EXTENDED METADATA FOR DATA WAREHOUSE SCHEMA

N. Parimala and Vinay Gautam
School of Computer & System Sciences, Jawaharlal Nehru University, New Delhi, India

Keywords: Data warehouse, E-Metadata, Ontology.

Abstract: We are concerned with providing support for identification of changes to the data warehouse schema. The approach involves, building an extended metadata, E-Metadata, using which we identify changes. In this paper we show the manner in which E-Metadata is built. E-Metadata consists of the technical metadata and an ontology. In the E-Metadata Development Process (EDP), first, the technical metadata is extracted from the metadata of the warehouse schema. In the next stage of ontology development process, the schema terms are extracted from the technical metadata. The data warehouse administrator is asked to provide business terms for the schema terms. We, then search the WordNet for synonyms, hypernyms etc. for these terms. Using this information we build the ontology.

1 INTRODUCTION

The Metadata is physical data and knowledge containing information about technical and business processes, rules and structure of data. It is a key success factor of data warehouse projects. It captures all the information necessary to analyse, design, build, use and interpret the data warehouse contents. It is widely used to improve effectiveness and efficiency of data warehouse environment. Typically metadata has two categories of data - technical metadata and business metadata. The technical metadata includes schema definitions and configuration specifications etc, which is used by the developer and technical people. Business Metadata contains the information for end user. Nowadays, organizations have started to use standard meta-model for defining the metadata. Common Warehouse Model and Open Information Model are two metadata standards developed by OMG & MDC respectively to represent and enable interchange of metadata. (Thomas Vetterliy, Anca Vaduvaz and Martin Staudty, 2000).

In our earlier work, we defined a system, called Change Identification System (CIS), for identification of changes in the data warehouse schema. (Parimala N., and Vinay Gautam, 2010). Once the changes are identified by CIS, the Data Warehouse Administrator (DWA) may incorporate some of these changes in the data warehouse schema and its metadata. The corresponding changes must be now available for CIS. To support this, in this paper we define an extended version of the metadata of the data warehouse schema. This extended metadata, E-Metadata, is constructed using technical metadata and is enhanced with an ontology.

Ontology has been widely addressed in literature. The ontology is a specification of a conceptualization. The ontology specification is formally described. (Ahlemnabli, Jamel Feki and Farez Gargouri, 2009) (W. L. Lacy et al., 2005). Facts, features of the real world and their relationships are described with the help of a language in a document file. These expressions are in machine readable collection of terms. OWL (Web Ontology Language) which has been standardized by W3C has been adopted by many researchers. (www.w3.org/2004/ OWL). We define our ontology of E-Metadata using OWL.

The example used in this paper is the Insurance data warehouse schema shown in Figure 1. The metadata for this schema is contained in five files, Policy_holder.xsd, Policy.xsd, Claim.xsd and PolicyRevenue.xsd containing fact information Time.xsd containing the dimension information and Policy_Period defined as dimension attributes.

Claim.xsd contains the information about the type of claim. Time.xsd contains the time hierarchy.

1.1 Related Work

Data warehouse metadata is used for building, maintaining, managing and using the data

Parimala N. and Gautam V..
EXTENDED METADATA FOR DATA WAREHOUSE SCHEMA.
DOI: 10.5220/0003468202540259
In *Proceedings of the 6th International Conference on Evaluation of Novel Approaches to Software Engineering* (ENASE-2011), pages 254-259
ISBN: 978-989-8425-57-7

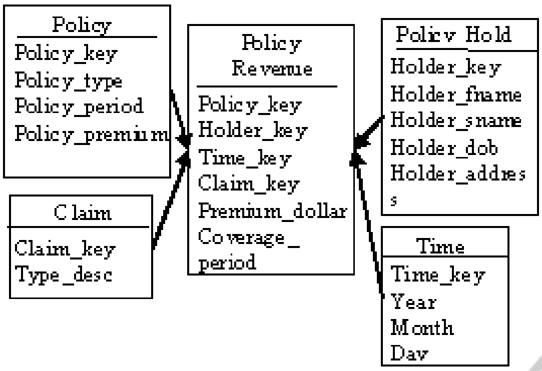

Figure 1: Insurance Schema.

