Ontology Building: An Integrative View of Methodologies

Gurinder Pal Singh Gosal
Department of Computer Science
Punjabi University
Patiala
e-mail: gosal.gps@gmail.com

Abstract—Ontologies are being developed and used in many disciplines now a day and they have become a key tool of data integration and knowledge representation in different domains of interest. The ontology building process identifies the stages through which the ontology should go through during its development. There is a certain set of activities to be performed in each stage of the ontology development process and different methodologies have been proposed by researchers for formalizing the different stages. The present paper investigates the most representative methodologies used in the ontology development to look at the different activities that are performed during the process of ontology development. The paper further attempts to provide an integrative view of the most representative methodologies used in the ontology development to look at the set of different activities that can be performed during the process of the ontology development.

Keywords-*Ontology, Semantic Web, Ontology Engineering, Ontology Life Cycle, Ontology Development.*

*****

I. INTRODUCTION

Ontology in the Computer Science has its origin in 1990's, when there were efforts for devising the new ways of building knowledge based systems with using the knowledge extracted from reusable components. The DARPA Knowledge Sharing Effort [1] was one of these endeavors in this direction. Ontologies have moved a long way ahead after these initial efforts of integrating the knowledge and many ontologies as well as applications based on them are widely available now.

One of the simplest and frequently used definitions of Ontology is given by Thomas Gruber in his extensively cited paper "*Toward Principles for the Design of Ontologies Used for Knowledge Sharing*". He defines ontology as an explicit specification of a conceptualization [2]. The reasons for building ontologies in a particular domain are, to make a common understanding of the structure sharable among different stakeholders in the domain, to enable reuse of the domain knowledge, to explicitly describe the domain, delineate the domain knowledge from operational commitment, and to explore the domain knowledge [3]. Ontologies are being developed and used in many disciplines, such as, Semantic Web, Artificial Intelligence, Natural Language Processing, Software Engineering, and various biological and bio-medical fields, so on and so forth.

The process for the development of ontologies can be viewed as a set of all activities that are involved in the building of ontologies, organized in the systematic and well defined steps or phases. In the development of ontologies, as well as in the applications based on those ontologies, there are variety of tools and technologies available which can help ontology engineering and management activities. The guidelines and principles for the ontology development process are not very definitive. Different researchers have identified diverse activities which are performed to accomplish the task of ontology development and they do that by following the different approaches. At present, as each group is trying to apply its own methodology, the need is to have a more integrated view of methodologies to arrive at a more pragmatic representation of ontology development process.

In the next sections of the paper, we look at what the ontologies are? What is the anatomy of ontologies? What classifications of ontologies exist and ontology languages. Then in the subsequent section for "Ontology Engineering", we look at the various methodologies put forward by different researchers to develop ontologies and in the last section the projected integrative view of set of activities involved in the ontology development is discussed.

II. ONTOLOGY

The term ontology originated from a branch of philosophy called metaphysics which defines it as a systematic way of existence and Computer Science borrowed this definition from it [4]. World Wide Consortium (W3C) (*http://www.w3.org/*) has defined ontology in the following words: *"An ontology defines the terms used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information (….) Ontologies include computer-usable definitions of basic concepts in the domain and the relationships among them (…). They encode knowledge in a domain and also knowledge that spans domains. Ontologies are considered to make that knowledge reusable*" [5].

Ontologies enable the integration, mining, and reasoning over diverse data sets by conceptually representing knowledge, which makes them distinct from relational databases [6, 7]. The ontologies have become a key tool of data integration and knowledge representation in different domains of interest. This is often seen that the data sources of interest to a community in a specific domain are often large, dissimilar in structure, format and content. They are frequently distributed across many resources, separately controlled, and rapidly changing. The ontologies are frequently used to deal with the heterogeneity of database schemas of different information sources by providing a shareable, consistent and formal description of the semantics [8]. A well identified, conceptualized, designed and populated ontology representing knowledge about a particular domain can be the basis for a range of applications serving that domain as well as the associated areas that are built upon the knowledge weaved in ontology [9].

III. ONTOLOGY STRUCTURE AND CLASSIFICATION

Ontologies are made up of classes, relationships and individuals to integrate and represent knowledge about the domain of discourse.

