



What's New in Cantata 7.0



Cantata 7.0, available from February 2016, is a major new release of Cantata. This document outlines the most important changes including completely new capabilities and a wide range of enhancements.

Introduction

Cantata 7.0, available from February 2016, is a major new release of Cantata.

This release contains significant new capabilities, a wide range of enhancements and some fixes. The full set of changes is documented in the Release Notes which track all changes since version 4.1. The most important changes are highlighted in the sections below.

An Add-on tool is also available with this release: Cantata Team Reporting enables distributed teams to work more effectively.

As with previous versions, once released, Cantata 7.0 will be independently certified by SGS TÜV SAAR to the highest level of integrity for the main software safety related standards:

- > ISO 26262 (Automotive)
- > IEC 62304 (Medical Devices)
- > IEC 61508 (Industrial)
- > IEC 60880 (Nuclear)
- > EN 50128 (Railways)

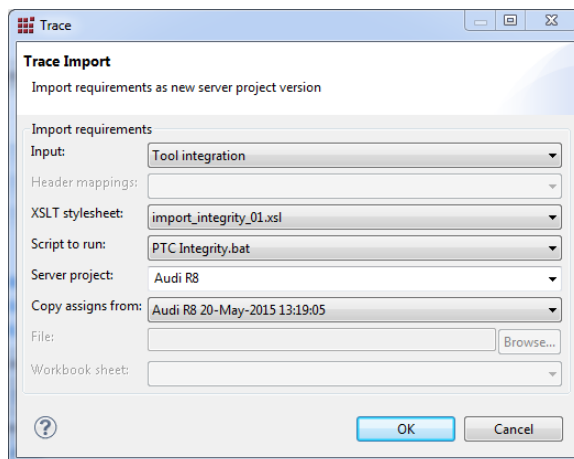


A full tool qualification kit will also be available for the avionics standards DO-178B/C and DO-330.

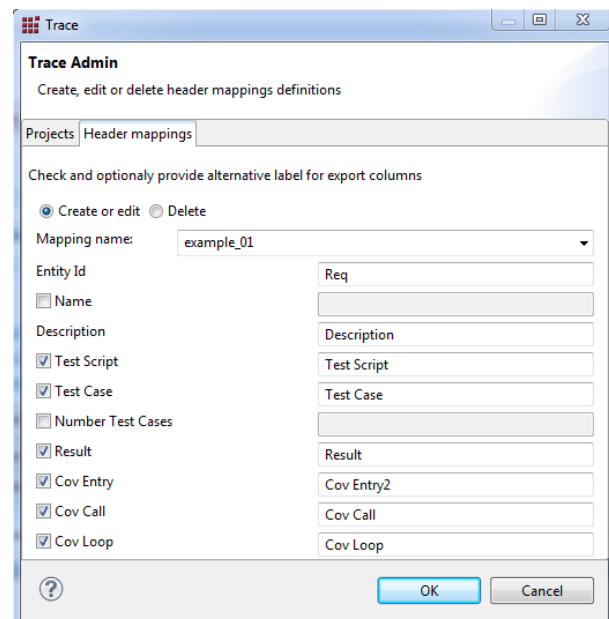
Cantata Trace

Bi-directional traceability of requirements or test plans with tests is required by all safety related standards. Cantata Trace is a completely new capability in Cantata 7.0, which makes it faster and cheaper for users to satisfy this bi-directional traceability. An intuitive drag-and-drop interface makes it easy to assign relationships and integrates with external Requirements Management Tools (RMT).

