International Journal of Computer Applications (0975 – 8887)
Volume 65– No.24, March 2013

Ontology Mapping Techniques and Approaches

K.Balachandar
Department of CSE,
Pondicherry Engineering College,
Puducherry, India.

E.Thirumagal
Department of CSE,
Pondicherry Engineering College,
Puducherry, India.

D.Aishwarya
Department of CSE,
Pondicherry Engineering College,
Puducherry, India.

R. Rajkumar
Department of CSE,
Pondicherry Engineering College,
Puducherry, India.

ABSTRACT
The Organisations may use heterogeneous ontologies for the same domain which results in the representation of different concepts to the same concept. The interoperability problem arises when two ontologies are trying to commune with each other and hence the ontology mapping is required. The ontology mapping discovers the relationship between the pair of concepts of heterogeneous ontology and depicts whether the concepts are similar or not. The mapping document will be generated on mapping two ontologies. The ontologies must respond to the change requirements which make the mapping document to be inconsistent. This paper reports various mapping tools for mapping two ontologies and depicts various techniques such as mapping composition, mapping adaptation and floating model. It also describes various mapping approaches such as KOAN approach, WISE approach and SKIS approach for making the mapping document to be consistent. The existing mapping techniques and approaches deals with the mapping document consistency. The requirement for reconciliation mapping approach is proposed which uses various scenarios for consistency maintenance. In addition to consistency maintenance, it also provides time efficiency, space efficiency and accuracy.

General Terms
Artificial Intelligence, Semantic Web, Knowledge sharing, Knowledge Representation.

Keywords
Change Requirements, Consistency Maintenance, Mapping Approaches, Ontology Mapping

1. INTRODUCTION
The ontologies provide the rich set of semantics about the particular domain of interest. Nowadays the ontologies are used in the variety of applications such as semantic web, information management, information integration, document management, knowledge management, etc. For applications like distributed architectures and business process reengineering have the framework which has the integration of the heterogeneous ontologies of different domains. The advantages of using ontologies in all these applications are interoperability, reusability, communication, reliability, etc. The mapping has to be established when integrating data in the ontologies.

Ontology mapping is used to link similar relationships or concepts between the ontologies by means of equivalence with some metric values [1]. The resulting ontology mapping documents are used for various integration tasks namely query answering system and data transformation process. In the ontology based data integration system, an important problem arises with the changing behaviour of the ontologies. Ontologies are dynamic data sources and subject to change in order to adapt the new development in the research or domain [2]. The change in the ontology leads the mapping document invalid and outdated.

The classic solution would be re-establishing the mapping between the source ontology and target ontology. This method of re-establishing of mapping is known as the blank sheet approach [3]. The ontology mapping tools are expensive in terms of the human effort as it needs more number of inputs from domain experts. For larger ontologies, manually maintaining ontology mapping is impractical. As the semantics of the original mappings are not considered during the reestablishment it provides no guarantee for the consistency of ontology mapping document. With the help of the above observations, the effort required for regenerating the mappings from scratch with the ontology evolution is costly and it is too problematic[4]. Instead of regenerating the mappings, can use the previously captured information. The captured information of the ontology has the changes, type of changes, description about the change, changed entities etc. The ontology development must be dynamic in behaviour as the domain knowledge may change frequently and the updating of ontology according to the change requirement is required.

In this paper, the problem of ontology based data integration for the dynamically changing ontologies is addressed. The need of ontology change information in ontology based data integration plan is discussed. The solutions proposed so far in the state of the art which try to reprocess the previously captured information of the ontology are reviewed. Most of the solutions concern database schema evolution to examine whether it can be applied to the ontology based data integration. The solutions in the state of the art are classified as two techniques. First, mapping composition technique is used to compose schema mappings and second, mapping adaptation is used to try adapting mapping with every primitive change operation. The techniques and its drawbacks which make those inefficient for the ontology based data integration are discussed. The ontology evolution approaches used in the literature for using the previously captured information of ontology in multiple relevant dimensions of changes occurs in the ontology are discussed. The latest approach which is efficient in terms of time and memory are highlighted. It responds well for the requirements of the ideal ontology based data integration system which handles the ontology evolution effectively and efficiently. The overall goal of the paper is to give the overview of the ontology mapping and also to provide the necessary insights for the practical understanding of the issues involved. The rest of the paper is organised as follows: in Section 2 discusses the preliminaries. Section 3 discusses the mapping tools. Section 4 discusses mapping techniques. Section 5 discusses mapping approaches. Section 6 discusses the proposed mapping approach.

