Metadata of Dashboard Data Source Based on Study of Pentaho Dashboard Metadata

Rosni Lumbantoruan, Agnes Juliana Siregar, Erikson Matondang and Marisa Helen Gultom

***Abstract*—In this paper, it is conducted an analysis of the two dashboards that applied metadata, they are "Metadata for Creating and Displaying Dashboard" and open source Pentaho dashboard. The analysis is to determine whether the metadata of the dashboard has met the metadata structure of data warehouse by comparing each metadata component of both dashboards to the metadata structure of data warehouse. Metadata component of each dashboard that meets the metadata structure of data warehouse will be combined to produce a new prototype named Meta ++.**

Besides Meta ++ as a blueprint of the combined metadata, there should be a generator for generating metadata based on inputs provided. Meta ++ will be used to assist the performance of metadata generator in generating real metadata. A metadata in the form of XML file will be generated by metadata generator that based on the information needs input from the user.

***Index Terms*—dashboard, generator, metadata, Pentaho.**

I. Introduction

A. Background

PREVIOUS research related to the development of metadata for dashboard has been considered in [1] using XML as metadata format because XML is flexible and supports writing of tags according to user requirements and can be used on a variety of platforms. The reference [1] uses descriptive metadata that plays an important role in describing the data. An important reason for creating descriptive metadata is to facilitate discovery of relevant information. In addition to resource discovery, metadata can help to organize electronic resources, to facilitate interoperability and legacy resource integration, to provide digital identification, and to support archiving and preservation [3]. Descriptive metadata is well in displaying the details of the entire data but missed data structure and data identification structure [1].

On the other hand, the second related work that will be analyzed and compared is an open source dashboard that applied metadata, named Pentaho. Pentaho dashboard provides an integrated features to do analysis, for this reason, it is widely used by businesses to produce data; in addition to that Pentaho is sturdy because it was built for a long time, and strong in terms of visualization and data unlimited sources and able to accommodate millions of lines of information [2].

The study is conducted to generate metadata that is capable of displaying data as a structured together with its identity that makes the data uniquely identified; this is consistent with the data warehouse metadata attributes. Hence, a comparison is done against the data warehouse metadata to get better metadata results. Each metadata attribute of both studied dashboards, namely 1) open source Pentaho dashboards and 2) dashboard results of the study entitled "Metadata for Creating and Displaying Dashboard" is analyzed and compared to the metadata structure of data warehouse. Metadata of the both dashboards that meets the criteria of data warehouse metadata structures will be combined to form a metadata prototype called the Meta ++. To support the creation of metadata that meets the requirements of users, a metadata generator is developed. Meta ++ is used as a guideline for metadata generator to generate an actual metadata in the form of XML. Furthermore, Meta ++ is applied to the previous developed application dashboard in [1].

B. Objectives

The aim of this study is to analyze the implementation of the conformity of open source Pentaho dashboards metadata and dashboard results of the study "Metadata for Creating and Displaying Dashboard" to metadata structure of data warehouse [9]. It is then to compile metadata prototype for the metadata improvement that will be applied to the dashboard resulted [1] and later it will be named Meta++.

II. Literature Study

A. Metadata

Metadata is often called data about data or information about information. Metadata is structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information resource. Metadata is a key to ensuring that resources will survive and continue to be accessible into the future [3]. Based on NISO, there are three main types of metadata as follow:

a) Descriptive Metadata: describes a resource for purposes such as discovery and identification. It can include elements such as title, abstract, author, and keywords.
b) Structural Metadata: indicates how compound objects are put together, for example, how pages are ordered to form chapters.
c) Administrative Metadata: provides information to help manage a resource, such as when and how it was

Manuscript received July 28, 2015. This paper with title "Metadata of Dashboard Data Source Based on Study of Pentaho Dashboard Metadata" was supported by Del Institute of Technology.