- **Class:** Classes in the ontologies represent the main concepts in the field of interest. The example of a class in an ontology in the biomedical domain can be a *Gene* class which captures genes of particular organism as a concept. The specific genes become the instances of *Gene* class.
- **Sub-Class Axioms:** For more steady representation of domain of discourse, there can be further hierarchy of subclasses under the parent classes and this hierarchy is usually described as taxonomic hierarchy. For example, class X, is subclass of class Y and we say that class Y subsumes class X. This subsumption, referred to as "extensional subsumption.", means all members of X are included in Y.
- **Relations:** To describe the internal organization of concepts and to provide further information, relationships or properties in ontologies are introduced, as the concepts in themselves are not able to fully describe the domain. Concepts represented in the form of classes can further be described with various features and characteristics of the concepts by using the properties in the ontologies. The properties in the ontologies can be the data properties or the object properties, used for further characterizing the instances of classes or showing the relationships between instances of various classes. For example, the *Mutation* class can have a property *hasPrimarySite* to show that a particular mutation has a primary site of the cancer occurrence such as "skin" and the same time instances of the Mutation class can have a property *implicatedIn* to show relationship with instances of other class *Disease.*
- **Individuals:** The elements or entities in an ontology are represented by individuals or instances or members. Defining an instance of a class requires selecting a class, forming an individual instance of that class, and loading the instance values. For example, we can create an individual instance "ABL", a gene, to represent an instance of the Gene class.
- ***Domain and range restrictions* (on relations):** For a certain relation the ontology might restrict which kind of entities can stand at the domain and range positions, respectively. For example, for relation *implicatedIn*, at the domain position we would expect a mutation, while at the range position we would expect some type of cancer.
- ***Cardinality constraints*:** We would for example like to say that each member of a *class* can be in relationship with the member of other class in at most once by attaching a cardinality constraint to the corresponding relation in the ontology.

The ontologies are classified by the researchers into various categories according to type. An upper ontology, also known as a top-level ontology, foundation ontology or generic ontology, is an ontology which describes very general concepts and that can be applicable to several domains [10]. This type of ontology is created to support very broad semantic interoperability between a large numbers of ontologies accessible under it. The Suggested Upper Merged Ontology (SUMO) [11] and its domain ontologies form the largest formal public ontology that are being used for research and applications in search, linguistics and reasoning. Basic Formal Ontology (*BFO)* [12] grows out of a philosophical orientation and is focused on the task of providing a genuine upper ontology which can be used in support of domain ontologies developed for scientific research, as for example, in biomedicine within the framework of the OBO Foundry.

Domain ontologies are developed to represent the knowledge about specific domains like biomedical ontologies [10]. Protein ontology (PRO) [13] is an example of the domain ontology. The ontologies can also be Task ontologies used to accomplish certain tasks and Application ontologies are used to extend specific applications [10]. Application ontologies are used to describe concepts that are often specializations of domain and task ontologies. There are other types of classifications of ontologies that vary in scope and description, level of hierarchy and the level of formalism and above types may appear in similar ways in these.

IV. ONTOLOGY LANGUAGES

The Web Ontology Language (OWL), which has been recommended by World Wide Consortium W3C, has made its place as a standard ontology language of the Semantic Web. OWL can be used to build ontologies that are explicit representations of terms and their interrelationships. The expressive power of OWL in terms of providing facilities to represent machine interpretable content on the Web is greater as compared to earlier recommendations of XML, RDF, and RDF-S. The Web ontology languages DAML and OIL were revised to give richer knowledge representation language to be known as the Web Ontology Language (OWL). The OWL language comes in three sublanguage groupings (shown in *Fig. 1)* that are differentiated on the basis of the expressive power provided by each group.

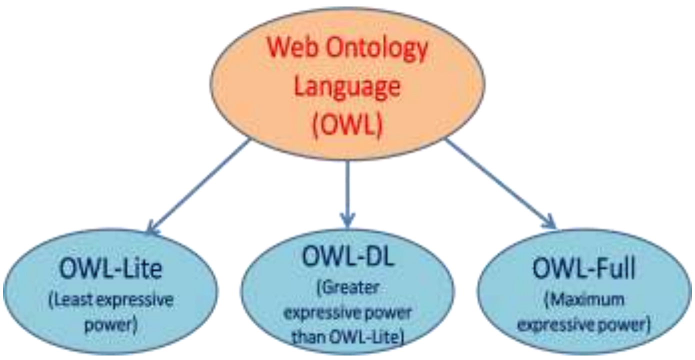

Figure 1. Web Ontology Language (OWL) and its three sublanguages

*OWL-Lite***:** This is the sublanguage of OWL which is having the least expressive power and it supports the community of users who require only a simple class hierarchy and simple constraints. Because of its lesser complexity, it is easier to provide tools for supporting OWL-Lite than the other two sublanguages.

