Graph Theoretical Algorithms For Structural Comparison Of Java Source And Byte Code

Submitted By

Artem Garishin

FRANKFURT UNIVERSITY OF APPLIED SCIENCES

FB2: Faculty of Computer Science and Engineering

This thesis presented for the degree of
Master of Science
in the

High Integrity Systems

Research Supervisor: Prof. Dr. Alekseev Sergej

Co-Supervisor: Prof. Dr. Matthias Wagner

August 2014

Legal Declaration

I declare that this thesis document is completely my own work and all used references have been clearly cited. I have not submitted this assignment in the context of an examination to any other examination board or person.

Signature:

―――――――――――――――――――――――――――――――――――

Location, Date:

―――――――――――――――――――――――――――――――――――

Abstract

Java code is compare we need for ...

Why we need it

This paper also explains the existing ...

Acknowledgments

I would like to take this time to thank FH Frankfurt, University of Applied Sciences for all of the resources which they provided me in order to pursuing my master study in computer science and make this thesis possible.

I would like to express my sincere gratitude to Prof. Dr. Andreas Orth and Prof. Dr. Egbert Falkenberg for their patient guidance, encouragement and advice which they provided me throughout this thesis work. I have been extremely lucky to have a supervisor who cared so much about my work, and who responded to my questions and queries so promptly. This continuous support and very productive discussions were very helpful for me to achieve the actual task and expected results.

My sincere thanks also goes to all of my colleagues and friends who supported and helped me in any sense to complete my work.

Last but not the least, I am thankful to my family members who experienced all of the ups and downs of my studies, specially my elder brother for his continued support and encouragement towards completion of my master study.

Contents

List of Figures

List of Tables

Abbreviations

IEDAE	**I**nteractive **E**xploratory **D**ata **A**nalysis Environment
SQL	**S**tructural **Q**uery Language
API	**A**pplication **P**rogramming **I**nterface
CI	**C**ontinuous **I**ntegration
JDBC	**J**ava **D**ata **B**ase **C**onnectivity
MVC	**M**odel **V**iew **C**ontroller
HTML	**H**yper **T**ext **M**arkup **L**anguage
XML	**E**xtensible **M**arkup **L**anguage
JAXB	**J**ava **A**rchitecture for **XML** **B**inding
UML	**U**nified **M**odeling **L**anguage
URL	**U**niform **R**esource **L**ocator
HTTP	**H**yper **T**ext **T**ransfer **P**rotocol
SCM	**S**ource **C**ode **M**anagement
CVS	**C**oncurrent **V**ersion **S**ystem

Chapter. 1

Introduction

This is 1st chapter

1.1 Section

This is section heading

1.1.1 Sub Section

This is sub section

Chapter. 2

Current Scenario

This is second chapter

2.1 Background

Here is way that how to reference from bibliography. Giving references like this
[3].

2.2 Flow of the Project

Here is way to put figure and its reference in document.

