



Development of high-quality software is a tough task

Quality software coding is only getting harder: more lines of code, more possibilities for introducing defects, lack of sufficient time for manual code reviews and testing. Yet software quality has never been so important. Software is the very heartbeat of safety-critical applications. The cost of defects can be devastating to a company's brand and bottom line. Even worse, one small error can lead to system failure, injury or even death.

Background

ECLAIR is a powerful platform for the automatic analysis, verification, testing and transformation of C and C++ programs. Applications range from coding rule validation, to the proof of absence of run-time errors or generation of counterexamples, and to the specification of code matchers and rewriters based on both syntactic and semantic conditions. The extreme flexibility of ECLAIR allows it to be tailored for any software verification problem or software development process.

ECLAIR is suitable for light verification tasks that can be run right on the developer's desktop, as well as for tough semantics-based analyses to be run in batch mode. ECLAIR has gone through a rigorous design process and has been checked with extensive internal and industry-standard validation suites.

ECLAIR is available on most modern flavors of UNIX®, Linux, OS X® and Windows®, including Cygwin and MinGW, and can be used with just about any development environment.

Popular Supported Applications

Coding Standards Enforcement

ECLAIR provides support for automatically checking conformance with respect to a number of widely used coding standards, among which: MISRA C:2012, MISRA C:2004, and MISRA C++:2008.

Coding rules are enforced using very general and *accurate* checkers, which operate on precisely the same sequences of tokens and abstract syntax trees that are manipulated by the compiler. Coupled with the fact that ECLAIR always checks each rule in the appropriate context (at the token, declaration, translation unit, whole program or project levels), this ensures that for decidable rules the checkers are *exact* (neither false positives nor false negatives).

ECLAIR can be configured to produce all sorts of violation reports: for immediate or later browsing using the very powerful web-based browsing technology provided with ECLAIR, or within popular IDEs, or with any suitable editor; for the automatic insertion into issue-tracking systems or any other database; for the automatic production of compliance reports required to meet industrial standards and guidelines such as IEC 61508, ISO 26262, CENELEC EN 50128, DO-178B/C, IEC 60880, IEC 62304 and FDA norms.

Metrics

For assessing the complexity, readability and maintainability of software, ECLAIR provides comprehensive code metrics including the fundamental set defined by the *HIS Source Code Metrics* document and many of the requirements of the *SQALE Method* quality model. Metrics can be output in Excel format for further processing.

Proper Integration with the Toolchain

ECLAIR automatically interacts with the toolchain in order to obtain reliable results and save your time. Thanks to its ability to intercept all toolchain components (compilers, assemblers, librarians, linkers), it supports all sorts of makefile-based, script-based or hybrid build systems. Most popular C/C++ compilers and cross compilers are supported, including ARM®, CodeWarrior™, Cosmic Software, CrossWorks™, GCC, Green Hills®, HighTec, IAR™, Intel®, Keil Software®, MPLAB®, Microsoft®, QNX™, Renesas Electronics, SOFTUNE™, TASKING®, Texas Instruments™, Wind River®, and clang/LLVM.

ECLAIR Response to Increased Industry Demands

| Improved Feature | Benefit |
|---|--|
| Ease of integration in the existing development environment | The usual build process is run within an ECLAIR environment, with direct interaction with the toolchain |
| Usability | So-called “personalities” can be forgotten; interactive reports are intuitive and provide complete information |
| Setup time | Basic analysis of the code can start in minutes |
| Ease of configuration | A graphic configuration wizard can be used to carry out most common configuration tasks (rule selection, source selection, ...) |
| Consistency | Analysis reports are strictly tied to the running code, i.e., ECLAIR analyzes the real thing, not something else |
| Quality of violation messages | Extremely high (tracking of macro expansion and header file inclusion, view sources before and after preprocessing, ...) |
| Command line functionality | Everything can be done from the command line for automation and reproducibility |
| Extensibility | Checkers and metrics can be configured by the user; specific checkers for internal coding/style rules can be easily developed thanks to the supplier support |
| Deviation process management | ECLAIR provides advanced functionalities for deviations and legacy code management |
| Support | Issue-tracking system; email and telephone support; training |

Build Better, More Reliable Systems

That is why we created ECLAIR — to help developers build better, more reliable systems, and to help managers ensure code safety, portability and reliability.

Developed by renowned industry experts and based on more than 25 years of research into techniques for automated software analysis and verification, ECLAIR delivers high precision and high recall, so less time is wasted chasing nonexistent bugs, and there is less exposure from undetected bugs.

It is easy to make the ECLAIR decision. It is inexpensive to acquire, easy to integrate into the daily work life of developers, and produces vastly superior results that result in cost savings and improved productivity, as well as improved overall quality.

For More Information

BUGSENG srl
Parco Area delle Scienze 53/A
I-43124 Parma, Italy
Via Lenin 132/F
I-56017 San Giuliano Terme (PI), Italy
Email: info@bugseng.com
Web: <http://bugseng.com>


**no shortcuts,
no compromises,
no excuses:
software verification done right**