Class, having

978-1-4673-7211-4/15 $31.00 © 2015 IEEE
DOI 10.1109/UIC-ATC-ScalCom-CBDCom-IoP.2015.322

these two event element classes (action, status) and Static Event Element Class, having the other four event element classes (object, time, place, language), denoted as:

$$EventClass ::=_{def} (dynElementClass,\ staticConcept)$$
$$dynElementClass ::=_{def} (action,\ status)$$
$$staticConcept ::=_{def} (object,\ time,\ place,\ language)$$

Event element classes defined as follows:

- Object: Object is a class of objects O which are involved in events class, including actors and general objects.

$$Object ::=_{def} (actor,\ generalObj)$$

- Place: Place is a class of places P where events happen, including real places in the real world, such as New York and Beijing as well as virtual place in the virtual world, such as forum and online store.

$$Place ::=_{def} (realPlace,\ virtualPlace)$$

- Language: Language is a class of language expressions L, including *elementExistence* which shows whether 6 elements exit in an event or not, *literal* which shows the exact word to represent an event, and *representRule* which shows a rule for representing events, such as collocation.

$$Language ::=_{def} (elementExistence,\ literal,\ representRule)$$

- Action: Action is a class of action A which is an essential part of event, a key element to reflect event dynamics, including thoughtEvent, statement, instantAction, movement, operation, stateChange and perception. Generally, each event has only one action, and both subjects and objects of action belong to the event element class Object. When the action finishes, the statuses which belongs to this event will change from preStatus to postStatus.

$$Action ::=_{def} (thoughtEvent,\ statement,\ instantAction,\ movement,\ operation,\ stateChange,\ perception)$$

- Status: Status is a class of status S, including preStatus and postStatus.

$$Status ::=_{def} (preStatus,\ postStaus)$$

C. Expression of Event in OWL

According to the definition of event, event has 6 elements, while resources in web are stored in RDF triple. In order to express the six-tuple event, we adopt 6 triples to express event elements by using object properties. Event elements will not be considered as event subclass, but be comprised by event, because they do not have parent-child relationship. Equivalence between event class and element classes is showed in Table 1.

TABLE 1 Equivalence between Event and Elements

Event Equivalent To
(hasTime some Time)
and (hasObject some Object)
and (hasPlace some Place)
and (hasAction min 1 Action)
and (hasAction max 1 Action)
and (hasStatus min 2 Status)
and (hasLanguage min 1 Language)

It can be seen in table 1 that there are 6 object properties, *hasobject*, *hasTime*, *hasAction*, *hasStatus*, *hasPlace* and *hasLanguage*, which are linking between *EventClass* and 6 event element classes respectively. The domains and ranges of these 6 object properties are showed in Table 2.

TABLE 2 Domain and Range of 6 Object Properties

Object Property	**Domain**	**Range**
hasObject	*staticConcept*	*object*
hasTime	*staticConcept*	*time*
hasAction	*dynElementClass*	*action*
hasStatus	*dynElementClass*	*status*
hasPlace	*staticConcept*	*place*
hasLanguage	*staticConcept*	*language*

III. MAPPING WITH AVAILABLE ONTOLOGY

In this section, SKOS mapping method and the ontologies available online are introduced. Then according to the definition of event ontology model and available ontologies, a method for mapping existing ontologies to event ontology is proposed in Figure 1.

A. SKOS Mapping

Simple Knowledge Organization System (SKOS) [3] is a W3C recommendation built upon RDF and RDFS and an area of work developing specifications and standards to support the use of knowledge organization systems (KOS) such as thesauri, classification schemes, subject heading systems and taxonomies within the framework of the Semantic Web. The SKOS defines the classes and properties sufficient to represent the common features found in a standard thesaurus. It is based on a concept-centric view of the vocabulary, where primitive objects are not terms, but abstract notions represented by terms. Each SKOS concept is defined as an RDF resource.

