A Cross-Ontology Mapping Framework for Semantic Web Agents

Osun Olufemi A.
Department of Computer Science
Lagos State Polytechnic
Lagos, Nigeria
E-mail: femiosun@gmail.com
Phone: +2348036709808

ABSTRACT

Heterogeneous ontologies are one of the factors mitigating the proliferation of agents based computing on the Semantic Web. This research proposes a framework for ontology mediation on the Semantic Web that utilizes both direct ontology mapping and upper level ontologies. At the core of the framework is a layer of interconnected Ontology Mapping Agency (OMA) that host agents that map ontologies on the fly on a need based basis. It proposes two set of API specifications, one for connecting agents of Application Agent Platforms (AAP) needing ontology mediation to the OMA and the other for interfacing OMA agents to upper level ontologies. AAPs may connect to more than one OMA for fail-safe redundancy. The framework uses matured Internet technologies and sits right within the Semantic Web infrastructure to provide on the fly ontology mediation as needed. Implementation of the framework with JADE platform is proposed for validation.

Keywords: Cross-Ontology, Mapping, Semantic Web, Agents, JADE, OMA, AAP, Frameworks

iSTEAMS Proceedings Reference Format
Osun, O.O. (2019): A Cross-Ontology Mapping Framework for Semantic Web Agents. Proceedings of the 17th iSTEAMS Multidisciplinary Research Nexus Conference, D.S. Adegbenro ICT Polytechnic, Itori-Ewekoro, Ogun State, Nigeria, 21st – 23rd July, 2019.P 141-151
www.isteams.net - DOI Affix - https://doi.org/ 10.22624/AIMS/iSTEAMS-2019/V17N2P18

1. INTRODUCTION

Tim Berners-Lee, the creator on the World Wide Web (WWW) in a ground breaking article envisioned an extension of the WWW (Berners-Lee, Hendler & Lassila, 2001), where intelligent software agents head out in the WWW doing useful work for their human owners, like automatically book flights and hotels for our trips, make doctor's appointments for us and do searches for us from a variety of source and provide us with a single customize answer (Hendler, 2001). The WWW as we know it is a web of documents, the Semantic Web on the contrary, is a web of data that enables people to create data stores on the Web, build vocabularies and write rules for handling data – it is a web of data with relationship established between them (W3C, 2015), that explicitly restructures the Web in an explicit machine-readable way (Berners-Lee and Fischette, 1999).

Machine-readability makes it possible to delegate a lot of routine works, bank transactions, time-tabling agreements, making appointment, to autonomous software agent – Semantic Web agents or simply agents in this case. (Genesereth and Ketchpel, 1994), (Wooldrige and Jennings, 1995) and (Russell and Norvig, 2003) all agree that an agent is essentially a specialized autonomous software component that provides interoperability interface to an arbitrary system and or behaves like a human agent, working for some client, in pursuit of its own agenda (Bellifemine, F., Claire, G.& Greenwood, D., 2007). The important aspect of an agent is autonomy, which also implies intelligence.

Though the Semantic Web is slowly growing, both in technology and applications, for it to really take-off certain things needs to be put in place, principal of which are a common language for representing data that could be understood by all agents (Agent Communication Language, ACL), set of statements that translate data from disparate data sources to common terms and can establish relationship between the terms (Ontology language), rules that allow agents to reason about the information described by those terms (Inference rules), and systems for establishing trust among cooperating and competing agents in the Semantic Web ecosystem.

An ontology is an explicit specification of shared conceptualization of a domain (Gruber, 1993). An ontology consist of a vocabulary - word, called terms of the domain of discourse, their meaning and the relationships between. There are a number of languages for specifying ontologies, more important is that an agent will commit to an ontology and will not ordinarily "understand" a different ontology, even though it might be specified in the same language as the ontology it is committed to. For an agent to effectively collaborate with other agents, as envisaged on the Semantic Web, it must not only be able to parse the syntax of the ontology language of the other agents, it must also be able to understand the meanings embedded in the other agent's ontology – the semantics - and be able to reason over them. W3C offers a range of languages and techniques for building ontologies, which include Resource Description Language (RDF), RDF schema, Simple Knowledge Organisation System (SKOS) and Web Ontology Language (OWL), depending on the complexity and rigour of a specific application (W3C, 2015).