Rosni Lumbantoruan, is with Del Institute of Technology, Indonesia. She is now with the Department of Information System (e-mail: rosni@del.ac.id).

Agnes Juliana Siregar, Erikson Matondang and Marisa Helen Gultom were students in Informatics Engineering Department, Del Institute of Technology, Indonesia.

created, file type and other technical information, and who can access it.

B. Metadata Format

Metadata has a specific format that is used based on its needs. The metadata format may consist of UML (Unified Modeling Language), XML (Extensible Markup Language), XMI (XML Metadata Interchange) and others. The metadata considered in [1] uses XML while Pentaho dashboard uses XMI.

Developed in 1996, XML stands for Extensible Markup Language, which means flexible, scalable, and adaptable. XML can be anything as required by the document for the distribution of information over the Internet between applications. XML is a markup language that contains the code in the form of certain signs with specific rules for formatting a document with its own tag to be understood. XML also provides a structured format for describing data [10].

On the other hand, XMI (XML Metadata Interchange) is a metadata format that becomes a standard for programmers and other users to exchange information of metadata. Specifically, XMI uses UML (Unified Modeling Language) which is a different programming languages and development tools for the exchange of information of data warehouse. XMI allows the exchange of models in various forms. XMI is a parallel mapping, where the mapping is done by MOF (Meta Object Facility) meta model and XML DTD and other mapping conducted by MOF meta model and XML documents [9].

XMI is an OMG standard for exchanging MOF based on meta model. The standard consists of:

a. A set of XML Document Type Definition (DTD) which generates rules to change the shape of MOF based on meta model into XML DTD
b. DTD document that generates a set of rules for encoding / decoding MOF based on metadata
c. Design principles for XMI based DTD and XML flows
d. DTD foundation for UML and MOF

XMI has two main components, namely:

1) XML DTD Production Rules generate XML DTD. XML DTD provides syntax specification for XMI document, and allow common XML tools that are used to transform and validate XML documents
2) XML Document Production Rules encodes metadata to a compatible XML. Production rules can be applied to decode of XMI to reconstructs metadata

XMI is an OMG modeling standards for metadata exchange using CWM models that allow users to follow specific data structures. CWM (Common Warehouse Model) is format that applied XMI concept as mechanism to interchange metadata that means flexible and support XMI ability in exchanging warehouse metadata and CWM meta model.

DTD standard for CWM meta model was produced by XMI main component, XML DTD Production Rules. Data warehouse metadata can be encoded; XML document standard for CWM meta model can be generated using XML Document Production Rules by the MOF DTD.

C. Data warehouse Metadata

There are many factors that contribute to the quality of the metadata. Here is a framework of guidance for building good digital collections according to NISO (National Information Standards Organization) to implement good metadata [3].

a) A good metadata has to be in accordance with the material collection and user collection
b) A good metadata supports interoperability.
c) A good metadata uses a controlled standard vocabulary to reflect what, where, when, and from whom the available content.
d) A good metadata contains a clear statement on the conditions and terms of use of digital object.
e) The note of a good metadata is the object itself; therefore, it must have quality in archiving, persistence, and unique identification. Good metadata should be authorized and verified.
f) A good metadata supports long-term management of objects in a collection.

Moreover, given the dashboard as one of the Business Intelligence that uses data warehouse as a data source, then the metadata that should be used must be a metadata that describes the data source. As described by David Haertzen, a chief enterprise architect, metadata is the control panel to the data warehouse, which describes the data from the data warehouse and business intelligence systems such as reports, cubes, tables (records, segments, entities), column (fields, attributes, the data element), keys, indexes [4].

Dashboard with objectives as data analytic needs data warehouse so that metadata created should be able to gather data from many data sources, necessitating id which resulted in each data source in the metadata is unique, so that the tag `<DataSource id = "en"> .. </ DataSource>` is required and `<DBMS name = "SQLServer" />` tag is needed in its connection string to determine the proper driver and ensures dashboard application to connect to database. `<HOST name = "localhost" />` tag is used to identify the framework of running DBMS, `<USERNAME name = "root" /> || <PASSWORD name = "root" />`, both of these tags are used for user authentication.

