

What's New in Cantata 25.07?

Cantata 25.07 is available from July 2025.

The main feature of this release is extended support for modern toolchains and IDEs, including Visual Studio 2022, GCC/G++ versions 14 and 15, and Eclipse 4.34. A new test directive, usability enhancements, and numerous bug fixes further strengthen the capabilities of Cantata.

This version is also certified as suitable for use with the new EN 50716:2023 railway software standard which replaces the previous standards EN 50128:2011 and EN 50657:2017, helping users meet the updated compliance requirements.

This document details all the most important changes in Cantata version 25.07.

Contents

Copyright Notice	2
Introduction	3
Full CLI Test Script Generation	3
New Set Test Input Macro	4
CPPGETCOV extended for Coverage by Context	5
Bug Fixes	5
Core Components	5
Ancillary Tools	5
Updated Platform Versions Support	6

Copyright Notice

Subject to any existing rights of other parties, QA Systems GmbH is the owner of the copyright of this document. No part of this document may be copied, reproduced, stored in a retrieval system, disclosed to a third party, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of QA Systems GmbH. Google, GoogleTest, GTest, GoogleMock and GMock are registered trademarks of Google LLC.

© Copyright QA Systems GmbH 2025

Introduction

Cantata 25.07 is available from July 2020. This release adds further capabilities for our code-editor centric users with full CLI test script creation and execution and new test case editing, updates the tool certification for the latest railway software standard EN 50716:2023, enhanced code coverage reporting and further extends the supported platforms. In addition, many bugs in the Core Components and ancillary tools have been fixed in this release.

Cantata 25.07 also contains many other productivity and flexibility enhancements as well as fixes. The full set of changes are documented in the Release Notes which track all changes in Cantata since version 4.0. The most important changes in this release are highlighted in the sections below.

Full CLI Test Script Generation

For many code-editor centric Cantata users (e.g. those developing tests in editors such as VSCode), being, a new in Cantata 25.07 Command Line Testscript Generator utility application will be a welcome productivity enhancement. This new headless Eclipse utility generates a Cantata test script entirely from the command line using some configurable options and other default settings.

Provided there is an existing Eclipse workspace, the utility provides all the options needed to generate working test scripts including:

- > Test location, absolute & relative paths
- > Test script name
- > CSI file location
- > Stub and wrapper generation
- > Use of Cantata makefiles to build the tests or placing them next to the source being tested
- > Test case generation (no tests, 1 test per function, AutoTest and using/not test steps)
- > Code Coverage Ruleset

New Set Test Input Macro

In the 25.07 release, a new Cantata Library macro **SET_TEST_INPUT** has been added for code-editor centric users. This adds the capability when editing test scripts to both directly set an input value and log that input value into the generated Cantata Test Results file (.CTR) without code duplication. This new macro takes a single argument, **var** which should be a valid variable assignment. The following example shows a test case and the test results report section generated.

```
START_TEST("test_isTemperatureCritical",
           "Max temp value is critical");

/* Expected Call Sequence */
EXPECT_CALL("", "", "");
SET_TEST_INPUT(testTemperature = TEMP_CRITICAL_MAX);
/* Call SUT */
returnValue = isTemperatureCritical(testTemperature);

/* Test case checks */
CHECK_CHAR(returnValue, TRUE);
/* Checks on global data */
check_global_data();
END_CALLS();
END_TEST();
```

Test case

```
----- Start Test: test_isTemperatureCritical

Max temp value is critical

      Start Call Sequence Validation:
..... Test input: testTemperature = TEMP_CRITICAL_MAX
PASSED: Check: returnValue = TRUE
        value: 0x00    <NUL>

      End Call Sequence Validation:

PASSED: Function:

----- End Test: test_isTemperatureCritical
```

Test results file section

CPPGETCOV extended for Coverage by Context

The Cantata **CPPGETCOV** certified supporting tool is an executable which is run on one or more Cantata Code Coverage (.COV) files to generate certification suitable Cantata Test Results files (.CTR). Traditional code coverage measures execution of source code constructs, but does not take account of the context in which that code object executes. The same source code may behave differently depending on this object context (e.g. running in a multiple threads, polymorphic base class code in multiple inheritances, or state machine code in different states). Cantata provides the ability to overlay (measure and filter) the structural code coverage with its execution context. This allows users to differentiate code coverage by context. The applications of Cantata context coverage are:

- > Inheritance
- > State machines
- > Threads

Code coverage by inheritance context is gathered automatically by Cantata. Code coverage by state context or thread context can be set using a **user_context_function** added to the test script, which defines the possible different contexts for the execution of the source code under test.

In Cantata 25.07 **CPPGETCOV** has been enhanced so that it can do coverage reporting on different code contexts. A **--context:<string>** parameter specifies a context string for coverage reporting while the **--function:<name>** parameter specifies a function or method to report on. Both parameters default to **"*"**.

Bug Fixes

Core Components

More than 30 defects have been resolved in the certified core components. Highlights include:

- > Fixes for call-return coverage inconsistencies with template functions, lambdas, and constructors.
- > Improved compatibility with Boost and C++20 constructs.
- > Enhancements to CTR2HTML conversion logic.
- > Stability improvements in license logging, test harness buffer management, and expression evaluation.

Ancillary Tools

Improvements across complementary tools. Highlights include:

- > Enhanced AutoTest handling of complex data types and union access
- > Fixes to Eclipse UI elements, test case generation, and deployment tool reliability
- > Correction of missing or incorrectly reported tutorial content for international users

Updated Platform Versions Support

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

Cantata is tightly integrated with leading Integrated Development Environments which are Built-on-Eclipse®, and toolchains available as Eclipse-Ready® plug-ins. Cantata 25.07 is built on the Eclipse 2024-12 (4.34) and is also available to install as an Eclipse-Ready plug-in set for the following versions:

- > Eclipse 4.6 (Neon)
- > Eclipse 4.7 (Oxygen)
- > Eclipse 4.9 (2018-09)
- > Eclipse 4.14 (2019-12)
- > Eclipse 4.15 (2020-03)
- > Eclipse 4.16 (2020-06)
- > Eclipse 4.19 (2021-03)
- > Eclipse 4.23 (2022-03)
- > Eclipse 4.24 (2022-06)
- > Eclipse 4.30 (2023-12)

Support for the Microsoft Visual Studio® 2022 and GNU GCC for C/C++ 14 and 15 compiler versions have been added in Cantata 25.07. The full list of platforms supported is now as follows:

On Windows (32 & 64 bit) versions 7, 8/8.1, 10, 11:

- > Microsoft Visual C++ (32-bit): 2010, 2013, 2015, 2017, 2019, 2022
- > GNU GCC for C/C++: 4.6.2 to 13.2.x

On Linux (32 & 64 bit) kernel distribution agnostic versions: 3.x, 4.x, 5.x, 6.x:

- > GNU GCC for C/C++: 3.4.6 to 15.1.x

For more information or support, please visit: [QA Systems Cantata Support](#)