Even when the ontology languages are the same, for agent to interact effectively on the Semantic Web on behalf of their human owners, they must not only able to reasonable "understand" the ontology of other agents they intend to interact with, the "comprehension" should happen transparently at agent run time, without taking the agent's attention away from the task at hand.

1.1 Statement of the Problem

For the Semantic Web to really take off, a number of current limitations need to be overcome, principal of which is the difficulty of agents to effectively collaborate with other agents that use ontologies different from their own. A number of approaches have been used to address this problem, principal of which are direct ontology mapping and mapping using high order ontologies. These approaches map heterogeneous ontologies either manually or automatically before agents initialization. The problem with these approaches is that agents must know the ontologies of all agents they will collaborate with beforehand and have their ontologies mapped ahead. Ideally an agent need not be aware of the ontology of any other agent until collaboration is initiated, in fact agents need not know the identity of other agents they might need to collaborate with until their services are needed. Since many agents committed to different ontologies may provide the same service, if a requesting agent is tied to a particular agent, it limits its versatility to solve problems and draws back on the vision of "a Web of autonomous agents".

This work proposes a framework for Semantic Web infrastructure that combines direct mapping with upper ontology, and allows agents to initiate collaboration with any other agents at the point it needs their services, while it undertakes its own task, without being aware of their ontology beforehand. It maps the ontologies on the fly, to provide an end to end mapping process that does not interrupt the performance of agents' tasks.

1.2 Aim and Objectives

As useful and desirable as the Semantic Web is, it cannot attain its full potential without reasonable high agentry (population of software agents active and doing useful work consistently). Agentification (the rate at which software agents populate a system) is hindered by a number of things, cardinal of which is the difficulty of agents to collaborate with other agents that are committed to ontologies different from their own. The aim of this work is to further advance the vision of Semantic Web by facilitating cross-ontology collaboration among agents in the Semantic Web.

The Objectives are:

i. To extensively review existing literatures on the Semantic Web, agent systems and agent communications
ii. To identify advances that have been made by previous researchers with respect to the Statement of the Problem
iii. To identify the specific failures and successes of earlier works
iv. To build on the successes of earlier works to design and implements a system that facilitate cross-ontology collaboration among Semantic Web agents
v. To design tests to validate the implemented system.
vi. To implement the tests to determine the success of the work.

1.3 Significance of the Study

Heterogeneity of ontologies has contributed in no small way to the limitations of agents' interactions on the Semantic Web. Though there are tools and techniques, of varying efficiency, for mediating ontologies, such have not been complete integral parts of the Semantic Web infrastructure. What we have so far are some tools like super ontologies like WordNet, DbPedia, Linked data Objects and upper level ontologies like Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) and Suggested Upper Merged Ontology (SUMO) that can be used to aid the process.

This work proposes a complete system for mediations and alignment for cross-ontology collaboration between agents that is designed as an integral part of the Semantic Web infrastructure, which does not require agents to consult with off-Web processes to meet their ontology mediation needs, much like the Domain Name System.

It is believed that this system will not only provoke further researches in this area, but will move the Semantic Web nearer to its goal of a web of autonomous agents collaborating to get useful work done for humanity.

2. LITERATURE REVIEW

2.1 Ontology

(Studier, Benjamins & Fensel, 1998), extended (Gruber, 1993) definition of ontology with "machine-readable", defining ontology as an explicit machine-readable specification of shared conceptualization. An ontology typically consist of terms (words describing concepts in the domain), a taxonomy, usually arranged in hierarchy of relationship between the terms, axioms and some inference rules for reasoning over the ontology. Ontologies are constructed using ontology languages that have their own syntaxes, semantics, degrees of expressiveness and reasoning capabilities.

2.2 Ontology languages

In order to be useful for detailing a precise formal specification of a domain, the main requirements for an ontology language are:

- A well defined syntax
- A well defined (formal) semantics
- A efficient reasoning support
- Sufficient expressive power
- Convenience of expression

(Antoniou, Franconi & Harmelen, 2005).

The desirability of the first and the last are well researched and understood from the field of programming languages. A well defined syntax is a necessary condition for machine processing and convenience of expression is an intuitive assistance to developers and contributes to uptake of a language. Nevertheless ontology languages should provide primitives for specifying concepts (usually organized in taxonomies), relations, functions, axioms and instances (Gomez-Perez and Corcho, 2002). While the ontology is largely a static structure, the domain model is a specific instantiation of the ontology. Semantics embodies the knowledge contained in an expression, its meaning. Formal semantics does not leave room for subjectivity or intuition; it defines precisely the meaning of the knowledge expressed by enforcing it as a set of constraints over the domain, such that any possible instantiation of the domain should necessarily conform to the constraints expressed in the ontology. Given a statement in an ontology, semantics evaluates which of the domain model the statement is compatible with. The statement is true in a domain model (i.e. an instantiation of the domain) if the instantiation is compatible with the statement (Harmelen and Fensel, 1999).