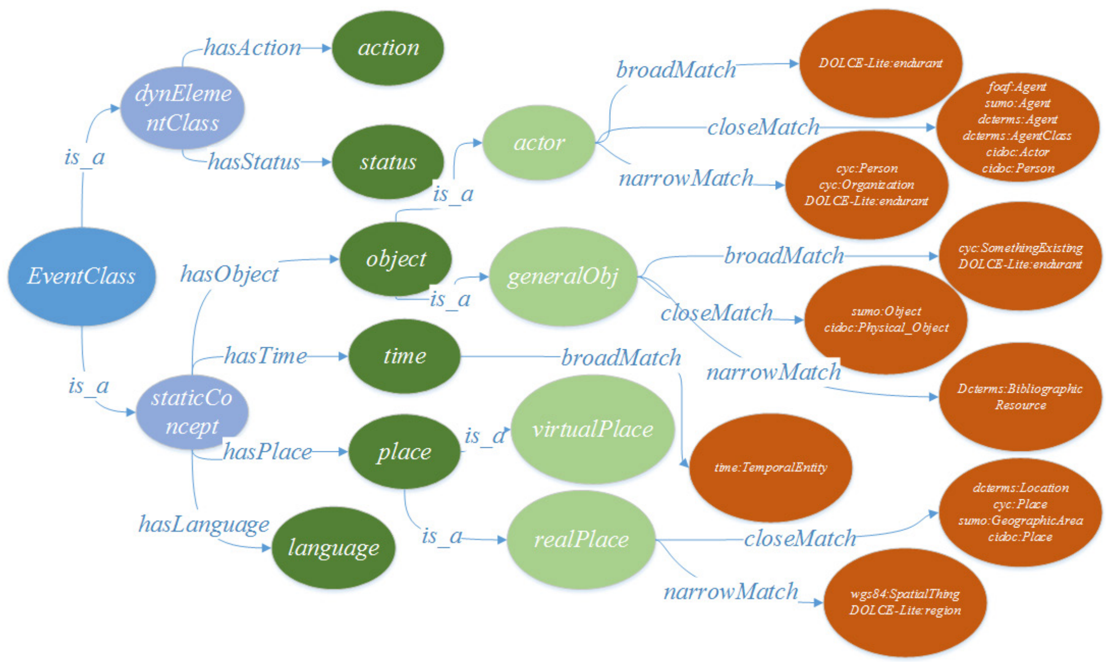

Figure 1 Mapping Method

TABLE 3 Properties and Meanings of SKOS

Property of SKOS Mapping	Meaning
skos:closeMatch	that maps two similar concepts, being used interchangeably sometimes, not transitive
skos:exactMatch	that is a transitive property which maps two concepts which have high confidence, as a sub property of skos:closeMatch
skos:broadMatch	a convenience, with the same meaning as the semantic properties broader
skos:narrowMatch	a convenience, with the same meaning as the semantic properties narrower
skos:relatedMatch	a convenience, with the same meaning as the semantic properties related

B. *Available Ontologies*

There are numerous ontologies available and some of them have already been recommended by W3C, such as FOAF which describes individuals, Time Ontology which describes time, GeoNames Ontology which describes place and so forth. The concepts and properties in available ontologies are matched by even elements class in Event Ontology. Here are the introduction of available ontologies that are be reused in the description of Event Ontology.

1) FOAF[1]*:* Friend-of-a-Friend, namely FOAF, is a XML/RDF vocabulary, which is a network of documents describing a network of people (and other stuff) in the Web by using the form that computer can understand. FOAF describes basic information about people, relationships among people, and also interests such as RSS Feed, books that are being read, movies that are watched and so on. FOAF is a basic element of virtual community, and describes different Web service need. FOAF has 19 categories and 62 properties. In FOFA, the class *Agent* with subclass *person*, *group*, *organization* is closely matched the description of *actor* in Event Ontology.

2) OpenCye[2]: OpenCye [4] is a collection of ontology terms, one of the biggest general knowledge base and common sense reasoning engine. OpenCye is one of artificial intelligence program Cyc results, and the purpose is to reason knowledge by using the rich and diverse collection of real-world concepts and ontologies in all fields. The current version of OpenCye includes 239000 terms and 2093000 triples, and builds link with OWL: sameAs and multi-ontologies. In OpenCyc, the class *Person* and *Organization* are narrowly matched the description of *actor* in Event Ontology, the class *SomethingExisting* is broadly matched the description of *generalObj* in Event Ontology, the class *Predicate* is narrowly matched the description of *statement* in Event Ontology, and the class *Place* is closely matched the description of *realPlace* in Event Ontology.

