■ 265

A Survey on Knowledge Transfer between Knowledge-based Systems

Nyoman Karna*, Iping Supriana, Nur Maulidevi
Institut Teknologi Bandung, Ganesha 10, Bandung, Indonesia
Corresponding author, e-mail: bogi@students.itb.ac.id, iping@stei.itb.ac.id, ulfa@stei.itb.ac.id

Abstract
The paper aims to clarify differences in knowledge sharing mechanisms between Knowledge-based Systems, including knowledge management system, web page-based knowledge, and expert system, in the hope that we can establish an automatic knowledge transfer between autonomous systems. This study lays the foundation for knowledge transfer mechanism where an autonomous Knowledge-based System may enhance its knowledge by using other system's knowledge. To design a knowledge transfer mechanism, the paper do a literature study by comparing three well-known protocols for knowledge sharing, OAI-PMH for knowledge management system, SPARQL for web page-based knowledge, and KQML for the expert system. The object of comparison is within three aspects, first is the ability to find another system, the second is knowledge retrieval from chosen system and third is how to add new knowledge into the system. The paper suggests that each protocol has its own strength and weakness, but when it comes to knowledge transfer, KQML covers more features. Therefore, based on this finding, the paper proposes a new model for autonomous knowledge transfer using KQML to enhance one Knowledge-based System's own knowledge.

Keywords: knowledge transfer, knowledge-based system, OAI-PMH, SPARQL, KQML

1. Introduction

Knowledge-based System (KBS) is a system that benefits from its own knowledge to provide solution or recommendation for a specific problem [1, 2]. Just like KBS, every intelligent system has the ability to learn and remember. In respect of human, the ability to remember identifies thinking process, which is how we use experiences stored in our memory [3]. Knowledge engineer designs each KBS for a specific purpose by assimilating certain expert's knowledge, which makes it applicable only in a specific knowledge domain. Although each KBS has a specific knowledge domain, some are on a similar domain, which means these KBS have the opportunity to transfer its content to enhance other system's knowledge. To provide ability in knowledge transfer, this research studies and compares several knowledge-sharing protocols, to enable a platform where KBS can share autonomously and automatically [4].

There are three characteristics that defines a KBS, (1) learning (knowledge acquisition), (2) retaining (knowledge organization), and (3) recalling (knowledge retrieval). Learning explains how any external knowledge re-engineered to fulfill KBS's purpose. Retaining explains how to represent the knowledge for KBS still able to read them. Recalling explains how KBS able to understand and conclude from its knowledge [2-3], [5]. Figure 1 describes this three processes.
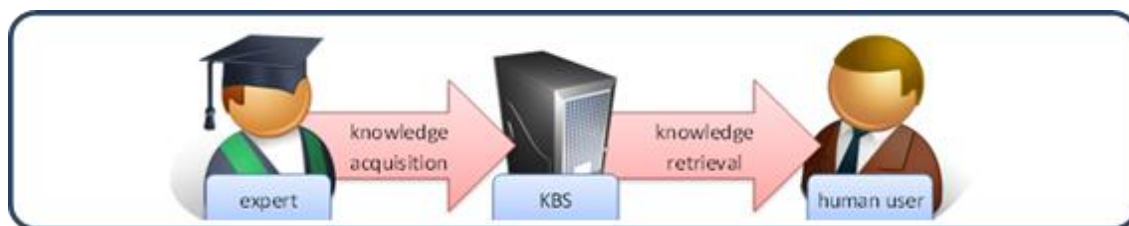