2. PRELIMINARIES

Definition 1: Ontology. The most common definition for ontology is "An ontology is an explicit specification of a conceptualisation"[5]. The conceptualisation means an abstract and representing the simplest view to the world for some purpose. It also consists of identification of concepts and their descriptions, relations and properties between them. The specification means description of domain of discourse, accurate, consistent, meaningful, solid and detailed. The explicit means the exteriorization of conceptualization and the representation of conceptualisation in way that it can be understood and used by agents. Another definition is "Ontology is a formal specification of shared conceptualisation"[6]. In this definition, the formal means that the ontology is understandable, processable and readable not only by people and also by machines. Shared implies that ontology is accepted as consensus in a group. Ontology concept is used in several contexts like multi-agent system, e-commerce, semantic web, natural language processing, etc.

Definition 2: Ontology Mapping. Ontology mapping is the process where the semantic relations are defined between two ontologies. The semantic relations will be looked at conceptual level and it is applied at data level. It transforms source ontology instances into the target ontology instances. The information in the semantic web is heterogeneous and distributed. Using distinct ontologies for same domain also can turn the exchange of information between software agents more difficult. The interoperability is defined as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged". Ontology mapping proves to be an efficient solution to solve/minimize the interoperability problem[7]. The ontology mapping process generate a formal mapping document contain semantic relations between source ontology entities to target ontology entities. Ontology mapping process is not a trivial process, requiring a good understanding of both ontology concepts and their semantic similarities. Concepts may have the similar meaning but different names e.g. Person and human, Concepts may have similar name but different meaning e.g. Keyboard in music context and Keyboard in computer context, Properties may have the similar meaning, but with different names e.g. gender and sex, Properties with the similar name, but different meaning, Terms with the similar meaning, but different writing e.g. color and colour, One source term has equal meaning to more than one target term, More than one source instance correspond to one target term.

3. MAPPING TOOLS

This section discusses the various mapping tools such as Falcon and MAFRA. The ontology mapping is used for establishing the relationship between the pair of concepts of different ontologies.

3.1 Falcon

Falcon is an ontology mapping tool which is used for finding the alignments between two different OWL or RDF ontologies[8]. The mapping library of Falcon consists of light weight linguistic mapping techniques such as V-Doc and I-Sub, structural mapping technique called GMO and the partition based mapping technique which is used for finding the mappings between the various blocks of the large ontology using divide and conquer method. The key steps of partition based mapping technique are the two input ontologies are partitioned independently and the pair wise mapping taken from two ontologies are executed. There are three phases of Falcon such as partitioning ontologies, mapping the blocks and discovering the alignments. The partitioning ontologies phase is that the similarity between the entities are calculated and based on the similarity the clusters are formed which are called as the blocks of the ontologies. Mapping the blocks phase is used for finding the alignments between the blocks of the large ontology which are formed by the partitioning ontologies phase. The alignments are then discovered. The entities of the ontology are grouped into the bag of words and the matching entities between the bags of words of entities are discovered. The drawbacks of this mapping tool are the ontologies are pre-processed to find the anchors and so the efficiency is reduced. The clustering algorithm used will be terminating abruptly when it reaches the maximum and this will be leading to the poor clustering.

3.2 MAFRA

MAFRA is the Mapping Framework toolkit which is used for ontology mapping i.e. for establishing the semantic relations between the ontologies[9]. The ontology mapping has three characteristics such as incremental, interactive and continuous. The incremental characteristic is that the ontology is improvised at the every stage of the process. The interactive characteristic is that the human being interaction during the flow of the process. The continuous characteristic is that the ontologies are improved continuously when the changes are applied. There are two types of dimensions such as horizontal dimension and the vertical dimension. The horizontal dimension is that the entities are raised and then mapped to the representation of ontology and they are normalised. The similarity is measured between the entities. Based on the similarity, the relationship is being established and they are executed. Then the post-processing is done which checks the results whether the quality has been improved or not. The vertical dimension is that the management of the mapping document for solving the interoperability problem. The graphical user interface is used for promoting the better ontology mapping.

4. TRADITIONAL MAPPING TECHNIQUES

This section discusses the three traditional mapping techniques which involves the mapping of the database schema. As various problems are addressed in maintaining the materialised view after the redefinition done by the user, the E-SQL has been proposed for redefining process which prevents the manual human interaction[10]. The below subsections discusses the three traditional mapping techniques for enabling the mapping at the database schema level.

4.1 Mapping Composition

This section discuss about the mapping composition. The semantic mapping composition problem was first identified by Madhavan and Halevy[11]. Here the adaptive mappings are considered. The schema level mapping composition is given by the Fig. 1.

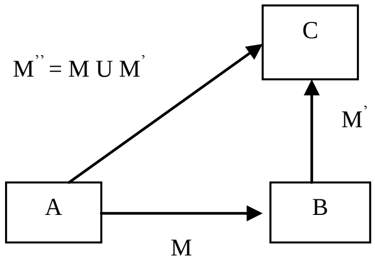

Fig. 1: Schema level mapping composition

In the Fig.1, A, B and C are the datasources. The mapping between the datasources A and B is given by M. The mapping between B and C is given by $M^{'}$. Instead of establishing the mapping between A and C via B, the direct mapping can be established between A and C which is given by $M^{''}=M \cup M^{'}$. Instead of establishing two mappings, the single mapping is sufficient which both yields the same result. This $M^{''}$ composition yields significant results but it also suffers from various drawbacks. The fact, semantics must be relative to the given class of queries must be satisfied which leads the mapping composition to suffer from limitations.

