Advanced Science and Technology Letters
Vol.29 (ASEA 2013), pp.246-251
http://dx.doi.org/10.14257/astl.2013.29.52

Model-Driven Concept Extraction from Procedural Source

Napas Methakullawat and Yachai Limpiyakorn,

Department of Computer Engineering, Chulalongkorn University,
Bangkok 10330, Thailand
Napas.M@student.chula.ac.th, Yachai.L@chula.ac.th

Abstract. PL/SQL is a programming language used for developing business software, which is the connection between oriented database technology and application procedures used in Oracle Developer Tools. PL/SQL has been in existence for decades. Applications implemented with PL/SQL can be considered legacy systems, and they are potentially problematic for several reasons. It would be worth the corporate investment to rebuild legacy applications in the modern business computing environment. This paper presents an approach for concept extraction from procedural source, PL/SQL, as the initiative to enable program comprehension activity during legacy transformation. Model driven is applied as the architectural framework for the model transformation. The specification of PL/SQL source is recovered and transformed to the metamodel of class diagram as a Platform Independent Model, which can be augmented with particular technologies and used for later implementation of the new system.

Keywords: reengineering, model driven architecture, legacy transformation, metamodel.

1 Introduction

A legacy system is any application based on older technologies, languages, and hardware. Legacy systems are considered to be potentially problematic for several reasons. The systems may have vulnerabilities in data security or data management due to older operating systems or applications. Enhancement of the systems to satisfy the business needs would be difficult to handle or manage. Integration with newer systems may also be difficult because new software may use completely different technologies. In addition, there would be a shortage of developers who can write and maintain the archaic code. Hence, migration of legacy systems to the modern business computing environment is worth the corporate investment. A technique of reengineer-ing is required to rebuild legacy applications in a new technology or platform.

Model Driven Architecture (MDA) is an approach for using models in software development. Beyond the notion of Platform Independent Model (PIM) and Platform Specific Model (PSM), the two key concepts of MDA are *models* and *transformations*. The general pattern of model transformation is applying a set of transformation rules to the source model, resulting in the target model. This pattern can be repeatedly applied to successive models, each one playing the role of either a PIM or a PSM.

ISSN: 2287-1233 ASTL

Kulandaisamy et al. [1] proposed an aspect based reengineering approach for using MDA based techniques in reengineering procedural applications, but the research did not cover the recovery of abstractions from the traditional system. Masiero and Braga [2] proposed an approach for reengineering the outdated system by using software patterns to augment the system completely. Rather than using software patterns, metamodel of four main sections of Oracle Developer Tools is used in this work.

This paper presents an approach for legacy transformation using Model Driven Architecture. For program comprehension, the specification of the PL/SQL legacy code is recovered and represented with the metamodel as a PIM. The visualization tool can be used for rendering the UML class diagram from the resulting metamodel.

2 Background

2.1 Procedural Language/ Structured Query Language (PL/SQL) [3]

PL/SQL is a programming language and procedural extension language for SQL that was developed by Oracle Corporation in 1988. The language serves as the connection between oriented database technology and procedure applications. The syntax resembles that of Pascal or Ada. PL/SQL is available in Oracle Database and Oracle Developer Tools. The main sections of Oracle Developer Tools include:

1. Form Builder: create and design forms.
2. Report Builder: create reports.
3. Graphic Builder: create diagrams.
4. Designer: layout designs form and reports.

2.2 Model Driven Architecture (MDA) [4]

Model Driven Architecture is a standard of Object Management Group (OMG). *Model driven* describes an approach to software development whereby models are used as the primary source for documenting, analyzing, designing, constructing, deploying and maintaining a system. A *model* is a formal specification of the function, structure and behavior of a system within a given context. It is this formalism, which allows the model to be expressed in a format such as XML, in accordance with a well-defined schema (XMI), rather than a typical combination of drawings and text. MDA supports the Platform Independent Model (PIM), which is a software system model independent of a particular technology or platform. The model can explain the system by hiding the details essential for a specific platform. According to MDA, a PIM can be converted to a Platform Specific Model (PSM), which augments a platform independent viewpoint with details relating to the use of a specific platform. The Platform Specific Model is necessary for the actual implementation of the system.

