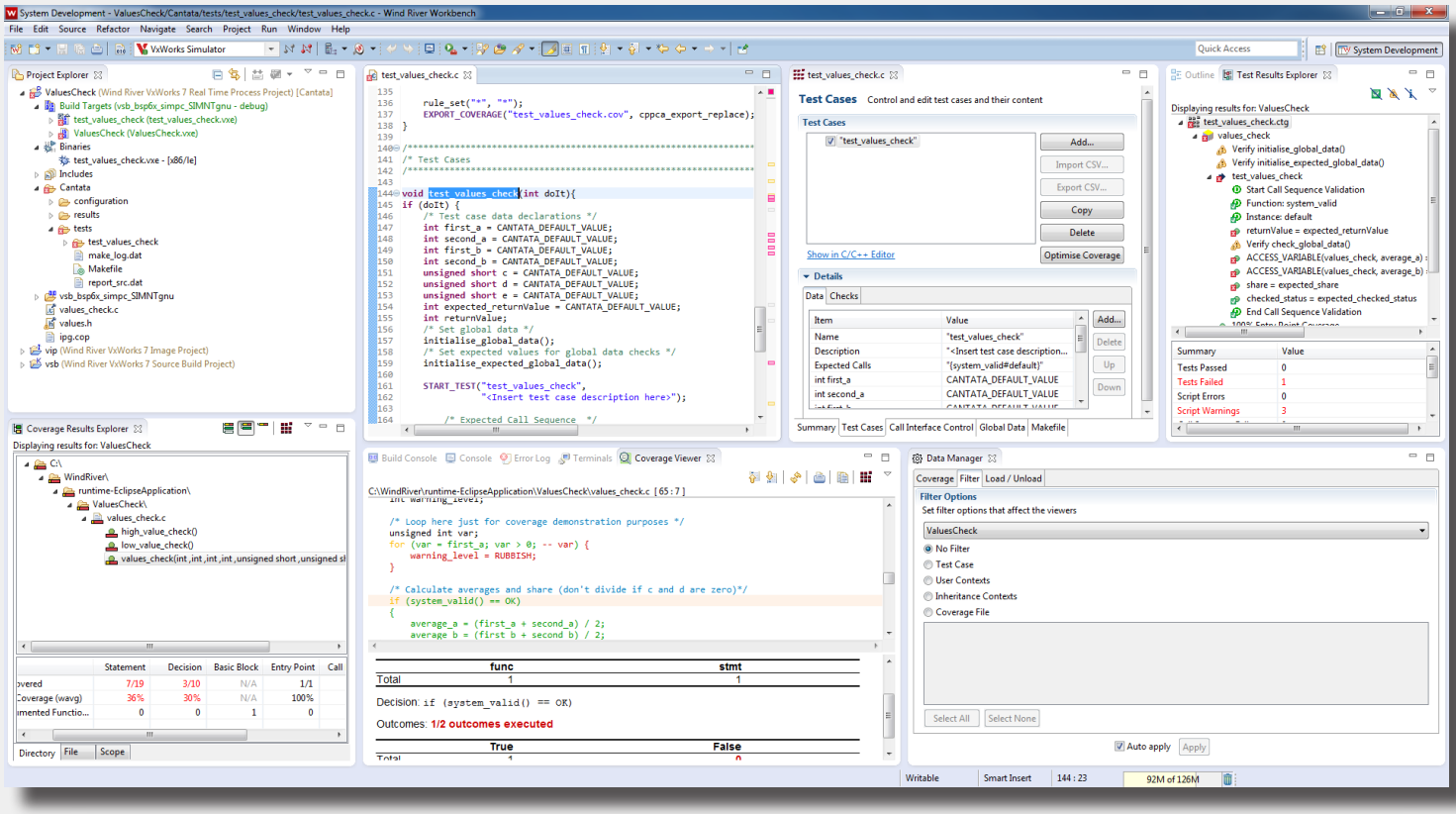
QA SYSTEMS

CANTATA for VxWorks 7

Technical Overview



- Cantata is seamlessly integrated with VxWorks 7, providing C/C++ developers with unit and integration testing and integrated code coverage. Regulatory safety standards compliance is achieved faster and cheaper.