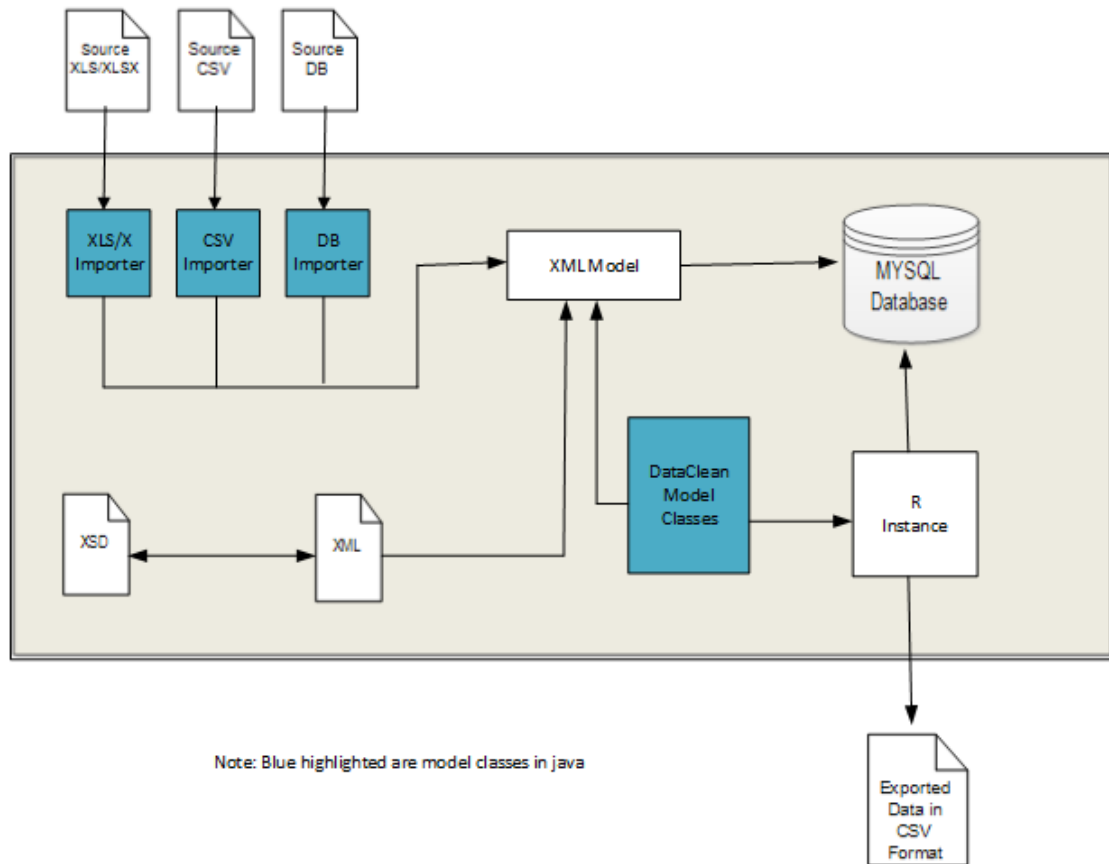Figure 2.1 shows the overall flow of the project.

Figure 2.1: Flow of the Project

Here is a way to design table and put reference of a table Table 2.1 gives visual overview of sequence in import process.

Items	Action
DataImportCSVXLS (doimport)	fileImport()
DataImportCSVXLS (fileimport)	newLoad().doLoad()
Load (Constructor)	ImportFactory.getFrameWork(tcm)
ImportFactory (getFrameWork)	csv =>return new ImportCSV(tcm) xls/xlsx =>return new ImportExcel(tcm) accdb/mdb =>return new ImporFromDB(tcm)
Load (Constructor)	Load has subclass instance of ImportFrameWork instance.getInsert()
ImportFrameWork (getInsert)	getInsert is responsible for the insert and returns the list of insert commands back generateLoadStatement()
ImportXXX,(generateLoad,Statement)	An instance of sub class, generates List and returns it back
ImportFramework,(getInsert)	A list will be returned
Load (Constructor)	List will be imported into database, iterate over the list super.execute(command)

Table 2.1: Classes and functions in import process

Here is way to itemize the points with numbers

1. point 1

2. point 2

Here is way to itemize the points with bullets

- point 1

- point 2

Here is way to put code inside listing

```java
public class Main(){

    public static void main(String[] args){


    }
}
```

Bibliography

[1] George H.L. Fletcher and Catharine M. Wyss, "Data Mapping as Search", Computer Science Department, School of Informatics, Indiana University, Bloomington, USA.

[2] Bogdan Alexe, Laura Chiticariu, Renee J. Miller, Wang-Chiew Tan, "Muse: Mapping Understanding and deSign by Example", University of California, Santa Cruz, University of Toronto.

[3] Dorian Pyle, "Data Preparation for Data Mining".

[4] Erich Teichmann, Eren Demir, Thierry Chaussalet, "Data preparation for clinical data mining to identify patients at risk of readmission", Department of Information Systems and Computing, School of Informatics, University of Westminster, London, W1W 6UW, UK.