OWL-DL: This sublanguage of OWL has greater expressive power than OWL-Lite and gets its name, DL, because of its use of Description Logic. This sublanguage of

 ISSN: 2321-8169
4677 - 4683

OWL supports the user community which requires a highly expressive language that is at the same time computationally complete and decidable. The computational completeness means that all computations can be completed in finite time and decidable means that all conclusions are computable.

OWL-Full: This sublanguage of OWL has the maximum expressive power among all OWL sublanguages and is preferred by the user community who want maximum syntactic freedom of RDF, although it may not be guaranteed for computational completeness. It is not possible to do automated reasoning on OWL-Full ontologies.

The practical realization of OWL-Full is not available to provide different conceptualizations defined in it in terms of reasoning tools. Which sublanguage of OWL is to be used depends upon the requirements of the user. If more expressive power is required, then OWL-DL may be preferable to OWL-Lite. Owl-Full may be advisable when users require even more meta-modeling capabilities.

V. ONTOLOGICAL ENGINEERING

Building ontology is not a trivial task especially if the ontology has to be truly representative of the domain of discourse. The overall ontology development process has to pass through many stages before quality ontology can be produced. There have been many efforts in the past by the researchers to formalize the set of activates involved in the development process of ontologies by introducing different methods and methodologies. This has given the birth to a new field of Ontological Engineering.

Ontological Engineering is a relatively new field and it refers to the set of activities that concern the ontology development process, the methods and methodologies for building ontologies, and the tool suites and languages that support them [14]. The first of the efforts to formalize the process of developing ontologies was proposed in 1995 by Uschold and King [15] and that was further refined later by Uschold and Grüninger [16]. Methodology by Uschold and King was based on their understanding in developing the *Enterprise Ontology*. The methodology proposed by them covered many aspects of the lifecycle for ontology development. The guidelines provided by them for developing ontologies included four phases: (i) Identify Purpose, (ii) Build ontology, a step further divided into three steps of ontology capture, coding and integrating existing ontologies, (iii) Evaluate, and (iv) Document. About identifying the main concepts in the ontology, Uschold and King have described three approaches: a top-down approach, a bottom-up approach and a middle-out approach. The approach in which abstract concepts are identified first and then specialized into more specific concepts is termed as a top-down approach and the approach in which the more specific concepts are identified initially and then generalized into more abstract concepts is said to be bottom-up approach. The third approach is a middle-out approach in which most important concepts are identified first and then specialized and generalized as per requirements. They point out the advantage of middle-out approach that it makes easier to relate terms in the different areas more precisely and also likely to reduce the potential for rework.

Another methodology was proposed by Grüninger and Fox [17] after developing the TOVE project ontology (Toronto Virtual Enterprise project ontologies at the University of Toronto Enterprise Integration Laboratory). The methodology proposed by them was motivated by usage of first order logic in the development of knowledge based systems. This methodology requires that initially an informal description is made of the specifications and then this description is formalized in the later steps. The TOVE (Toronto Virtual Enterprise) ontologies that are built following this methodology include: Project Ontology, Enterprise Design Ontology, Scheduling Ontology, or Service Ontology.

Bernaras et al. [18] presented a technique used to build an ontology on electrical networks at the 12th European Conference for Artificial Intelligence in 1996. This effort was a part of a European ESPRIT project, KACTUS, which had an objective to investigate the feasibility of knowledge reuse in complex technical systems and the usefulness of the ontologies in this task. In the method proposed in the KACTUS project an application knowledge base becomes the basis on which ontology is developed. This is done by following the bottom-up approach and when more applications are developed based on that knowledge base, the ontology becomes richer. This approach of developing ontologies is conditioned by the application development. So, every time an application is built, the ontology that represents the knowledge required for the application is built. This ontology can be developed by reusing others and can also be integrated into the ontologies of later applications.

METHONTOLOGY [19] is a methodology to create domain ontologies independent of the applications to be based on the ontologies. METHONTOLOGY was developed by the Ontological Engineering group at Universidad Politécnica de Madrid. METHONTOLOGY describes the whole ontology development cycle through a set of activities, such as, the specification, the conceptualization, the formalization, the implementation and the maintenance of the ontology and these activities are grouped as development activities "*Fig. 2*".