- Cantata uses the Build Specs and the Target Connection Framework within Workbench to integrate all aspects of testing into the project environment, allowing users to build, run and obtain their results with minimal effort.

- Cantata installs easily into Wind River Workbench using the powerful Eclipse P2 provisioning system and the built-in configurations for the VxWorks simulator, or deployment for any VxWorks target.

CANTATA
30-day Free Trial

Now Available...
WIND RIVER
MARKETPLACE

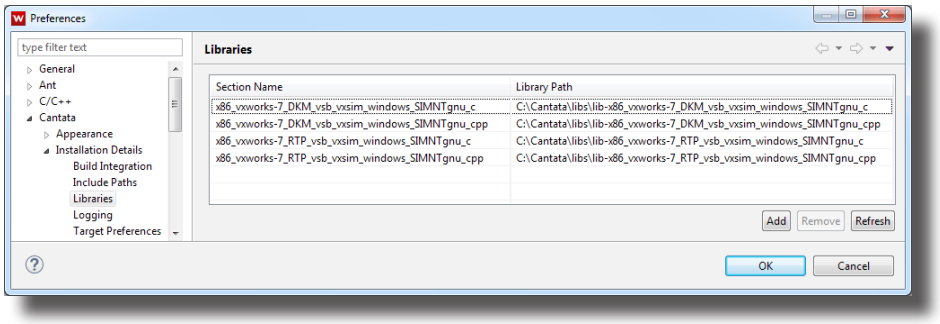
Test Planning

Static Analysis generates over 300 source code metrics (procedural and object oriented), in csv format, for maintainability and complexity aid estimating test effort.

Requirements Based Testing is supported through tagging of test cases for requirements traceability into reports.

Test Automation

Out-of-the-box Configurations for the VxWorks simulator. Fully integrated Deployment Wizard for custom build specs allows end users to configure Cantata for any VxWorks target.



Test Script Generation by parsing source code to derive parameter, calls and data information with call interface control (Stubs, Isolates and Wrappers) generated into the test script(s). Optional initial values for variables and returns enable tests to run immediately.

Test Script Manager provides graphical development, editing and management of tests synchronised with the C/C++ editor.

Re-use of xUnit Tests within Cantata tests, retains investment in open source scripts, allowing them to be re-used in and extended with Cantata features.

Intuitive Test Directives reduce manual scripting, by calling library of helper functions to provide structured, readable, independent, repeatable tests with clear and unambiguous results.

Test Execution

Cantata tests are fully integrated into Wind River Workbench projects. This allows Cantata to automatically create Build Targets for the Workbench build system, and so Cantata tests can be built and run with a single click. Test results are automatically obtained during test execution, and stored within the specific test folder.

Interface Control

All calls (external and internal to the compilation unit boundary) may be simulated by Stubs / Isolates or intercepted using the real call via Wrappers.

Interface Control Selection for method used on each call at test script creation (with defaults), and script editing, with call list is refreshed on each code build.

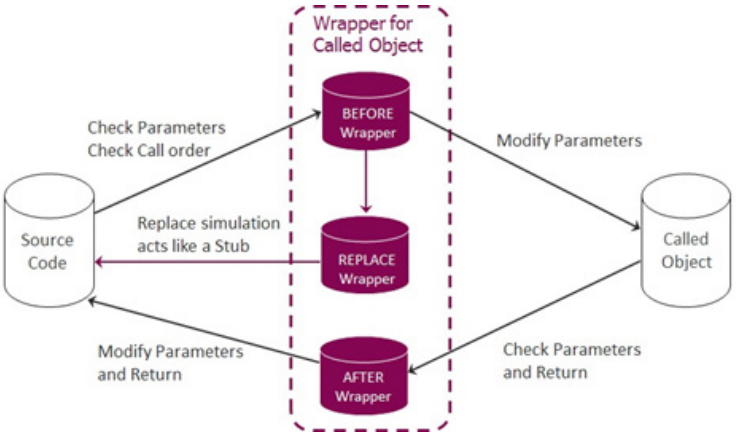
Optional Automatic Checks of all parameters/returns passed over call interfaces.

