PARASOFT

C++test™ 9.5

Release Notes

Parasoft C/C++test, the industry's most comprehensive development testing solution, automates a broad range of best practices proven to improve software development team productivity and software quality for C and C++.

C/C++test facilitates:

- Static code analysis, data flow analysis, and metrics analysis
- Peer code review process automation, preparation, notification, and tracking
- Unit test creation, execution, optimization, and maintenance
- Runtime error detection to find memory access errors, leaks, corruptions, and more

This provides teams a practical way to prevent, expose, and correct errors to ensure that their C and C++ code functions as expected. To promote rapid remediation, each problem detected is prioritized based on configurable severity assignments, automatically assigned to the developer who wrote the code, and distributed to his or her IDE with direct links to the problematic code and a description of how to fix it.

The 9.5 release introduces the following improvements:

Extended Environment Support

C/C++test includes support for Microsoft Visual Studio 2013 IDE

Extended Compiler Support*

C/C++test includes support for the following Windows-hosted compilers:

- IAR Compiler for ARM v. 6.6x
- Microsoft Visual C++ 2013
- Renesas M16C/R8C C Compiler 5.4x

New Static Analysis Rules

C++test includes new rules for additional MISRA C:2012 support

Rule ID	Description
MISRA2012-RULE-17_3	A function shall not be declared implicitly
MISRA2012-RULE-17_5	The function argument corresponding to a parameter declared to have an array type shall have an appropriate number of elements
MISRA2012-RULE-17_8	A function parameter should not be modified
MISRA2012-RULE-18_5	Declarations should contain no more than two levels of pointer nesting

Rule ID	Description
MISRA2012-RULE-20_4_a	A macro shall not be defined with the same name as a keyword
MISRA2012-RULE-20_4_b	A macro shall not be defined with the same name as a keyword
MISRA2012-RULE-20_7	Expressions resulting from the expansion of macro parameters shall be enclosed in parentheses
MISRA2012-RULE-20_9_a	All identifiers used in the controlling expression of #if or #elif preprocessing directives shall be #define'd before evaluation
MISRA2012-RULE-20_9_b	All identifiers used in the controlling expression of #if or #elif preprocessing directives shall be #define'd before evaluation
MISRA2012-RULE-20_12	A macro parameter used as an operand to the # or ## operators, which is itself subject to further macro replacement, shall only be used as an operand to these operators

The following rules were added to extend support for coding conventions:

Rule ID	Description
CODSTA-132	A function parameter should not be modified
CODSTA-133_a	A macro shall not be defined with the same name as a keyword in C90
CODSTA-133_b	A macro shall not be defined with the same name as a keyword in C99
CODSTA-134	The function argument corresponding to a parameter declared to have an array type shall have an appropriate number of elements

The following rules were added to improve analysis of preprocessor operation:

Rule ID	Description
PREPROC-17	A macro parameter used as an operand to the # or ## operators, which is itself subject to further macro replacement, shall only be used as an operand to these operators
PREPROC-18	The input/output library stdio.h shall not be included

PRs/FRs Resolved

Reported PR/FR ID	Description
105111	Added project importer for Renesas HEW and M16C projects
106975	Added support for Visual Studio 2013 (VS2013) IDE integration
108128	Added new rules to extend MISRA C:2012 support
108050	Added support for Visual C++ 2013 toolchain.
108058	Added support for Renesas M16C and R8C compiler V.5.4x
108551	Added static analysis support for IAR C/C++ Compiler for ARM v. 6.6x (iccarm_6_6)
104272, 106713, 107274	Updated static analysis rules
108036	Updated Fujitsu Softune branding to Spansion Softune
108245	Improved support for C++11 constexpr
103166	Improved generated code for VxWorks 6.8 and greater

Reported PR/FR ID	Description
99322, 106024, 107308, 107353, 107395, 107399, 107409, 107443, 107544, 107555, 107981, 108087, 108182, 108268, 108334, 108335	Fixed issues related to static analysis rules
102104, 103781	Fixed issues related to RVCT compiler
104402, 107848, 107851, 107852, 108165, 108413, 108462	Fixed issues related to RM16C compiler
107897, 108358, 108912	Fixed issues related to documentation
105693	Fixed issue related to IAR compilers
105927, 106648	Fixed issues related to auto-generated stubs
107420	Fixed issue related to GHS compilers
107730	Fixed issue related to ICC430 compilers
108184	Fixed issue related to TI compilers
108908	Fixed issue related to ICCARM compiler
107618	Fixed issue related to CL compiler
107661	Fixed issue related to VS2008 integration
108077, 108201, 108649	Fixed issue related to data flow analysis
108088	Fixed issued related to static analysis suppressions

* See C/C++test User's Guide for details on supported features and limitations.