Fernadez and Lopez [19] analyzed the different approaches and concluded that: None of the processes covers all the processes involved in ontology building. Most of the methods and methodologies for building ontologies are focused on development activities, such as ontology conceptualization and ontology implementation, and the other aspects are ignored. Some methods are used only for one development process while some are used within a group.

At present each group tries to apply its own methodology. This is exacerbated by the fact that none have reached maturity. Therefore, efforts are required along the lines of unifying methodologies to arrive at a more mature model of ontology development process.

ISSN: 2321-8169
4677 - 4683

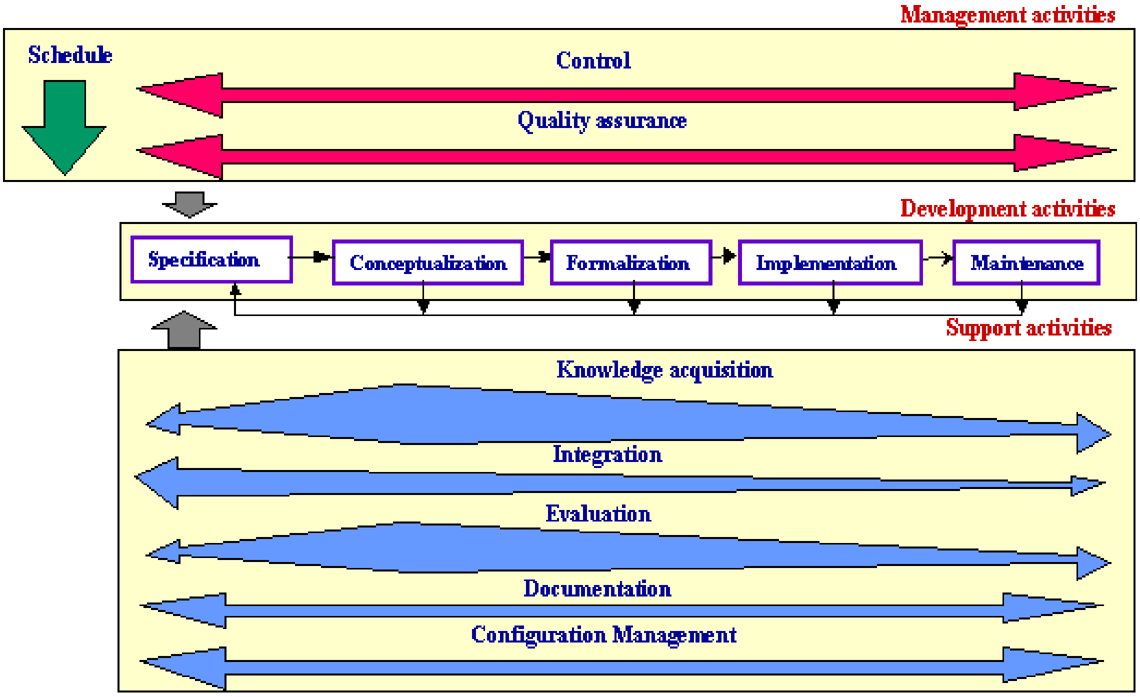

Figure 2. Ontology Lifecycle in METHONTOLOGY [19].

VI. INTEGRATIVE VIEW OF ONTOLOGY DEVELOPMENT PROCESS

Although there are a number of methodologies available for developing ontologies but usually they differ on several aspects in their approach and usage. Still it can be found that they agree on many lifecycle activities as the essential part of the development process. We have attempted to take a integrative view of these methodologies so that one can generically look at what goes into the stages of the ontology development and make use of those stages that are suitable to a particular development exercise. Main activities envisioned in this integrative view are shown in "*Fig. 3*".

A. Ontology Initiation

The various researchers working in the field of ontology development agree that the ontology development process is initiated, when a need is observed or an idea is coined by the stakeholders. These stakeholders can be the user community in the domain, the group active in the area, corporate organizations in the case of the corporate ontologies or any interested party that can invest time and resources for the effort.

B. Ontology Specification

In the beginning of the development process, it is undoubtedly very important to know what the need of developing the ontology is and what the scope of that ontology is. The uses and users of the intended ontology, should be characterized in this step of the development. Many researchers have also described this activity as *Specification.* What is the intended purpose and scope of the ontology greatly affects the design and the development of the ontology.

