

# **NPL Project – Remote Compiler**

## **Existing system:**

The typical method for compiling a program today is done by installing the language support software for a particular language in the workstation. This software consists of all the libraries and provides the complete execution environment for the language which also includes the compilation part in the execution of a program. Thus, the user writes a program in a language, compiles it and executes it using the language specific environment on his/her workstation to view the results.

## **Remote compiler:**

A remote compiler will enable a user to compile his/her program remotely (i.e. on another workstation, even one having a different hardware and software configuration, if necessary) and obtain the results of the same at his/her workstation. This project aims at enabling a client to send a program to the server machine and remotely invoke the application written to provide the compilation feature on the server. This will result in the compilers available on the server to be used to compile and execute the program sent by the client. The results of the execution will be sent back to the remote client. If the compilation is successful, then the outcome of the program will be sent. Else, the errors in the code will be returned. In order to accomplish the remote communication, we will be using the concepts of networking and Java's RMI (Remote Method Invocation). We also aim at setting up this communication between two workstations running on different operating systems. A block diagram of the project is given below for reference.

## **Purpose:**

- A user can develop a Solaris or Linux application from his/her workstation running Windows 7 or Mac OS X.
- User can test his/her application on different operating systems without even leaving the IDE.
- User can compile large applications on a fast multicore server rather than on his/her regular workstation.
- There could be few scenarios where in the compiler may not be there in the host machine.
- Support software of a newly developed language can be kept private using this technique, if necessary.
- Using a remote compiler improves the performance of the host machine to a certain extent as the time and resources used for the compilation can now be used for other purposes.

- Certain compilers may be available only on certain platforms and may not be supported on others. In such a scenario, the remote compilation technique can be used to make use of such compilers.
- It can be used for websites hosting tutorials for learning a programming language where a temporary compiler is provided for practicing the code.