3) CIDOC CRM[3]*:* CIDOC Conceptual Reference Model [5], namely CIDOC CRM, is object-oriented, and provides definitions and a formal structure for describing the implicit and explicit concepts and relationships used in cultural heritage documentation. It is intended to promote a shared understanding of cultural heritage information by providing a common and extensible semantic framework that any cultural heritage information can be mapped to. It is intended to be a common language for domain experts and

1 http://xmlns.com/foaf/spec/#sec-glance%E3%80%82FOAF

2 http://sw.opencyc.org/

3 http://erlangen-crm.org

implementers to formulate requirements for information systems and to serve as a guide for good practice of conceptual modeling. In this way, it can provide the "semantic glue" needed to mediate between different sources of cultural heritage information, such as that published by museums, libraries and archives. This model has tree structure, 85 categories and 283 properties. In CIDOC CRM, the class *Actor* and *Person* are closely matched the description of *actor* in Event Ontology, the class *Physical_Object* is closely matched the description of *generalObj* in Event Ontology, the class *Place* is closely matched the description of *realPlace* in Event Ontology, and the class *Language* is closely matched the description of *literal* in Event Ontology.

4) DOLCE and DnS: the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [6] is the first module of the WonderWeb foundational ontologies library. As implied by its acronym, DOLCE has a clear cognitive bias, in that it aims at capturing the ontological categories underlying natural language and human common sense. In DOLCE and DnS, the class *endurant* is broadly matched the description of *actor* and *generalObj* in Event Ontology, the class *process* is narrowly matched the description of *operation* in Event Ontology, the class *accomplishment* and *achievement* are narrowly matched the description of *stateChange* in Event Ontology, the class *region* is narrowly matched the description of *realPlace* in Event Ontology, and the class *state* is closely matched the description of *status* in Event Ontology.

5) DCMI[4]*:* Dublin Core Metadata Initiative, namely DCMI [7], is a small set of vocabulary terms that can be used to describe web sources as well as physical resources and objects, with 15 original classes. In DCMI, the class *Agent* and *AgentClass* is matched the description of *actor* in Event Ontology, and the class *BibliographicResource* is matched the description of *gerneralObj* in Event Ontology.

6) SUMO[5]*:* Suggested Upper Merged Ontology, namely SUMO [8], is owned by IEEE. Its aim is to be as a foundation ontology for a variety of computer information processing systems, to promote the abilities of interoperability, information search, automatically reasoning and natural language processing. It has 12263 terms, 62753 axioms, 5130 rules and 1383 relationships. SUMO is the only formal ontology that has been mapped to all of the WordNet lexicon. In SUMO, the class *Agent* is closely matched the description of *actor* in Event Ontology and the class *Object* is closely matched the description of *generalObj* in Event Ontology.

7) Time Ontology[6]*:* Time Ontology presents ontology of temporal conception. It has 9 categories, such as Instant, Interval as subclass of TemporalEntity, and 41 properties, such as before, after, years, weeks, days. In Time Ontology, the class *TemporalEntity* is closely matched the description of *time* in Event Ontology.

8) GeoNames Ontology[7]*:* GeoNames Ontology has geospatial semantic information, with all over 8.3 million geonames toponyms having a unique URL with a corresponding RDF web service. In GeoNames, the class *Place* is closely matched the description of *realPlace* in Event Ontology.

9) WGS84[8]*:* WGS84 is a basic RDF vocabulary that provides the Semantic Web community with a namespace for representing latitude, longitude and other information about spatially-located things. In WGS84, the class *SpatialThing* is narrowly matched the description of *realPlace* in Event Ontology.

C. Method of Mapping

According to SKOS Mapping, the alignment of event element classes and available ontologies can be completed. The reason to choose SKOS Mapping rather than OWL Mapping (such as owl:sameAs, owl:equialentClass) is that SKOS Mapping has a less tough relation than OWL Mapping so that it may be more easy for SKOS Mapping to reuse the possible classes in available ontologies. In existent ontologies, there are still numerous concepts which are able to be reused, and we can align the public ontologies online which are accepted by most scholars with Event Ontology.

1) actor: the mapping method of *actor* is showed below.

TABLE 4 Mapping Method of actor

Class	**Mapping Sentence**
foaf:Agent	<skos:closeMatch rdf:resource="http://xmlns.com/foaf/0.1/Agent"/>
cyc:Person	<skos:narrowMatch rdf:resource="http://sw.opencyc.org/concept/Mx4rvViAkpwpEbGdrcN5Y29ycA#Person"/>
cyc:Organization	<skos:narrowMatch rdf:resource="http://sw.opencyc.org/concept/Mx4rvVjVT5wpEbGdrcN5Y29ycA#Organization"/>
sumo:Agent	<skos:closeMatch rdf:resource="http://www.adampease.org/OP/SUMO.owl#Agent"/>
dcterms:Agent	<skos:closeMatch rdf:resource="http://purl.org/dc/terms/Agent"/>
dcterms:AgentClass	<skos:closeMatch rdf:resource="http://purl.org/dc/terms/AgentClass"/>
cidoc: Actor	<skos:closeMatch rdf:resource="http://erlangen-crm.org/ontology/ecrm/ecrm_current.owl#E39.Actor"/>
cidoc:Person	<skos:closeMatch rdf:resource="http://erlangen-crm.org/ontology/ecrm/ecrm_current.owl#E21.Person"/>
DOLCE-Lite:endurant	<skos:broadMatch rdf:resource="http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl#endurant"/>

2) generalObj: the mapping method of *generalObj* is showed below.

