

03 APRIL 2020



Change Documentation for

Testwell CTC++

Version 9.1

Features and Changes

Starting with this version 9.1 of Testwell CTC++, all components are synchronized in their version number.

New Help System

Testwell CTC++ is delivered with a new help, based on HTML instead of the PDF-based User Guide. In the \Doc folder of the installation, a file called "Testwell CTC++ Help" can be found to start this help.

New mechanism for build integration on Windows

A new tool called `ctclaunch` is released that can be used to integrate `ctc` easily in build systems like CMake or MSBuild. The purpose of `ctclaunch` is in particular the usage on build machines when `ctcwrap` cannot be used directly, and `ctcwrap` with option "-hard" is not acceptable in its consequences. Please see the Testwell CTC++ Help for a detailed description.

Instrumentation of "missing defaults" in switch-case-statements

To obtain full coverage, it is now also required to test the default case even if there is no explicit default statement given, comparable to a "missing else" branch in an if statement. Note that this may decrease coverage for existing projects after an upgrade.

Configuration dump for ctc

During instrumentation with `ctc`, the values of configuration parameters are set based on different configuration files (`ctc.ini`) and command line options. With the new `ctc` option "-dump-config", all these settings are written to a log file in a temp folder:

Windows ¹ :	%TEMP%\ctclog
Linux and macOS:	/tmp/ctclog

The filename starts with "ctclog" and contains a date and a process ID. This dump is useful to analyze cases with unexpected behavior.

Extended diagnosis for ctc with option "-V"

For the search of configuration files, every search position is given now independently from the concrete INI file, if available. If a license is available, detailed information is given about the location of the license, the owner of the license, and the expiration date. If no license is available, the same license errors are shown as before.

Rounding behavior for coverage ratios (TER)

Previously, different rounding rules were used to calculate the Test Effectiveness Ratio as a percentage, depending on the range of the TER. Now every ratio is rounded down to an integer. 100% coverage is still reached only if all counting points have been tested.

¹ using the environment variable %TEMP%

HTML report improvements

For the code view in HTML report (Execution Profile), the table header sticks now at the top when the page is scrolled down. In overview tables, the wrapping behavior is improved. The fonts used in default CSS are slightly modernized: Verdana is no longer used and Consolas is used for source code instead of Courier New.

Improved text report comparator `ctcdiff`

Functions are now compared even if their source file changed, as long as the function itself did not change.

Warning for misplaced line-ending characters

If source code files contain untypical line-end characters, for example the carriage return character in a Linux environment, a warning is now issued by `ctc2html` for this file. The HTML report is nevertheless generated. In some cases, it may show a corrupted source code view for these files, but all coverage figures are yet valid.

Starting of a browser with `ctc2html`

If `ctc2html` has to start the browser, it does no longer wait until the browser is closed.

Attribute `__declspec` for `armcc`

This compiler specific attribute is now also accepted for `armcc`.

Namespace support

Functions defined in namespaces are now identified by Testwell CTC++ with the name of the namespace included. Example: A (member) function `foo` in a class `A` in a namespace `n` was identified as `A::foo` before in the symbolfile and in all reports. Now it is identified as `n::A::foo`.

Note: If namespaces are used, the symbolfile `MON.sym` of a project shall be renewed after an upgrade to version 9.1, making a full rebuild of the instrumented program necessary.

Withdrawal of JSON report

The generation of a coverage report in JSON format by `ctcpost` is no longer provided. Customers using this format are kindly asked to contact Verifysoft before an upgrade.

Withdrawal of `ctc2excel`

The converter `ctc2excel` producing CSV files from the Execution Profile Listing is no longer part of the Testwell CTC++ package. Customers using this format are kindly asked to contact Verifysoft before an upgrade.

License and tool information in `ctc.ini`

The configuration file of Testwell CTC++ does by default no longer contain a restricted block with information about the tool version and the way of licensing.

Bug Fixes

Process hung of ctc

For C++ code having identically named types referring differently in different namespaces, it was possible that `ctc` got in an endless loop.

Rounding on directory level in HTML report

Test Effectiveness Ratios on directory level are now always rounded down, consistently to other reports. An issue showing 100% on directory level even if the real TER was below 100% is therefore fixed.

Fixes around SOURCE_IDENTIFICATION = basename

This setting caused crashes of `ctc` before.

Casts to user-defined types

Casting a decision like `x = (uint8)(a && b);` could prevent the instrumentation of this decision.

Uncompilable Code

Instrumenting certain while-loops directly followed by a ternary-? operator, `ctc` caused compilation errors.

Fixed inconsistency for decision coverage

Assignments containing a multicondition like `x = a && b;` were instrumented and taken into account if the instrumentation mode was multicondition and the report was reduced to decision coverage. This is now synchronized with the behavior if the instrumentation mode is decision coverage, giving the following result: This kind of assignment is not taken into account for decision coverage, but for multicondition coverage, MC/DC and condition coverage.

Source code comments in HTML report

After comments containing strings and multiple lines, the line coverage highlighting was corrupted.

Decltype in a typedef

The `decltype` specifier in a typedef are now processed correctly by `ctc`.

XML report with too many conditions

When generating an XML report with `ctcpost` for MC/DC coverage, an error could occur for decisions with too many conditions. This is fixed.

Warning for too complex multiconditions

If a multicondition would lead to more than 500 evaluation alternatives, `ctc` issues a warning during instrumentation and instruments the multicondition as a decision only. This warning was previously shown only as an information (appearing only if `WARNING_LEVEL` was set to "info"). Now it is shown also for the default setting `WARNING_LEVEL = warn`.