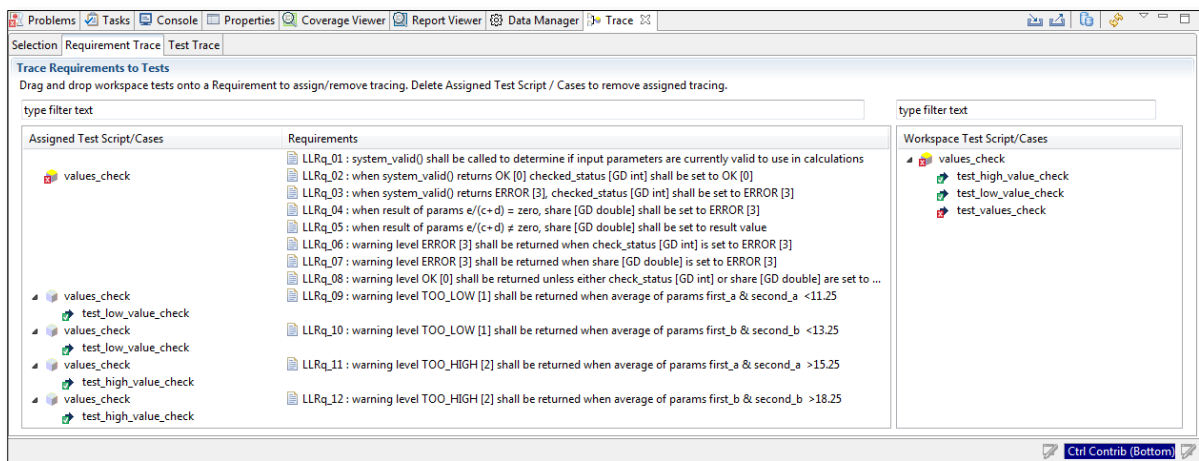
Requirements or test plan information is imported from a RMT such as DOORS®, PTC Integrity®, Polarion® or even Microsoft Excel®. This information is stored by Cantata on a server so that information can be accessed and synchronized across multiple workspaces.



Cantata Trace provides options for configuring Import and Export of requirements traced to tests and code coverage results.



Cantata Trace is used to assign trace relationships with tests scripts or test cases within them. The traced relationship is automatically updated with test results and code coverage status and exported to the RMT.



Cantata Trace with drag and drop association of requirements tests scripts / test cases, and visibility of all traced relationships in both the user's own workspace and on the server.

Cantata Trace provides out of the box support for import and export via XML and Microsoft Excel. An on-site direct tool API integration configures Cantata Trace to fit precisely with users' workflow and specific RMT (e.g. PTC Integrity, IBM DOORS, Polaron etc).

Enhanced Platform & Toolchain Support

Platforms are always changing, and integrating into a tool chain should not be a challenge. Cantata 7.0 continues to make it easy to stay up to date with current platforms and toolchains, easing developer workflow and saving time.

Host Operating Systems and Compilers

Cantata 7.0 includes upgrades for:

- > Windows 64-bit
- > Microsoft Visual Studio 64 bit
- > GNU tool chain up to version 4.9.x



QA Framework for QA-C and QA-C++



Cantata and QA Framework are installed in the same Eclipse IDE providing a seamless transition between dynamic and static testing. As the source code may be subject to change during the unit and integration stages of development, it is helpful for engineers to immediately know that such changes are compliant with the static analysis requirements.

To keep dynamic and static testing processes synchronised, Cantata 7.0 can automatically trigger QA-C/C++ re-analysis upon source code changes in the Cantata workspace.

Eclipse® Environments



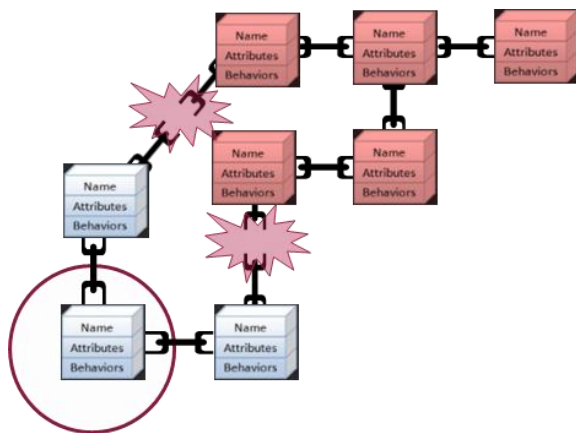
Cantata 7.0 is built on the Luna (4.4) release, and is also available to install as an Eclipse-Ready plug-in set for releases Galileo (3.5) up to the recently released Mars (4.5), giving instant access to the ecosystem's latest rich set of plug-and-play tool integrations (e.g. SCM, CI tools).