The mapping composition $M^{''}$ which is the composition of two schema mappings such as M and $M^{'}$. The query given may be class of conjunctive queries. The mapping composition of two mapping schemas when the query is the class of conjunctive queries will be leading to inequalities. Yu and Popa proposed the nested relational schema for supporting schema evolution. They proposed the composition algorithm for handling nesting and which provides the advantage of the size of the result is being reduced. The real time schemata are used for conducting experiments which demonstrates that the proposed composition algorithm reduces the size of the result efficiently[12]. Nash et al[13] proposed the mapping composition algorithm where the mappings are given by the embedded dependencies. This algorithm performs de-Skolemization which is used for obtaining the first order constraints from the second order constraints. Recursion and de-Skolemization are the two challenges faced by this algorithm. The recent research is to pose the scalability challenges for the mapping composition when compared to query redefining approach. The schema level mapping composition involves the access of the entire schema while the query level mapping composition involves the access of the part of the schema. The next subsection discuss about Mapping Adaptation.

4.2 Mapping Adaptation

The mapping adaptation involves for the schema change, the mappings are adapted incrementally. This technique is called as the mapping adaptation tool which is used for enabling the changes in the schema. The tool is used for detecting the inconsistencies while the schema changes[14]. When the schema changes the mappings are done in response i.e. the part of the schema will be modified. This technique has an advantage that it is used for tracking the semantic decisions that are made. The schemas are often ambiguous and the mapping choices cannot be made. Hence the semantic decisions are needed and these decisions can be reused appropriately. There are three types of updates done on the for the mapping adaptation policy. The three types of updates that are made on the mapping adaptation are move, delete and add which is given by Fig. 2.

In the Fig.2, the source schema is A and the target schemas are B_1, B_2 and B_3. The mapping of the source schema A and the target schema B_1 is given by $M^{'}$. The mapping of the source schema A and the target schema B_2 is given by $M^{''}$. The schema B_2 is obtained from the schema B_1 by using the update called add element. The mapping of the source schema A and the target schema B_3 is given by $M^{'''}$. The schema B_3 is obtained from the schema B_2 by using the update called move element. The consecutive mapping schemas are derived by using the mapping updates such as move, add and delete. This technique not only considers both the local changes to the schema and the changes that affect and transform the components of the schema. This technique is applicable for evolving only the small schemas. The modifications or updates are applied to the schemas. When the modifications are applied incrementally then the mappings will lead to the drawbacks in the source or the target schemas. When the schema is small, then this technique will best suit. When the schema is large then the incremental changes cannot be updated properly and this will lead to inconsistencies. The next subsection discusses the Floating model mapping technique.

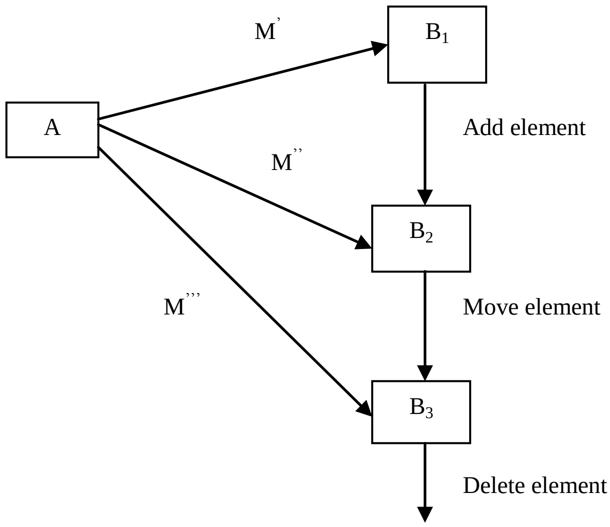

Fig.2: Mapping Adaptation

4.3 Floating Model

This section discuss in detail about the mapping technique called Floating Model. Xuan et al [15] proposed the mapping technique which deals with the varying versions of the schemas. This technique is used for integrating the schemas of various versions. This system works based on the assumptions given below.

- Every datasource which has to be integrated must have its own ontology
- The ontology is used for establishing the relationship between the schemas that are integrated.
- The shared ontology is extended as required.

The instances are considered and the semantics of the instances are added on them by using the implicit storage and hence the semantic keys are added on the instances. For adding the semantic keys to the instances, the universal identifiers are used for the properties. For the semantic keys the validation period is considered for each instance. The ontology continuity principle is proposed which does not falsify the axioms which are already proved to be true. This principle is used for managing the older instances using the newer version of the ontology. This floating model uses the ontology continuity principle which is used for integrating the ontologies of various versions. This technique is simple and the class or the property is deleted which is the common operation of the ontology evolution. Hence this technique is not applicable for the real world scenarios.