Figure 1. Interaction between expert–knowledge-based system–human user

In term of knowledge sharing, this paper highlights several characteristics on KBS including ability to identify its own knowledge limitation on a specific subject area, know where to obtain further advice, and ability to improve its own knowledge by learning [5, 6]. This means a knowledge sharing mechanism is needed to ensure knowledge enhancement via machine-to-machine communication. However, discussion on knowledge sharing between KBSs is not deep enough to allow knowledge become transportable autonomously from machine to machine without human interference [7]. The ability to transport knowledge, in return provides additional functionality where a KBS can enhance its own knowledge base from other KBS [4, 8]. To support transportable knowledge, this research proposes protocol for knowledge transfer between KBSs. From author perspective, there are three systems, which fall into KBS definition [4], they are:

a. Knowledge Management System provides a specific area of knowledge, although unstructured and unrelated, human can gain information from it and use it to create knowledge through a thinking process;

b. Web Page provides any type of knowledge, although usually unstructured, human can gain information from it and use it to create knowledge through a thinking process;

c. Expert System provides recommendation by using its knowledge for a specific type of problem through a question and answering mechanism between human and system.

Table 1. Comparison Among Three Types of Knowledge-based System

	Knowledge Management System	Web Page	Expert System
Information structure[1]	unstructured	semi-structured	structured
Reasoning location[2]	human side	human side	within system
Thinking process[3]	manually	manually	automatically

a. Information structure explains how the system stores its knowledge. Unstructured means the information (i.e. metadata or knowledge symbolism) has no relationship between them, while structured means there are links to explain specific relationship between information.

b. Reasoning location explains location of inference engine. Within system means the location of the inference engine is inside the KBS and human side means user does the reasoning.

c. Thinking process explains how the system conducts the reasoning to propose new knowledge. Manually thinking process means that the process is done by human, while automatically means that the process is done solely by the system.

2. Related Works

Establishment of three protocols for knowledge sharing marked the momentum in this domain. These three protocols are KQML, Semantic Web, and OAI-PMH. They establish an interoperability protocol to ensure knowledge sharing between knowledge-bases systems, i.e. expert systems, intelligent agents of web pages, and knowledge management systems.

a. Draft of KQML (Knowledge Query and Manipulation Language), based on DARPA Knowledge Sharing Initiative in 1993;

b. Semantic Web from World Wide Web Consortium in 2001;

c. OAI-PMH (Open Archives Initiative–Protocol for Metadata Harvesting) from The Coalition for Networked Information and The Digital Library Federation in 2001.

Three researchers also elaborate on knowledge sharing. Snowden [9] emphasizes the third generation of knowledge management, which requires the clear separation of context, narrative and content management. Implicitly, this requires knowledge sharing capability to support codification and creating relation. The last two researchers discuss the importance of knowledge sharing [10] and the difference between knowledge sharing and knowledge transfer [11].

2.1. Knowledge Management System

The history of Knowledge Management (KM) has passed for three decades. Although there is still no universally accepted definition, in technical perspective KM is the process through which organizations generate value from their intellectual and knowledge-based assets. It becomes apparent that KM is concerned with the process of identifying, acquiring, distributing and maintaining knowledge that is essential to the organization [4], [12-14]. There are five processes identified to generate value from knowledge:

a. Collecting knowledge;
b. Organizing knowledge;
c. Summarizing knowledge;
d. Analysing knowledge;
e. Synthesizing knowledge.

In the second process of organizing knowledge, a Knowledge Management System (KMS) may stores knowledge in PDF softcopy document. There is no relation between documents except a classification based on the knowledge's subject [13]. One classification method comes from librarian, which is Dewey Decimal Classification (DDC). DDC helps KMS to classify documents, although in order to simplify the classification we need to excerpt some information using metadata from each of the document [13]. To help metadata creation, KMS has the option to use resource description such as Dublin Core®. Using Dublin Core® and DDC, KMS may provide abilities such as organizing knowledge within KMS, similarity measurement between knowledge, and create relation between knowledge. All these benefits provide support for knowledge sharing mechanism between KMSs [15].

