



HSR

HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

FHO Fachhochschule Ostschweiz

Linticator Integrates Static Code-Analysis Tools in Eclipse CDT

Rapperswil, CH—IFS Institute for Software releases Linticator, a static code-analysis tool for Eclipse. Linticator integrates Gimpel Software's Flexe/PC-lint into Eclipse's C/C++ Development Tools (CDT). Static code-analysis helps the developer detect possible errors in the code during development. Linticator tightly integrates the command-line Flexe/PC-Lint tools into Eclipse and continuously analyzes the code to provide quick and accurate feedback to the developer. Linticator is available at CHF 500 per seat (plus 20% for updates per year). A free 30 days evaluation license is available on request at www.linticator.com.

A weakness of Lint is its lack of integration in development environments. The programmer needs to maintain the configuration, run Lint and analyze the results manually. Linticator hooks into Eclipse's build process and automatically runs Lint in the background whenever the code is compiled. The problems and warnings that Lint reports are continuously shown in the source code editor at the corresponding lines in the code. This way, the developer sees all the relevant messages for the files he is currently editing. Lint's configuration (the list of files that need to be analyzed, the include-path settings, etc.) are automatically retrieved from the Eclipse project settings and passed to Lint. Furthermore, Linticator shows the detailed message descriptions from the Lint documentation and thus making it easier to understand the analysis results.

An often occurring problem when using static code-analysis is the possibility of false-positive reports, that is, false alarms. For this purpose, Lint offers various ways to inhibit specific messages. Linticator can generate the appropriate inhibition directive at the push of a button and inserts it into the source code or into a separate configuration file. For a growing number of Lint message, Linticator contains so-called Quick-Fixes that automatically fix a problem in the source code on a key stroke instead of suppressing it.

About PC-lint/FlexeLint: The Lint tools, available as PC-Lint for Windows and FlexeLint for the remaining platforms, belong to the most commonly used static analysis tools and has been developed for over 25 years by Gimpel Software (www.gimpel.com). Lint analyzes the code for often-occurring programming mistakes like incorrect pointer dereferences, unsafe array access, potential memory leaks but also warns you about platform-dependent constructs or the compliance to standards like MISRA C/C++.

About the Institute for Software: The IFS Institute for Software is part of HSR University of Applied Sciences Rapperswil, Switzerland. IFS develops plug-ins for Eclipse CDT that make developing in C++ easier. Examples are CDT's C++ refactoring support, CUTE Unit Testing Support (www.cute-test.com), Includator (www.includator.com) and SConsolidator (www.sconsolidator.com)

Contact Person: Prof. Peter Sommerlad, Director of IFS, peter.sommerlad@hsr.ch



IFS

INSTITUTE FOR
SOFTWARE

Prof. Peter Sommerlad ■ peter.sommerlad@hsr.ch ■ T+41 55 222 4984
HSR Hochschule für Technik Rapperswil ■ IFS ■ Oberseestrasse 10 ■ CH-8640 Rapperswil
T+41 55 222 46 30 ■ F+41 55 222 46 29 ■ ifs@hsr.ch ■ www.ifs.hsr.ch