

From C to JAVA

Farhana Kausar¹, Karthik G S²

¹Assistance Professor, Atria Institute of Technology, Visvesvaraya Technological University, Bangalore

²Student, Atria Institute of Technology, Visvesvaraya Technological University, Bangalore

Abstract - As per current business needs and rapid growth in Automation, it became necessary for industries to provide migration solutions to client as early and more efficiently as possible to cope up with competition in market. For this purpose a method has been proposed for constructing JAVA programs from C programs. IBM mainframe programs, which written in language such as C, is automatically translated into equivalent JAVA programs. Our method features the documentation of translation process. Not only the translation, but the correspondence between source C program and resulting JAVA program is produced as document. This helps in reducing the overhead of manually rewriting the entire C code into JAVA language.

Key Words: Migration, Modernization

1. INTRODUCTION

This paper proposes a method of migrating C programs to equivalent JAVA programs, which focuses on generating supplementary information to understand the resulting JAVA programs and the translation process. JAVA programs with high flexibility are generated through intermediate representation extracted from the given C codes. Also, the comments in the C programs, that hold useful information to understand the program like, how the programs should be used, the motive of each part of code, etc., are placed at the correct position of the resulting Java programs.

Migration is basically shifting from one place to other or changing from one form to another. But, if we relate this concept to programming, code migration would be changing code from one programming language to another. Thus, our paper presents the idea for changing the code from C to Java. Although IT industry is growing rapidly, there are few organizations those are stuck to mainframe systems. Mainframe system uses programming languages like C, COBOL, etc. These organizations are thinking to shift towards more widely used and flexible languages like JAVA. So, if they get a tool which would readily convert them the code from one language to another, then that would be very beneficial to them in terms of time and efforts as well.

An automation tool based on the proposed method is implemented in Java, which has successfully converted some C programs into working JAVA programs.

2. BASIC IDEA

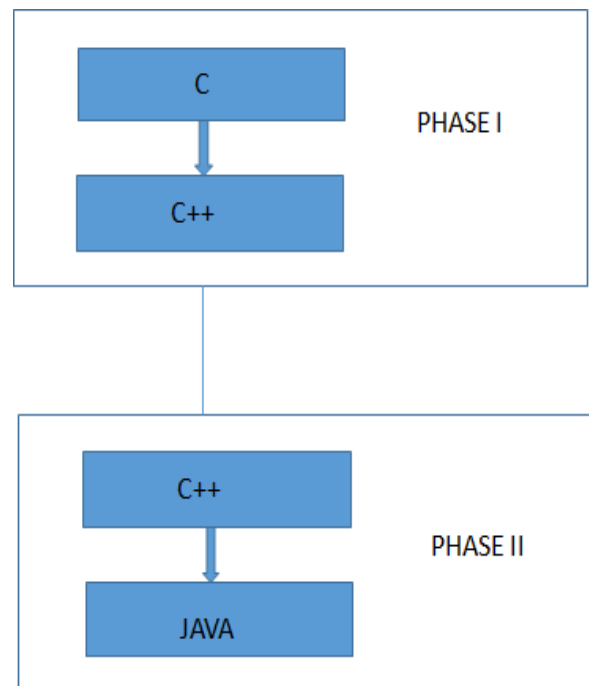


Fig -1: Translation Process

Thus, in order to get the code translated from C to Java, we have divided the approach into two phases. In the very first phase, program in C language will be converted into CPP language. This is because many of the syntaxes of C and CPP are similar, for example, the looping statements, the decision making statements etc. are similar in both C and CPP.

In second phase, the program in CPP language will be converted into JAVA. Converting CPP code into JAVA will be somewhat easier as features of both CPP and JAVA like classes, objects, inheritance etc. are similar. Thus, we can get the C code converted into JAVA.

The paper proposes approach to convert/translate the given source program from C to JAVA. But, if we try to convert code directly into java, then it might be difficult since, C is procedure oriented language and JAVA is Object Oriented language.