Open Archive Initiatives offer knowledge sharing between KMSs using OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting). OAI-PMH provides client-server messaging transaction, using request and response messages between harvester (client) and repository (server). There are six request messages in OAI-PMH [16]:

a. Verb=Identify, used to check an identity of a repository and how to retrieve its metadata;
b. Verb=ListIdentifiers, used to list all possible identifiers (header of record) in a repository;
c. Verb=ListMetadataFormats, used to list all possible metadata formats in a repository;
d. Verb=ListRecords, used to list all possible records in a repository using filtering;
e. Verb=ListSets, use to list all possible sets in a repository using filtering as parameter;
f. Verb=GetRecord, use to get all records from specific identifier

2.2. Web-paged Knowledge

A web page contains information from a simple and focus information such as a personal blog to a broad range of information such as Wikipedia. Due to the unstructured and high possibility of informal narration, World Wide Web Consortium (W3C) provides Semantic Web so a web page may have higher readability for a machine. Semantic Web uses metadata to identify the content of a web page. As Semantic Web arising, we can extend the metadata in RDF format to leverage the machine readability, which in return support knowledge sharing. RDF (Resource Description Framework) stores metadata within a web page and SPARQL Protocol and RDF Query Language provides the tool to retrieve this information. The result of a SPARQL query is an RDF graph (RDF triples) provided in SPO structure (Subject–Predicate–Object) as described in Figure 2.

Figure 2. Knowledge model in RDF

SPARQL distinguishes two systems in client-server interaction, they are SPARQL Protocol Service and SPARQL Protocol client. As a query language, SPARQL provides four types of queries, SELECT, CONSTRUCT, ASK, and DESCRIBE. A request and response pair in SPARQL can only interact with RDF and not on the metadata of a web page. To use the metadata as an RDF, we need to reformat the metadata into SPO structure. This gap can be

minimized by using Open Graph protocol, where several metadata already provided in RDFa format, such as *og:title*, *og:type*, *og:image* and *og:url* [17].

2.3. Expert System

The Stanford Heuristic Programming Project led by Edward Feigenbaum introduced this expert system, followed by Edward Shortliffe who tried to identify domains where expertise was highly valued and complex, such as diagnosing infectious diseases, in which they developed MYCIN in the early 1970s using IF-THEN mechanism. Later on, ARPA (Advanced Research Projects Agency) assembled a team and initiate Knowledge Sharing Effort to further extend the capability of expert system. One of the working groups in ARPA KSE, the External Interfaces Group, proposed KQML (Knowledge Query and Manipulation Language) document. This document defines the interaction including communication between two KBS [7, 18]. KQML uses pragmatic approaches that include two processes:
a. Knowing who to talk to and how to find them, and
b. Knowing how to initiate and maintain an exchange

Using KQML, a KBS can exchange knowledge base with other KBS. Knowledge exchange in KQML is defined using sequence of messages including *query*, *reply*, *subscribe*, *handle*, and *next*. There are several basic communication protocols defined in KQML, and three of them are described in Figure 3. All communication may use KIF (Knowledge Interchange Format) to transfer a specific knowledge [19].
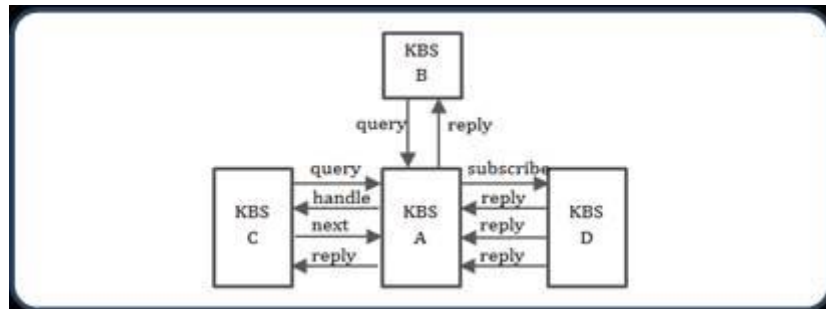

Figure 3. Basic communication protocol in KQML

Based on Figure 3, KQML provides three communication protocols between sender and receiver agent [20]. In A-B interaction, B sends a query to obtain a spesific knowledge and A reply with all requested knowledge in one delivery. In A-C interaction, C sends a query to obtain a specific knowledge, A request a handshake through handle before replying with all requested knowledge in one delivery. In A-D interaction, A sends a subscription message to obtain a specific knowledge, D reply with all requested knowledge in several deliveries.

