International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org Volume 3 Issue 2‖ February 2014 ‖ PP.35-37

XAML Based Mechanism with Closed Dataset Robust System Using HERM Algorithm

[1]P.M.Balaganesh M.E., (P.HD) DEAN/cse, [2,]D.Radha M.Tech, AP/cse,
[3]R.Sarojinidevi (ME)/cse
[1, 2, 3,] *Sembodai Rukmani Varatharajan Engineering College, Sembodai.*

ABSTRACT: *Robust system Hybrid extraction of robust model (HERM), a dynamic XAML based mechanism for the adaptive management and reuse of e-learning resources in a distributed environment like the Web. This proposed system argues that to achieve the on-demand semantic-based resource management for Web-based e-learning, one should go beyond using domain ontology's statically.*

KEYWORDS: *XAML, Dataset, Data Management, E-learning.*

I. INTRODUCTION

In many application domains (e.g., medicine or biology), comprehensive schemas resulting from collaborative initiatives are made available. This proposed system argues that to achieve the on-demand semantic-based resource management for Web-based e-learning, one should go beyond using domain ontology's statically. So the propose XAML based matching process involves semantic mapping has done on both the open dataset and closed dataset mechanism to integrate e-learning databases by using ontology semantics. It defines context-specific portions from the whole ontology as optimized data and an XAML based resource reuse approach by using an evolution algorithm. Such well established schemas are often associated with reliable data that have been carefully collected, cleansed, and verified, thus providing reference ontology-based data management systems (DMSs) in different application domains. A good practice is therefore to build on the efforts made to design reference DMSs whenever we have to develop our own DMS with specific needs. Easily that a robust module based DMS evolves safely both the schema and the data of the reference DMS. From a module robust to consistency checking, for any data update in a corresponding module-based DMS, we show how to query the reference DMS for checking whether the local update does not bring any inconsistency with the data and the constraints of the reference DMS .From a module robust to query answering, for any query asked to module-based DMS, It shows how to query the reference DMS for obtaining additional answers by also exploiting the data stored in the reference DMS.

II. RELATED WORK

Conventional Data Management Strategies:

In conventional information management principles, the stored records are normally identified by sets of key words or index terms, and requests for information are expressed by using Boole an combinations of index terms. The retrieval strategy is normally based on an auxiliary inverted-term index that lists the corresponding set of document references for each allowable index term. The Boolean retrieval system is designed to retrieve all stored records exhibiting the precise combination of key words included in the query: when two query terms are related by an and connective, both terms must be present in order to retrieve a particular stored record; when an or connective is used, at least one of the query terms must be present to retrieve a particular item.

Content-based Methods:

In content-based recommendation methods, the utility $u(c, s)$ of item s for user c is estimated based on the utilities $u(c, si)$ assigned by user c to items $si \in S$ that are "similar" to item s. For example, in a movie recommendation application, in order to recommend movies to user c, the content-based recommender system tries to understand the commonalities among the movies user c has rated highly in the past (specific actors, directors, genres, subject matter, etc.). Then, only the movies that have a high degree of similarity to whatever user's preferences are would get recommended.

Limited content analysis:

Content-based techniques are limited by the features that are explicitly associated with the objects that these systems recommend. Therefore, in order to have a sufficient set of features, the content must either be in a form that can be parsed automatically by a computer (e.g., text), or the features should be assigned to items manually. While information retrieval techniques work well in extracting features from text documents, some other domains have an inherent problem with automatic feature extraction. For example, automatic feature extraction methods are much harder to apply to the multimedia data, e.g., graphical images, audio and video streams. Moreover, it is often not practical to assign attributes by hand due to limitations of resource Another problem with limited content analysis is that, if two different items are represented by the same set of features, they are indistinguishable.

Proposed design:

This proposed system Hybrid extraction of robust model (HERM) is very efficient and reuse of e-learning resources in a distributed environment like the Web for better result. This proposed system argues that to achieve the on-demand semantic-based resource management for Web-based e-learning, one should go beyond using domain ontology's statically. The proposed approach for matching process in web cluster databases from different database servers can be easily integrated and deliver highly dimensional e-learning resource management and reuse is far from being mature. However, e-learning is also a widely open research area, and there is still much room for improvement on the method.

