not to use the new proposed mapping entry to replace the obsolete one in the case of deletion and change operations. If the concept ce referred to by the obsolete mapping entry has not been included in the extended concepts part of the referred ontology, it is included in that part with $A(ce) \equiv < c_1, v >$ where c_1 is the concept proposed by the mapper to replace ce in the mapping list and v is the version of the referred ontology where ce originated. If concept ce has been included in the extended concepts part, only the value of c in the annotation needs to be changed to c_1.

V. Evaluation and Discussion

The proposed method is evaluated by presenting a case study in which a health archetype is used as an example of the application. Following the case study is discussion on the efficiency of the use of the mapper compared to the common method and the maintenance of the semantic currency contained in the applications due to the new mapping entries proposed by the mapper.

A. Application of the Method to the Archetype Term Binding Process

In this section, an application of the proposed method is presented for managing the mapping between the application terms and the ontology concepts when changes occur in the referred ontology. We used an archetype as the representation of an application which refers to the referred ontology. An archetype is a model of specific domain knowledge, in this case, clinical knowledge. Each archetype describes a complete clinical knowledge concept such as 'diagnosis' or 'test result' [26]. For the referred ontology, a sub-ontology, which is derived from SNOMED CT as the main ontology, is developed. The method proposed in [27] is used to build the sub-ontology.

For this work, an archetype is created as a representation of an application. The archetype was named *tooth_care_summary*. The archetype was created by considering the concepts changed in SNOMED CT so that the types of change operations in the sub-ontology can be shown in the archetype. Thus, the archetype itself is a modified version to fulfill the above requirement. The definition of the *tooth_care_summary* archetype is shown in Figure 5, which is previewed using the Archetype Editor -arc built by Ocean Informatics. It contains 40 terms, each of which is bound to a SNOMED CT concept. Based on the concepts required by the binding, a sub-ontology is built. The sub-ontology was extracted from the 20110131 version of the International SNOMED CT edition. The sub-ontology contains 557 concepts and 645 relationships of SNOMED CT. Based on the archetype terms and the sub-ontology concepts, the mapper created a mapping list.

Figure 6 shows part of the initial binding between the archetype terms and the 20110131 version of SNOMED CT concepts viewed by the Archetype Editor. The node column presents the term names, the code column refers to the bound SNOMED CT concepts, while the release column shows the version of SNOMED CT in which each of the concepts is initially included. Note that this is not the actual mapping list.

After the 20110131 version, SNOMED CT has been changed and the newer version is the 20110731 SNOMED CT. To accommodate the changes, the sub-ontology has been changed as well. Some change operations were performed. The mapper checked the mapping entries to see if the change operations were affected by the change operations. Apparently, there were seven operations which were related to the mapping entries. The effect of each change operation to the related mapping entry is summarised in Table II. In Figure 6, the highlighted rows contain the concepts which are affected by the change operations.

The highlighted rows in Figure 7 are the binding between archetype terms and the SNOMED CT concepts which are affected by the sub-ontology changes. It can be seen that the rows contain different concepts ids and release versions of SNOMED CT from the concept ids and release versions contained in Figure 6.

B. Discussion

There are several issues related to the proposed method which will be discussed in this section. These issues are elaborated as follows.

The use of sub-ontologies to increase efficiency in change propagation process

In the proposed method, applications refer to sub-ontologies instead of the base ontology. The number of concepts in a sub-ontology is smaller than the number of concepts in a base ontology. When the base ontology changes, only the relevant changes are propagated to each of the sub-ontologies. In this way, the number of changes in each of the sub-ontologies is also smaller than the number of changes in the base ontology.

After updating its components according to the changes propagated by the base ontology, a sub-ontology then propagates these changes to the applications referring to it. This process is performed by the mapper by updating its mapping list according to the rules of sub-ontology to application change propagation. Since the number of change operations in the sub-ontology is smaller than the number of change operations in the base ontology, the mapper does not need much time to examine all the change operations which occurred in the sub-ontology. The time needed to examine the change operations will be longer if the sub-ontology does not exist, which implies that the mapper must look up all the change operations in the base ontology. In the case of SNOMED CT used in the evaluation of the method, the number of changes which occurred to Version 20110131, which is included in Version 20110731, is 8,697 operations. It will take time for the mapper to check whether each of the operations falls into one of the rules for propagating the changes to applications.