All the mapping techniques which are discussed above are used only for the schema mapping evolution. These techniques are not used for the ontology level mapping. This paper attempts to use the above mapping techniques for the ontology mapping which is discussed in the design of the proposed system section.

5. ONTOLOGY EVOLUTION

Ontology evolution can be defined as the timely adaptation of ontology to the new changes and the consistent management of the changes occurred. Since a new change in the ontology may cause inconsistencies in various parts of the ontology, as well as the dependent artifacts. The ontology evolution is considered as a process but it is not a trivial process, due to the consequences of changes and variety of sources. Thus the ontology evolution cannot be performed manually by an ontology engineer. The ontology evolution process has three relevant dimensions such as change representation, consistency maintenance and change storage. Change representation is one of the characteristic of ontology evolution approaches which defines how to represent ontology changes and it depends on the ontology language used. The change representation allows understanding ontology changes and the relation between them. Consistency is the degree of standardization, uniformity and free from contradiction among the parts of a component or system. The consistency maintenance allows understanding how the ontology changes are performed in order to keep ontology consistent. The way how the change is stored plays as an important role in ontology evolution systems. The ontology changes need to be stored in a particular format that can be used in future and easily accessed. Change storage allows understanding how the changes can be interpreted, accessed and understood by other systems.

5.1 Ontology Mapping and Updating Approaches

This section discusses the ontology mapping approaches which is the solution for adapting the ontology mapping document when there is an evolution of related ontology. The ontology adapting to the change requirements at varying times causes inconsistencies in ontology and hence the mapping and updating approaches are needed. There are three ontology mapping updating approaches such as KOAN approach, WISE approach and SIKS approach. The detailed description of the three ontology mapping and updating approaches is given by Fig.3. The below subsections discuss the three ontology mapping and updating approaches.

5.2 KOAN ontology mapping and updating approach

This section discusses the KOAN approach which was proposed by Stojanovic [16]. The detailed design of the KOAN approach is given by Fig.3. In this KOAN approach, the explicit change requirements are captured or the change requirements can be captured using the change discovery mechanisms. The changes that are captured are then represented as ontology changes. The semantics of changes are used for preventing the inconsistencies when the new changes are updated. The semantics of captured changes gaurentees the consistency state of the ontology. After the change requirements are being updated, the projection phase will make all the dependent applications to be in consistent state. During the changes implementation phase, the attributes of the requested changes are informed to the ontology engineer and the changes tracks are preserved. The results are being evaluated and these processes will be continued for the subsequent change requirements.

5.3 WISE ontology mapping and updating approach

This section discusses the WISE approach which was proposed by Plessers [17]. The detailed design of the WISE approach is given by Fig.3. The change requirements are specified first. Then it is checked whether the change requirements when applied to the ontology maintains the consistency of the ontology or not. If the ontology likely to have inconsistencies, then those changes are said to be change request for which the inconsistencies must be solved. Then the change requirements that are captured are applied to the ontology. When the ontology is modified then it is checked for the occurrence of the other changes than the changes that are requested. The change requests are checked for revision. The changes that are listed in the evaluation logs are projected to the dependent applications. The key element of this approach is that version log. The log is used for keeping track of the ontology from its creation to its retirement.

5.4 SKIS ontology mapping and updating approach

This section discusses the SKIS approach which was proposed by Klein [18]. The detailed design of the WISE approach is given by Fig.3. According to Klein the ontology change is defined as the changes updated in the ontology which will be resulting in the ontology which is different from the older version ontology. For updating the changes to the ontology, initially the relevant information regarding the ontology version must be selected. Then the change requirements which are to be applied for the particular ontology must be specified. The information about the

