2016 International Conference on Platform Technology and Service (PlatCon)

Proceedings

15-17 February 2016
Jeju, Korea

IEEE Catalog Number: CFP16F03-ART (Xplore)
ISBN: 978-1-4673-8685-2 (Xplore)

IEEE Catalog Number: CFP16F03-CDR (CD)
ISBN: 978-1-4673-8684-5 (CD)

A Method of Handling Metamodel based on XML Database for SW Visulization

Hyun Seung Son
SELab., Dept. of Computer Information Communications
Hongik University
2639, Sejong Campus, 30016, Korea
son@selab.hongik.ac.kr

R. Young Chul Kim
SELab., Dept. of Computer Information Communications
Hongik University
2639, Sejong Campus, 30016, Korea
bob@hongik.ac.kr

***Abstract*—Software visualization shows to extract the architecture from source code. With software visualization, we can improve the software quality due to software modularization and also reconfiguration. To do this, we compose of parser, data analyzer, database management system, and visualizer like a tool-chain for realizing the software visualization. In this tool-chain, we develop the parser named xCodeParser that generate the OMG's Abstract Syntax Tree Metamodel (ASTM) for various language such as C, C++ or Java. In this paper, we propose the metamodel based XML databases to save the ASTM files into the repository. The proposed database is able to handle the data which organized metamodel based on XML database. Improving quality through realizing this visualization helps easily to develop high quality software, to increase reusability, and to reduce maintenance cost.**

Keywords—metamodel; Structured Query Language (SQL); XML databases; SW Visualization; SW quality

I. INTRODUCTION

The software has the nature of invisibility, complexity and changeability. Also the software depends on each individual maturity level of software developer [1]. Due to this reason, the software is difficult to develop a complete product which spends much more cost and time consuming with huge scale project. Even this software product should be continuously modified and changed, which spends more than 60 percent of the development costs for just maintenance [2]. This implies to happen the large number of software error from the software expansion and modification. Such software errors are represented as the Bathtub Curve [1].

Recently, the software is required to manage the systematic quality as for increasing the software complexity and frequently changing the user requirements. To improve software quality, we have various methods such as testing [3-5], formal method [6] or Test Maturity Model (TMM) [7]. But, these methods needs additional personnel and the cost, and does not provide a solution to improve software on identifying the error. Therefore, it is necessary to have the improving method of more fundamental software.

The software visualization is a method that makes an architecture from program codes through reverse engineering. It is easy to find & fix errors due to visualizing the defect from code. In NIPA's Software Engineering Center [8], to execute the visualization, they organize the open source based tool-chain using Source Navigator [9], Graphviz [10], SQLite [11], Jenkins [12], and etc. It is require to combine diverse tools such as parser, database and visualizer. To solve the problem, we proposed the parser [13] based on the OMG's Abstract Syntax Tree Metamodel (ASTM) [14], in the previous tool-chain for software visualization. In this paper, we propose the metamodel based XML databases to generate ASTM files from the parser. Actually, the previous metamodel oriented integration data repository have existed as the CDO Model Repository in eclipse [15]. The CDO Model Repository is Java based model repository for Eclipse Modeling Framework model and metamodel [16]. Also, the CDO supports persistence and distributed framework for EMF based application system, and stores the various databases such as JDBC, Hibernate, Objectivity DB, MongoDB, DB4O or etc.

But the CDO has two problems: 1) the large-scale framework speeds not fast, 2) the data communication is frequently broken with unstable connection. Also, It is a complex structure and not load ASTM files to process the data. To solve this problem, we use XML databases because the basic structure of model and metamodel is XML, and the model and metamodel is XML Metadata Interchange (XMI) [17].

Therefore, we propose the metamodel based XML databases to save the ASTM files in the repository. The proposed database handles the data which is organized metamodel based on XML database. With the software visualization, we help to improve quality, and to reduce the reuse & maintenance cost.

This paper is organized as follows. Chapter 2 explains related works, including the XML databases. Chapter 3 mentions a design of metamodel based databases. Last chapter mentions the conclusion and future work.