[4] http://dublincore.org/documents/2012/06/14/dcmi-terms
[5] http://www.adampease.org/OP/
[6] http://www.w3.org/TR/owl-time/#summary
[7] http://www.geonames.org/ontology/documentation.html
[8] http://www.w3.org/2003/01/geo/

TABLE 5 Mapping Method of generalObj

Class	Mapping Sentence
cyc:SomethingExisting	<skos:broadMatch rdf:resource="http://sw.opencyc.org/concept/Mx4rvVi255wpEbGdrcN5Y29ycA#SomethingExisting"/>
sumo:Object	<skos:closeMatch rdf:resource="http://www.adampease.org/OP/SUMO.owl#Object"/>
dcterms:BibliographicResource	<skos:narrowMatch rdf:resource="http://purl.org/dc/terms/BibliographicResource"/>
cidoc:Physical_Object	<skos:closeMatch rdf:resource="http://erlangen-crm.org/ontology/ecrm/ecrm_current.owl#E19.Physical_Object"/>
DOLCE-Lite:endurant	<skos:broadMatch rdf:resource="http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl#endurant"/>

3) *time*: the mapping method of *time* is showed below.

TABLE 6 Mapping Method of time

Class	Mapping Sentence
time:TemporalEntity	<skos:broadMatch rdf:resource="http://www.w3.org/2006/time#TemporalEntity"/>

4) *realPlace*: the mapping method of *realPlace* is showed below.

TABLE 7 Mapping Method of realPlace

Class	Mapping Sentence
wgs84:SpatialThing	<skos:narrowMatch rdf:resource="http://www.w3.org/2003/01/geo/wgs84_pos#SpatialThing"/>
dcterms:Location	<skos:closeMatch rdf:resource="http://purl.org/dc/terms/Location"/>
cyc:Place	<skos:closeMatch rdf:resource="http://sw.opencyc.org/concept/Mx4rvVjTtJwpEbGdrcN5Y29ycA#Place"/>
sumo:GeographicArea	<skos:closeMatch rdf:resource="http://www.ontologyportal.org/SUMO.owl#GeographicArea"/>
cidoc:Place	<skos:closeMatch rdf:resource="http://erlangen-crm.org/ontology/ecrm/ecrm_current.owl#E53.Place"/>
DOLCE-Lite:region	<skos:narrowMatch rdf:resource="http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl#region"/>

IV. THE SCHEMA

The Event Ontology Model is described in OWL with 6 event element classes mapped with existent ontologies. Due to the space limitations, the parts of event ontology description framework, object property sample and event element sample, are showed below.

A. Part of Object-Property

Taking *hasObject* as an object property example, it is one of object-properties, a subobject-property of *hasClass*, which includes 6 subobject-properties, *hasobject*, *hasTime*, *hasAction*, *hasStatus*, *hasPlace* and *hasLanguage*. *hasObject* links classes between *staticConcept* class and *object* class. The details are showed below.

```
<owl:ObjectProperty rdf:ID="hasObject">
    <rdfs:SubObjectPropertyOf
rdf:resource="#hasClass">
    <rdfs:domain rdf:resource="#staticConcept"/>
    <rdfs:range rdf:resource="#object"/>
</owl:ObjectProperty>
```