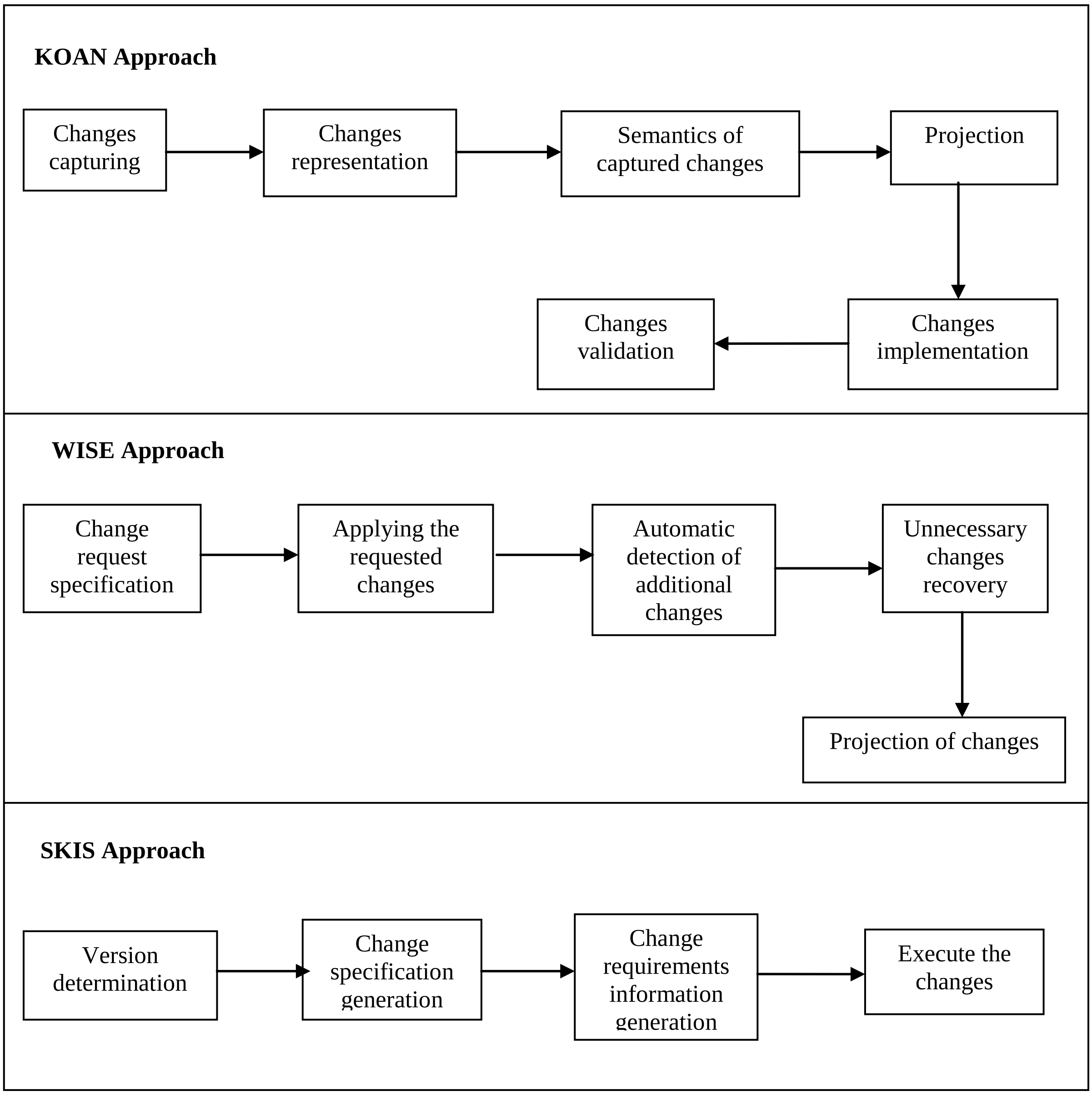

Fig 3: Ontology Evolution Approaches

change requirements is generated and then the changes are applied to the ontology for removing the inconsistencies that occur in the ontology. The key element of the SKIS approach is the change log which is used for keeping track of the changes that are made in the ontology. The changes applied may be conceptual change, specification change or the representation change. The conceptual change is the change that is applied at the conceptual level or during conceptualisation. The specification change is the change in the specification of the conceptualisation. The representation change is the changes that are done to the specification of conceptualisation.

5.5 Summary

The ontology mapping and updating approaches which are stated above are used for removing the inconsistencies while applying the change requirements to the ontology. For the space efficiency the ontology mapping and updating approaches are needed. The changes that are applied to ontology are of three namely such elementary changes, composite changes and the complex changes. The elementary changes are the modifications such as adding or removing are applied to the single entity alone. The examples of elementary changes are AddConcept, AddSubConcept, AddProperty, RemovePropertyRange, etc. The composite changes are the changes that the modifications will be applied to the neighbourhood entities of the ontology entity. The composite changes are the more powerful since the ontology engineer need not undergo the sequence of basic changes. The examples of composite changes are Pull concept up, Pull concept down, Split concept, Group concepts, Merge concepts, Concept copy, Concept generalisation, Inheritance extension, Concept specialisation, etc. The complex changes are the combination of the elementary changes and the composite changes. The examples of the complex changes are Move Concept, Move sibling Concepts, Deep Concept Copy, etc. When applying these changes the ontology may go to the inconsistent state. For avoiding the inconsistencies the mapping and updating approaches are used. All the three approaches maintain the logs for storing the changes, keeping track of the changes and the versions. The KOAN approach maintains the storage log for storing the changes that are applied to the ontology. The WISE approach maintains the version log which is used for keeping track of the versions of the ontology as the changes are updated to the ontology. The SKIS approach uses the change log which is used for keeping track of the changes applied to the ontology. Thus the above stated approaches are used for removing the inconsistencies and to improve the space efficiency.