Reasoning is the process of deriving valid deductions from an ontology. For a deduction to be valid, it must be true for every model of the ontology (Antoniou, Franconi & Harmelen, 2005). Reasoning and semantics are based on some logic systems, Predicate Logic, First Order predicate Logic and Description logic have been used in Ontology languages (Obitko, 2007). Formal semantics and reasoning support is usually provided by mapping an ontology to any of the selected logical formulations. Ontology languages should provide the following standard reasoning supports:

- Check ontology and knowledge consistency
- Check for unintended relationship between classes
- Derive explicitly all the statements that are true in the ontology to better understand its properties
- Reduce redundancy in an ontology by discovering equivalence, reusing concept descriptions and refining definitions
  (Antoniou, Franconi & Harmelen, 2005).

Note worthy is the tradeoff between the expressiveness of an ontology language and its reasoning support. Less expressive languages can support more powerful reasoning capabilities, while the more expressive ones are less computational capable.

Over time many ontology languages have emerged for the Semantic Web, each serving a particular need relevant to its specific period in the evolution of the Semantic Web. XML Ontology Language (XOL) is based on XML and was developed by American bioinformatics community for exchange of ontology definition between heterogeneous software system (Karp, Chaudhi & Timere, 2002), Simple HTML Ontology Extension (SHOE) was first used to embed semantics into HTML pages before blossoming into a full fledge ontology language in its own right (Luke and Heflin, 2002), Ontology Management Language (OML) is based on SHOE and shares many features with it (Kent, 2002), Resource Description Framework (RDF) (Lassila and Webick, 1999) and RDF Schema (Brickley and Guha, 1999) are two languages specified by and adopted by W3C, Ontology Inference Language (OIL) was developed based on RDFS by the European Union and DARPA Agent Management Mark up Language (DAML) by the American Department of Agency. Both were submitted to the W3C and merged to produce a standard language DAML+OIL. Further work on DAML+OIL specification by W3C produced Web Ontology Language (OWL) and its versions. RDF, RDFS and OWL are W3C standards for specifying ontologies for the Semantic Web.

Ontologies require different levels of modeling requirements, expressiveness and reasoning. These are shown for the standard Semantic Web ontology languages in Table 1.

Table 1. W3C Standard Semantic Web Ontology Languages

Language	Model, reasoning and expressiveness
RDF	It provides primitives to explicitly model classes, their properties and their taxonomy, but cannot model relationship between properties and resources. It can also recognize non-explicit information. Based on semantic network
RDFS	It can in addition to RDF define relationship between resources and attributes. Its has good reasoning ability, its formal semantics and reasoning are based on description logic.
OWL LITE	Adds the possibility to express axioms with limited use of properties to define classes in addition to features of RDFS.
OWL DL	Supports maximum expressivenes with good computational abilities
OWL FULL	Supports full expressiveness with computation no guaranteed

2.3 Ontology mediation and aligning and mapping

Inspite of W3C standardization on RDF, RDFS and OWL, there are other languages for ontologies, even if one language is used, it cannot be expected that individuals and organizations on the Semantic Web will agree on a common vocabulary, ontology or technology. (Uschold, 2000). Information heterogeneity consists of syntactic, structural and semantic heterogeneity (Sluckenschmidt and Harmelan, 2005). (Wiederhold, 1992) discussed many advanced solutions for syntactic heterogeneity, in the field of Semantic Web, (Gruber, 1993) also discussed a technique used in Ontoligua; Ontoligua is a system that translates an ontology from one syntactic representation to another, while preserving the semantics.

Ontology mediation is the reconciliation of structural and semantic differences between ontologies. Ontology mediation entails three processes; ontology aligning, ontology mapping and ontology merging. Ontology aligning is the discovery of correspondences between ontologies, ontology merging is concerned with the creation of s single ontology from two or more different ontologies, whereas ontology mapping is the representation of semantic correspondence between similar elements in different ontologies. (Noy, 2004) and (Doan and Halevy, 2005), seems to consider aligning as a part of mapping when they defined mapping as establishing correspondences among ontologies and determining the set of overlapping concepts, concepts that are similar but have different names and or structure.