Cantata 7.0 is tightly integrated with leading Integrated Development Environments Built-on-Eclipse and toolchains available as Eclipse Ready plug-ins.



Dependency Resolver

The efficiency of isolating testing C/C++ is limited by a chain of linker dependencies required to build the software under test. For C++ although these referenced methods in classes are not directly called by the software under test, their definitions are still needed. Iteratively and manually creating these definitions in stubs, when identified by the linker as an unresolved symbol, can be so time consuming that isolation testing C++ may not be cost effective.

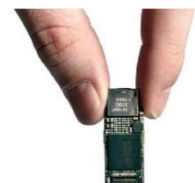


Dependency Resolver automatically reads the linker output to find any undefined references that have been identified, it then adds those undefined references (data or routines) to the build to create a complete test executable. This breaks the manual dependency chain, freeing testers to efficiently isolation test C++ classes. This increases efficiency as testers only need to generate call interface controls (stubs, isolates and wrappers) for calls which they wish to test.

Reduced Target Footprint

With Cantata 7.0 target libraries, testing on small targets can be an order of magnitude faster. A new subset without the `sprintf()` function, reduces ROM requirements by 40%, and uses integer IDs not string literals. The subset provides:

- > More target memory available for source code or Cantata test cases
- > Support for smaller targets and safety RTOSs (e.g. GreenHills Integrity-178)
- > Users' environment can be compliant with the MISRA 2012 guidelines, which prohibits `sprintf`
- > Formatting test results on the host results in faster data transfer from target to host



Local Static Data Accessibility

Cantata 7.0 new automatic Accessibility Instrumentation provides direct test script access to function local static data for even more efficient White-Box unit tests. It allows testers (read and write) access to data declared static to local functions (not just a file) and unlocks it for testing.

The most common use of local static data is for storing the logical state within state machines. This new accessibility instrumentation provides the Cantata test script the ability to:

- > Use white-box testing within state machines
- > Read local static data to check machine state between transitions
- > Write local static data to efficiently set test case trigger pre-conditions prior to state transitions