Benefits of the use of the mapper

There are several advantages of the use of the mapper in the change propagation process to applications as follows.

- The mapper maintains the history of the application term references to ontology concepts.
  The mapper never deletes its entries. Invalid (not used) entries are only marked by the 'invalid' value of the

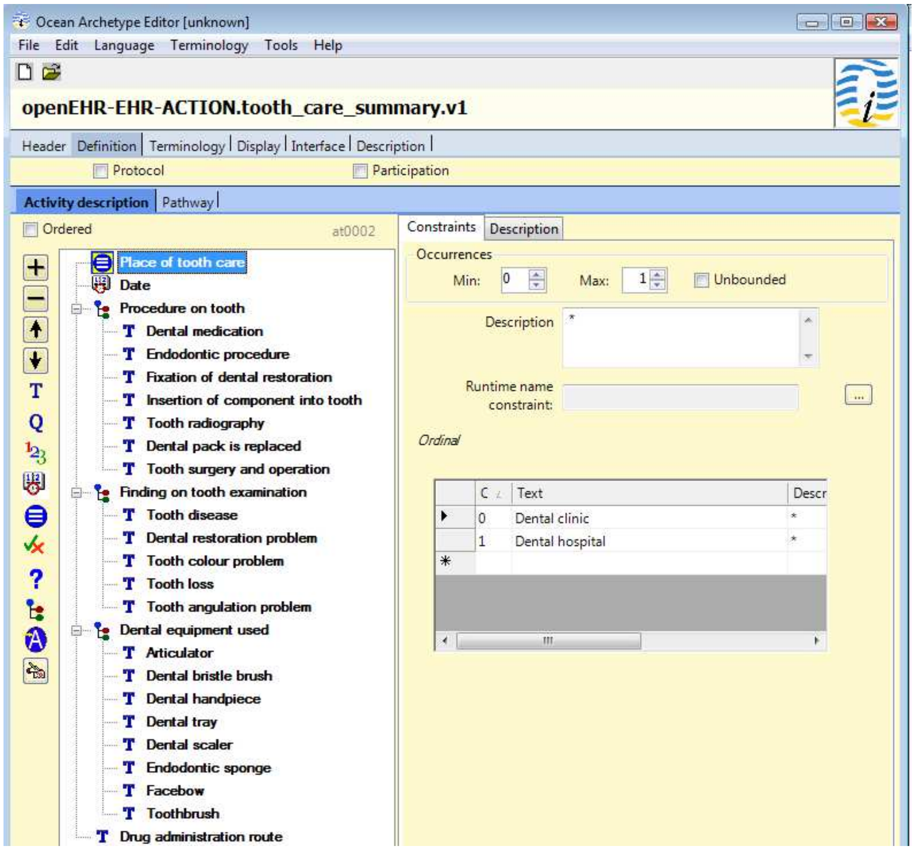

Fig. 5: The definition of *tooth_care_summary* archetype.