Programmable Test Control Points are available at each call interface control through named instances to set and check data, raise exceptions etc.

Automated Stubs provide replacement simulation of called object interface.

Automated Isolates provide simulation of linked called object interface, and support system and variadic functions.

Automated Wrappers provide interception of linked called object interface with Before/After and Before/Replace modes to intercept or simulate. More powerful and flexible control over interfaces than stubs or isolates.



Call Sequence Verification provide full control over call order and use of Stub, Isolate or Wrapper instances in each test case, with exact sequence or any time matches.

Black & White-Box Testing

Cantata provides a high degree of automation at unit and integration levels for both black-box and more efficient/ thorough white-box testing.

Per Function Testing generates a complete template test case for each function/method in the code with all parameters, accessible data and calls to control identified.

Table-driven Testing provides multiple input value ranges, combinatorial effect calculator and CSV import/export for large data set black-box verification.

Robustness Testing with pre-defined values for basic data types, in table driven test cases.

Automated Global Data Checks to verify expected and unexpected changes to all global data accessible to the software under test.

Automatic White-Box Access provides the ability to efficiently call functions or methods, and set/check data directly from the test script without conditional compilation:

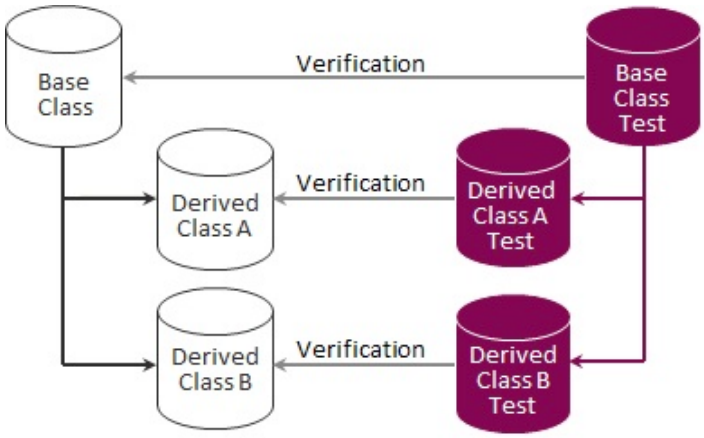
- Static functions and private methods
- Private data and data static to the file

Automated C++ Exception Verification of multiple expected and unexpected exceptions.

Object Oriented Testing

Cantata unit and integration testing supports: C++ functions, templates, classes or clusters of classes in single or multiple source files. Test scripts for classes are written in C++ as a test class.

Automated Test Case Re-use via parallel inheritance hierarchy, retains benefit of code re-use for testing, and verify that a derived/specialised class has the same semantics as a base or template class (the Liskov Substitution Principle).



Automatic Test Class Inheritance structure created for inherited classes, so child class tests inherit from parents.

Automatic Implementation of abstract base classes (ABCs) or pure virtual methods, with generation of appropriate stubs in the script. Supports use of the factory method.

Code Coverage

Coverage analysis of C/C++ and Java provides objective measurement of how effective testing has been in executing the source code.