3. Comparison among Protocols

This research surveys among protocols that support knowledge sharing [15] between KBS. They are OAI-PMH, originally designed for document in digital catalog, SPARQL, originally designed for Semantic Web, and KQML, originally designed for expert system. In term of knowledge sharing, a KBS is said to support knowledge sharing when they have the ability to do three tasks:
a. How to find the right KBS that can provide knowledge needed by origin system;
b. How to retrieve a specific knowledge from previously chosen KBS;
c. How to use previously retrieved knowledge to enhance origin system's knowledge.

All three protocols are compared base on above three tasks, in the hope that an extension from those protocols can be proposed for future enhancement. This extension will enable automatic knowledge enhancement using machine-to-machine communication. As an example of application of this enhancement, say there are two KBS with knowledge domain on diabetes mellitus treatment, one with knowledge on medical treatment approach and one with knowledge on herbal treatment approach. Using automatic knowledge enhancement, KBS with

medical treatment approach can also have the knowledge about herbal treatment, thus it can provide recommendation with both medical and herbal approach.

3.1. Finding the Right System

Finding the right system means every KBS should provide information about its knowledge domain. Providing information about knowledge domain can be done as per request by origin KBS (pull mechanism) or by broadcasting to a specific aggregator service (push mechanism). Besides knowledge domain, KBS is necessary to provide other information, like how to retrieve its content and in what format the information would be delivered.

To support finding another system, OAI-PMH provides *eprints* record to store information that describes the KBSs knowledge domain. To extract this information, harvester should send an *Identify* request to the repository. Repository should respond with an XML contains three description containers, *oai-identifier*, *eprints* and *friends*. However, the *eprints* and *friends* record is optional. The only mandatory description container is the *oai-identifier*. Therefore, there is a probability that a harvester could not find a repository with *eprints* description container. Figure 4 shows an *Identify* request to CiteSeerX. The response contains a text message "*Computer and Information Science Publications collected by CiteSeerX.PSU*" in *eprints* record. This text might be used to explain the knowledge domain of the repository.

```
<request verb="Identify">http://citeseerx.ist.psu.edu/oai2</request>
  <Identify>
    <repositoryName>
      "CiteSeerX Scientific Literature Digital Library and Search Engine"
    </repositoryName>
    <description>
      <eprints xmlns="http://www.openarchives.org/OAI/1.1/eprints"
      xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/eprints
      http://www.openarchives.org/OAI/1.1/eprints.xsd">
      <content>
        <text>
        Computer and Information Science Publications collected by
CiteSeerX.PSU
        </text>
      </content>
      </eprints>
```

Figure 4. Example of *eprints* description container for knowledge domain identification

To support finding another system, a simple SELECT query in SPARQL is enough to find the right web page that contains needed knowledge; however, SPARQL Protocol is not much for a job here since the information needed is already there within the metadata of the web page. Information regarding the knowledge domain of a web page is available in the subject description in Dublin Core® metadata *dc.subject*. Figure 5 shows an example of *dc.subject* as a result of HTTP GET from a journal of Emerald Group Publishing. This request provides response from the web page where in *dc.subject* contains the knowledge domain of the page "*Knowledge Management; Experience; Return on investment; Investment appraisal; Investment in knowledge management; Knowledge management value*". This information can be further explained using *dc.description* metadata.

```
<meta name="dc.Title" content="Does knowledge management produce practical
outcomes?" />
<meta name="citation_journal_title" content="Journal of Knowledge Management" />
<meta name="dc.Creator" content=" Peter Rex Massingham " />
<meta name="dc.Creator" content=" Rada K Massingham " />
<meta name="dc.Subject" content="Knowledge Management; Experience; Return on
investment; Investment appraisal; Investment in knowledge management; Knowledge
management value" />
```

