Soot GUI Documentation

Patrice Pominville
patrice@sable.mcgill.ca

March 24, 2000

Soot has a simple GUI; this document explains its usage.

1 Synopsis

Usage:

`java soot.JMain`

The class `JMain` must be in your `CLASSPATH`. Note that under Unix-like systems, you should run `java` under native threads, i.e. `java -native soot.JMain`.

2 Description

The Soot GUI provides a simple interface to the Soot optimization system. It can be used by those who wish to experiment with Soot; it allows one to quickly and easily use Soot as a program optimizer.

This is a simple front-end to Soot, with a quick-and-dirty implementation. As such, it does not support all of the command line functionality. The power user will invariably find it limiting, and revert to the command line.

The GUI interface is briefly detailed in this document. Please refer to the command line documentation for more information about the exact meaning of these, and other, Soot options.

3 Look and Feel

The GUI is implemented in Java, using `Swing`, so it should run correctly on all platforms where Java 2 is installed.

In the online version, available at

`http://www.sable.mcgill.ca/soot/tutorial/soot-gui`

a snapshot of the GUI would appear here.

Class to Optimize This pane contains a text field where the user can enter the name of a class to process with Soot. The name must be that of a fully-qualified Java class. For example, a user would enter `java.lang.String` to process the class `String` found in the `java.lang` package.

Next to the text field is a check box; if checked, Soot is run in *application mode*.

Finally, two pull-down menus are provided to choose a default input representation for the classes to process (`jimple` or `class`), and an output representation for the processed classes.

Optimization Flags This pane provides two check boxes to determine if Soot is to perform scalar optimizations (-O) and/or whole program optimization (-W). There is also a pulldown menu to specify the final internal representation (`baf` or `grimp`) used before emitting bytecode.

Miscellaneous Options The user can provide a *SOOT_CLASSPATH*. The default value is the user's `CLASSPATH`. There are also check boxes which specify whether Soot is to be run verbosely, and whether Soot is to be run in debug mode.

Execution Output The output of the compilation is displayed in this text component. If an error occurs it will be reported here. Note that this text component does not auto-scroll.

Runtime Options This pane provides the *Sootify Now* button. Once the user has configured all the above option, clicking on it starts the compilation.

History

- March 24, 2000: Initial version.