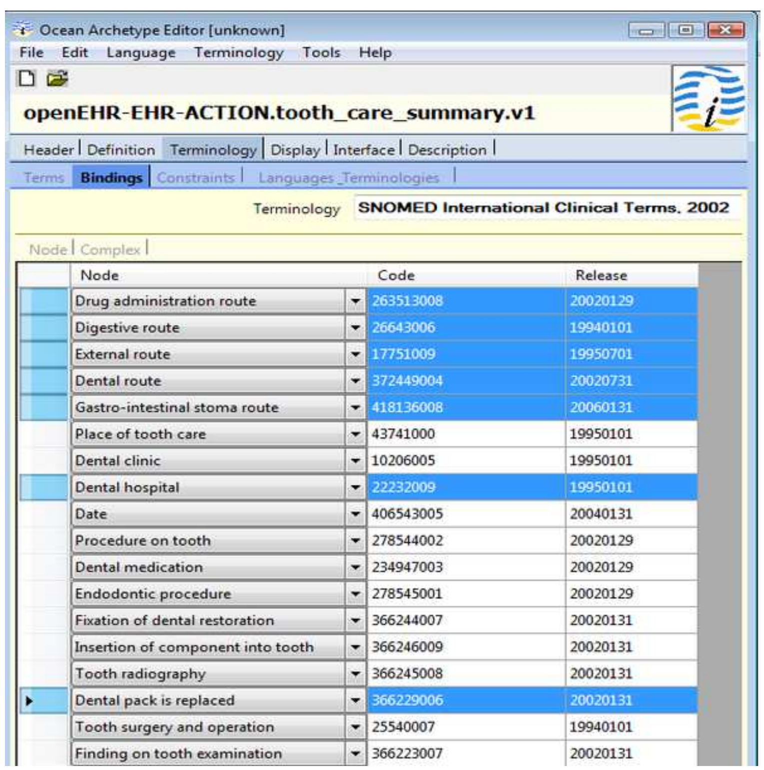

Fig. 6: The binding between the terms in the archetype and the SNOMED CT concepts before the sub-ontology changes.

TABLE II: Change operations found in the sub-ontology which are related to the mapping entries

Change operation in sub-ontology	Related mapping entry and the change					
	Term id	Term name	Concept id of the existing entry	Decision by the mapper	Reason	Concept id of the new entry
MergeCon (263513008, 410675002)	at0037	Drug admin-istration route	263513008	Status is changed to obsolete, new entry is proposed	Concept is merged	410675002
InsertCon (447964005)	at0038	Digestive route	26643006	Status is changed to obsolete, new entry is proposed	The term name is more similar to the name of the new concept	447964005
DelCon (17751009)	at0039	External route	17751009	Status is changed to obsolete, new entry is proposed	The deleted concept has only 1 parent concept (284009009), but no child concept	284009009
MoveCon (372449004)	at0040	Dental route	372449004	No need to update	The moved concept is still included in the sub-ontology	-
InsertCon (372454008)	at0041	Gastro-intestinal stoma route	418136008	No need to update	The term name is not more similar to the name of the inserted concept	-
AddLeaf (448399001)	at0004	Dental hospital	22232009	Status is changed to obsolete, new entry is proposed	The term name is more similar to the name of the new leaf concept	448399001
AddLeaf (447896001)	at0012	Dental pack is replaced	234718000	Status is changed to obsolete, new entry is proposed	The term name is more similar to the name of the new leaf concept	447896001

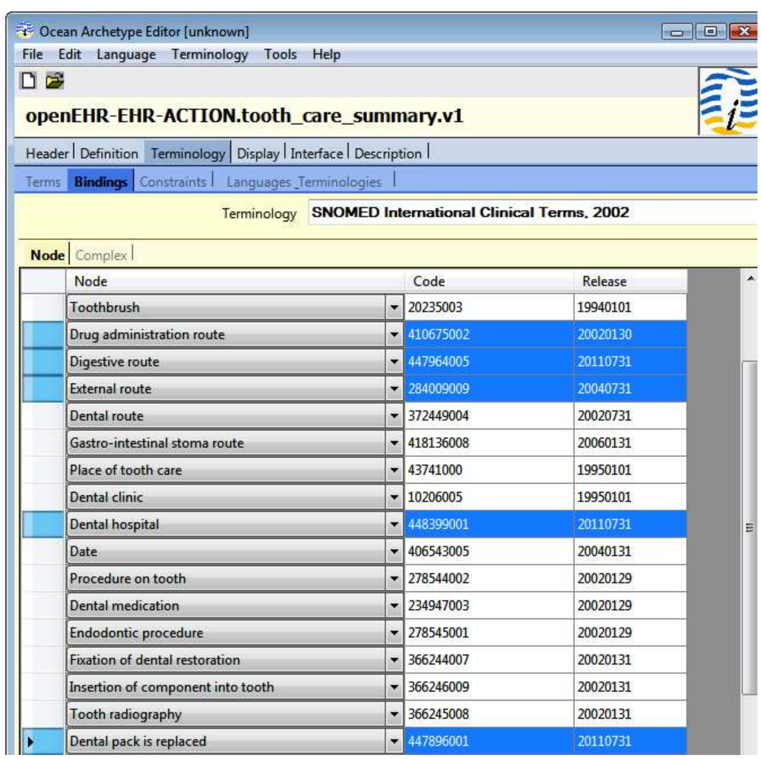

Fig. 7: The binding between the terms in the archetype and the SNOMED CT concepts after the sub-ontology changes.