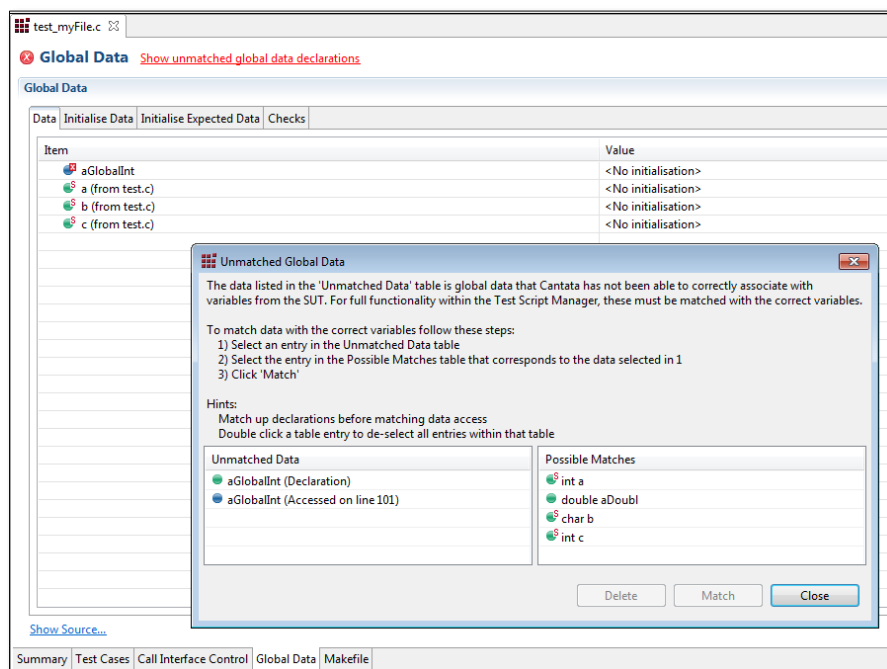
```
state_machine.h | state_machine.h
#include "state_machine.h"

/* State Machine - main control function */
static void state_machine() {
    static EState vState = EInitialise;
    EReturn vReturn_e = EOK;
    switch (vState) {
        case EInitialise:
            vStatus_e = initialise();
            if (vStatus_e == EOK) {
                vState_e = EGetData;
            } else {
                vReturn_e = EError;
            }
            break;
        case EAction:
            vStatus_e = action();
            if (vStatus_e == EOK) {
                if (vData_e.state == EAction) {
                    /* Perform action again */
                    vState_e = EAction;
                } else if (vData_e.state == EFinished) {
                    vState_e = EFinished;
                } else {
                    /* Unexpected state - return an error */
                    vReturn_e = EError;
                }
            } else {
                vReturn_e = EError;
            }
            break;
        default:
            vReturn_e = EError;
            break;
    }
    if (vReturn_e == EError) {
        vState_e = EFinished;
    }
    return vReturn_e;
}
```

Improvements to Test Script Manager

In Cantata 7.0 Test Script Manager has been improved to ease workflow and enhance test productivity.

- > A new Test Script Model provides more robust synchronisation of the C/C++ test script with its graphical representation in the Test Script Manager.
- > Text selection and sorting of lists has been added to the Call Interface Control tabs in the test generation wizard and within the Test Script Manager.
- > Test script synchronisation with the software under test, to update for code changes:
 - Improved matching of call interface controls (stub/isolate/wrapper)
 - Enhanced matching of static data and global data



Cantata Team Reporting Add-On

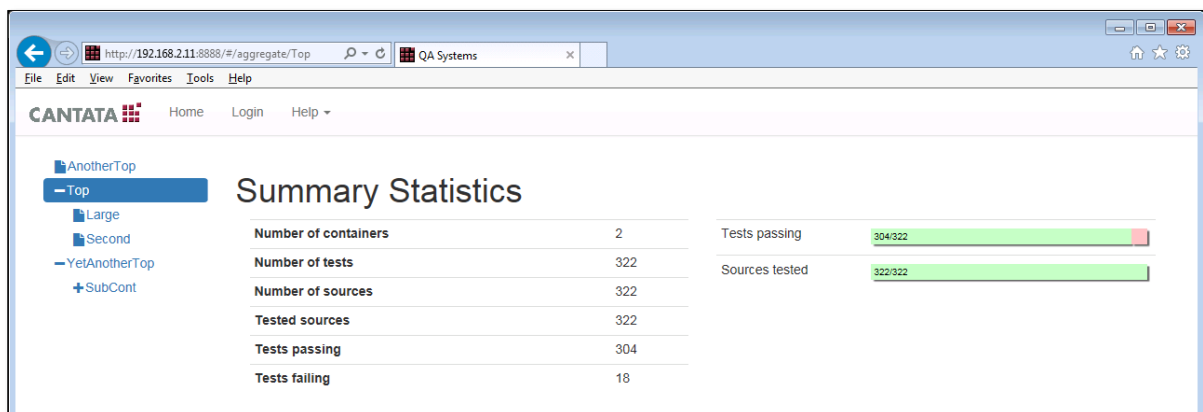
Cantata Team Reporting enables distributed teams to work more effectively together. It is available as an optional Add-on for Cantata 7.0.