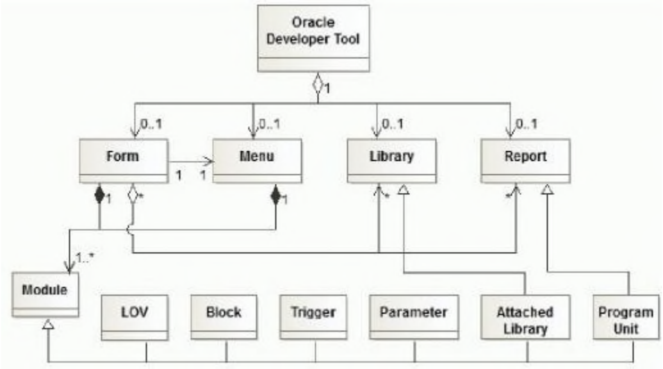


Fig. 1. Metamodel of four main sections of Oracle Developer Tools.

3 Research Methodology

Model-driven is applied as the architectural framework of this research. The archaic PL/SQL source of Oracle Developer Tools will be transformed to a PIM by the concept extraction method presented in this work. The *concept* is defined as the relationships between the problem domain and the main sections of PL/SQL legacy code, combined with the relationships between each main section and its parts. Fig. 1 illustrates the metamodel representing the conceptual design of a PL/SQL program via UML Class diagram.

The process of recovering the conceptual design of PL/SQL code consists of three main steps (Fig. 2) as described in the following subsections. Initially, the input of PL/SQL code needs be transformed into the working formats using the function ‘form2XML’ of Oracle Developer Tools. The PL/SQL forms, menus, and reports will be transformed to XML format, while libraries will be transformed to PLD format.

3.1 Element Extraction

The data extracted from the input XML files of forms, menus, and reports are those following the tags defined in Table1. The Document Object Model (DOM) [5] is used as a means of data extraction from XML files, while searching for the required words is used for data extraction from PLD format files.

3.2 Transformation Mappings

The target platform model, PIM, determines the nature of the mapping. A set of rules is defined for mapping the data elements extracted from the previous step in order to create the metamodel of those four main sections of the PL/SQL source. Example of the resulting XMI file is illustrated in Fig. 3. The UML visualization tool, Modelio, is used for rendering the class diagram (Fig. 4) associated with the XMI file (Fig. 3).

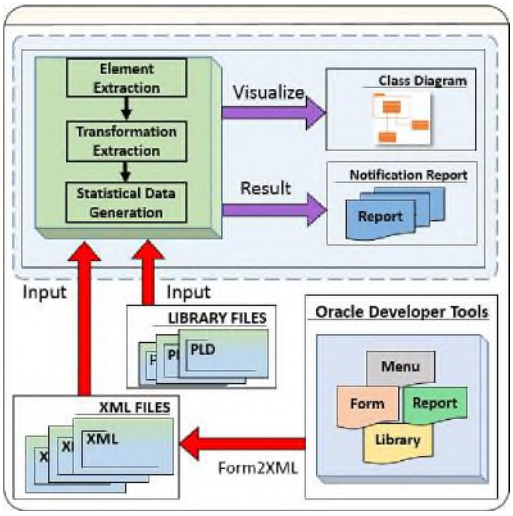
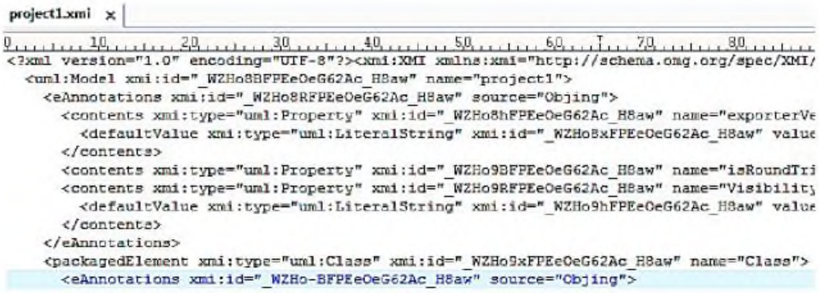


Fig. 2. Process of concept extraction from PL/SQL code.