The data warehouse has specific metadata requirements. For example, metadata that describes the table should have the followings [4].

d) Physical name: the name of an existing table in the database
e) Logical name: Business name table used
f) Type: Type tables in the database
g) Role: The role of the database table
h) DBMS: DBMS types used in metadata
i) Location: The connection string used in the metadata
j) Definition: A description of any existing table in the database.

In addition, metadata that describes the column should contain the followings [4]:

a) Physical name: the name of an existing column in the table in the database;
b) Logical name: business name of the columns used in the table;
c) Order of the table: the order of the columns in the database).
d) Data type: type of data used on each column;
e) Length: length of the characters used in a string;
f) Decimal position: decimal position of a column with integer values;
g) Nullable/required: requirement on the value of a column;
h) Default: default value of a column, and

i) Definition: description of each column in a table

As CWM (common warehouse meta model) is metadata standard that plays a role in the exchange of metadata of data warehousing and metadata of business intelligence and distributed in different environments. Thus, it can be said that the CWM is a standard metadata for data warehouse and Business Intelligence as dashboard.

D. Analyzed Result of "Metadata for Creating and Displaying Dashboard"

Metadata on the dashboard is used to describe the design, the result of design changes, and relationship amongst tables in the data source. The aims of this dashboard are to easily differentiate the information that needs to be displayed from different data sources and to save the best practices in displaying the information to the user to minimize ambiguity perceptions [1].

Descriptive metadata was used to describe the data structure. Descriptive metadata consist the explanation of the elements owned by the table as a field, data type, primary keys, foreign keys and tables that has relationship with the table itself that will finally form the metadata that describes a whole set of relationships that the database has. By then, from this metadata, the application can determine what information is held by the database.

This metadata has also function in determining the description of a good design for a data visualization in a dashboard and store the results of the changes made by the user to the visualization (as if user made modification to the visualization for a better result).

E. Metadata Description of "Metadata for Creating and Displaying Dashboard

Dashboard result in [1] is considered by using a component of the XML metadata stored in a CWM package that is XML Schema. XML Schema aims to explain the types of XML documents, it is usually expressed in terms constraints of structure and content of the document. XML schema is an alternative to XML-based DTD. XML Schema document called the XML Schema Definition (XSD). XML Schema is used to define the range of values for XML elements or attributes.

An XML Schema works as follow [5]:

a) Defining the elements that might present
b) Defining the attributes that might present
c) Defining child elements
d) Defining the sequence of child elements
e) Defines the number of child element
f) Defines whether an element is empty or can contain text
g) Defines the data types of elements and attributes
h) Defining the default value and the fixed value of the elements and attributes

XML Schema is used for the following reasons:

1) XML Schema is more extensible for future changes
   a. Reusability of existing XML Schema to the others XML Schema.
   b. Make the data type out of the standard data type. Referring to others XML Schemas from one XML document.
2) XML Schema is written using XML rules
   a. No need to learn a new language
   b. Can use existing XML editor
   c. Can use existing XML parser to process an XML schema
   d. Can be transformed with XSL
3) XML Schema supports data types
   a. It is easier to define the content that is allowed in a document
   b. It is easier to validate the data
   c. It is easier to work with the data type of database
   d. It is easier to define the constraints of data
   e. It is easier to define data patterns
   f. It is easier to convert between data types
4) XML Schema is more complete and powerful than DTD
5) The XML Schema supports namespace
6) XML Schema is a W3C recommendation

Elements in the XML Schema are defined in several types [5]:

1) Simple elements: Elements that contain only text
2) Complex element
3) Empty element: Element is empty
4) Element content: Elements that contain other elements
5) Simple content: Elements containing text and attributes
6) Mixed content: Elements that contain other elements, text, and attributes

The metadata in [1] did not store metadata of each object in database and it also required data in order to connect to database. Information regarding the database connection was stored in different file. Each time a data source defined, it should rewrite it in the metadata. This makes it is impossible for this metadata to connect to different data sources as need in data warehouse. This drawback should be overcome in Meta++.

