Static analysis tool C-STAT

Ensuring code quality through static analysis

Static analysis helps you to find potential issues in your code by doing an analysis on the source code level. The static code analysis tool C-STAT is completely integrated in the IAR Embedded Workbench IDE and provides an easy way to make sure your application complies with the coding standards defined by MISRA and hundreds of other checks derived from CWE and CERT.

Key features

- Analysis of C and C++ code
- Includes almost 700 checks in total, some comply with rules as defined by MISRA C:2012, MISRA C++:2008 and MISRA C:2004
- More than 250 checks mapping to issues covered by CWE
- Checks compliance with the coding standard CERT C for secure coding
- Fully integrated with the IAR Embedded Workbench IDE
- Comprehensive and detailed error information
- Fast execution
- Available for most IAR Embedded Workbench products

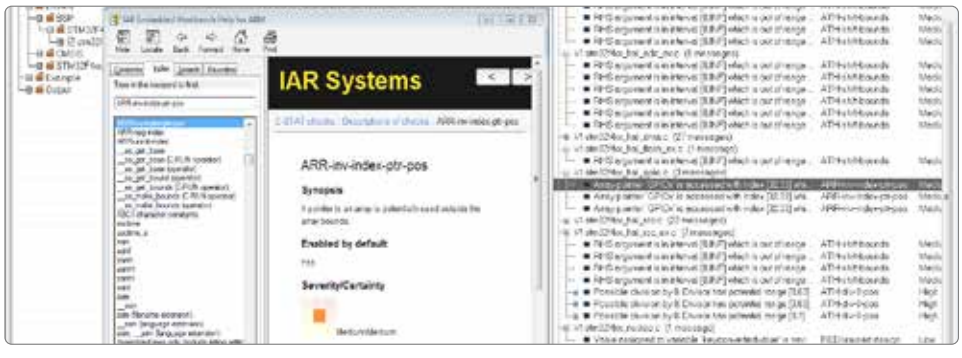
C-STAT checks code compliance with industry standards MISRA, CWE and CERT C/C++

C-STAT performs a number of security checks for compliance with the MISRA rulesets and rules as defined by the CERT C/C++ Secure Coding Standards as well as for a number of weaknesses as defined by CWE. To further simplify compliance tasks, C-STAT provides output that is consistent with the naming of weaknesses in CWE.

MISRA (the Motor Industry Software Reliability Association) rules has spread over the world and into different industry segments, and the ruleset is now the most widely used C subset in the embedded industry. MISRA-C:2012, which is the latest version, contains 143 rules and 16 so called directives. The rules are classified as mandatory, required or advisory and cover such areas as avoiding possible compiler differences like integer size, avoiding using functions and constructs that are prone to failure, limiting code complexity, and ensuring that the code is maintainable for example by using naming conventions and commenting.