warehouse. It is widely considered as promising driver for improving effectiveness and efficiency of data warehouse system. The metadata is used to support data warehouse developers and business people. It is managed by metadata repository. (Anca Vaduva, 2001). As brought out above it consists of business metadata and technical metadata. The business metadata is defined to support end-user. Some examples of systems that use metadata. (Guotong Xie, Yang Yang, Shengping Liu, Zhaoming Qiu, Yue Pan and Xiongzhi Zhou, 2010) (Veronika Stefanov and Beate List, 2006) (N. L. Sarda, 2006). The business metadata is used to provide flexibility to data mart deployment from data warehouse. (Guotong Xie, Yang Yang, Shengping Liu, Zhaoming Qiu, Yue Pan and Xiongzhi Zhou, 2010). The business metadata is used to integrate data warehouse and enterprise goals by building a model to provide links between enterprise goals and data warehouse. (Veronika Stefanov and Beate List, 2006). Temporal object oriented business metadata model is developed to provide context to business management and decision support. (N. L. Sarda, 2006). The technical metadata, on the other hand, is defined as consisting of schema definitions and configuration specifications, physical storage information, access rights, executable specification like data transformation, plausible rules and run time information like log files.(Won Kim, 2005). The technical metadata is defined to support developer and technical people. Some examples of systems that have used technical meta data (Wita Wojtkowski, Gregory Wojtkowski, Stanislaw Wrycza and Joze Zupancic, 2010) (Wua, Millera and Nilakantab, 2001) (Katic, Quirchmay, Schiefer, Stolba and Tjoa,1998). The technical metadata is used to provide support to incorporate the changes in data warehouse. (Wita Wojtkowski, Gregory Wojtkowski, Stanislaw Wrycza and Joze Zupancic, 2010). It is used to design data warehouse and generate the required sets of relational queries in (Wua, Millera and Nilakantab, 2001). The technical metadata is used to provide a security model for data warehouse. (Katic, Quirchmay, Schiefer, Stolba and Tjoa,1998). Here, the technical metadata contains the information such as access rules, classification of security objects or clearances of security subjects. We use the technical metadata to build E-Metadata.

The ontology is a specification of shared conceptualization. The static as well as dynamic ontology is used in information systems. It is expressed using many approaches and languages and has been studied in detail. (Igor Jurisica, 2004) (Paulheim and Probst, 2010). Ontology has been used in different applications. The ontology is developed to support OLAP operations for analysis in a multidimensional system. (Kurze, Gluchowski, Bohringer, 2010). The ontology is combined with database metadata to construct a data space to tackle the issues of data management in complicated scientific studies. (Ting Wang, 2010). We use the ontology with the technical metadata for identification of changes in a data warehouse schema.

The layout of the paper is as follows. Section 2 deals with the E-Metadata development process. In Section 3 an example is considered. Section 4 is the concluding section.

2 E-METADATA DEVELOPMENT PROCESS

In our earlier work, Change Identification System (CIS), we defined an ontology to help in identification of changes in the data warehouse schema. (Parimala N., 2010). To support identification of changes and evolution of the ontology along with the evolution of the data warehouse schema, we define and build, in this paper, an extended metadata of the data warehouse schema. This extended metadata, E-Metadata, consists of the technical metadata of the data warehouse schema and an ontology. The ontology itself consists of data warehouse business terms, domain terms etc. The ontology is built starting from the terms that exist in the technical metadata. The E-Metadata development process is explained in section 3. Formally, we define E-Metadata as follows:

EM : <O,M>

where

EM: E-Metadata
M: Technical metadata for data warehouse.
O: Ontology

It maybe queried as to why the technical metadata is maintained in E-Metadata. When changes are made to the warehouse schema, the corresponding metadata will also undergo a change. We extract the new technical metadata and the difference identifies the changes. This drives version management of E-Metadata. Version management is, however, not addressed in this paper.

2.1 Building the E-Metadata

The data warehouse metadata consists of technical metadata and business metadata. The E-Metadata is built by first, extracting the technical metadata. In the next step, the technical metadata is used to build the Ontology. To start with the names of facts, attributes and dimensions in the metadata are extracted. These may not necessarily be business terms. The corresponding business terms have to be identified by the DWA. These can be enriched with other terms. These other terms are additional domain terms. Once this information is available, we search the WordNet to add synonyms etc.

The E-Metadata Development Process (EDP) is a two stage process as shown in Figure 2. In the first stage, the technical metadata is extracted from the metadata of the data warehouse schema. The second stage is the Ontology Development Process (ODP).

Stage 1: Technical Metadata Extraction Process: Technical metadata (data warehouse schema definition) is imported from metadata definition.