Code Coverage Metrics supported:

- Entry Points
- Call Returns
- Statements
- Basic Blocks
- MC/DC (Masking and Unique Cause)
- Relational Operators
- Loops
- Conditions
- Decisions (branches)

Coverage by Context provides coverage by:

- Each Cantata Test Case
- Derived inheritance contexts
- User defined context for multiple states, threads and data contexts (for DO-178B/C data coupling analysis)
- Any non-Cantata test run

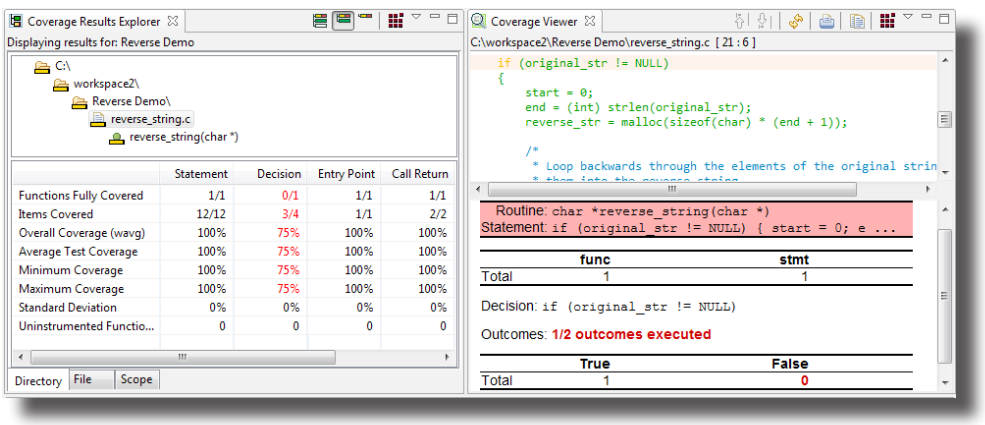
Configurable Coverage Requirements are easily defined in simple Rule Sets. Predefined rule sets can be used for standard compliance.

Coverage Target Checks on metrics are integrated into dynamic tests resulting in Pass/Fail for coverage requirements.

Automatic Test Case Optimisation provides option to remove/disable test cases which do not contribute additional coverage.

Project Code Coverage Trees for multiple projects with filters for coverage types/fully covered code items and drill down to highlighted source code.

Source Code Coverage Views highlight individual code constructs (not just by line) with additional diagnostics by test case, test run and metric type.



Automatically Test Legacy Code

Cantata AutoTest automatically generates complete passing unit test scripts to:

- Reduce reliance on system tests
- Support continuous integration
- Automatically close gaps in coverage
- Identify testability problems in code
- Easily change unit testing tools

Configurable Test Depth with code paths determined by selecting the metric types in a code coverage Rule Set.

Configurable Automatic Verification using standard workspace preferences for passing checks on function(s) in each file of:

- Return value from function
- Output parameters from function
- Accessible global data values
- Order of function calls made
- Parameter values to function calls

IDE Test Generation for selected source files or directly into existing test scripts.

CLI Test Generation for larger code bases selected by source directories, files and functions.

Separate Standard Licence Feature allows controlled use of AutoTest.

Testability Issues Identified with warning messages for problems in code preventing generation of complete passing tests:

- Dynamically unreachable code
- Crash Scenarios
- Data uninitialised or function static
- Implicit function declarations
- Compiler type truncation

AutoTest Generation Report created as HTML report for fully, partially or untested files and functions.

Automatic Regression Suite of AutoTests created using Cantata Makefiles for CLI invocation.

Ongoing Maintainability of AutoTest is simplified as test scripts are standard Cantata style, with re-use of call instances and detailed path solving description for each generated test case.

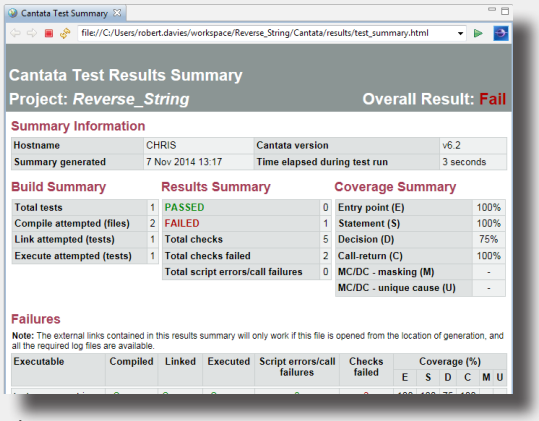
Diagnostics & Reports

Cantata offers powerful filterable diagnostics of test and code coverage results within the UI, and flexible user configurable reports. Detailed Test Diagnostics for all checks of expected against actual results by test case for:

- Global data
- Parameters and returns
- Exceptions
- Call order
- Code coverage targets

Printable Views from within Cantata UI for all test and coverage results displays.