status field. In this way, the history of references of application terms can be preserved. All references to a particular application term have the same values of *application_id* and *term_id* fields. Each invalid reference has a value for its *reason* field which records the reason why the reference is not used. Furthermore, the user can also check if there are application terms which are out-of-date and may need to be changed. The mapping entries with the most current value of the *version* field and 'not used' value of *reason* show that the corresponding application terms still refer to older ontology concepts.

- The mapper enables the semi-automatic update process of the mapping list.
  A fully manual update is error prone and obviously takes a longer time. In the sub-ontology used in the evaluation, there are 40 terms to be mapped to the sub-ontology concepts. In the case of a manual process, when the sub-ontology changes, the user must check each term to see whether it refers to a deleted sub-ontology concept or not. Since there are 557 concepts in the sub-ontology, the manual checking process is hardly feasible. The mapper can do the checking faster because it has the mapping list which includes the concept ids. Hence, the mapper only needs to find the concept ids which are included in the list of sub-ontology changes given by the central sub-ontology manager.
- The mapper facilitates the management of the mapping between the application terms and the sub-ontology concepts such that the sub-ontology changes can be handled without the need to modify the applications.
  This can be done because the mapping is managed outside the applications. The separation between the mapping mechanism and the applications is important because, unlike an ontology which can be changed automatically, the decision to change an application cannot be made instantly because it may affect the existing data and raise some technical issues related to the application development.
- The mapper can propose a better mapping of the application terms
  In the example, there are three terms which are mapped to the new concepts which are semantically more similar to the terms. In the previous mapping, the terms are mapped to the less similar concepts because they are the best choice in the previous sub-ontology. The new concepts added to the sub-ontology apparently have better similarity to the terms. Again, a manual examination takes time because there are 40 terms to be checked. The mapper can quickly propose the new mapping based on the rule of mapping list changes due to an addition of a leaf concept or an insertion of a concept. This will improve the quality of the mapping between the archetype terms and the ontology concepts. Otherwise, the application terms will maintain their references to less similar concepts, while there are actually concepts which have better similarity to them. At this moment, a mapping is improved by changing the referred concept to its new child or parent concept only in the changed sub-ontology. In the future, improvements might be made by changing the referred concept to any new concept in the sub-ontology or even any new concept in the base ontology. However, a discussion on this issue is beyond this thesis.

Validity of the mapping entries with regard to the current sub-ontology

A concept deleted from the sub-ontology indicates that the concept is not available in the current sub-ontology. In some ontologies such as SNOMED CT, a concept merged to another one is also considered a deleted concept. A reference to a non-existent concept is obviously not valid, and hence, should not exist in the mapping list. In the sub-ontology, the concept to be deleted is concept 17751009, while the concept to be merged is concept 263513008. In Figure 6, those two concepts are listed in the binding list between the archetype terms and the sub-ontology concepts. However, in Figure 7, these two concepts are not listed in the binding list. This is correct since the two concepts do not exist in the current sub-ontology. The terms previously bound to the two concepts are now binding to other concepts which exist in the sub-ontology. This shows that the proposed method is able to keep the application terms referring to the valid concepts in the current sub-ontology.

Possibility of application change due to sub-ontology change

The changes to the sub-ontology suggest that the knowledge has changed as well. For an application which has a very high requirement for knowledge update, the changes in the sub-ontology can be interpreted as an indication that the application needs to be updated. For instance, if a concept referred to by an application term is deleted from the sub-ontology, it may be the case that the term should be deleted from the application due to its obsoleteness. The mapper can give notification to the applications with regard to the changes, and it is the decision of the applications to update the terms included in them. If the terms are updated, the sub-ontology must be updated too because the selected concepts which are referred to by the application terms might be changed. This leads to another process of sub-ontology changes.