Table 1. XML elements and attributes of form, menu, and report required for data extraction

XML Elements	Attributes
MenuModule	Name, MainMenu
MenuItem	Name, SubMenuName, MenuItemCode, CommandType, Label
FormModule	Name, ConsoleWindow, MenuModule, Title
ModuleParameter	Name, ParameterInitializeValue
LOV	Name, RecordGroupName
LOVColumnMapping	Name, ReturnItem, Title
RecordGroup	Name, RecordGroupType, RecordGroupQuery
RecordGroupColumn	Name, ColumnDataType
AttachedLibrary	Name, LibrarySource, LibraryLocation
Program Unit	Name, ProgramUnitType, ProgramUnitText
Block	Name
Trigger	Name, TriggerText
Canvas	Name, WindowName
Graphics	Name, GraphicsText



```
<?xml version="1.0" encoding="UTF-8"?><xmi:XMI xmlns:xmi="http://schema.omg.org/spec/XMI/
  <uml:Model xmi:id="_WZHo8BFPEeOeG62Ac_H8aw" name="project1">
    <eAnnotations xmi:id="_WZHo8RFPEeOeG62Ac_H8aw" source="Objing">
      <contents xmi:type="uml:Property" xmi:id="_WZHo8hFPEeOeG62Ac_H8aw" name="exporterVe
        <defaultValue xmi:type="uml:LiteralString" xmi:id="_WZHo8xFPEeOeG62Ac_H8aw" value
      </contents>
      <contents xmi:type="uml:Property" xmi:id="_WZHo9BFPEeOeG62Ac_H8aw" name="isRoundTri
      <contents xmi:type="uml:Property" xmi:id="_WZHo9RFPEeOeG62Ac_H8aw" name="Visibility
        <defaultValue xmi:type="uml:LiteralString" xmi:id="_WZHo9hFPEeOeG62Ac_H8aw" value
      </contents>
    </eAnnotations>
    <packagedElement xmi:type="uml:Class" xmi:id="_WZHo9xFPEeOeG62Ac_H8aw" name="Class">
      <eAnnotations xmi:id="_WZHo-BFPEeOeG62Ac_H8aw" source="Objing">
```

Fig. 3. Example of mapped XMI file.