6. DESIGN OF PROPOSED MAPPING APPROACH

As shown in the previous sections the mapping techniques proposed so far have several drawbacks and it cannot form a ideal solution to the ontology evolution which leads to the ontology mapping document invalid or outdated. Almost all the approaches deal with database schema more than the ontology schema. The ontology evolution needs more efficient mapping technique which adapts the changes happen in the ontology in any one of the approaches described in previous section. Thus the recent and efficient mapping technique which is efficient in terms of computation time and space are highlighted. This technique extracts the recent changes from the Changes History Log which stores all the changes happen in the ontology and then it computes mapping for the changed resources. The computed mapping will be updated with the original mapping later. The methodology of the mapping reconciliation technique is shown in the Fig.4 and the detailed describtion about the technique is in below section.

The proposed design is based on the recent mapping technique called mapping reconciliation which is more effective in terms of time and space.

6.1 Mapping Reconciliation

Reconciliation of mapping in ontologies deletes the mustiness of the existing mappings when there is an evolution of one or both the mapped ontologies and it is efficient in terms of computation space and time. This technique uses the concept of CHL, with which the changes are obtained during the ontology evolution[19]. The mapping reconciliation needs CHL to know the affected resources and staled part of mapping due to changes occurred in ontology. This technique contains two main components: i) Change History Ontology (CHO) and ii) Reconciliation of mappings.

The change history ontology specifies the changes appropriately in order to handle implicit and explicit change requirements correctly. The Change History Ontology (CHO) has three uses such as changes in log ontology, agents change and the reasons for the change which are used for keeping the track of the change history of ontology that are in use. The key elements of CHO are OntologyChange and Change-Set. The OntologyChange has secondary class Atomic Change which characterize all the property level and class level changes at the atomic level. Another core element the ChangeSet has all the changes happened in a specific time period in a coherent manner. It is responsible for managing the ontology changes happened and arrangement of changes in time indexed manner. This method of storing ontology changes differs with the traditional approaches by storing with preserving the dependencies and interlinking of changes with other changes. The ontology changes list is stored in the CHL. The changes list in CHL has its conformance to CHO. Each entry that is maintained in the log is an instance of OntologyChange or ChangeSet class of the CHO. The mapping reconciliation is possible with the idea of interval type ChangeSet to bundle all the changes respectective to a particular change session and to use the particular set of changes. The time ordering of each element level change is followed for every step of reconciliation procedure. The element level change information and ChangeSet collectively helps to achieve the objective of mapping reconciliation.

The reconciliation of mapping approach is more suitable for large ontologies with thousands of resources. The time efficiency of this technique can be compared better in large ontologies with traditional techniques. Assume two ontologies are mapped and information are exchanged based on the established mappings. The one or both of the ontologies are changed to other state and that now the mappings become invalid and not reliable. The mapping documents of these ontologies are also needed to evolve with the changed ontologies to make mappings valid. The mapping document also have to update according to the new mapping for the changed resources and eliminating staleness from the original mappings. It identifies the changed entity in both ontologies using the CHL entries of it and reconciles the mappings in time efficient manner. The mapping taking place in the reconciliation mapping technique establishes only for the changed entities and it will be updated with the original mappings. This technique calculates Semantic Affinity by providing particular resources using CHL's change information. The changes are extracted from the CHL using the SPARQL queries. As the ChangeSet instances can be sorted in descending order with the time stamp in CHO, the topmost ChangeSet instance can be selected to determine recent change.

This technique decreases the time taken for the mapping reconciliation. But it is also very important to test the accuracy of reconciliation of mapping performed. The accuracy of the mapping document is more critical when the information systems or services deal with information of the healthcare domain. To investigate its accuracy with the traditional techniques, the ontologies from health domain is taken. The testing of these ontologies are done by using Lily, H-Match, Falcon. Then the results obtained from these systems are compared with the reconciliation of mapping technique implemented system. But fewer mappings are missing than the results from the original systems and it not mentioned about the consistency of the mapping document after the updation of the reconciled mapping with the original mapping document. Even though it misses fewer mappings, it reduces the amount of time consuming to generate mapping using CHL. It proves to be effective and time efficient ontology mapping technique.

As this technique is efficient in terms of time and space by its experimental results provided in the literature, it can be implemented in the mapping techniques for the large ontologies. It also more helpful and time saving technique for the dynamic ontologies which need the updation of mapping document in short time to make the information provided in it reliable and useful to the world. In this technique, fewer things have to be considered in order to make it more efficient in terms of accuracy and consistency. We proposing the requirement of the technique to make it efficient and effective, idea to reduce the limitations of this technique and also proposed a framework to implement the idea. According to the recent technique emerged with its advantages, it also has its limitations as described above. This technique required for improving its accuracy of the mapping document and also it has to ensure the consistency of the mapping document. In this paper we discuss about the way of improving the reconciliation of mapping technique to get rid of its limitations. This technique process is about computing mapping for the changed resources and updating it with the original mapping document. In this case, it just considers only the changed resources and mapping document will be generated for those entities only. This technique ignores the deduced changes happen because of the simple and complex changes in the ontology and these deduced changes also have to be considered when computing mapping for the changed entities.