C. Ontology Development Planning

Ontology development as a whole is a not a trivial task and involves a series of systematic and well organized steps. To put all the things in the perspective, a good ontology development plan has to be devised. This ontology development plan can include the scheduling of the activities to be performed in the development process, the resources required and the allocation of these resources, documenting the plan, and other related management activities.

D. Ontology Requirement Analysis

In this step, domain experts and ontology engineers can analyze the requirements for the ontology being developed. The methodologies following the Software Engineering oriented approaches emphasize on the need of having Ontology Requirement Specifications Document (ORSD), very similar to the Software Engineering Specification Document. The goal of the ontology requirements specification activity is to specify about the purpose of the ontology, about its intended users and uses, and about its requirements to be fulfilled [20]. If there is an Ontology Requirements Specification Document (ORSD) that identifies intended scope and purpose of the ontology along with other specifications, it greatly helps in overall development activities that follow afterwards.

E. Ontology Design

This stage includes identifying the key concepts and relationships in the domain of interest. The domain experts do play a decisive role in this activity. In the beginning the focus can be on the specification to identify the concepts rather than committing to the words that will represent them. Then one

 ISSN: 2321-8169
4677 - 4683

can move further by getting more precise and unambiguous text definitions for such concepts and relationships. The terms are decided afterwards to represent these concepts and relationships that can be used in ontology. After deliberations and revisions along with modifications, an agreement is obtained on the conceptualization for structuring the domain knowledge. The identification of the already available and reusable ontologies and other sources, such as taxonomies or database schemas is done here. Also how to use the identified ontologies and sources is considered. It is also proposed by Gruber [2] to make a list of the concepts to be contained within the ontology and explore other ontologies for reusing their conceptualizations and terminologies. In the conceptualization, the orientation is towards getting an implementation independent informal terminology.