Desk top users have a simple client interface to push (automatically or selectively) tests and additional data about them into a folder hierarchy onto a Cantata server. This additional test data for each container in the hierarchy can include:

- > Environment variables requested by their names (e.g. PATH)
- > Java property values, requested by their property name (e.g. operating system name)
- > Cantata Properties (e.g. Cantata version, Target Specific Configuration)
- > User defined properties

Cantata Team Reporting provides a web interface for teams to share tests avoiding the need to replicate tests over multiple workspaces. It also enables managers to monitor test status and progress. Highlights of the web interface include:

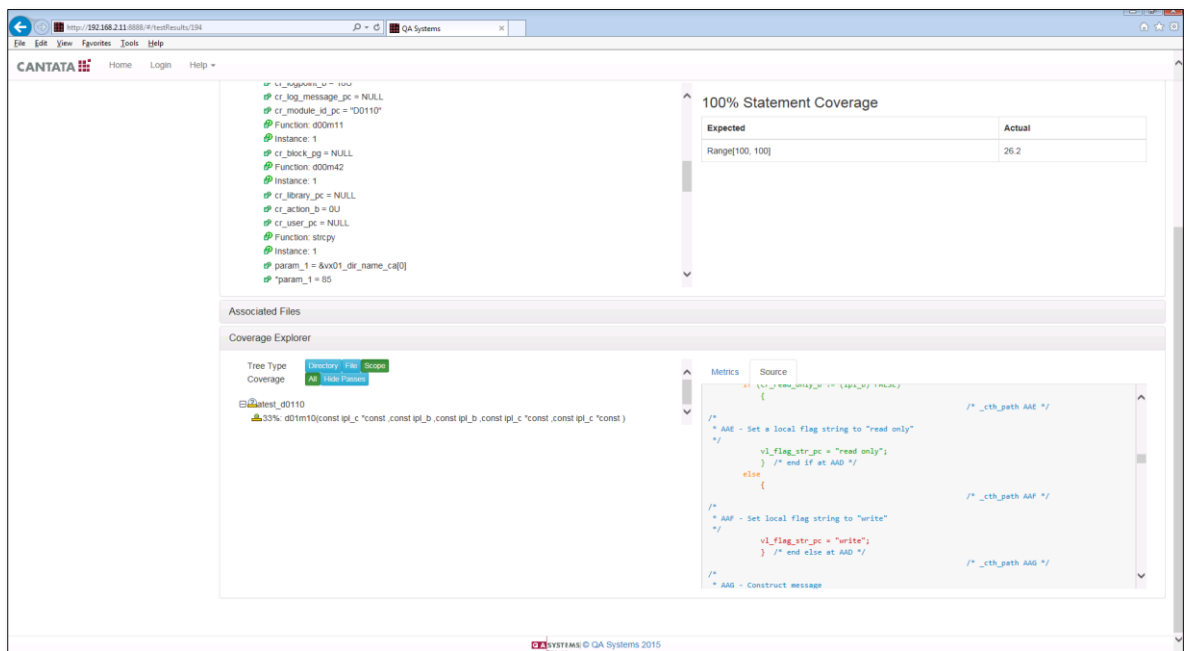
- > On-line summaries of test execution and results status (Cantata and any other tests) with filters, ordering and history
- > Differentiation of tests by Additional Test Data (e.g. by user, tests run, different system configuration variants)
- > Drill-down through a hierarchy of containers from summaries to detailed results
- > Interactive diagnostics (similar to Cantata Test Results Explorer, Coverage Explorer and Coverage Viewer) visible for all test results
- > Aggregation of code coverage data over multiple tests
- > A REST API providing a JSON formatted response to queries, so results data can be directly matched and provided in any 3rd party interface



Cantata Team Reporting Summary View

What's New in Cantata 7.0

Page | 7



Cantata Team Drill-Down Summary Views