Test Summary Reports for test build and execution results of all tests with Makefiles.


Cantata Test Summary

file://C:/Users/robert.davies/workspace/Reverse_String/Cantata/results/test_summary.html

Cantata Test Results Summary

Project: *Reverse_String* **Overall Result: Fail**

Summary Information

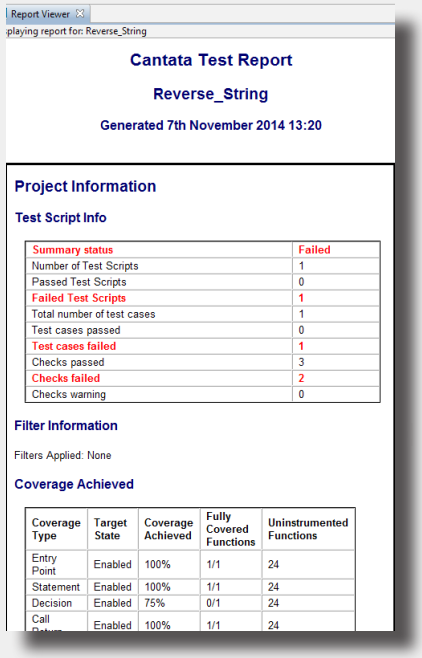
Hostname	CHRIS	Cantata version	v6.2
Summary generated	7 Nov 2014 13:17	Time elapsed during test run	3 seconds

Build Summary		Results Summary		Coverage Summary	
Total tests	1	PASSED	0	Entry point (E)	100%
Compile attempted (files)	2	FAILED	1	Statement (S)	100%
Link attempted (tests)	1	Total checks	5	Decision (D)	75%
Execute attempted (tests)	1	Total checks failed	2	Call-return (C)	100%
		Total script errors/call failures	0	MC/DC - masking (M)	-
				MC/DC - unique cause (U)	-

Failures

Note: The external links contained in this results summary will only work if this file is opened from the location of generation, and all the required log files are available.

Executable	Compiled	Linked	Executed	Script errors/call failures	Checks failed	Coverage (%)					
						E	S	D	C	M	U


Report Viewer

playing report for: Reverse_String

Cantata Test Report

Reverse_String

Generated 7th November 2014 13:20

Project Information

Test Script Info

Summary status	Failed
Number of Test Scripts	1
Passed Test Scripts	0
Failed Test Scripts	1
Total number of test cases	1
Test cases passed	0
Test cases failed	1
Checks passed	3
Checks failed	2
Checks warning	0

Filter Information

Filters Applied: None

Coverage Achieved

Coverage Type	Target State	Coverage Achieved	Fully Covered Functions	Uninstrumented Functions
Entry Point	Enabled	100%	1/1	24
Statement	Enabled	100%	1/1	24
Decision	Enabled	75%	0/1	24
Call Return	Enabled	100%	1/1	24

Configurable XML reports with summary or full test details consolidate across multiple projects.

Project Level Tree Views of test pass/fail results with drill down hyperlinked navigation to individual test cases and checks.

ASCII Text Report available with passes and fails highlighted inside Cantata with outline view, or as plain text results file for high integrity certification needs.

Platform Availability

VxWorks 7

HOST PLATFORMS

- Windows XP, Vista, 7, 8
- Linux 2.4, 2.6, 3.x Kernel

OTHER WIND RIVER PLATFORMS

- VxWorks 6.x
- VxWorks 653
- VxWorks 5.4, 5.5
- VxWorks DO-178B and 61508 (6.6 Cert)
- Linux 3.0, 4.0
- Workbench 3.3.x, 4.0