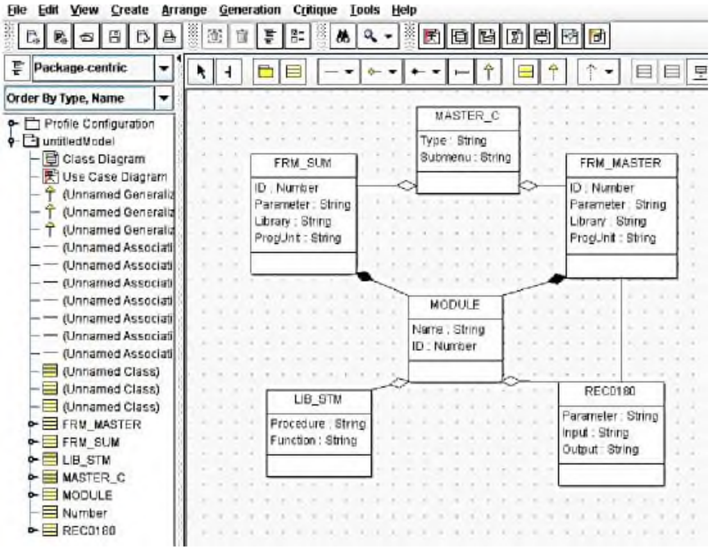


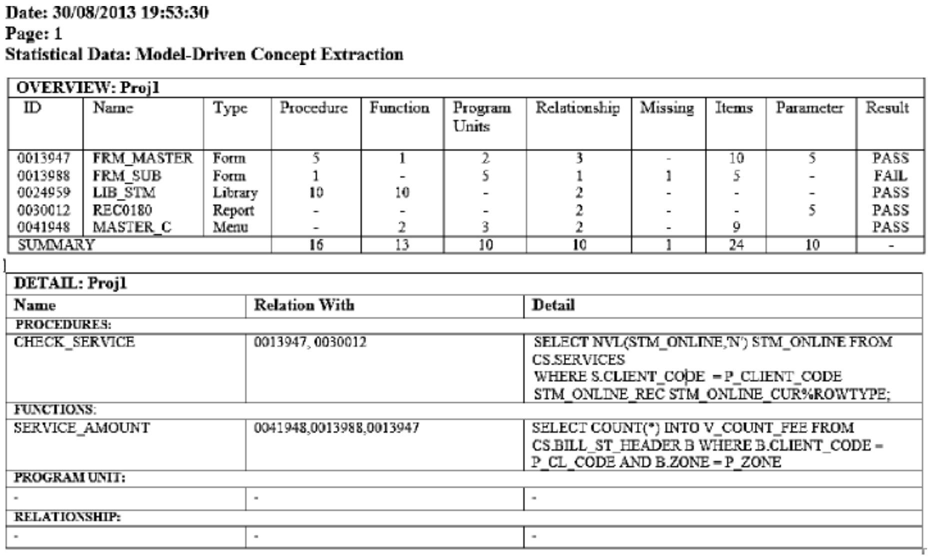
Fig. 4. Example Class diagram associated with the concept extracted from PL/SQL code.

3.3 Statistical Data Generation

To validate the result of concept extraction, the statistical report (Fig. 5) is generated to summarize and describe the details of each module of main sections, such as the constituent procedures, functions, relationships etc.

4 Conclusion

This paper presents an approach to extracting the conceptual design from the PL/SQL source code as the initiative of legacy transformation. The result is the UML class diagrams describing the relationships between each main section (forms, menu, report, and library) and its parts. To serve the inspection purpose, the statistical report is also generated to summarize and describe the details of each main section instance, for example, the number of constituent procedures, functions, relationships etc. Model Driven Architecture is applied for legacy transformation in this work. The first step when constructing an MDA-based application is to create a Platform Independent Model expressed via UML. Within a PIM, a target model is created containing only the data elements defined in the conceptual model. Based on this framework, the metamodel representing the specification of the PL/SQL source is obtained as the PIM, which can be further transformed to the PSM for the actual implementation of a modernized system.



Date: 30/08/2013 19:53:30
Page: 1
Statistical Data: Model-Driven Concept Extraction

OVERVIEW: Proj1										
ID	Name	Type	Procedure	Function	Program Units	Relationship	Missing	Items	Parameter	Result
0013947	FRM_MASTER	Form	5	1	2	3	-	10	5	PASS
0013988	FRM_SUB	Form	1	-	5	1	1	5	-	FAIL
0024959	LIB_STM	Library	10	10	-	2	-	-	-	PASS
0030012	REC0180	Report	-	-	-	2	-	-	5	PASS
0041948	MASTER_C	Menu	-	2	3	2	-	9		PASS
SUMMARY			16	13	10	10	1	24	10	-

DETAIL: Proj1		
Name	**Relation With**	**Detail**
PROCEDURES:		

CHECK_SERVICE	0013947, 0030012	SELECT NVL(STM_ONLINE,'N') STM_ONLINE FROM CS.SERVICES
		WHERE S.CLIENT_CODE = P_CLIENT_CODE
		STM_ONLINE_REC STM_ONLINE_CUR%ROWTYPE;

FUNCTIONS:		
SERVICE_AMOUNT	0041948,0013988,0013947	SELECT COUNT(*) INTO V_COUNT_FEE FROM CS.BILL_ST_HEADER B WHERE B.CLIENT_CODE = P_CL_CODE AND B.ZONE = P_ZONE
PROGRAM UNIT:		
-	-	-
RELATIONSHIP:		
-	-	-

Fig. 5. Example of statistical data report.

References

1. Kulandaisamy, P.D., Nagaraj, N.S., Thonse, S.: Representing Procedural Source in UML, In: 3rd Workshop on UML For Enterprise Applications: Model Driven Solutions For The Enterprise, California (2002)
2. Masiero, P.C., Braga, R.T.V.: Legacy Systems Reengineering Using Software Patterns, In: 19th International Conference of the Chilean Computer Science Society, pp. 160—169 , Talcal (1999)
3. FeuerStein, S., Probyl, B.,: Oracle PL/SQL Programming. O'Reilly Media, California (2009)
4. OMG, Object Management Group: Model Driven Architecture (MDA). http://www.omg.org/cgi-bin/doc?omg/03-06-01
5. World Wide Web Consortium (W3C) : DOM and XML DOM, from : http://www.w3c.org