Figure 5. Example of *dc.subject* for knowledge domain identification

Differ with OAI-PMH and SPARQL, KQML does not provide language to ask knowledge domain. By using KQML, an agent should know which agent would be queried prior sending the query itself. Therefore, KQML provides a special class of agent called communication facilitator. Figure 6 describes facilitator functionality to find a specific knowledge from an agent. An agent A is trying to find an agent that holds sentence X by sending *subscribe* message to communication facilitator. Knowing that agent A is looking for sentence X, agent B *tell* the facilitator about X to be forwarded to agent A by a facilitator.
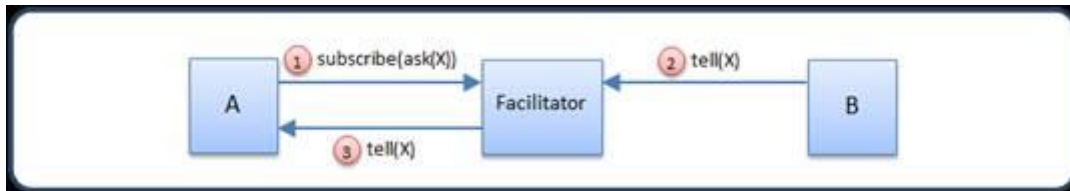

Figure 6. Sample mechanism for finding correct agent in KQML

3.2. Finding Correct Knowledge

Finding correct knowledge means a KBS should understand how to retrieve specific knowledge from other KBS. This is the second step after a KBS found the correct KBS. This step is crucial for knowledge sharing so a KBS can use other KBS's knowledge. Therefore, in this step, a simple pull mechanism is needed to allow a KBS retrieves the intended knowledge.

Using OAI-PMH, we can use *GetRecord* request and obtain the information from the response. This information is in *dc:title*, which describes the title of knowledge, *dc:creator*, which describes the author of the knowledge, *dc:subject*, which describes the subject of knowledge, and *dc:description*, which can be used to describes the abstract of the knowledge. Applying GetRecord request on CiteSeerX gives response like in Figure 7.

```xml
<request identifier="oai:CiteSeerX.psu:10.1.1.1.1484" metadataPrefix="oai_dc"
         verb="GetRecord">http://citeseerx.ist.psu.edu/oai2</request>
                              <GetRecord>
                               <record>
                               <header>
           <identifier>oai:CiteSeerX.psu:10.1.1.1.1484</identifier>
                     <datestamp>2009-05-24</datestamp>
                               </header>
                               <metadata>
            <oai_dc:dc xmlns:dc="http://purl.org/dc/elements/1.1/"
           xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/"
          xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/
                 http://www.openarchives.org/OAI/2.0/oai_dc.xsd">
                               <dc:title>
     Winner-Take-All Network Utilising Pseudoinverse Reconstruction Subnets
     Demonstrates Robustness on the Handprinted Character Recognition Problem
                               </dc:title>
                  <dc:creator>J. Körmendy-rácz</dc:creator>
                      <dc:creator>S. Szabó</dc:creator>
                        <dc:creator>...</dc:creator>
      <dc:subject>Correspondence and offprint requests to</dc:subject>
                 <dc:subject>J. Kormendy-Rácz</dc:subject>
                              <dc:description>
     Wittmeyer's pseudoinverse iterative algorithm is formulated as a dynamic
       connectionist Data Compression and Reconstruction (DCR) network...
                            </dc:description>
                             </oai_dc:dc>
                             </metadata>
                              </record>
                            </GetRecord>
```

Figure 7. Using *GetRecord* to retrieve knowledge from a repository

Sending HTTP GET to a page in Emerald Group Publishing, we can find knowledge based on *dc:description*. Like Figure 8, in *dc.description* we can expect to have excerpt of the knowledge within the page "Purpose–The paper examines ways that Knowledge Management (KM) can demonstrate practical value for organizations. It begins by... (etc.)".

```
<meta name="dc.Title" content="Does knowledge management produce practical outcomes?"
/>
<meta name="citation_journal_title" content="Journal of Knowledge Management" />
<meta name="dc.Creator" content=" Peter Rex Massingham " />
<meta name="dc.Creator" content=" Rada K Massingham " />
<meta name="dc.Subject" content="Knowledge Management; Experience; Return on
investment; Investment appraisal; Investment in knowledge management; Knowledge
management value" />
<meta name="dc.Description" content="Purpose – The paper examines ways that Knowledge
Management (KM) can demonstrate practical value for organizations. It begins by..."/>
<meta name="keywords" content="Knowledge Management, Experience, Return on
investment, Investment appraisal, Investment in knowledge management, Knowledge
management value" />
```