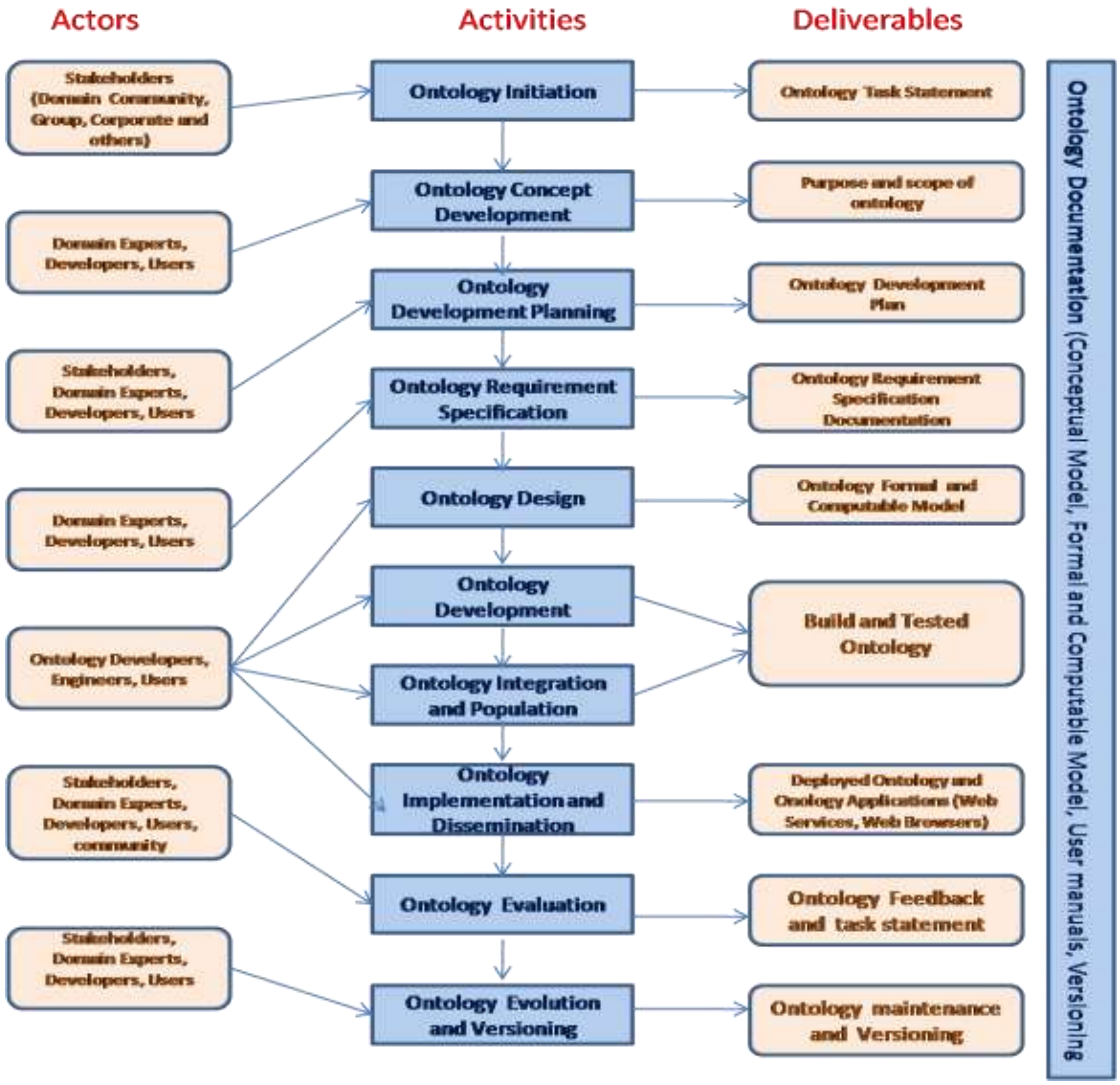

Figure 3. Integrative View of Ontology Development Process

Formalism is representing the conceptualization in some formal language, e.g. frames, object models or logic. For certain ontologies the representation of knowledge can be informal but for more technical or software oriented ontologies the formalism is more rigorous as the semantics of ontology must be more precise [21]. The degree of formality by which the vocabulary of an ontology is specified varies from informal definitions expressed in natural language to definitions stated in a formal language, such as first-order logic with a strictly defined syntax and semantics [18].

The different groups or stakeholders have devised varied design principles that can be followed in their respective areas, but certain commonalities exist amongst them. These design principles, if followed during the design development activities, result in more acceptable and valuable ontologies. The design principles of standardization, modularity, extensibility, portability, user friendliness, availability, and modifiability are often followed for optimal results.

F. Ontology Development

This is the process which transforms the conceptualized and formalized design into a reality. One approach of developing the ontologies is to reuse the existing ontologies available in the public domain and share the already captured knowledge using a variety of ways. One way of reusing ontologies can be to include an ontology into another in the form of a combined ontology while the other method can be to refine a generic ontology to fulfill the specific needs of new ontology. So if possible, the approach is to always consider reusing the existing relevant ontologies instead of duplicating the effort. In this ontology development step, the reusable ontologies found are modified to suit the ontology being developed.

Some researchers have argued that the approaches of backward and forward engineering, as used in the Software Engineering, can be applied to the process of reusing ontologies. The usage of design patterns is also proposed by many researchers in reusing the ontologies. In Software Engineering, patterns are an accepted way to facilitate and support reuse and in the field of Ontology Engineering similarly we can identify a scheme for ontology patterns and can divide ontology patterns into five levels: Application Patterns, Architecture Patterns, Design Patterns, Semantic Patterns, and Syntactic Patterns [21]. But if there is no ontology that can be reused, the alternative and best approach is to build ontology from the scratch.

One of the primary activities in the development process is data or knowledge acquisition. The data acquisition activity involves acquiring the data from the identified sources that were specified in the earlier phases and integrate that knowledge to be populated in the ontology as per the formalized model. The sources of knowledge can be many for a particular ontology and these can be in heterogeneous formats. Data collection is a typical activity followed for knowledge acquisition. Several techniques for collecting the data can be followed and the analysis of the collection can prompt further collection. The data collection not only helps in using the acquired knowledge for ontology building but also serve as the basis for knowledge bases. One of the main principles of data collection for ontology engineering is to never prevent users from saying what they want to say, but encourage them to say things in a way that it is easy to work with [16]. Data acquisition can be manual, semi-automatic, or fully automatic to collect and organize the knowledge.

G. Ontology Integration and Population

When we acquire data by any of the methods of data collection, such as manual, semi-automatic, or automatic, it may not be of the desired quality. We may have to do lot of processing of that data, to integrate and make it fit for populating in the ontology. If the cleansing of data is not done, it may lead to inconsistencies. These inconsistencies must have to be resolved to have good quality results. To populate ontologies, substantial overhead may be imposed to the user when all instance data has to be created manually and this overhead can be reduced with the help of semi-automatic or automatic population of data wherever possible [22].