F. Pentaho Dashboard Metadata

Metadata on Pentaho is described with the term *concept* to mean a collection of metadata properties that can be applied to a given business object. Metadata is wrapped in one area, called domains, which describes the entire created, stored, and used metadata business object on metadata layer. A domain consists of one or more connections, models, information security, business table, business view, categories, columns, and concept. Domains can be described as metadata documents that are published in the form of files in .XMI extension. A domain metadata accessed by BI server by publishing or to exporting a domain into .XMI file, and move the file to the Pentaho storage [6].

Metadata type that is used on this dashboard is structural metadata. Dashboard metadata describes the relationship between two different business objects where the object will inherit metadata from the other objects. Business model of metadata is a major component in the Domain Metadata Pentaho. The domain encapsulates descriptions of physical or logical database object models (business models). It aims to simplify the user by not informing the location of these objects through dashboard visualization.

Some business models can be created on a domain. But the business model can only be connected to a reference, which means it cannot combine the physical table from two different connections on a business model. This prevents business model in using more than one data source. But with

the ability of Pentaho Metadata Editor that allows the table to exchange connection, this feature should switch from database development to production database.

Pentaho Metadata format is in XMI (XML Metadata Interchange). Effectively, XMI format standardizes how different sets of metadata are described and require the user through many industries and operating environments to see the data in the same way [6].

G. *Metadata Pentaho Description*

Pentaho Dashboard uses standard metadata CWM (common warehouse meta model), XMI. CWM specifies the interface used to allow exchanges between intelligence data warehouse metadata and metadata business intelligence platform with the tools in the warehouse. Furthermore, metadata repositories data warehouse is distributed in different environments. XMI is an OMG modeling standards for metadata exchange using CWM models that allow users to follow specific data structures

Metadata storage architecture at the Object Management Group (OMG) can be seen in Fig. 1.

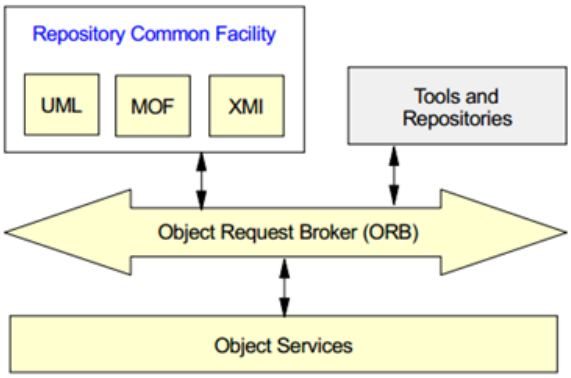

Fig. 1. Object Management Group (OMG) Metadata Storage

The followings are some XMI syntax in CWM model used on the dashboard Pentaho metadata [12]:

1) CWMTFM (Common Warehouse Metamodel Transformation)
   CWMTFM consists of classes and relationships that describe the common metadata transformation used in data warehouse. Transformation changes set of source objects into a set of the target object. For examples, tables, columns, or other model elements are depicted in the memory as a physical object.
2) CWMMDB (Common Warehouse Meta Model Mining Data Base)
   CWMMDB is a data mining technique that provides a description of information that shows the innate (inherent pattern) as well as the relationships among the data.
3) CWMRD (Common Warehouse Meta model Relational Database)
   Relational illustrates the description of the data that can be accessed via the interface as a relational DBMS, ODBC, or JDBC. Relational-based SQL or a standard focuses on RDBMS.
4) CWMOLAP (Common Warehouse OLAP Meta Model)
   a. Determine the things that are important about the concept of OLAP metadata in general be used on OLAP system
   b. Provide a facility where OLAP meta model instance mapped to the structure of the development, including data sources such as the physical models and Package Relationships CWM Multidimensional
   c. Providing navigation through OLAP logic hierarchy models and diverse sources models that displayed simultaneously.