Architecture diagram:

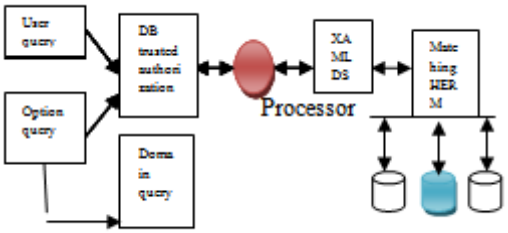

Fig.1 System Architecture of HERM

Fig.1.This architecture explains the context aware based evolution algorithm for dynamic e-learning resource reuse in detail. This system is going to conduct a simulation experiment and evaluate the proposed approach with a xaml e-learning scenario.

HERM Algorithm:

The extraction and matching process has been implemented step by step with initial requirements.

String [][] fillCLabMatrix (Tree of Nodes source, target);
String [][] cLabsMatrix;
String[] matchers;
int i,j;
Matchers=**getMatchers** ();
For each sourceAtomicConceptOfLabel in source
i=**getACoLID**(sourceAtomicConceptOfLabel);
for each targetAtomicConceptOfLabel in target
j= **getACoLID**(targetAtomicConceptOfLabel);
cLabsMatrix[i][j]=**getRelation**(matchers,
sourceAtomicConceptOfLabel,targetAtomicConceptOfLabel);
return cLabs Matrix
String **get Relation**(String[] matchers,
Atomic ConceptOfLabel source, target)
String matcher;
String relation Idk ;
int i=0;

while ((i<sizeof(matchers))&&(relation==Idk))
matcher= matchers[i];
Relation=**execute Matcher** (matcher, source, target);
i++;
return relation;

The major work of this module is perform the data matching by getting the user query the data request can be done in optimized fashion, that is, the data request can come from the data agent of any type of user query. Data Agent can request for explicit request with a specific condition, where in this application, it is modeled as patient information, here the agent can request for specific name. Agent can also request for sample request, that is, they can request for specific number of clustering by sequence. These requests are raised by the data agents during runtime and the open datasets are also allocated uniquely for each query during runtime. This Dataset accompanied by necessary additions depending upon the requirement. Open data set used to read, write, matching the data from more than one different database services. If your application requires working with data, you can load the data into a dataset, which provides your application with a local in-memory cache of the data to work. Information about loading data into a dataset.

III. CONCLUSION

The proposed technique HERM have given better results for solving the problem of safe personalization of modules built from an existing reference DMS. This raises new issues to check easily that a module-based DMS evolves independently but coherently the reference DMS from which it has been built. It have introduced two notions of module robustness that make possible to build locally the relevant queries to ask to the reference database in order to check global consistency (possibly upon each update), and to obtain global answers for local queries.

IV. ACKNOWLEDGEMENT

I would like to thank Ms. D.Radha for give excellent insights and suggestions.

REFERENCES

[1] Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization, H. Stuckenschmidt, C. Parent, S. Spaccapietra, eds. Springer, 2009.
[2] R. Kontchakov, L. Pulina, U. Sattler, T. Schneider, P. Selmer, F. Wolter, and M. Zakharyaschev, Minimal Module Extraction from DL-Lite Ontologies Using QBF Solvers," Proc. 21st Int'l Joint Conf.
[3] Artificial Intelligence (IJCAI), 2009.
[4] B. Konev, D. Walther, and F. Wolter, "Forgetting and Uniform Interpolation in Extensions of the Description Logic EL," Proc. 22nd Int'l Workshop Description Logics, 2009.
[5] B. Konev, D. Walther, and F. Wolter, "Forgetting and Uniform Interpolation in Large-Scale description Logic Terminologies," Proc. 21st Int'l Joint Conf. Artifical intelligence (IJCAI), 2009.
[6] K. Wang, Z. Wang, R.W. Topor, J.Z. Pan, and G. Antoniou, "Concept and Role Forgetting in ALC Ontologies," Proc. Eighth Int'l Semantic Web Conf. (ISWC), 2009.
[7] M.Y. Vardi, "The Complexity of Relational Query Languages," Proc. 14th Ann. ACM Symp. Theory of Computing (STOC), 1982.
[8] A. Cali, G. Gottlob, and T. Lukasiewicz, "Datalog+-: A Unified Approach to Ontologies and Integrity Constraints," Proc. Int'l Conf. Database Theory (ICDT), 2009.
[9] R. Cattell, "Scalable Sql and Nosql Data Stores," SIGMOD Record, vol. 39, no. 4, pp. 12-27, 2010.
[10] B. Cuenca Grau, I. Horrocks, Y. Kazakov, and U. Sattler, "Extracting Modules from Ontologies: A Logic-Based Approach," Proc. Third Int'l Workshop OWL Experiences and Directions (OWLED), 2007.