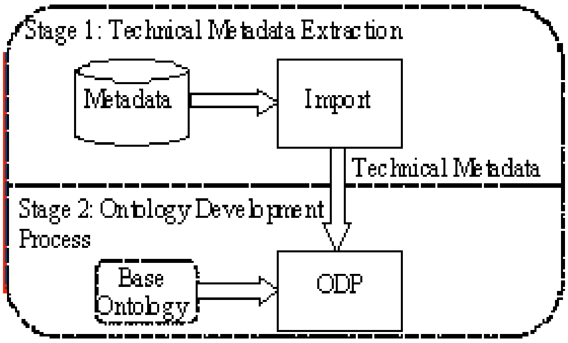
Stage 1: Technical Metadata Extraction
Metadata
Import
Technical Metadata
Stage 2: Ontology Development Process
Base Ontology
ODP

Figure 2: Metadata Development Process.

Stage 2: Ontology Development Process: ODP starts with a base ontology and the output from stage 1. Base ontology contains the core concepts or classes of data warehouse such as fact, dimension and attribute etc.

A. Extraction: In this step, Java API is used to extract data warehouse constructs from technical metadata expressed in XML format. In this process, each extracted construct known as 'token', represents a small piece of information.

The base ontology is shown in Figure 3.

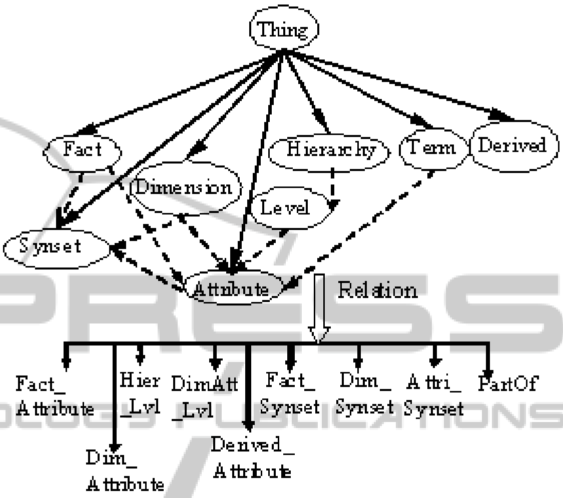

Figure 3: Base Ontology.

ODP consists of three steps as shown in Figure 4.

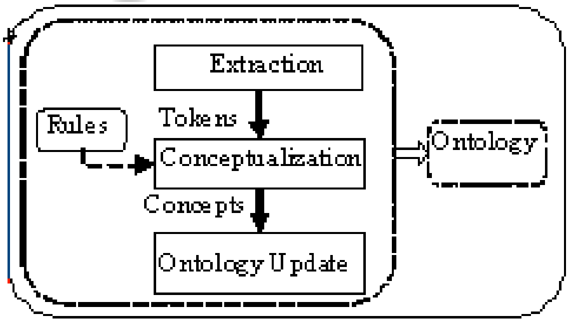

Figure 4: Ontology Development Process (ODP).

B. Conceptualization: The 'conceptualization' is an important step in ODP. Here, the base concept for each 'token', from among the base ontology concepts, is identified. Subsequently, the domain concepts are added. The two steps are shown in Figure 5.

a) Identify Base Concept for each Token: The 'token' is identified as an instance of Fact, Dimension and Attribute among base ontology. The following rules are used to identify the base concept for a given token.

Rules: To identify base concepts for each token

```
R1:If (fact) then "Base Concept of
token is the Fact"
R2:If (dimension) then "Base Concept
of token is the Dimension"
R3: If (simple element) then "Base
Concept of token is the Attribute".
```

b) Identify Domain Concept for each Token: The token represents data warehouse schema term. So the DWA is asked to add a business term corresponding to each token. The domain concept consists of this business term and other information extracted for this business term from the WordNet. For a given word in the WordNet we look for the following information:

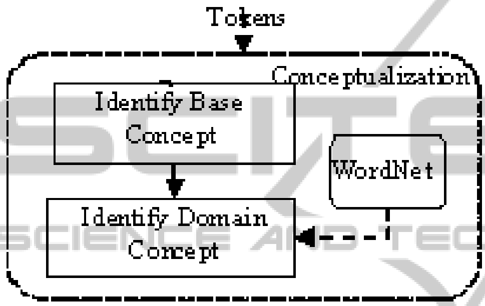

Figure 5: Conceptualization.

- Synonyms – words with similar meaning.
- Hypernyms-Hierarchical relationship of a word.
- Meronyms- Contains Partof relationship of a word.