III. DISCUSSION

This chapter below describes the conformity of two previous related studies to data warehouse metadata standard.

A. *Metadata Attributes based on NISO*

Based on the type of metadata by NISO (National Information Standards Organization), metadata that aims to describe the physical or logical structure of a complex object is structural metadata. The followings are the attributes that, a proposed metadata, Meta++ must have, they are:

Connection String

The connection string is an attribute to connect the application to the database. Remembering that data warehouse is a collection of databases that might use different DBMSs, so it is necessary to have an attribute that might store id of connection and DBMS type uniquely. The attributes can be shown through the example of metadata for the connection string as shown in the Metadata 1.

```
<Connection name=" ">
    <Host value=" " />
    <Type value=" " />
    <TargetDatabase value=" "/>
    <Username value=" " />
    <Password value=" " />
    <Port value=" " />
 </Connection>
```

Metadata 1. Connection string attributes

Tables

Schema to portray table is divided into two parts: physical and logical. Physical table stores table name and data source; while logical table stores table as part of the business. For business table, the attributes that describe the column also envisaged, the existing columns will be sorted in accordance with the conditions of the actual table. ID is needed to ensure physical and logical objects table unique.

Examples of metadata attributes for physical table displayed in the Metadata 2.

```
<PhysicalTable name="" tblRef="" />
```

Metadata 2. Physical table attributes

Examples of metadata attributes for a logical table can be seen in the Metadata 3.

```
<BusinessTable name=" ">
   <PhysicalTable name="" tblRef="" />
   <Columns>
     <Column name=" " id=" " />
     <Column name=" " id=" " />
   </Columns>
</BusinessTable>
```

Metadata 3. Logical table attributes

Columns

In the schema, the column will also store information regarding the names, both physical and logical. Examples of metadata attributes for the column can be seen in the Metadata 4.

```
<BusinessColumns>
   <BusinessColumn name=" ">
      <PhysicalName name=" " />
    <BusinessTable name=" " />
   </BusinessColumn>
   <BusinessColumn name=" ">
      <PhysicalName name=" " />
      <BusinessTable name=" " />
   </BusinessColumn>
</BusinessColumns>
```

Metadata 4. Column Attributes

Examples of metadata attributes for the behavior/description of the column can be seen in the Metadata 5.

```
<ColumnsDescription Id=" " Type=" "
DataType=" "
Length=" "
Definition=" "
Nullable=" "
DefaultValue=""
/>
```

Metadata 5. Description Attributes

Based on the results of the analysis, it can be concluded that important attributes in metadata are connection, physical table, physical column, business models, business table, business column, and relations

B. Comparison of Metadata Attributes

Metadata Pentaho has eminence compared to "Metadata for Creating and Displaying Dashboard" in [1], but this metadata has superiority in easily understandable way in describing object by using a common tag for a database metadata designer so that anyone with a basic knowledge of the introduction of the database can understand the structure of a database by just reading the metadata. Table I below illustrates the comparison between the two metadata are analyzed.

TABLE I
METADATA ATTRIBUTES COMPARISON SUMMARY

Metadata data *warehouse*	**Metadata for Creating and Displaying Dashboard**	**Metadata** Pentaho
Multiple connection	-	✓
DBMS	-	✓
Metadata object definition logically or physically	-	✓
Table and column definition in metadata	✓	✓
Relationship	✓	✓

To generate metadata standard that meets all the metadata structures, it takes parser that can read XML file into an object. On this study, it is constructed a useful parser to read each metadata attribute that is stored in XML file into an object that can be used. Integration of each metadata attribute that has been analyzed would be combined as a metadata concept and used as the basis for generating metadata based on performance metadata prototype generator.