H. Ontology Implementation and Dissemination

The ontology is not just a theoretical concept; it has to be implemented so that it can be made available to the users of that ontology. Once ontology is developed and contains all the required knowledge in it, the next step is to deploy the ontology so that it can be disseminated afterwards. Dissemination includes activities performed with an ontology after it has been engineered. The users may have to be trained to use the knowledge in the ontologies in the case of corporate ontologies. For public ontologies the user manuals and other support (usually online) may have to be provided. The use of individual web services, browsers, and utilizing well established collaborative tools of browsing and visualization (for example, BioPortal in the case of biomedical ontologies), all are the commonly followed practices in making ontologies available.

I. Ontology Evaluation

When the ontologies are developed, they can be considered as good quality and valuable ontologies, if they are able to serve the intended purpose. To look for the ontologies that can be viewed as complete is actually an unrealistic goal. As ontology building is an expensive and time consuming process and lot of resources are invested in the development of ontologies. Once the ontology is developed by following any of the development approaches, it must be evaluated for its quality by following a certain evaluation criteria. The ontologies are usually disseminated in the form of applications that we built on the knowledge contained in these ontologies. Other criteria of evaluating the quality of an ontology can be determining the appropriateness of it for its intended applications.

The simple measures of precision and recall are not that easy to be applied to the ontologies, as is the case of other knowledge extraction methods. In the case of specialized ontologies these metrics may assume different notions in different kind of applications. Evaluation is done practically, by evaluating the capability of the ontology to satisfy the requirements of its application and assessment for some common evaluation metrics of domain coverage, usability, the quality of content, consistency, completeness, conciseness, documentation and support. There may be certain evaluation metrics that may be suitable for one type of ontology, but may not be that important for other type. For example, a specialized biomedical ontology may give more emphasize on the accuracy and correctness rather than on any other metric. An ontology has its ultimate evaluation of quality and success, to find whether it is used and accepted widely by the community or not.

J. Ontology Evolution and Versioning

The ontologies are bound to evolve with the time. To make the ontology up-to-date and consistent, any changes in the sources of knowledge of the domain, the schemas of representation of knowledge or changes in the requirements of users, all have to be incorporated in the ontology. There is the evolution of ontologies with the time to adapt for these changes. The new and earlier unaddressed needs may also warrant changes. Ontologies do change management to deal with these kinds of changes, by providing versioning of ontologies. Ontology versioning is the concept of keeping multiple versions of ontologies to mange changes and evolution in ontologies. The compatibility of versions must be checked for instance-data preservation (no data lost between versions unless explicitly required), ontology preservation (query is satisfied in both versions), consequence preservation (all the facts could be inferred equally from the new version as inferred from older one) and consistency preservation (no logical inconsistencies) [23].

K. Ontology Documentation

Documentation is an important activity in the overall development process of an ontology which is done throughout the different stages involved in the ontology lifecycle. Uschold and Gruninger [16] have *the intended distinctions to humans who design agents. This means that ambiguity should be minimized, distinctions should be motivated, and examples should be given to help the reader understand definitions that lack necessary and sufficient conditions. (...). In all cases, definitions should be documented with natural language and examples to help clarify the intent.*

When the ontology creation is started by specifying the problem in some clear and concise statements, the activity of documentation comes into play and is carried out throughout the development stages. The documentation facilitates the appropriate use and re-use of an ontology by documenting the defining, more expansively than is possible within the ontology, the exact meaning of terms within the ontology. The documentation has to be done for conceptual model, formal and computable model, data acquisition, design and coding, evaluation, versioning, testing methods, user manuals and every other aspect involved in the development process. The developers and engineers may move out with the time, but the documentation remains.

VII. CONCLUSION

The ontologies are being developed and used in many disciplines and are considered as an important tool for the data integration and the knowledge representation. Ontologies in the past have been made by the developers by following the different methods and methodologies and there are number of methodologies that can be used as a reference point for developing ontologies. In this paper the different methodologies of ontology development process put forward by different researchers or stakeholders in the past were investigated. The various methods and approaches have their specific relevance, but it has been observed that they do not comprehensively cover all the activities involved in the ontology development task. Some approaches concentrate only on the implementation aspects ignoring all other important activities, while some are just utilized only for a single domain or used within a single group of developers.

