What's New in Cantata 8.0?

Cantata 8.0, available from May 2018, is a major new release providing significant new functionality as well an improved user interface.

This document outlines the most important changes in Cantata version 8.0.

Introduction

Cantata 8.0, available from May 2018, is a major new release providing significant new functionality. This version automates test maintenance, makes it easier to test across multiple embedded targets, and provides greater flexibility when instrumenting code.

Version 8.0 also continues the journey started in version 7.2 to provide a user interface that is accessible and intuitive for novice users while still retaining the advanced functionality that existing users have come to expect.

Cantata 8.0 also contains many other enhancements and fixes. The full set of changes are documented in the Release Notes which track all changes in Cantata since version 4.1. The most important changes are highlighted in the sections below.

Code Change Analysis

Testing earlier in the software development lifecycle improves quality & reduces code rework later-on. However, even well-planned projects will need some changes to source code after initial testing. Unit & integration tests are very dependent on the source code structure, so can be brittle (i.e. not even build) when code is changed, and fixing them can be time consuming. The new Cantata Code Change Analysis feature identifies code changes that will affect tests and helps automate test maintenance. Detailed guidance is provided on resolving these issues and suggestions can be selected triggering automatic test refactoring for many types of code change.

Thorough analysis of code changes and test dependencies identify:

- Changes to source code that affect existing test scripts
- Which test scripts are affected by code changes
- Code changes that may affect the code coverage previously achieved by the existing tests

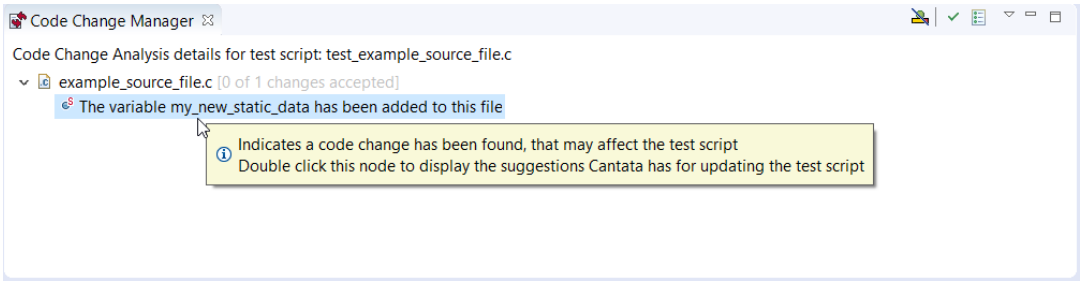
The Code Change Analysis results are clearly identified in a new dedicated Code Change Manager. Information is retained when code was last tested and compared to the current code.

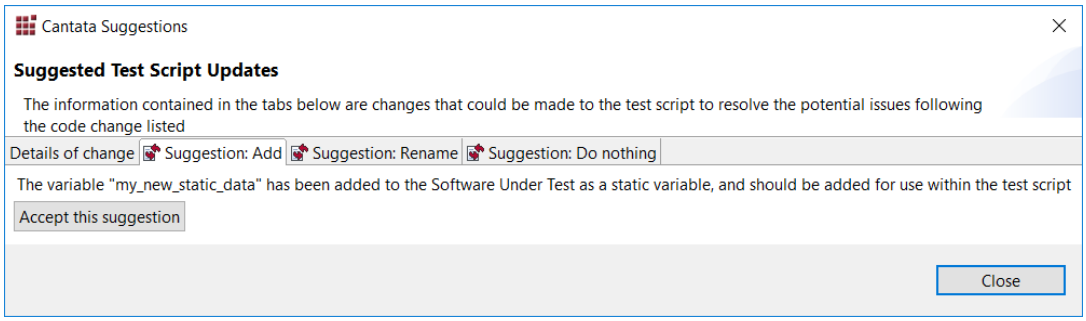
Changes detected in source code which affect existing test scripts include:

Source Files	• added / removed / renamed
Header Files	• added / removed / renamed
Functions & Methods	• added / removed • signature changed (scope, return type, name & parameters) • Call changed - which may affect stubs / wrappers • body added/removed • body changed - which could affect the code coverage
Variables	• Added / removed • Changed (scope, type & name etc)

Guided choices are provided for automatic updates to re-synchronise tests with the changed code:

- Details of the code changes affecting existing test scripts are provided for each source file
- Suggestions to update the test are provided with guidance on the effects of accepting the suggestion
- Test scripts are automatically refactored throughout to implement the suggested test update

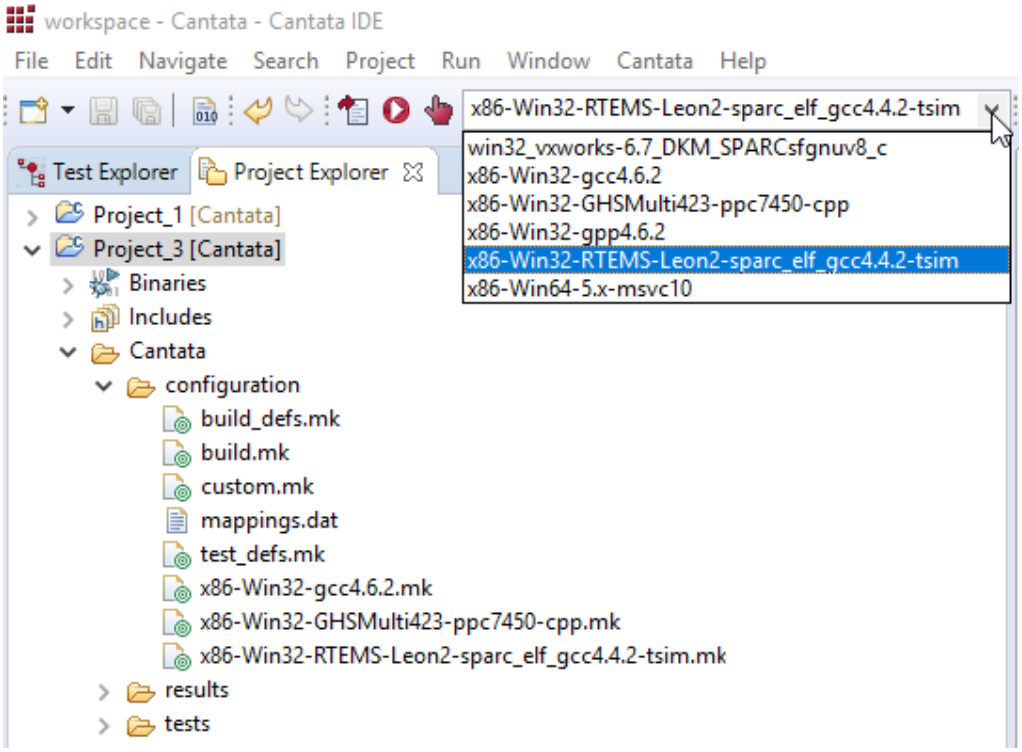




Target Deployment Switching

The new deployment switching feature in Cantata version 8.0 makes it easier to reuse the same tests for multiple targets and toolchains. This is particularly useful when code is reused across multiple environments for different product variants.

Separate environment specific make files are created for each target deployment. Switching between them is as easy as selecting a deployment from a dropdown menu on the main toolbar.



Reusing the same tests with different deployments makes testing across multiple environments faster, and avoids replicating whole projects just for testing. Test maintenance effort is reduced by maintaining only one set of tests. The risks associated with managing multiple copies of projects are eliminated.

The Cantata test results (.ctr) file records the specific deployment with which tests were executed, so that separate test certification ready evidence can be provided for each environment.

In addition to the built-in target deployments inside Cantata for user customisation, as new deployments are completed between releases, these are now made available to download from QA Systems.