C. Comparison of Metadata for Creating and Displaying Dashboard to Data Warehouse Metadata Standard

Compared to data warehouse metadata standard, this metadata has attributes characteristics as described below:

1) Tables and Column Object are not specified

There is no explanation whether tables or column physical or logical. This criterion does not fulfill metadata standard that should specify the identity of the each object.

2) One data source for each metadata file

As described in this schema `<xs:element name="DATABASE">` concluded that metadata file consists objects from one data source only.

3) Metadata does not have information about the host and DBMS.
4) Metadata does not have description of tables of database.
5) Column properties such as length, decimal position, nullable/required, default value, edit rules, and definition are not described in metadata.
6) Metadata defined data ordered and type of each column.

Another characteristic of this metadata that is not part of the metadata standard is:

7) Metadata has described the relations amongst tables.

D. Comparison of Pentaho Metadata to Data Warehouse Metadata Standard

Compared to data warehouse metadata standard, Pentaho metadata has characteristics as follow:

1) Supports multi-connection to databases

Pentaho supports many connection types such as Apache Derby, MySQL, MS SQL Server, Oracle, MS Access, PostGreSQL, SQLite, etc.

2) Metadata has described the physical and logical name for each business object.
3) Metadata supports detail information of each column objects such as aggregate function, data type, length, definition, and column order.
4) Metadata has data source that can connect to different DBMSs.
5) Object location and metadata definition have been covered.
6) The role such as legacy, OLTP and stage are defined in object table.
7) There is no information regarding decimal position, nullable/required, default value, and edit rules in object tables.

Other characteristics of this metadata that are not part of the metadata standard are:

8) Metadata classifies objects based on its importance in business analytics. Metadata for business analytics is classified as business model.
9) Metadata has described the relations amongst tables.
10) Metadata uses the same data type of column object with data type in programming language.

11) Metadata contains object view. Like OLAP feature, this object collects attributes of various related transactional tables and makes it as one table.
12) Metadata describes the integration of many data sources in a given format.

E. Meta++

Based on the comparison above, it is designed a metadata prototype called as Meta++ that consists of 201 taglines with 8 metadata attributes value, they are connection, physical table, physical column, business models (consists of business table, business column, business view, and relation). Root of this metadata is WarehouseMetadata. WarehouseMetadata has elements such as BusinessModels, Table, Connection dan ColumnsDescription. In relation, there are elements such as TableParent, TableChield, TableParentField, TableChildField and JoinType.

F. Metadata Generator

After defining the mandatory attributes of metadata that are covered in Meta++, this prototype will be used as a blueprint of metadata that will be generated by metadata generator. Fig. 2 shows the main interface of the metadata generator.

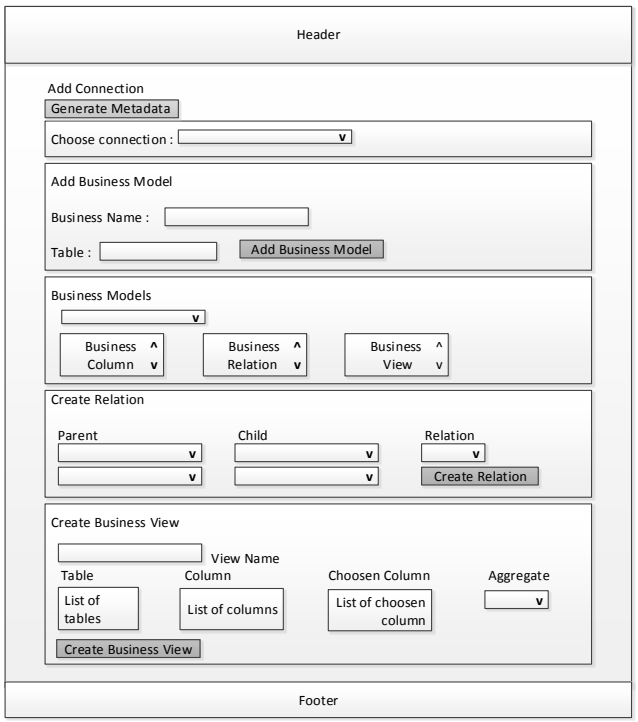

Fig.2. Interface of Metadata Generator

Before, user has to connect to data sources, choose the tables that will be involved in business analytics and then provide details as in Fig. 2 above.