Figure 8. Using HTTP GET to retrieve knowledge from a web page

In term of expert system, KQML provides communication language for retrieving knowledge from other expert system. The knowledge can be retrieved in any format because KQML only manages the communication between expert systems. As described in Figure 9, we can use query *performatives* like *ask-one*.

```
ask-one
    :content <expression>
    :aspect <expression>
    :language <word>
    :ontology <word>
    :reply-with <expression>
    :sender <word>
    :receiver <word>
```

Figure 9. Using specific performative to retrieve knowledge from an expert system

3.3. Adding New Knowledge

The third mechanism is the last step in knowledge sharing between two KBSs. When a KBS found out that it does not have the necessary knowledge to answer user's question, it should find another KBS that withheld such knowledge and retrieve that necessary knowledge. The last step should explain how the first KBS uses the retrieved knowledge. For this last step, there are three possible mechanisms for knowledge manipulation, add the new knowledge, delete existing knowledge, and modify existing knowledge.

For this last step, using the retrieved knowledge to enhance the original knowledge base is not the scope of the OAI-PMH, SPARQL, and KQML. However, dealing with KMS and web page that has no relation between knowledge, adding new knowledge and deleting existing knowledge can be straightforward. While adding and deleting knowledge can be straightforward, modifying an existing knowledge is not. Some restrictions should be considered before modifying an existing knowledge, like which knowledge in the knowledge base should be modified and is the new knowledge contains better knowledge?

Although working on KMS and web page can be straightforward, unfortunately, this does not happen in Expert System (ES). This is due to the knowledge structure within ES that has a locked relation among knowledge that makes it difficult to alter. As an example, an Expert System implementation called CLIPS (*C Language Integrated Production System*) provide an example to determine a type of animal based on user's answers. Figure 10 explains a portion of

the knowledge structure in *animal.clp* example file, where a question and answer pair will determine the path where the Expert System goes.

```
(rule (if species is horns and one.horn is yes)(then type.animal is hippopotamus))
(rule (if species is horns and one.horn is no)(then subspecies is nohorn))
(question one.horn is "Does your animal have one horn?")
(rule (if subspecies is nohorn and fleece is yes)(then type.animal is sheep/goat))
(rule (if subspecies is nohorn and fleece is no)(then subsubspecies is nofleece))
(question fleece is "Does your animal have fleece?")
(rule (if subsubspecies is nofleece and domesticated is yes)(then type.animal is cow))
(rule (if subsubspecies is nofleece and domesticated is no)(then type.animal is
deer/moose/antelope))
(question domesticated is "Is your animal domesticated?")
(answer is "I think your animal is a " type.animal)
```

Figure 10. A portion of knowledge base in Expert System using CLIPS

4. Conclusion and Future Work

This research tries to explore the linkage among the family of KBS, which are KMS, web page, and ES. The exploration is done on knowledge sharing capability perspective, using specific protocol originally designed for each of the system, OAI-PMH, SPARQL, and KQML. The knowledge sharing in this term includes three steps, finding a correct system, finding correct knowledge and adding new knowledge.

Based on the previously discussion, all OAI-PMH, SPARQL, and KQML provide support for the first and second steps of knowledge sharing. However, for the third step, only KMS and web page provide the possibility of knowledge manipulation. This is due to the loosely coupled between knowledge entities, which make an insertion of new knowledge or modification of existing knowledge is possible. However, this is not the same case with Expert System, where there is a relation between knowledge entities. This relation exists for reasoning purposes, so there is no simple way to alter the existing knowledge inside the Expert System. To solve this problem, an intelligent agent is added as a patch to an Expert System to provide knowledge sharing capability. An intelligent agent is supposed to understand KQML and has the ability to enhance the origin Expert System by addition, deletion and updating knowledge [5].