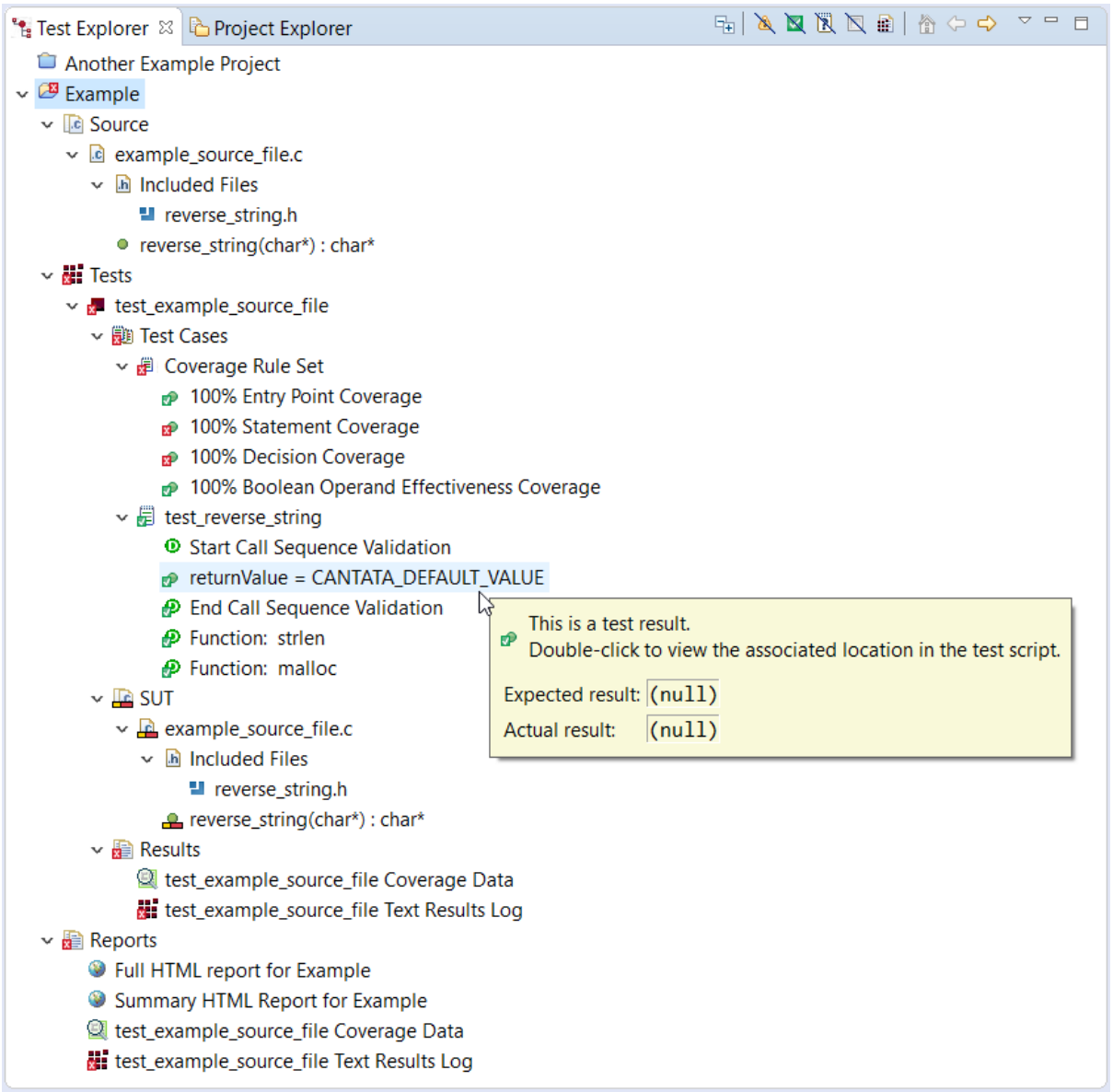
Test Explorer

Cantata 8.0 introduces a new Test Explorer which makes it easy to see code, tests, results and reports in one place. It is a test centric view rather than a project focused one. This is clearer and easier to navigate, showing only the information relevant to testing with Cantata.