Inputs by users will be saved as metadata attributes in XML file, however, the application will not be able to read those inputs before converted to an object format. In order to able to read each input as an object, parser is needed. In this study, this parses was created using LINQ. Parses was stored as MetaReader.cs.

In other cases, in order to able to generate Meta++, by metadata generator, the metadata attributes were mapped as programming code. Below are the flows of metadata generator in generating Meta++.

1) User defines the connection to databases
2) User defines the physical tables that should be stored in metadata file.
3) User defines the name for business model, selecting business tables that will directly adding the information to XML file.
4) User defines tables' relation; they are parent tables, child tables, and common field of related tables. This information will be in XML file in business model.
5) User defines business view. As in databases, view is used in order to combine attributes from different tables that later on will be part of data sources. This information will be in XML file in business model.
6) Generate Metadata button will generate and saved Meta++ in XML file as defined by the user.

IV. CONCLUSION

The structure of metadata on each of the different dashboard, as well as information stored in the metadata structure, resulted in the type of metadata that is used is also different. Based on the analysis, the structural metadata was a good type of metadata used to store data information which will be used in decision support. The proposed metadata, Meta++ was adapted to the standard attributes for metadata structure data warehouse. Meta++ was generated by metadata generator and saved as XML file.

REFERENCES

[1] L. Rosni, "Metadata for Creating and Displaying Dashboard in XML Schema", *in Proc. Information System International Conferences* (ISICO), Bali, Dec. 2013.
[2] *Why Pentaho* (2012) [Online]. Available: http://www.bizcubed.com.au/our-solutions/why-Pentaho
[3] G. Rebecca and R. Jacqueline, *Understanding Metadata* (Book style), USA: NISO Press, 2004, pp.1 – 16.
[4] *Metadata for Data Warehousing and Business Intelligence*, First Place Software, Inc., 2010. Available: http://www.tutorialspoint.com.
[5] XML Course website (2012) [Online]. Available: http://xml.constructive-learning.info/?p=105.2012
[6] Work with Relational Data Models (2013), Pentaho. Available: https://help.pentaho.com/Documentation.
[7] H. Marthadinata, S. Januar, and T. Winda. "Metadata untuk Pembuatan Dashboard", Dipl. Thesis, Dept. Information Sys. Del Polytechnic of Informatics, Indonesia, Jun.2012. (In Indonesian)
[8] NISO, *Understanding Metadata*, NISO Press. ISBN 1 -880124-62-9.
[9] OMG, Inc. *Common Warehouse Metamodel (CWM) Specification*, Version 1.1, Volume 1 formal, Mar. 2003.
[10] W. Norman, M. Leonard (1999, October). DocBook: The Definitive Guide (1st ed.) [Online]. Available: http://www.oreilly.com/openbook/docbook/book/ch01.html.

Rosni Lumbantoruan was born in Tarutung, Indonesia 24 October 1982. She granted the bachelor degree of engineering informatics from Bandung Institute of Technology, Indonesia in 2007 and master degree of information system development from HAN University of Applied Sciences, The Netherlands in 2010.

She has been working as a lecturer in information system department at Del Institute of Technology, Indonesia since 2007 till present. She also assigned as the head of database systems and information management cluster of study area at her institute since 2010, her interests are data/text mining, business intelligence application, data warehousing, information extraction, and information system development.