VI. Conclusion

In this paper, a change propagation mechanism from an ontology to the depending application has been proposed. A mechanism which is able to manage the continuous access of the applications to the ontology they refer to is proposed. Using the mechanism, the applications keep referring to the up-to-date ontology even when the ontology changes.The heart of the mechanism is the component called the mapper, which includes a mapping list. The mapping list contains mapping entries, each of which represent the mapping between an application term and an ontology concept. The task of the mapper is to manage the mapping list in the event of ontology changes so that the reference to a non-existing concept is avoided and the quality of the mapping entries can be improved by proposing new mapping entries which contain more relevant pairs of application terms and ontology concepts. Some rules, which are based on the ontology change operations, have been created for the mapper to update the mapping list.

The proposed method has been applied to an application together with its referred ontology. An archetype is used to represent an application, while an ontology has been developed based on the SNOMED CT ontology and the archetype terms. It is shown that the mapper offers more efficient change propagation than the commonly used method in terms of the number of change operations which should be propagated to the applications. The use of the mapper also enables semi-automatic updating to the mapping list which is obviously faster than manual inspection. Moreover, the rules used by the mapper can maintain the semantic currency of the mapping because the mapper can propose a new mapping entry in which the application term is semantically more similar to the new concept than the previous one.

For future work, the application of the method in other domains, such as bioinformatics and Internet of Things, can be observed. Similar to the application of the method in health domain, the application to other domains requires the availability of a standardised ontology, the distributed environment nature, and the reference of application terms to ontology concepts. Technical aspects of the application of the approach may also be interesting for future work. The performance, reliability and scalability of the deployment of the approach in distributed environment needs to be examined.

REFERENCES

[1] D. E. Oliver and Y. Shahar, "Change management of shared and local health-care terminologies," Methods of Information Medicine, vol. 39, pp. 278–290, 2000.

[2] D. E. Oliver, Y. Shahar, E. H. Shortliffe, and M. A. Musen, "Representation of change in controlled medical terminologies," Artificial Intelligence in Medicine, vol. 15, pp. 53–76, 1999.

[3] N. Stojanovic, L. Stojanovic, and S. Handschuh, "Evolution in the ontology-based knowledge management systems," in Proceedings of the 10th European Conference on Information Systems, Gdansk, Poland, 2002, pp. 840–850.

[4] M. Klein, D. Fensel, A. Kiryakov, and D. Ognyanov, "Ontology versioning and change detection on the web," in Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW02), LNAI 2473, A. Gomez-Perez and V. Benjamins, Eds. Springer-Verlag, Berlin, Heidelberg, 2002, pp. 197–212.

[5] L. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic, "User-driven ontology evolution management," in Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, ser. EKAW '02. London, UK: Springer-Verlag, 2002, pp. 285–300.

[6] A. Maedche, B. Motik, L. Stojanovic, R. Studer, and R. Volz, "Managing multiple ontologies and ontology evolution in ontologging," in Intelligent Information Processing. Kluwer, 2002, pp. 51–63.

[7] N. F. Noy, A. Chugh, W. Liu, and M. A. Musen, "A framework for ontology evolution in collaborative environments," in Proceedings of the 5th International Semantic Web Conference, ser. LNCS Volume 4273. Springer, 2006, pp. 544–558.

[8] F. Zablith, "Dynamic ontology evolution," in International Semantic Web Conference (ISWC) Doctoral Consortium, Karlsruhe, Germany, 2008.

[9] F. Zablith, M. Sabou, M. d'Aquin, and E. Motta, "Ontology evolution with evolva," in Proceedings of the 6th European Semantic Web Conference (ESWC) LNCS 5554. Springer-Verlag, Berlin, Heidelberg, 2009, pp. 908–912.

[10] T. Kirsten, A. Gross, M. Hartung, and R. Erhard, "Gomma: a component-based infrastructure for managing and analyzing life science ontologies and their evolution," Journal of Biomedical Semantics, vol. 2, 2011.