(Bruijn et al, 2006) identified the major issues in ontology mediation as the location and specification of overlapping and mismatched of concepts, relations and instances in different ontologies. While correspondences are easy to deal with (the more correspondences we are able to establish, the easier the work of mediation and the more consistent the result of the process would be), mismatches are more difficult to resolve, (Klein, 2001) identified mismatch in style of modeling, terminological mismatch and encoding mismatch as the three possible types of mismatches that can occur in ontologies.

Modeling style mismatch may be in terms of using different paradigms to describe the same concept (e.g modeling time as intervals rather than points in time) or describing concepts differently (for example using attributes rather than subclasses for a concept). Terminological mismatches occur as synonyms (when different terms are used to equivalent concepts) and homonyms (when the same term is used to for different concepts). Encoding mismatch occur when values in ontologies are encoded differently (for example using litres rather than cubic centimeter). Another type of mismatch not mentioned by (Klein, 2001) is when a concept in an ontology is not modeled at all in the other ontology. Ontology mapping attempts to express one ontology in the vocabulary of another, but may also produce a new ontology that is a merger of the mapped ontologies.

2.4 Ontology mapping techniques

Ontology mapping techniques are based on centralized (external resources) and decentralized (direct) architecture (Ramar and Gurunthan, 2016). Centralized mapping approaches use background knowledge external supers ontologies, like WordNet, DBPedia, Linked Open Data (LOD). Because of the size of these super ontologies it is assumed that correspondences for the entities in the subject ontologies will be found in them, and thus used to negotiate agreements. Upper level ontologies like Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) (Gangemi and Guarino, 2003) and Suggested Upper Merged Ontology (SUMO) (Pease, Niles & Li, 2001) are other central external resources that are used. The assumption with upper levels ontologies is that at topmost hierarchies of the taxonomies of ontologies of multiple domains yield a small number of categories that are the same in all these domains, if this categories are defined and axiomised as a upper level ontologies, they serve as reference for mapping ontologies of that domain (Hoendorf, 2010).

The main categories of decentralized mapping approaches are, structure based, terminology based, instance based, semantic reasoning, hybrid method and ontological mapping using background knowledge. The common denominator in all the decentralized approaches is that they use information (annotations and structural information like subclasses, superclasses, relationship, domain and range, instances of classes and graph structure) available in the ontologies to determine correspondences and mismatches. (Ramar and Gurunathan, 2016). Different algorithms boasting varing degrees of successes have been proposed for all these methods (Brahma and Refoufi, 2015).

2.5 Agents and agents systems

The goal of Artificial Intelligence had always been to replicate autonomous, intelligent and useful entities. As far back as 1977 (Hewitt, 1977) espoused the concept of “actors”; self contained concurrently–executing objects that encapsulate internal state and could respond to messages from similar objects. In recent times we have seen the emergence of software systems that behave if not exactly but quite close to this vision. Jasper (Davies, Weeks, & Revett, 1997), CMU Visitors’ Hosting System (Sycara, 1995) and ADEPT (O’Brien and Wiegand, 1996) are points in reference.

(Nwana and Ndumu, 1997) identified the important features of agents as collaboration, learning and autonomy and put agents into types based on the emphasis they put on these characteristics. Collaborative agents, that emphasis autonomy and collaboration ahead of learning (O’Brien and Wiegand, 1996) and collaborate with other agents in its ecosystem to get useful work done for its owner. Interface agents emphasis autonomy and learning, they act as assistance to their owners and go between their owners and a third system the owner is trying to learn or use, providing useful guides and information (Maes, 1994), an example of interface agent is the ill-fated Microsoft Office Assistant. Mobile agents are agents that can leave their host systems and migrate to foreign hosts to perform tasks for their users and “return home”. Issues of privacy, security and rogue behavior are of importance consideration with this type of agents. Information gathering agents gather, manage and process information for their owner from distributed source without leaving their native hosts. Jasper is such an example.

