1 Introduction

This document describes the high-level architecture and code layout of the NextGen prototype compiler at Rice University. The NextGen compiler was developed as an extension to the GJ compiler under special license from Sun Microsystems. This same compiler was extended independently by Sun Microsystems to form the JSR-14 prototype compiler, scheduled for inclusion in J2SE 1.5. In the process of developing NextGen, we have refactored the original GJ compiler substantially, and no attempt has been made to maintain compatibility with the JSR-14 source code. Nevertheless, the reader may find that some of the architectural features of NextGen described here are helpful when deciphering the JSR-14 code base (modulo class, package, and variable name changes).

Throughout this document, it is assumed that the reader is familiar with the GJ, NextGen, and MixGen language designs, as well as the published descriptions of how these languages can be compiled to the JVM so as to maintain compatibility with existing compiled binaries. Readers not already familiar with this material are referred to [10, 9, 3].

2 The NextGen CVS Repository

The NextGen source code is maintained under the javaplt CVS repository at Rice University, and is available to all members of Rice JavaPLT. You can access
this repository from any machine with access the the Rice CS network. To do so, set your CVSROOT to <user-name>@cs.rice.edu:/home/javaplt/.cvsroot. If you are accessing the repository remotely, set your CVS_RSH variable to ssh. Then perform the following steps to set up your environment for NextGen development:

1. In your home directory, create a new subdirectory called javaplt.

2. In your javaplt directory, enter the following commands:

   $ cvs checkout bin
   $ cvs checkout packages
   $ cvs checkout public_html
   $ cvs checkout java/<your-platform>

References