II. RELATED WORKS

XML database is a database system for storing the data in XML format directly. The XML databases have RDBMS to store the XML, and native XML databases. RDBMS has various product such as IBM DB2, Microsoft SQL Server, Oracle and PostgreSQL. It has advantageous to store the data

978-1-4673-8685-2/16/$31.00 ©2016 IEEE

of XML with the existing database, but needs to use the existing SQL statements and XQuery to query XML data at the same time, which will not be saved in a specific format.

The native XML databases extract the data through XQuery or XPath using only data of XML, which have various types such as BaseX [18], Qizx [19], eXist [20] or MarkLogic Server [21]. The RDBMS must write a query twice, but native XML databases is available with a single query. In this paper, we use the native databases. The specific features of native XML database of each tool is shown in table 1.

TABLE I. THE COMPARISON OF NATIVE XML DATABASE

Property \ Name	BaseX	Qizx	eXist	MarkLogic Server
Native Language	JAVA	JAVA	JAVA	C++
XQuery 3.0	Supported	Supported	Partial	Partial
XQuery Update	Available	Available	Commercial	Commercial
XQuery Full Text	Supported	Supported	Commercial	Commercial
EXpath Extension	Available	No	No	No
EXqueryExtension	Available	No	Available	No
XSLT 2.0	Available	Available	Available	Available

Through using native XML databases, we choose the BaseX that can use all the features. We need to transform the BaseX based on metamodel for XML. In this paper, we implement an integrated data repository for metamodel through developing an additional programs based on the BaseX.

III. A DESIGN OF METAMODEL DATABASES

To implement the software visualization as shown figure 1, the developer is necessary to process three step: 1) the parser to generate the Abstract Syntax Tree (AST) from program code, 2) the data analyzer to store data using databases, and to extract the related data of the source code from AST, 3) the data visualizer to represent various graphs from the analyzed data.

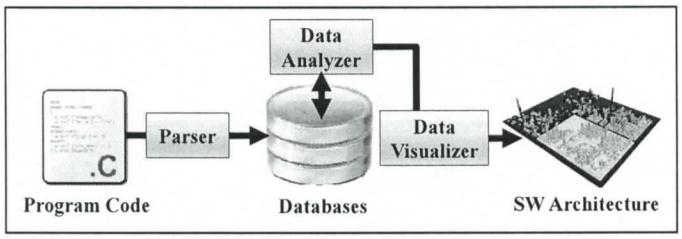

Fig. 1. The basic steps of SW visualization

In this paper, we propose the metamodel based integratation data repository to save the ASTM files and to use the data analyzer in the repository. The figure 2 show the structure of metamodel based integratation data repository.

To achieve the purpose of visualization, we need to convert to the abstract syntax tree from the program code. The xCodeParser can generate ASTM file from a program code. To store the data in an XML database, the ASTM file loader changes the extension name of the ASTM file to xml, and loads the data into the database using the BaseX library.

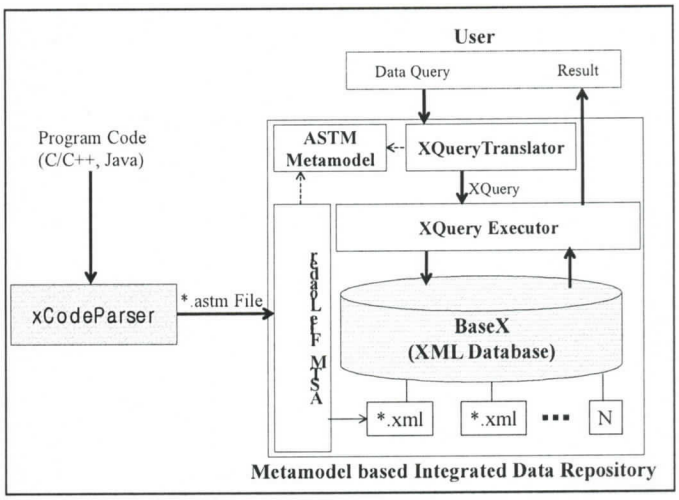

Fig. 2. The structure of metamodel based integratation data repository

This data is collected in database which may perform a search using XQuery. But, it is difficult to retrieve the metamodel because the XQuery searches the only data of XML. Thus, we propose a new query language named XQuery Translator that allows the user to easily view the data in the code. The XQuery Translator converts to XQuery from them. It does easily view the program code information with a simple command line to define the pattern of the metamodel of the ASTM. Finally, the XQuery executor performs the converted query statement to XML databases, and transfers the results to the user. Through this process, the user can easily process the desired data.