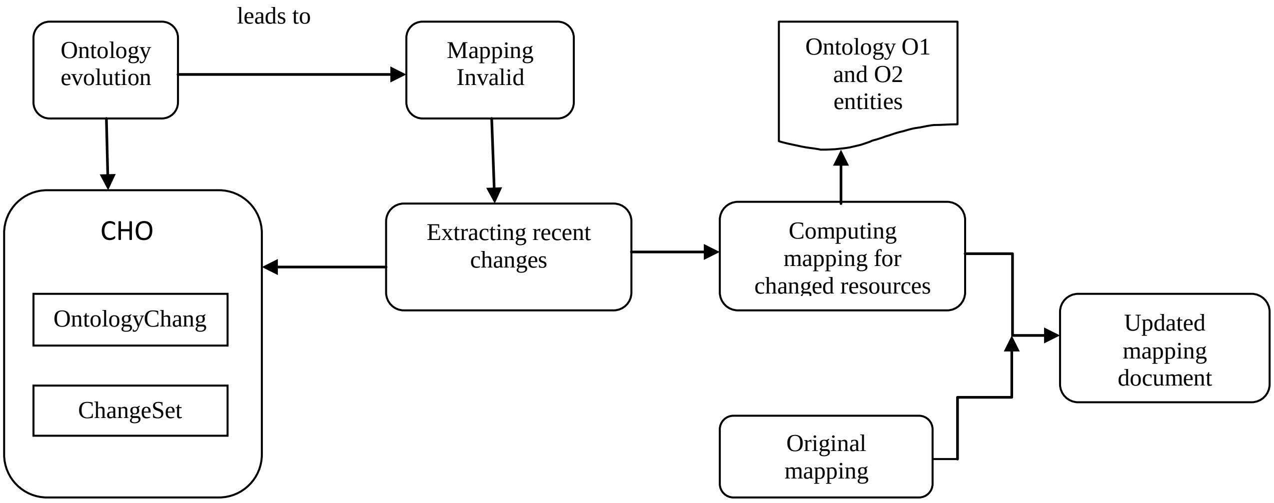

Fig 4: Reconciliation of mapping

This is the first requirement of this technique to increase the accuracy of the mapping computed with the changed resources from the CHL.

The second requirement is to have the particular order of changes for computing mapping document. The changes storing in the CHL is about the sequence of changes happening in the ontology.

When computing mapping for the changed resources, the changes has to be applied with some priority to the type of change happened. For example, the delete changes in the CHL has to be applied first before going for the add element change as it may consider the entity of the ontology which may be removed later by the effect of the delete element change which results in the limitation of missing accuracy of the mapping document. So the priority level has to be set for the type of changes occurring in the ontology to reduce its accuracy problem. It has to be considered in the process of computing mapping for the changed resources and required to maintain the accuracy of the mapping document.

When updating the computed mapping with the original mapping document, lot of things have to be considered in order to maintain the consistency of the mapping document. The idea of appending the computed mapping with the original mapping is the cause of inconsistency problem occurs in the updated mapping document. Here the requirement is about to consider the changes happened in the ontology and has to apply the changes related updation when merging the computed mapping with the original mapping. The updation of mapping has to adapt the ontology changes with respective to its type and also include the mapping occurred due to the new changes in the ontology. Thus the requirement we specify is to consider the changes happened in the updation of the mapping document to maintain the consistency. The ideal solution to implement all these requirements is possible by optimizing the changes log which stores all the ontology changes. The ontology change log has to capture each and every instance change happen during the ontology evolution.

It has to capture the deduced changes occur due to the effect of the changes happen in the ontology with more clarity. Then the ontology changes has to set some priority to make correct impact in the ontology mapping document due to forward and backward verification of the presence of entity in the ontology when applying it for computation of mapping. Finally the important aspect to consider is that implementing the updation of mapping document with the effects of changes happens due to ontology evolution. Thus the ontology mapping document reconciled using reconciliation of mapping technique ensures the accuracy and consistency which is a crucial thing in the data representation or data integration using ontology data sources.

