



UNIVERSITY OF SOFIA "St. KLIMENT OHRIDSKI"
FACULTY OF MATHEMATICS AND INFORMATICS

Abstract

University of Sofia "St.Kliment Ohridski"
Department of Information Technologies
Faculty of Mathematics and Informatics

Thesis title:

Graphic User Environment for processing and extracting of meta data from a Java byte code.

Graduate student: Rositsa Panayotova Panayotova, specialty "Computer Science", Specialization "Information Systems", faculty number M-21332

Supervisor: assoc.prof Boyan Bonchev, PhD, Sofia University, Department of Information Technologies, Faculty of Mathematics and Informatics

Defense of master's thesis date: 27.07.2007

Key words: JDK, Swing, JVM, Java Bytecode, Annotations, Java EE 5, JEE Application, EJB, XML, XSLT

Annotation:

The rapid development of the software systems and technologies, imposed Java as a powerful mean, that can be utilized for implementing almost every architectural software solution. The variety of solutions could span from mobile devices such as cellural phones, through elaborate corporate productive systems capable of processing millions of clients' requests worldwide. The growth of the Java technologies, the unceasing development of new standards in every software businesses, the increase of abstraction level of software products - all of these indicators are consequences of the evolutoin of Java as a programming language and also as reliable platform for developing state of the art software.

In the newest version of the Java language - Java Development Kit (JDK) 1.5, Java Enterprise Edition (Java EE) 5, the Java software vendor has provided an opportunity to embed meta info directly into the source code of the applications (these embedded code snippets are called "annotations"). This meta data provides the software developers with means to link additional data to a class, field, method, method parameters or arbitrary

piece of coding. In previous releases of the Java language, this data was stored in additional Extensible Markup Language (XML) files.

The main concept behind this thesis comes from the circumstance that it is not always possible to use the built in Java methods in order to access the meta data of an application (Reflection API). To achieve this, one should load all classes (Class Loading) in the Java Virtual Machine (JVM), which directly or indirectly are references in the class, which contains the meta data. This implies restrictions on the various scenarios where the meta data model should be used. For example, a class in an application cannot be loaded from the virtual machine, if it depends on another class, which cannot be loaded for some unexpected reason.

This imposes the need of a system, which can extract the meta data from the Java Byte Code and structure it in convenient format, independently from the way in which data is loaded in the Java Virtual Machine. In the same time, this system must offer an alternative manner to work with it, besides its programming interface (API). In that way the users will not be forced to develop programming code in order to use it. Namely the implementation of graphical user interface which can be utilized from users to work with the system interactively. This graphical system will give an opportunity to choose the input files (*.class, *.jar, directories etc.), to use special filters to search for concrete annotations in the selected set of input files and also to save the results into independent format - XML.