B. Part of Event element

The event element class *object*, is showed below. *object* class is a collection, union of two subclasses, *actor* and *generalObj*. According to part III, *actor* class is mapped with 10 classes of available ontologies, and *generalObj* class is mapped with 7 classes of available ontologies based on SKOS Mapping techonology. The details are showed below.

```
<owl:class rdf:ID="object">
    <rdfs:comment xml:lang="en">
        'object'is one of six event element
classes,involved in events class, including actors and
general objects.
    </rdfs:comment>
    <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="actor">
        <owl:Class rdf:ID="generalObj">
    </owl:unionOf>
</owl:class>
<owl:class rdf:ID="actor">
    <rdfs:subClassOf rdf:resource="#object">
    <skos:narrowMatch
rdf:resource="http://sw.opencyc.org/concept/Mx4rvVi
AkpwpEbGdrcN5Y29ycA#Person"/>
    <skos:narrowMatch
rdf:resource="http://sw.opencyc.org/concept/Mx4rvVj
VT5wpEbGdrcN5Y29ycA#Organization"/>
    <skos:broadMatch
rdf:resource="http://www.loa-cnr.it/ontologies/DOLC
E-Lite.owl#endurant"/>
    <skos:closeMatch
rdf:resource="http://purl.org/dc/terms/Agent"/>
    <skos:closeMatch
rdf:resource="http://purl.org/dc/terms/AgentClass"/>
    <skos:closeMatch
rdf:resource="http://www.adampease.org/OP/SUMO.o
wl#Agent"/>
    <skos:closeMatch
rdf:resource="http://erlangen-crm.org/ontology/ecrm/
ecrm_current.owl#E39.Actor"/>
    <skos:closeMatch
rdf:resource="http://erlangen-crm.org/ontology/ecrm/
ecrm_current.owl#E21.Person"/>
    <skos:closeMatch
rdf:resource="http://xmlns.com/foaf/0.1/Agent"/>
    <skos:narrowMatch
rdf:resource="http://daml.umbc.edu/ontologies/cobra/
0.4/agent#Agent"/>
</owl>
<owl:class rdf:ID="generalObj">
    <rdfs:subClassOf rdf:resource="#object">
```

<skos:narrowMatch rdf:resource="http://purl.org/NET/c4dm/event.owl#Factor"/>
<skos:narrowMatch rdf:resource="http://purl.org/NET/c4dm/event.owl#Product"/>
<skos:broadMatch rdf:resource="http://sw.opencyc.org/concept/Mx4rvVi255wpEbGdrcN5Y29ycA#SomethingExisting"/>
<skos:broadMatch rdf:resource="http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl#endurant"/>
<skos:narrowMatch rdf:resource="http://purl.org/dc/terms/BibliographicResource"/>
<skos:closeMatch rdf:resource="http://www.adampease.org/OP/SUMO.owl#Object"/>
<skos:closeMatch rdf:resource="http://erlangen-crm.org/ontology/ecrm/ecrm_current.owl#E19.Physical_Object"/>
</owl>

V. Conclusion and future work

Reusability of the available ontologies is a decent way to describe event ontology when there are semantical relations between these ontologies and some elements of event. This paper gives specific definition of Event Ontology Model and its 6 event elements. By utilizing SKOS Mapping technology, this paper also proposed a general event ontology description framework, which aligns 6 event element classes with available ontologies, in order to reuse the existent resources. Finally, we give part of event ontology description framework in OWL.

When it comes to specific domain, there are still some extensions about event elements, such as *object* class. Because numerous events belong to different fields, general*Obj* class has various objects, mapping to different domain ontologies. For instance, given an event class, Environment Emergency, in this event class, generalObj class could be mapping to EnvO [9]. In the future, we can use this event ontology description framework into specific domains to enlarge this event ontology.

ACKNOWLEDGMENT

This paper is supported by the Natural Science Foundation of China, No.61305053 and No.61273328, and the Natural Science Foundation of Shanghai, No.12ZR1410900.

REFERENCES

[1] FOAF Vocabulary Specification, http://xmlns.com/foaf/spec/

[2] Time Ontology in OWL, http://www.w3.org/TR/owl-time/

[3] SKOS Simple Knowledge Organization System, http://www.w3.org/2004/02/skos/

[4] Foxvog D. Cyc. Theory and Applications of Ontology: Computer Applications [M]. London, New York. Springer, Dordrecht Heidelberg New York. 2010: 259-278.

[5] Crofts N, Doerr M, Gill T, et al. Definition of the CIDOC Conceptual Reference Model [R]. Paris: ICOM/CIDOC CRM Special Interest Group, 2003.

[6] Claudio Masolo, Stefano Borgo, Aldo Gangemi, et al. WonderWeb Deliverable D18 Ontology Library (final) [M] http://www.loa.istc.cnr.it/old/Papers/D18.pdf, 2003.

[7] Dublincore.org. DCMI Metadata Terms. http://dublincore.org/documents/dcmi-type-vocabulary/index.shtm. 2013

[8] ChuRen Huang, Nicoletta Calzolari, Aldo Gangemi, et al. ONTOLOGY AND THE LEXICON A Natural Language Processing Perspective [M]. Beijing: PEKING UNIVERSITY PRESS, 2014: 27-29.

[9] Environment Ontology, http://environmentontology.org/Browse-EnvO