Test Explorer provides a context sensitive control hub for all Cantata functions and views. It combines and enhances the control capabilities of Project Explorer, with the diagnostics previously shown in Coverage Results and Test Results Explorers.

New capabilities available in Test Results Explorer include:

- Direct creation and management of Cantata tests with context menus
- Filters to focus on relevant source files, test status and results
- Tool-tips display extra information and relevant results for each tree node

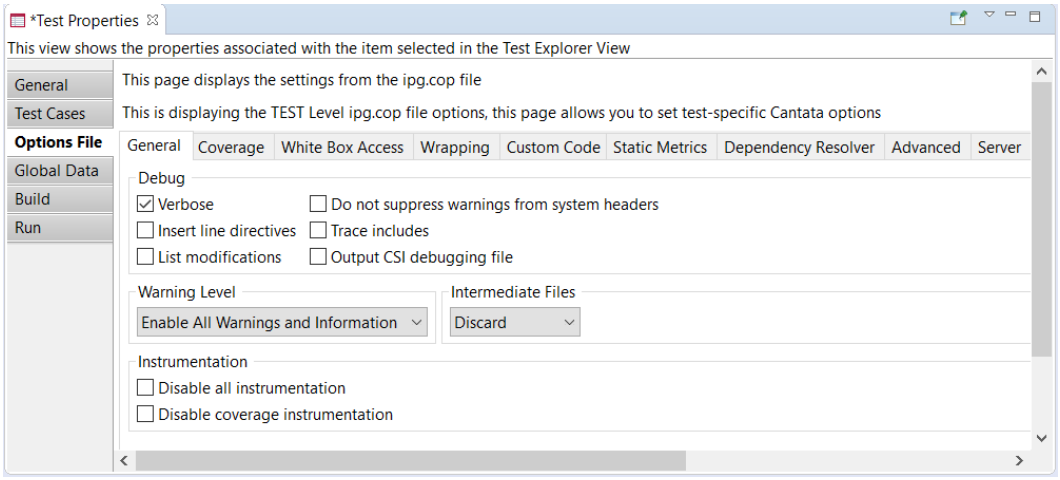


Test Properties

The new Test Properties view provides a single point of control for the test environment. It displays and can set the properties for any test item selected in the Test Explorer. This view enhances the control previously shown in Test Script Manager.

Test Properties allows the user to manage:

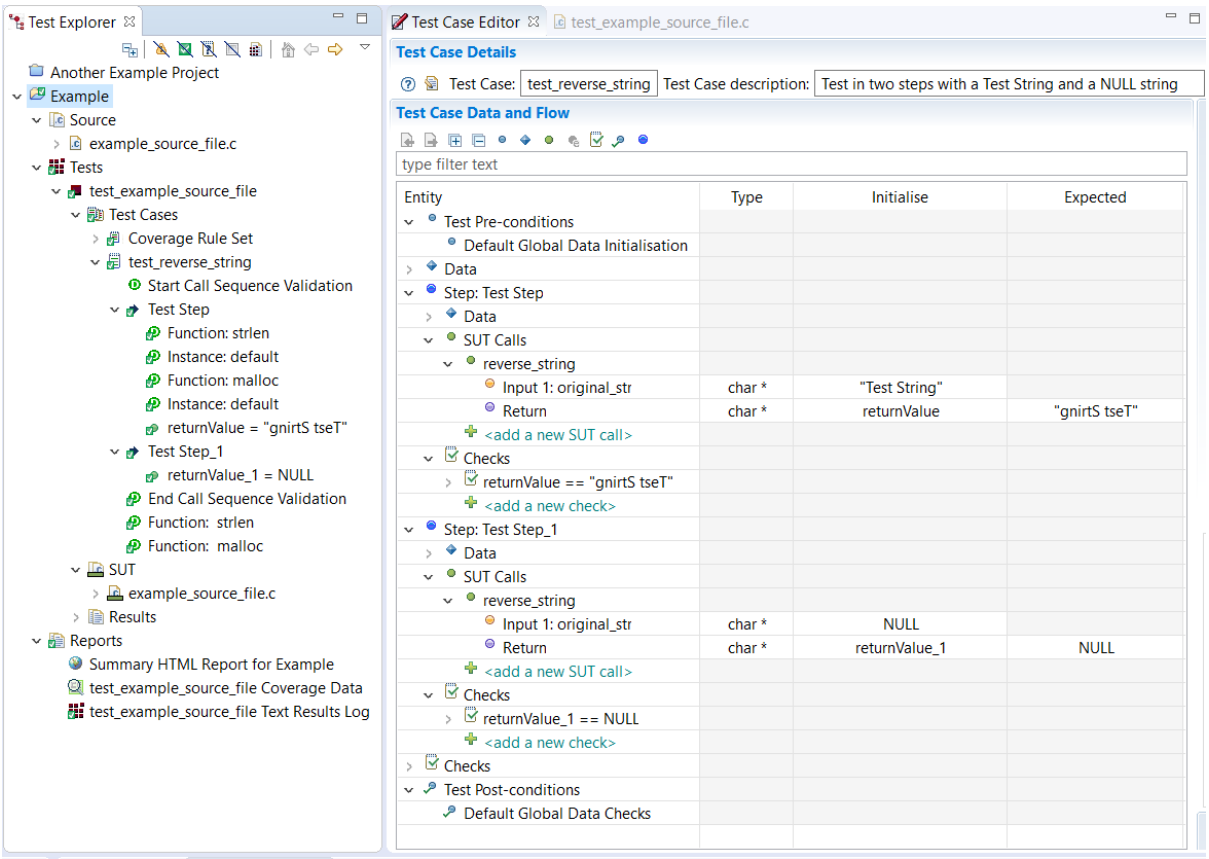
- Project and Test script level options (ipg.cop files)
- Global Data settings for testing
- Test Case function name and comment description
- Control of test build settings (make file, include path, code coverage, and static memory)
- Control of test Run Configurations (including enabling / disabling test case execution)



Test Case Editor

The Test Case Editor has been improved, new Test Steps add a deeper level of control within a test case. Test cases are separated into one or more Test Steps, each step can have its own Data setup, calls to the software under test (SUT) and checks. By default, each new test case is generated with a single test step and further test steps can be added. The order of test steps can be changed using drag & drop.

These improvements to Test Case Editor simplify test scenario creation, as multiple steps all share initial test set-up and final tear-down. Scenarios which involve retaining the state of the SUT between test steps, make it easier to verify object oriented classes or inter-dependent functions.



Other Test Case Editor improvements include: layout improvements to the Call Interface Control panel, and suggestions for accessible global variables which could be checked.

Custom Code Injection

It is now possible to inject extra code with instrumentation anywhere in the software under test for testing purposes.