Reactive agents provide no internal representation of their environment or maintain a percepts history of interaction with the environment, but are simple stimulus-reaction devices (Russel and Norvig, 2003), they do not have a fixed architecture, the most popular architecture is one proposed by (Brooks, 1999), which is based on Augmented Finite State Machine (AFSM). Very few applications outside of simple games exist for reactive agents. Hybrid agents combine two or more of these architectures and leverage on their strengths to achieve its tasks. With the prolificacy of agent technologies, ontologies and increasing demands for agents to collaborate, (Genesereth, and Ketchpel, 1994) argues the need heterogeneous agent that combine the architecture of one or more agent types. Agent-based software engineering is a field that has emerged for developing heterogeneous agents.

Agents by their nature should be autonomous, proactive and social (Bellifemine, Caire, G. & Greenwood, 2007). Autonomy is the ability to carry out independent tasks of varying complexities, proactivity makes it possible for agents to take initiative and carry out needed task without explicit directives from their users, sociability implies it must be able to communicate with other agents to achieve its goal. Along with this core requirements are needs for agent registrations, service advertisement and life cycle management for agents, communication and coordination. Agent systems exist to provide these low level "plumbing" activities, so developer can concentrate on the business logic of the agent. Agent systems that support many agents are called Multi Agent Systems (MAS). The most important services provided by MAS are communication and coordination. Agents need to communicate with the user, other agents and system resources. Inter-agent communication is by special Agent Communication Languages (ACL), that are based on Speech Act theory of (Searles, 1969).

An ACL provides means for the exchange of information and knowledge between agents, ACLs must be both agent and semantics independent. (Genesereth and Ketchpel, 1994) reduced agency simple to the ability to communicate with an ACL. Knowledge Query and Manipulation Language (KQML) was the first ACL developed (Mayfield, Labrou & Finn, 1996). The new standard for ACL is a Foundation for Independent Physical Agents (FIPA) specification which is built on KQML called FIPA-ACL; this is the most commonly used ACL today (Labrou, Finn, & Peng, 1999). FIPA-ACL is fully implemented in Java Agent Development Framework, a MAS, also by FIPA.

FIPA has defined a specification for MAS for easy interoperability. FIPA MAS specification is implemented by Java Agent Development (JADE) Framework.

3. METHODOLOGY

This work proposes a system of independent but interconnected ontology mapping services called Ontology Mapping Agencies (OMA). An OMA is an MAS platform consisting of agents that implements ontology mapping algorithms. All the agents may implement the same algorithm or different algorithms. The OMA also will also be connected to an upper level ontology (ULO). The purpose of the upper level ontology is to refine mappings done by an OMA agents to give much better fitted mappings. The AAPs are also MAS platforms that host application agents that collaborate on behave of their owners.