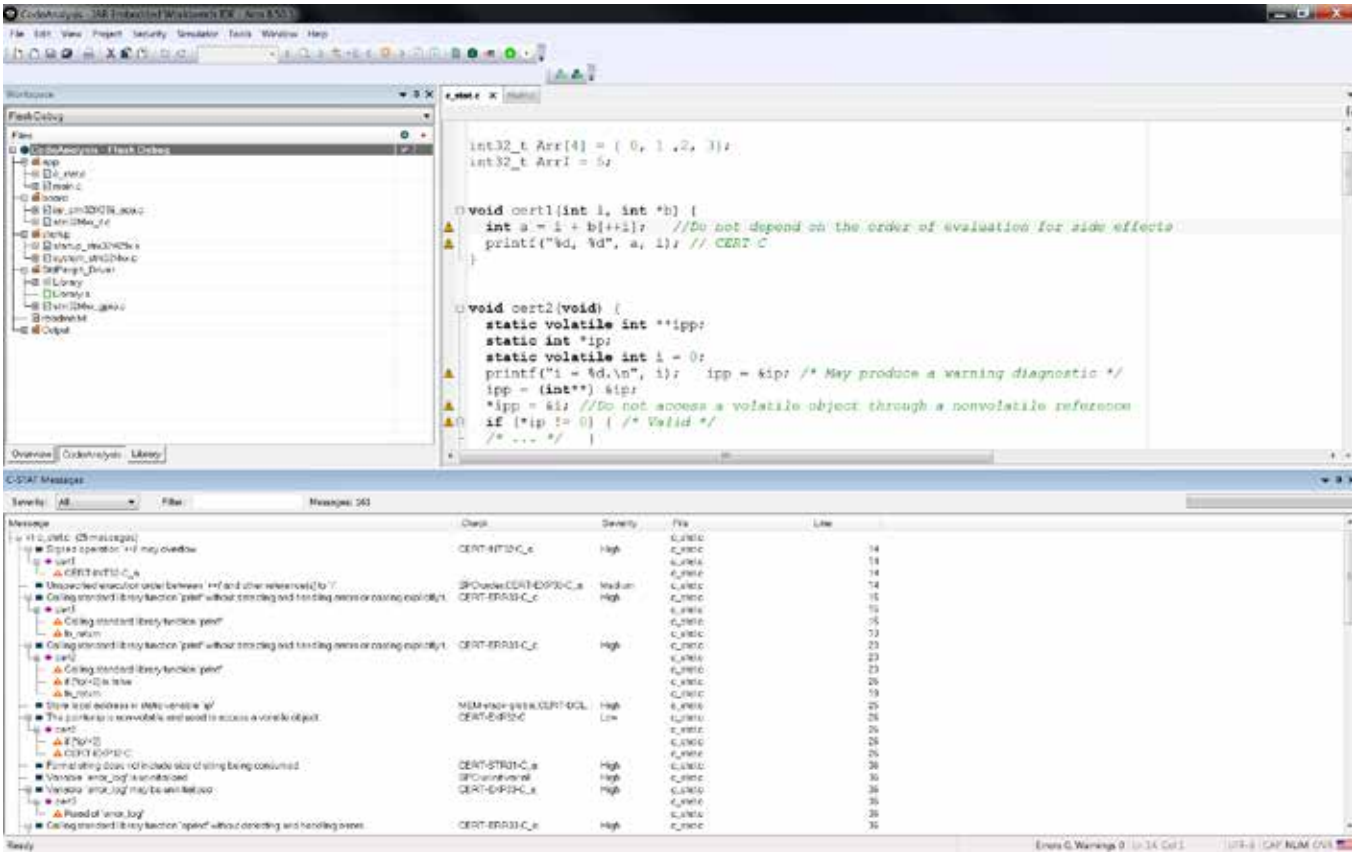
CWE (the Common Weakness Enumeration) is a community-developed dictionary with descriptions of the weakness and its potential consequences as well as potential mitigations, code samples, taxonomies and references. It is intended to help gain a better understanding and management of software weaknesses as well as to enable more effective selection and use of software security tools and services that can find these weaknesses.

CERT provides rulesets for secure coding in C as well as in C++. Each guideline consists of a title, a description, a non-compliant code example and examples of compliant solutions. The standards include guidelines for avoiding coding and implementation errors, as well as low-level design errors. The aim of the standards is to eliminate insecure coding practices and undefined behaviors that can lead to exploitable vulnerabilities.



Fully integrated into the IDE

C-STAT is fully integrated into the IDE and is as simple to use as the regular build tools. No need for complex tool setup and no struggle with language support and general build issues. The rules in the different standards overlap and complement each other. No coding standard includes all listings in CWE since not all of them are present in one coding language. Because of their mutually supportive roles it is wise, or even necessary, to consult all these instances to make sure that the software is safe and secure. Regardless of which of the rule-sets you are working with, C-STAT will check that your code is compliant and all checks in C-STAT are thoroughly documented with references to the corresponding entries in CWE and in the MISRA and CERT standards. You can select to check your code against rulesets as well as against individual rules.



IAR Embedded Workbench

IAR Embedded Workbench is a complete C/C++ development toolchain for embedded applications. The tool-chain offers leading code quality, outstanding optimizations for size and speed, as well as extensive debug functionality with a fully integrated debugger with simulator and hardware debugging support. C-STAT is fully integrated with the IAR Embedded Workbench IDE, which helps developers to ensure their code is safe and of high quality at an early stage, which also aids companies to shorten their time to market as impact of errors further down the line might be very time consuming and expensive.