The above information can be added provided the business term is found in the WordNet. If the business term is not found in the WordNet then the DWA is asked to specify an equivalent term. WordNet is searched for this equivalent term. If found, then as before the synonyms etc are picked up. If it is not found then the DWA is asked to specify some synonyms if possible. The approach of asking for an equivalent term was prompted by studying different example schemas. (http://merc.tv /img/fig/ Adventure WorksDW2008.pdf, http:// www.information-management-architect.com /star-schema .html). The schema term CalenderYear has no entry in WordNet. However, the word Year is present in WordNet. If the DWA can specify Year as an equivalent term then it enhances the ontology. The extracted concepts and base concepts are shown below in Table 1.

C. Ontology Update: The concepts extracted in the previous step are new concepts, which are added as instances of base concepts among the base ontology concepts. A token or a business terms refers either to a fact or a dimension or attributes thereof. The business term, therefore, is added either as an instance of Fact or Dimension or Attribute. The synonyms are added as instances of Synset; the hypernyms are added as instances of Hierarchy and the meronyms or part-of are added as instances of Term in the base ontology. Table 2 below shows a few examples.

Table 1: Instance and base concepts.

Concept	**Base Concept/Class**
Business term	Fact/Dimension/Attribute /Hierarchy/Level
Synonym	Synset
Meronym	Term
Hypernyms	Hierarchy

Table 2: Instance and base concept.

Token/Business term/WordNet term	**Instance of Concepts**
Policy	Dimension
Policy Revenue	Fact
Holder name	Attribute
City	Term
Customer is synonym of Policy Holder	Synset
Year-> month-> day WordNet hierarchy	Hierarchy
Year, month, day	Level

There are different types of relations defined in the base ontology as shown in Figure 3. After identifying the concept to which a token belongs, we identify the relations. The classes to which the tokens belong are used to determine the relations. Below are a few examples:

- Policy is an instance of Dimension. Policy_type is an instance of Attribute. Now, the relation between Policy and Policy_type is an instance of Dim_Attribute.
- Policy Revenue is an instance of Fact. Premium_in_dollar and Claim_limit are instances of Attribute. Now, the relation between Policy Revenue and Premium_in_dollar, Claim_limit is an instance of Fact_Attribute.
- Premium is an instance of Synset and it is a synonym of Policy_premium. Policy_Premium is itself an instance of Attribute. Thus, the relation between premium and policy_p remium is an instance of Attri_synset.
- City is an instance of Term. State is an instance

of Attribute. The relation between City and State is an instance of PartOf.

- Year, Month and day are instances of Attribute. Now, the relation between Year & L1, Month & L2 and day &L3 is an instance of DimAttr_Lvl. L1-> L2->L3 (H1) is an instance of Hierarchy. The relation between L1, L2 and L3 are instances of Hier_Lvl.

3 CASE STUDY

As an example, we demonstrate the application of this approach for the warehouse schema shown in Figure 1. We now trace the steps outlined above.

Stage 1: Technical Metadata Extraction: In this stage, the technical metadata is imported from Insurance data warehouse metadata repository. This metadata (in the form of XML schema) is an input to next stage of this approach.

Stage 2: Ontology Development Process

A. Extraction: This is used to extract tokens from the XML file. For example, from Policy.xsd the tokens that are generated are Policy, Policy_type, Policy_period and Policy_premium.

B. Conceptualization

a. Identify Base Concept for each Token: The 'token' is identified as an instance of Fact, Dimension and Attribute among base ontology. For example, as per the rules above, Policy is a Dimension and Policy_type, Policy_period and Policy_premium are Attributes. So the base concept for Policy is Dimension and Attribute for Policy_period, Policy_type and Policy_premium.

b. Identify Domain Concept for each Token: This contains two steps shown below:

For example, Policy, Policy_period are schema terms. The business terms corresponding to them are policy, policy period. Now, these words are searched for in WordNet for synonym, hypernym and meronym shown below in Table 3. Policy is present in WordNet but policy period is not present. So DWA is asked to add some specific term for further search.

C. Ontology Update: Policy, Policy type, Policy period and Policy premium are added as instances of base concepts in the base ontology as identified in step B. Policy is added as instance of Dimension and Policy type, Policy period and Policy premium are added as instances of Attribute. The synonym, hypernym and meronym are added as instances of Synset, Hierarchy and Term.

Table 3: Synonym, Herpernym and Meronym from WordNet.