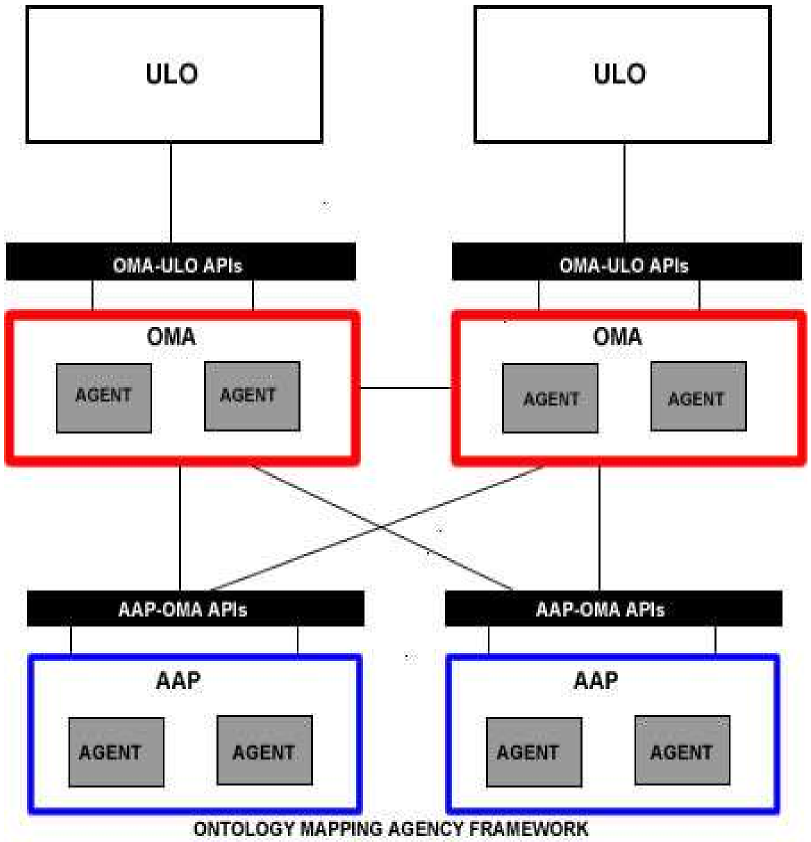

Figure 1. OMA Framework

An agent in the AAP that requires collaboration with another agent in the same of different AAP will send an ACL message to the AAP's Directory Facilitator (DF) asking for an agent that can service its request. The ACL message among other things will contain the ontology used by the requesting agent. If the DF has an agent registered that can service the request, it sends back an ACL message that contains among other things, the identity and ontology of the agent. If the ontologies are the same, collaboration is initiated. If the ontologies are different, the requesting agents requests for the ontology of the servicing agent and packages it with its own ontology to an OMA for mapping, using AAP-OMA APIs. The OMA gives the job to one of its mapping agents; the mapping agent uses its local mapping algorithm and uses OMA-ULO APIs to access ULO resources to map the two ontologies then sends the mapped ontology back to the two requesting agents. The two agents can then collaborate using this common ontology. This framework will be validated by building an OMA using JADE. The OMA will host three mapping agents that implement different direct mapping algorithms, and connect to SUMO for mapping refinement.

The AAP will be another JADE MAS platform that hosts four other agents, a client agent, an auction agent, and two bank agents. The client agent shall represent a client who desires to purchase some items from the auction house, the auction agent represents an auction house that may have the desired item and each of them has one of the bank agents. All the agents except the two bank agents have different ontologies implemented in OWL DL. The client agent shall attempt to buy some item for its owner by collaborating with these other agents, using the OMA to mediate their ontologies.

JADE is an open source project MAS platform released under Library Gnu Public License (LGPL). (LGPL) gives the privilege to use the source code and amend it as needed, so far the final work is released back into the public domain. It is programmed in Java, a Apache Application server will be used at both OMA and AAP sites. All these tools are available free under open source licenses.

4. EXPECTED OUTCOME

The following outcomes are expected from this work:

- An extended JADE implementation to carry out the functions of an OMA
- A set of APIs that facilitate open and transparent connection with an upper level ontologies
- A set of specifications for APIs that AAP can use to negotiate with an OMA in transparent and consistent manner.
- An implementation of the set of the AAP APIs for JADE.
- A set of specifications for APIs to connect OMA to a ULO.
- An implementation of the OMA-ULO for JADE
- Possible a set of communication protocols for the inter components connections in the framework, if the native protocols of the test agent platform proves inadequate.

5. CONCLUSION

It is expected that an OMA by combining both direct and an upper ontology mapping facilities will provide by far by aligned and mapped ontologies than using just one approach. The framework sits right within the Semantic Web infrastructure and uses matured and tested Internet technologies, this promises a high likelihood of success and take up. The API specifications are open to encourage tool makers provide API implementations for their own implementation of the framework.

REFERENCES

1. Antoniou, G., Franconi, E. & Harmelan, F. (2005). Intorduction to web ontology languages. First International summer School Conference, Msida. (2005).
2. Bellifemine, F., Caire, G. & Greenwood, D. (2007). Developing Multi-Agent Systems with JADE. John Wiley & Sons, Ltd. (2007)
3. Berners-Lee, T and Fischette, M. (1999). Weaving the web: the original design and ultimate destiny of the web by its inventor. Hapers San Francisco (1999).
4. Berners-Lee, T., Hendler, J and Lassila, O. (2001).The semantic web.The Scientific America, May 2001, 33-43.
5. Brahma, B., Refoufi,.A. (2015). Ontology matching algorithms. IPAC' 15, Betna, Algeria(Nov. 23-25, 2015).
6. Brickley, D. and Guha, R. (1999). Resource Description Framework (RDF) Schema W3C Recommendation. Retrieved from http://www.w3.org/TR/PR-rdf-schema in January 2015.
7. Brooks, R.A. (1999). Intelligence without representation. Artificial Intelligence, 47, 139-159 (1999).
8. Bruijn, J., Ehrg, M., Feier, C., Martin-Recuerda, F. , Sharfffe, F., Weiten, M. (2006). Ontology mediation, mapping and aligning.
9. Davies, N.J., Weeks, R., Revett, M.C. (1997). Information agents for the world wide web.BT Laboratories, Martlesham Heath, Ipswich, Suffolk, IP57re, UK.