References

[1]   Smith R. *Knowledge-Based Systems: Concepts, Techniques, Examples*. Presented at the Canadian High Technology Show, Schlumberger-Doll Research. 1985.
[2]   Brachman R, Levesque H. Knowledge Representation and Reasoning. San Francisco: Morgan Kaufmann. 2004: 25-28.
[3]   Halpern D. Thought & Knowledge: An Introduction to Critical Thinking. Fourth Edition. New Jersey: Lawrence Erlbaum Associates, Inc. 2003: 24-28.
[4]   Karna N, Supriana I, Maulidevi N. *Knowledge Sharing between Similar Domain Knowledge Management Systems*. AIP Conference Proceedings of the Conference on Fundamental and Applied Sciences for Advanced Technology (ConFAST). Yogyakarta. 2016: 020063: 1-8.
[5]   Karna N, Supriana I, Maulidevi U. Intelligent Interface for a Knowledge-based System. *TELKOMNIKA Telecommunication, Computing, Electronics and Control*. 2014; 12(4): 1096-1104.
[6]   Curtis G, Cobham D. Business Information Systems: Analysis, Design and Practice. Sixth Edition. London: Prentice Hall. 2008: 631-677.
[7]   Karna N, Supriana I, Maulidevi N. *Autonomous Knowledge-based System for Sensor Network*. Proceeding of 3rd International Conference on Information and Communication Technology (ICoICT). Denpasar. 2015: 92-95.
[8]   Karna N, Supriana I, Maulidevi N. *Knowledge Representation for Image Feature Extraction*. Proceeding of Intelligence in the Era of Big Data: 4th International Conference on Soft Computing, Intelligent Systems, and Information Technology (ICSIIT). Denpasar. 2015: 174-181.
[9]   Snowden D. Complex acts of knowing: paradox and descriptive self-awareness. *Journal of Knowledge Management*. 2002; 6(2): 100-110.
[10] Smith P. Knowledge Sharing and strategic capital: The importance and identification of opinion leaders. *The Learning Organization*. 2005; 12(6): 563-573.

[11] Paulin D, Suneson K. Knowledge Transfer, Knowledge Sharing and Knowledge Barriers–Three Blurry Terms in KM. *The Electronic Journal of Knowledge Management*. 2012; 10(1): 81-90.
[12] Uriarte F. *Introduction to Knowledge Management*. ASEAN Foundation. Jakarta. 2008.
[13] Yuliant R, Karna N. *Knowledge sharing filtering on OAI-PMH*. Proceeding of International Conference on Information Technology Systems and Innovation (ICITSI). Bandung. 2016: 1-5.
[14] Metaxiotis K, Psarras J, Papastefanatos S. Knowledge and information management in e-learning environments; the user agent architecture. *Information Management & Computer Security*. 2002; 10(4): 165-169.
[15] Ardichvili A, Page V, Wentling T. Motivation and barriers to participation in virtual knowledge-sharing communities of practice. *Journal of Knowledge Management*. 2003; 7(1): 64-76.
[16] The OAI Executive. Open Archives Initiative - Protocol for Metadata Harvesting - v.2.0. 2015.
[17] Recordon D. *The Open Graph Protocol Design Decisions*. Presentation at the W3C's Linked Data Camp at WWW 2010. 2010.
[18] Finin T, Fritzson R, McKay D, McEntire R. (1994) *KQML as an agent communication language*. Proceedings of the third international conference on Information and knowledge management - CIKM 1994. 1994: 456-462.
[19] Yu C, Zhou W. Knowledge Interchange in Task-Oriented Architecture: For Space Robot Application. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(4): 1857-1867.
[20] Finin T, Weber J, Wiederhold G, Genesereth M, Fritzson R, McKay D, McGuire J, Pelavin R, Shapiro S, Beck C. *Specification of the KQML Agent-Communication Language*. The DARPA Knowledge Sharing Initiative External Interfaces Working Group (draft). 1993.