Business terms	**Synonym**	**Hypernym or Hierarchy**	**Meronym or partof**
Policy	Sense 3 insurance policy;	Sense 3 insurance policy	No
policy period	Not found	---------	------------

4 CONCLUSIONS

In this paper, we have proposed an extended metadata, E-Metadata, which is a combination of the technical metadata and the ontology. The benefits of extending the metadata are twofold: firstly, it can be used to identify the changes to the data warehouse schema. Secondly, all changes mode to the schema and the corresponding changes to the ontology are maintained in one place, thus ensuring consistency.

The E-Metadata Development Process (EDP) is used to build the E-Metadata. First the technical metadata is imported from metadata represented in XML format. Subsequently, the ontology development process, starting with the base ontology and the technical metadata builds the Ontology.

The ODP has three steps - extraction, conceptualization and Ontology update. In the extraction step, tokens are extracted from the technical metadata. The Wordnet is used to find synonyms etc in the second step. Subsequently, the base ontology is updated with the new terms of the previous steps.

We explored the possibility of using WSD. For example, after the DWA specifies the word sense of policy_type whether it is possible to pick up the sense of policy_period considering that both the attributes belong to the same dimension. However, since there is no relationship between the two words WSD could not be used. We considered other schemas as well for the applicability of WSD algorithm. Except for Time attribute, it was not possible to use WSD for disambiguation of senses of attributes of any other dimension or fact.

The technical metadata of E-Metadata is generated automatic. The only effort to be put by DWA is to define the ontology. This however is a onetime effort. We expect that this approach can be adapted in large data warehouse schema as well.

The system is being implemented using Java to ccess WordNet, Oracle for metadata management and OWL API's for updating ontology.

REFERENCES

Ahlemnabli, Jamel Feki and Farez Gargouri., 2009. An ontology based method for normalization terminology, LNCS, Springer, Vol., 235-246.

Anca Vaduva, 2001, Metadata Management for Data Warehousing: Between vision and reality, *Database Engineering & Applications*, 129-135.

George Angelos Papadopoulos, Wita Wojtkowski, Gregory Wojtkowski Stanislaw Wrycza, Jo¿e Zupancic, 2010, Metadata support for data warehouse evolution, *Information Systems Development*.

Guotong Xie, Yang Yang, Shengping Liu, Zhaoming Qiu, Yue Pan and Xiongzhi Zhou, 2010, EIAW: Towards a Business-Friendly Data Warehouse Using Semantic Web Technologies, *2nd Asian conference on Asian semantic web conference*.

Igor Jurisica, 2004, Ontology's for Knowledge Management: An Information Systems Perspective, *Knowledge and Information Systems* Volume 6, Number 4, 380-401.

Katic, N., Quirchmay, G., Schiefer, J., Stolba, M., Tjoa, A. M.,1998, A Prototype Model for Data Warehouse Security Based on Metadata, *International Workshop in DESA*.

Kurze, C. Gluchowski, P. Bohringer, M., 2010, Towards an Ontology of Multidimensional Data Structures for Analytical Purposes, HICSS.

L. Wua, L. Millera, S. Nilakanta, 2001, Design of data warehouses using metadata, *Information and Software Technology*,109-119.

N. L. Sarda, 2006, Structuring Business Metadata in Data Warehouse Systems for Effective Business Support, *International Enterprise Object Computing Conference Workshops*.

Parimala N., Vinay Gautam, 2010, CIS: Change Identification System, *proceeding Knowledge Engineering and Ontology Development SciTePress*.

Paulheim, H., Probst, F, 2010, Ontology-Enhanced User Interfaces: A Survey, *International Journal on Semantic Web and Information Systems*.

Peter Spyns, 2008, An Ontology engineering methodology for DOGMA, Applied Ontology archive.

Ting Wang, 2010, Constructing a Dataspace Based on Metadata and Ontology for Complicated Scientific Data Management, *Pervasive Computing and Applications*.

Thomas Vetterliy Anca Vaduvaz Martin Staudty, 2000, Metadata standards for data warehousing: Open Information Meta-model Vs Common Warehouse Meta-model, ACM SIGMOD.

Veronika Stefanov, Beate List, 2006, Business Metadata for data warehouse – weaving Enterprise goals and models, *Journal of Object Technology* 4(2), 41-48.

W. L. Lacy, 2009. Representing Information Using the Web Ontology Language, *Trafford Publishing*.

Won Kim, 2005, On Metadata Management Technology: Status and Issues, *Journal of Object Technology*.

OWL, 2004, www.w3.org/2004/OWL.