10. Doan, A., Halevy, A.Y. (2005). Semantic integration research in database community: a brief survey. AI Magazine, vol. 26, 83-94.
11. Gangemi, A. and Guarino, N. (2002) Sweetening ontologies with DOLCE.13th International Conference on Knowledge Engineering and Knowledge Management , EKAW 2002.
12. Genesereth, M. and Ketchpel, S. (1994). Software agents. Communications of ACM, 37(7). 48-53.
13. Gomez-Perez, A., and Corcho, O. (2002). Ontology languages for the semantic web. IEEE Intelligent Systems (2002)
14. Gruber, T. (1993). A translational approach to portable ontology specification. Knowledge Acquisition vol. 5 issue 2. 199-220. (1993).
15. Harmelen, F. and Fensel, D. (1999). Practical knowledge representation for the web. Proceedings of IJCA '99 Workshop of Intelligent Information Integration. (1999).
16. Hendler, J. (2001). Agents on the semantic web. IEEE Intelligent System, vol. 16 Issue 2, (March 2001), 30-32.
17. Hewitt, C. (1997). Viewing control structures as patterns of passing messages. Artificial Intelligence, 8. No.3 323-364 (1997).
18. Hoehndorf, R. (2010). What is an upper level ontology. Retrieved from http://ontogenesis.knowledgeblog.or/740 on 15/07/19.
19. Labrou, Y., Finn, T. and Peng, Y. (1999). Agent Communication languages: the current landscape. IEEE Intelligent Systems vol 14, issue 2, 45-52.
20. Lassila, O and Webick, R. (1999). "Resource Description Framework (RDF)model and syntax, W3C recomendation. Retrieve from http://www.w3.org/TR/PR-rdf-syntax in January 2015.
21. Luke and Heflin. (2002). SHOE 1.01 proposed specification, SHOE Project. Retrieved from http://www.cs.umd.edu/projects/plus/SHOE/spec01.htm in January 2002.
22. Maes, P.(1994). Agents that reduce work and information overload. Communications of ACM 37, No. 7, 31-40.
23. Mayfield J., Labrou, Y and Finn, T. (1996). Evaluating KQML as an agent communication language. InWoodridge, M., Muller, J. and Tambe, M. (1996) (eds.) Intelligent Agents II (LNA 1037, Springer-Verlag, Heidelberg, 347-360.
24. Noy, N.F(2004). Semantic integration: a survey of ontology-based approaches. ACM sigmod Rec., 33, 65-70.
25. Nwana, H.S., Ndumu, D.T. (1997). An introduction to software agent technology. Chapter in book
26. O'Brien, P. and Wiegand, M.(1996). Agent of change in business process management. BT Technology Journal, 14, No.4 . (1996).
27. Obitko, M. (2007). Formal representation. Retrieved from http://www.obitko.com/tutorial/ontologies/sematicweb/ on 19/07/19.
28. Pease, A., Niles, I., Li. J.(2001). Suggested upper level merged ontology: a large ontology for the semantic web and its applications. AAAI Technical Report WS-02-22 (2002).
29. R. Kent(2001). Conceptual Knowledge Modeling language. Retrieved from http://www.ontologos.org/CKML on 15/06/2005.
30. Ramar, K. and Gurunthan, G .(2016). Technical review of ontology mapping techniques. Asian Journal of Information Technology vol. 15 issue 4, 676-688, (2016).
31. Russell, S and Norvig, P (2003). Artificial Intelligence: a modern approach, 2nd edition. Prentice Hall (2003).
32. Searles, J. (1969). Speech act. Cambridge University Press, Cambridge MA.
33. Sluckenschmidt, H., and Harmelan, F. (2005). Information sharing on semantic web. Springer, New York, ISBN: 97830540205944.

34. Uschold, M. (2000). Creating, integrating and maintaining local and global ontologies. Proceedings of the 1st Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI-2000), Berlin, Germany.
35. W3C (2015). Ontology. Retrieved from http://www.w3.org/standards/semanticweb/ontology/ on 12/07/19.
36. W3C (2015). Semantic web. Retrieved from http://www.w3.org/standards/semanticweb/ o12/07/19.
37. Wiederhold, G. (1992). Mediators in the architecture of future information systems. Computer,25, 38-49.
38. Wooldridge, M and Jennings, N. (1995). Intelligent agents: theory and practice. Knowledge Engineering Review, vol. 10, issu e 2. 115-152 (1995),