3. IMPLEMENTATION

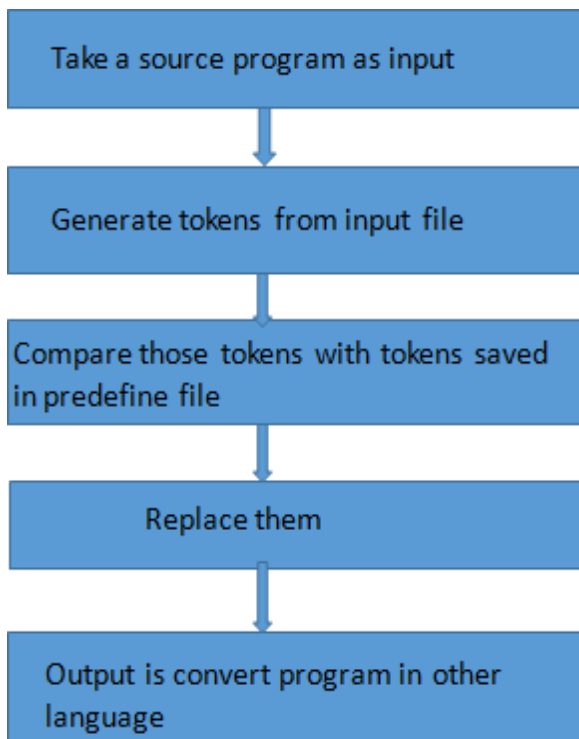


Fig 2. Flow of Implementation

The main goal of the approach is to exploit the power of data mining and genetic algorithms in aiding source code migration. The genetic algorithm is capable of learning complex mappings that exist between classes, methods and API usages between two languages. For example, the source code for outputting a string to the console screen in C is,

```
printf("Hello World!");
```

The corresponding statement in JAVA

```
is System.out.println("Hello World!");
```

For example, when printf word is found in any of the statements from input C program, then that word will be searched against its replacement in JAVA, which will be System.out.println. Once the corresponding keyword is found, then it will be replaced. Similarly, the entire code is searched line by line and the output is appended line by line into a paragraph. Thus, the resulting code is converted JAVA program.

4. CONCLUSIONS

A Genetic Algorithm based approach was proposed for the problem of translation of the given source program from one language to another. The technological changes take place so drastically that many

more migration projects are likely to come up and hence the development of an effective system for such migration goes important. The developed system has the problem of mapping from C to JAVA as C belongs to procedure oriented paradigm, and JAVA belongs to object oriented paradigm. More efforts need to be taken for migration between different paradigms (as from C to JAVA).

Also, the proposed system can be applied to applications that are freely available in order to improve the functionality of resultant outcomes. Along with this, there can be advancement in Computer Science field like improvement in neural networks and natural computations can be applied as it is mandatory that no efforts should be left in order to improve the efficiency of translation tasks.

REFERENCES

- [1] Daisuke Fujiwara and Nagisa Ishiura, "Reverse Engineering from Mainframe Assembly to C Codes in Legacy Migration", IIAI International Congress on Advance Applied Informatics, IEEE, 2016.
- [2] Andrea Fornaia and Emiliano Tramontana, "Is My Code Easy to Port? Using Taint Analysis to Evaluate and Assist Code Portability", International Conference on Enabling Technologies, IEEE, 2017.
- [3] S. Geetha and Dr. K. Iyankutti, "A Genetic Algorithm Based Source Code Mining Approach for Language Migration", International Journal on Recent and Innovation Trends in Computing and Communication, Vol:2, ISSN:2321-8169, August 2014.
- [4] Suvam Mukharjee and Tamal Chakrabarti, "Automatic Algorithm Specification to Source Code Translation", Indian Journal of Computer Science and Engineering (IJCSE), Vol:2, ISSN:0976-5166, Apr-May 2011.
- [5] Zhong, Hao, Thummalapenta, Suresh, Xie, Tao, Zhang, Lu, Wang, Qing, Mining API mapping for language migration, Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering, Volume 1, 2010.