[11] M. Hartung, A. Grob, and E. Rahm, "Conto-diff: generation of complex evolution mappings for life science ontologies," Journal of Biomedical Informatics, vol. 46 (2013), pp. 15–32, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.jbi.2012.04.009

[12] A. M. Khattak, K. Latif, S. Khan, and N. Ahmed, "Managing change history in web ontologies," in Proceedings of the Fourth International Conference on Semantics, Knowledge and Grid, China, 2008.

[13] A. M. Khattak, Z. Pervez, S. Lee, and Y.-K. Lee, "After effects of ontology evolution," in Proceedings of the 5th International Conference on Future Information Technology (FutureTech), 2010, pp. 1 – 6.

[14] A. M. Khattak, K. Latif, and S. Lee, "Change management in evolving web ontologies," Tsinghua Science and Technology, vol. 37 (2013), pp. 1–16, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2012.05.005

[15] J. Mapoles, C. Smith, J. Cook, and B. Levy, "Strategies for updating terminology mappings and subsets using snomed ct," in Proceedings of the 3rd international conference on Knowledge Representation in Medicine, R. Cornet and K. Spackman, Eds., 2008.

[16] A. Shaban-Nejad and V. Haarslev, "Bio-medical ontologies maintenance and change management," in Biomedical Data and Applications, ser. Studies in Computational Intelligence Volume 224, A. Sidhu and T. Dillon, Eds. Springer, 2009, pp. 143–168.

[17] R. Palma, O. Corcho, A. Gmez-Prez, and P. Haase, "A holistic approach to collaborative ontology development based on change management," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 9, pp. 299–314, September 2011.

[18] A. K. Sari and W. Rahayu, "A methodology for change propagation in health ontology," in Proceedings of the 15th Pacific Asia Conference on Information Systems, Brisbane, Australia, 2011.

[19] A. K. Sari, W. Rahayu, and M. Bhatt, "An approach for sub-ontology evolution in a distributed health care enterprise," Information Systems, vol. 38, pp. 727–744, July 2013.

[20] L. Stojanovic, A. Abecker, N. Stojanovic, and R. Studer, "On managing changes in the ontology-based e-government," in Proceedings of the 3rd International Conference on Ontologies, Databases and Application of Semantics (ODBASE 2004), ser. LNCS Volume 3291. Springer Verlag, 2004, pp. 1080–1097.

[21] Z. Huang and H. Stuckenschmidt, "Reasoning with multi-version ontologies: A temporal logic approach," in Proceedings of the 4th International Semantic Web Conference (ISWC), 2005, pp. 398–412.

[22] Y. Liang, H. Alani, D. Dupplaw, and N. Shadbolt, "An approach to cope with ontology changes for ontology-based applications," in Second Advanced Knowledge Technologies DTA Symposium, Aberdeen, Scotland, 2006.

[23] Y. Liang, H. Alani, and N. Shadbolt, "Changing ontology breaks the queries," in Doctoral Symposium of The 5th International Semantic Web Conference, ser. LNCS Volume 4273. Athens, GA, U.S.A: Springer-Verlag, 2006, pp. 982–985.

[24] A. C. Yu and J. J. Cimino, "A comparison of two methods for retrieving icd-9-cm data: The effect of using an ontology-based method for handling terminology changes," Journal of Biomedical Informatics, vol. 44, pp. 289–298, 2011.

[25] H. Gu, J. Geller, L.-m. Liu, and M. Halper, "Using a similarity measurement to partition a vocabulary of medical concepts," in Proceedings of the 10th International Conference on Database and Expert Systems Applications (DEXA '99). London, UK: Springer-Verlag, 1999, pp. 712–723.

[26] H. Leslie and S. Heard, "Archetypes 101," in HIC 2006, J. Westbrook and J. Callen, Eds. Sydney: Health Informatics Society of Australia Ltd (HISA), 2006.

[27] A. K. Sari, W. Rahayu, and M. Bhatt, "Archetype sub-ontology: Improving constraint-based clinical knowledge model in electronic health records," Knowledge Based Systems, vol. 26, pp. 75–85, February 2012.