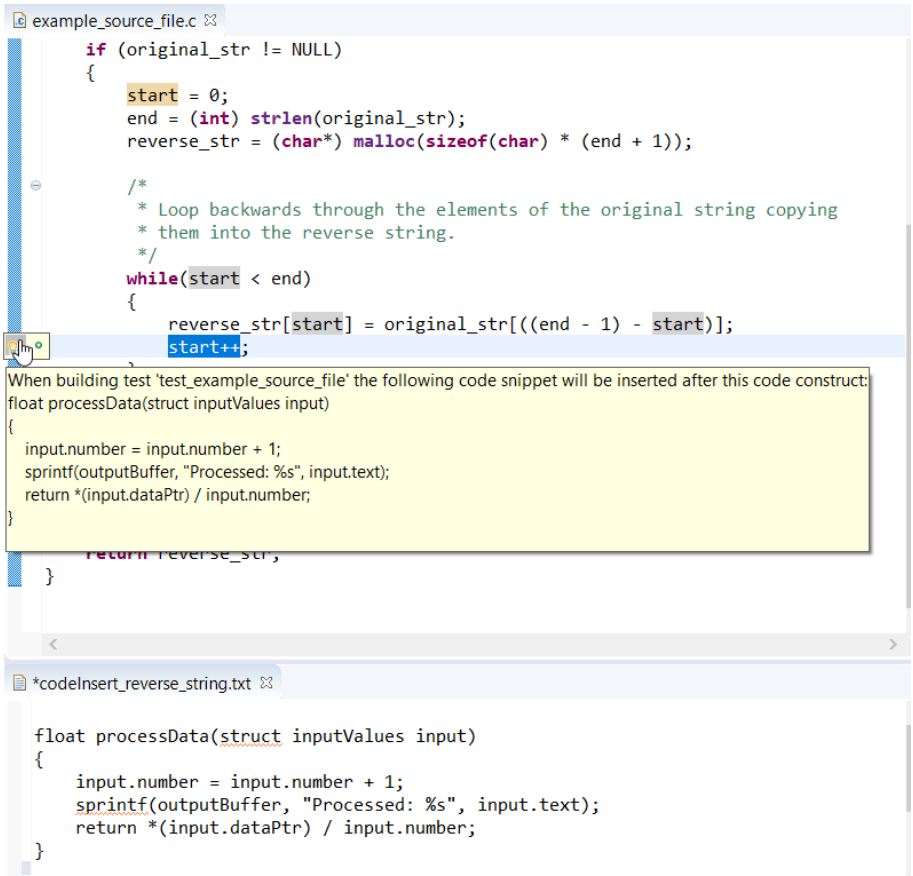
This avoids modifying the shipped production code using conditional compilation, which can cause:

- Potential unintended behaviour in production environment
- Code bloat from residual test code
- Reduced clarity of code for maintainability

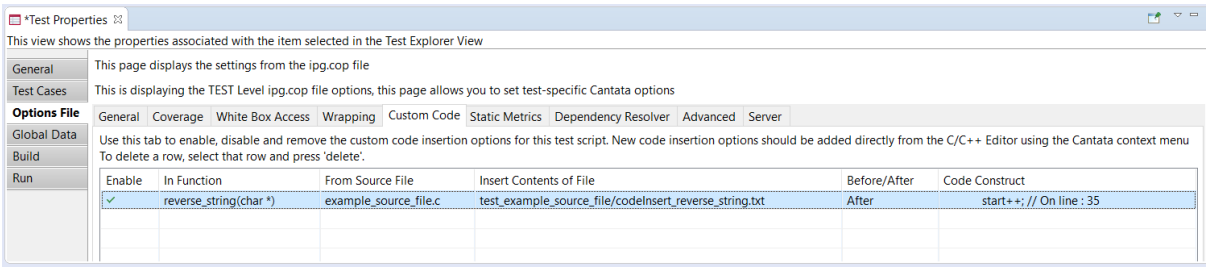
Cantata Wrapping provides the automated ability to modify code behaviour at the function call boundary, however Custom Code Injection can be used anywhere to:

- Inject errors
- Avoid use of debug jumps
- Change values of local variables
- Change values of variables and registers
- Breaking out of infinite loops
- Fire an interrupt
- Verify and store potential code fixes without modifying the production code

Custom code can be injected using the C/C++ editor to pinpoint exactly where to put it. The code snippet can be added before or after any statement.



The injected code is stored in a separate file and its use for one or more test scripts can be controlled from the new Cantata Test Properties view and the C/C++ editor.



Extended Platform Support

As with every version of Cantata, support for platforms has been extended.

Cantata is tightly integrated with leading Integrated Development Environments which are Built-on-Eclipse®, and toolchains available as Eclipse® Ready plug-ins. Cantata 8.0 is built on the Neon (4.6.3) Eclipse release, and is also available to install as an Eclipse-Ready plug-in set for releases Galileo (3.5) up to Oxygen (4.7), giving instant access to the ecosystem's latest rich set of plug-and-play tool integrations (e.g. ALM, CI and SCM tools).

Support for the GNU GCC and g++ compilers has been extended to version 7.1 (Windows) and 7.3 (Linux).

C++ 11 & 14 support has been enhanced to provide support for language features which were not covered in Cantata version 7.2. For a full list of supported C++ 11 or 14 features please contact sales@qa-systems.com