IV. CONCULSION

The software visualization is a method that extracts an architecture from program codes. In NIPA's Software Engineering Center, they have organized the open source based tool-chain using Source Navigator, Graphviz, SQLite, Jenkins, and etc. to execute the visualization. It is required to combine diverse tools such as parser, database and visualizer. To solve the problem, we proposed the parser based on the OMG's Abstract Syntax Tree Metamodel (ASTM), which is applied in the existing tool-chain for software visualization. In this paper, we propose the metamodel based XML databases to handle and process the generated ASTM files from the parser. We use XML databases because the basic structure of model and metamodel have XML although the model and metamodel have XML Metadata Interchange (XMI). Through visualizing software, we help to improve quality, and to reduce reuse & maintenance cost. Further research should be conducted, which is not dealt in this study on visualizer to represent the various graph.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2011601) and the Human Resource Training Program for Regional Innovation and Creativity through the Ministry of Education and National Research Foundation of Korea (NRF-2015H1C1A1035548).

REFERENCES

[1] R. Pressman, Software Engineering: A Practitioner's Approach, 7th ed., McGraw Hill, 2010.

[2] B. P. Lientz, E. B. Swanson, G. E. Tompkins, "Characteristics of application software maintenance," Communications of the ACM, Vol. 21, No. 6, pp. 466-471, June 1978.

[3] H. S. Son, R. Y. C. Kim, "Test case extraction process for validation of real time transactions in finaancial value added network for trusworthy customer's credit card," ASIA LIFE SCIENCES, Supplement 11, pp. 277-288, July 2015.

[4] H. S. Son, R. Y. C. Kim, "A Case Study on Metamodel of Cause-Effect Graph Based on Model Transformation for Mobile Software Testing," Advanced and Applied Convergence & Advanced Culture Technology, AACL 03, pp. 5-7, Nov. 2014.

[5] W. Y. Kim, H. S. Son, R. Y. C. Kim, "A Study on Test Case Generation Based on State Diagram in Modeling and Simulation Environment," Advanced Communication and Networking, CCIS 199, pp.298-305, Aug. 2011.

[6] J. M. Wing, "A Specifier's Introduction to Formal Methods," Computer, pp.8-24, Sept. 1990.

[7] I. Burnstein, A. Honyen, R. Grom, C.R. Carlson, "A Model to Assess Testing Process Maturity," Crosstalk: The Journal of Defence Software Engineering, pp.26-30, Nov. 1998.

[8] NIPA's Software Engineering Center, http://www.sw-eng.kr

[9] Source Navigator, http://sourcenav.sourceforge.net

[10] Graphviz, http://www.graphviz.org

[11] SQLite, http://www.sqlite.org

[12] Jenkins, http://jenkins-ci.org

[13] H. S. Son, Y. S. Kim, Y. B. Park, W. Y. Kim, R. Y. C. Kim, "Abstract Syntax Tree Metamodel for SW Visualization," International Conference on Convergence Technology (ICCT), Vol.4, No.1, pp. 157-158, July 2014.

[14] OMG's ASTM, "Architecture-driven Modernization: Abstract Syntax Tree Metamodel (ASTM) Version 1.0", OMG Document Number: formal/2011-01-05.

[15] CDO Model Repository, http://www.eclipse.org/cdo

[16] D. Steinberg, F. Budinsky, M. Paternostro, E. Merks, EMF: Eclipse Modeling Framework (2nd Edition), Addison-Wesley, 2009.

[17] OMG's XMI, "MOF 2.0/XMI Mapping version 2.1.1", OMG Document Number: formal/2007-12-01.

[18] BaseX, http://www.basex.org

[19] Qizx, http://www.axyana.com

[20] eXist, http://exist-db.org

[21] MarkLogic Server, http://www.marklogic.com