The paper looks at an integrative view of the most representative methodologies used in the ontology development to describe the set of different activities that can be performed during the process of the ontology development. The type of ontology being developed, the domain of discourse to be captured, the expertise of the developers of ontology, all these factors drive the selection of one method over another while building ontologies. Further what type of applications of ontology have been envisioned, the time and resources available for the task, and other related factors, also influence the choice of the approach. The set of activities in the integrative view can be adapted to be applied in different settings of ontology development process.

REFERENCES

[1] Ramesh Patil , R.E.F., Peter F. Patel-Schneider , Don Mckay , Tim Finin , Thomas Gruber , Robert Neches The DARPA Knowledge Sharing Effort: Progress Report 1992.

[2] Gruber, T.R., Toward principles for the design of ontologies used for knowledge sharing. International Journal of Human-Computer Studies, 1995. Vol. 43(4-5): p. 907-928.

[3] Noy, N.F. and D.L. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology. 2001.

[4] Gruber, T.R., A translation approach to portable ontology specifications. Knowledge Acquisition, 1993. Vol. 5: p. 199-199.

[5] W3C: Ontologies in Semantic Web Context. 25 March 2010]; Available from: http://www.w3.org/2001/sw/SW-FAQ#whont.

[6] P. Spyns, et al., "Data modelling versus ontology engineering," SIGMOD Record, vol. 31, pp. 12-17, 2002.

[7] B. Motik, et al., "Bridging the Gap Between OWL and Relational Databases," Journal of Web Semantics, vol. 7(2), pp. 74-89, 2009.

[8] N. F. Noy and M. Klein, "Ontology Evolution: Not the Same as Schema Evolution," Knowledge and Information Systems, vol. 6, pp. 428–440, 2004.

[9] Janik, M. and K.J. Kochut, Wikipedia in Action: Ontological Knowledge in Text Categorization, in Proceedings of the 2008 IEEE International Conference on Semantic Computing. 2008, IEEE Computer Society. p. 268-275.

[10] Guarino, N. Some ontological principles for designing upper level lexical resources. In Proceedings of First International Conference on Language Resources and Evaluation (LREC), Granada, Spain, pp. 527–534, 1998b.

[11] The Suggested Upper Merged Ontology (SUMO) 25 March 2010, date last accessed]; Available from: http://www.ontologyportal.org/.

[12] Basic Formal Ontology. 25 March 2010, Date last accessed]; Available from: http://www.ifomis.org/bfo.

[13] Natale DA, Arighi CN, Barker WC, Blake JA, Bult CJ, Caudy M, Drabkin HJ, D'Eustachio P, Evsikov AV, Huang H et al: The Protein Ontology: a structured representation of protein forms and complexes. Nucleic Acids Res 2011, 39(Database issue):D539-545.

[14] Gómez-Pérez A, F.-L.M., Corcho O, Ontological Engineering. Springer–Verlag, London, United Kingdom, 2003.

[15] Mike Uschold , M.K., Towards a Methodology for Building Ontologies. In Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95, 1995.

[16] Uschold M, G.M., Ontologies:Principles Methods and Applications. Knowledge Sharing and Review, 1996. 2.

[17] Michael Grüninger , M.S.F., Methodology for the Design and Evaluation of Ontologies IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing, 1995: p. 6.1-6.10.

[18] A Bernaras, I.L., J M Corera. Building and reusing ontologies for electrical network applications in 12th Eur. Conf. on Artificial Intelligence (ECAI'96). 1996.

[19] FERN, et al., Overview and analysis of methodologies for building ontologies. The Knowledge Engineering Review, 2002. 17(02): p. 129-156.

[20] Su\, M.C., et al., How to Write and Use the Ontology Requirements Specification Document, in Proceedings of the Confederated International Conferences, CoopIS, DOA, IS, and ODBASE 2009 on On the Move to Meaningful Internet Systems: Part II. 2009, Springer-Verlag: Vilamoura, Portugal. p. 966-982.

[21] Gandon, F., Ontology Engineering: a Survey and a Return on Experience.

[22] Studer, T.T.a.P.H.a.H.L.a.Ó.M.-G.a.A.G.-P.a.R., Lifecycle-Support in Architectures for Ontology-Based Information Systems. 2007.

[23] Natalya F. Noy, M.K., Ontology Evolution: Not the Same as SchemaEvolution. Knowledge and Information Systems, 2004. 6: p. 428–440.