7. CONCLUSION AND FUTURE ENHANCEMENT

Thus this paper describes about the traditional mapping techniques which processes mapping in database schema level. The traditional mapping techniques such as mapping composition, mapping adaptation and floating model processes mapping at the database schema level suffer from various drawbacks which are described in this paper. The ontology level mapping techniques needs more features and functional things to adapt to the ontology schema. According to the ontology evolution, the mapping document needs to be updated very often. The techniques we described in this paper deals with the mapping adaptation needs to store the ontology changes in the log which can be implemented rather in the ontology mapping. The ontology evolves by the new changes occurring in the domain and in order to maintain the ontology data source reliable and useful to the outside world users. So it needs some approaches to handle this ontology evolution and managing the ontology changes. The ontology mapping document also evolves as the ontology evolves with the new changes. The ontology mapping evolution approaches are described in this paper to deal with the ontology changes and its manageability. The ontology mapping evolution approaches such as KOAN, WISE, SIKS approaches captures the ontology changes and carries the changes to the mapping document for updating the changes with maintaining the semantic of the mapping document. With this traditional mapping technique and mapping evolution approaches results with the emergence of the recent mapping technique called mapping reconciliation. The reconciliation of mapping computes mapping for the changed entities of the ontology and updates it with the original mapping document using the concept of Change History Log. The Change History Log is used to store the changes occur in the ontology during the ontology evolution. This technique is efficient in terms of the computation time and space needed for the computation. But this mapping reconciliation technique lacks in the mapping accuracy and also it not ensures the consistency of the mapping document which is very crucial in the health domain. In this paper, we highlighted the requirements of the mapping reconciliation technique to enhance its functionality and to increase the mapping accuracy. These requirements reviewed the mapping reconciliation technique fully and found its functionality requirement to reduce its limitations. Thus the requirement which has to be considered in future to enhance this technique is described in this paper. The mapping reconciliation technique which adapts the requirement described in this paper will play major role in the ontology evolution which needs mapping updation very often. It also very useful in the dynamic ontologies and large ontologies which evolves and needs more time to regenerate the mapping document. Thus the reconciliation mapping technique with applied the requirements will be time efficient, space efficient, mapping accuracy and also ensures the consistency of the mapping document.

8. REFERENCES

[1] Klein, M.: Combining and relating ontologies:an analysis of problems and solutions. In:IJCAI (2001)

[2] Flouris, G., Manakanatas, D., Kondylakis, H., Plexousakis, D., Antoniou, G.: Ontology change: Classification and survey. Knowl. Eng. Rev. 23, 117–152 (2008)

[3] Yu, C., Popa, L.: Semantic adaptation of schema mappings when schemas evolve. In: Proceedings of the 31st international conference on Very large data bases. VLDB Endowment, Trondheim, Norway (2005)

[4] A.Y. Halevy, Z.G. Ives, M. Jayant, P. Mork, D. Suciu, I. Tatarinov, The Piazza peer 464 data management system, IEEE Transactions on Knowledge and Data 465 Engineering 16 (2004) 787–798.

[5] T.R. Gruber, "A translation approach to portable ontology specifications", Knowledge Acquisition, vol. 5,no. 2,pp. 199—220,1993.

[6] Brost, Willem Nico, " Constroction of Engineering Ontologies for knowledge Sharing and Reuse", PhD Thesis(1997).

[7] Madhavan, Jayant, Bernstein, Philip A. and Rahm, Erhard, " Generic Schema Matching with Cupid" . In The VLDB Journal. Roma , 2001.

[8] W. Hu, Y. Qu, Falcon-AO: a practical ontology matching system, Journal of Web Semantics 6 (3) (2008) 237–239. doi=http://dx.doi.org/10.1016/j.websem. 2008

[9] A. Maedche, B. Motik, N. Silva, R. Volz, MAFRA – a MApping FRAmework for distributed ontologies, in: Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, London, 2002, pp. 235– 250

[10] Lee, A.J., Nica, A., Rundensteiner, E.A.: The EVE Approach: View Synchronization in Dynamic Distributed Environments. IEEE Trans. on Knowl. and Data Eng. 14, 931–954 (2002).

[11] Madhavan, J., Halevy, A.Y.: Composing mappings among data sources. In: Proceedings of the 29th international conference on Very large data bases, VLDB Endowment, Berlin, Germany, vol. 29 (2003)

[12] Fagin, R., Kolaitis, P.G., Popa, L., Tan, W.-C.: Composing schema mappings: Secondorder dependencies to the rescue. ACM Trans. Database Syst. 30, 994–1055 (2005)

[13] Nash, A., Bernstein, P.A., Melnik, S.: Composition of mappings given by embedded dependencies. ACM Trans. Database Syst. 32, 4 (2007)

[14] Velegrakis, Y., Miller, R.J., Mylopoulos, J.: Representing and Querying Data Transformations. In: Proceedings of the 21st International Conference on Data Engineering. IEEE Computer Society, Los Alamitos (2005)

[15] Xuan, D.N., Bellatreche, L., Pierra, G.: A Versioning Management Model for Ontology- Based Data Warehouses, DawaK, Poland (2006)

[16] Stojanovic, L. (2004). Methods and Tools for Ontology Evolution. Ph.D. Thesis, Karlsruhe University, Germany.

[17] Plessers, P., De Troyer, O,"Ontology Change Detection using a Version",.The Semantic Web – ISWC 2005. s.l. : Springer Berlin / Heidelberg, 2005.

[18] Klein, Michel. 2004. Change Management for Distributed Ontologies. 2004. PhD Thesis.

[19] Asad masood Khattak, Zeeshan Pervez, Khalid Latif, Sungyoung Lee, " Time efficient reconciliation of mappings in dynamic web ontologies", Knowlwdge- Based Systems,vol.35, pp. 369—374,November, 2012.