

Sicherheitsstandards schnell
und sicher erfüllen.
Mit automatisierten Unit- und
Integrationstests für C & C++.



Wir geben Gas, damit Ihre
Software besser wird



Auf einen Blick: Gute Gründe für Cantata

CANTATA 



Standards einhalten: einfach und kostengünstig

Cantata ist die Gesamtlösung, welche die Testanforderungen führender Sicherheitsstandards an Unit- und Integrationstests erfüllt.

Cantata ist vom TÜV Saar geprüft und zertifiziert. Das Cantata Zertifizierungskit hilft Ihnen, Ihre Software einfacher zu zertifizieren.

Wie werden Sicherheitsstandards bei der Softwareentwicklung eingehalten? Das herauszufinden, ist schwierig und mit viel Zeitaufwand verbunden – zumindest ohne Cantata! Die Leitlinien der Standards sind in Cantata enthalten. Sie erfüllen also automatisch alle Anforderungen an dynamische Tests. Das macht standardgemäßes Programmieren schneller und kostengünstiger.

Zertifiziert für:

- › ISO 26262:2018 (Automobilindustrie)
- › EN 50128:2011/A2:2020 (Bahntechnik)
- › EN 50657:2017 (Bahntechnik)
- › IEC 62304:2006 (Medizintechnik)
- › IEC 61508:2010 (Allgemeine Industrie)
- › IEC 60880:2006 (Atomkraft)

Qualifizierbar für:

- › DO-178B
- › DO-178C / DO-330
- › Weitere Standards auf Anfrage

Softwareausfall? Gefahr gebannt!

Was ist schädlicher für den Ruf eines Softwareunternehmens als eine Rückrufaktion? Und teuer wird es auch. Denn der damit verbundene Aufwand übersteigt häufig die Entwicklungskosten. Mit Unit-Tests vermeiden Sie Fehler, bevor diese wirken können. Das spart Aufwand, Nerven und Geld!

Mit Cantata sind Sie schneller am Markt. Denn je früher Fehler entdeckt und eliminiert werden, desto sicherer halten Sie Entwicklungspläne ein. Und desto besser lassen sich einzelne Komponenten integrieren.

Cantata verschafft Rechtssicherheit. Mit Cantata getestete Software besteht auch vor Gericht. Denn sie entspricht dem State of the Art der Softwareentwicklung.

Kostenvorteil Automatisierung

Tests mit Cantata können hochgradig automatisiert werden. Das senkt die Kosten und den Aufwand auf ein Mindestmaß. Das gilt für alle testrelevanten Bereiche:

- › Testrahmengenerierung
- › Testfallgenerierung
- › Baseline-Tests auf Legacy Code
- › Tests auf Host- und Zielplattformen
- › Ergebnisdiagnose und Berichte
- › Regressionstests für die Continuous Integration

Auch das hilft Ihnen, Kosten zu senken: Cantata ist eng verzahnt mit Cross-Compilern und Entwicklungsumgebungen sowie DevOps-Workflows. Das macht es Entwicklern einfach, Cantata in jede beliebige Toolchain zu integrieren. Zusammen mit den intuitiven C/C++-Codetests in Eclipse GUI oder Code Editoren ist Cantata die beste Lösung: kostengünstig und schnell zu erlernen!

Schneller am Markt

Bessere Software oder schneller am Markt? Mit Cantata geht beides! Wer qualitativ hochwertige Software schnellstmöglich marktreif haben möchte, kommt an Cantata nicht vorbei.

- › Cantata verbessert die Zusammenarbeit Ihrer Entwicklungsteams. Und auch deren Effizienz. Und es verkürzt die Integrationszeit der einzelnen Softwarekomponenten.
- › Bessere, effektivere Zusammenarbeit plus schnellere Integration der Komponenten – das bringt den Zeit- und Kostenvorteil!

Cantata Highlights auf einen Blick

- > Automatische Erstellung von Testgerüsten und Testfällen
- > Unterstützung für unterschiedliche Plattformen und integrierte Tool-Chains
- > Anwenderfreundliche Eclipse-GUI und in C/C++ geschriebene Tests
- > Anforderungen können bidirektional rückverfolgt werden
- > Einzigartige Funktionsaufrufsteuerung (Wrapping) zum Simulieren, Abfangen und Überprüfen von Aufrufen
- > Flexible User Code Einbindung
- > Unterstützung von Test Driven Development (TDD)
- > Integrierte Analyse der Codeabdeckung
- > Automatisierte Regressionstests
- > Automatische Aufbereitung der Tests nach Codeänderungen
- > Kostenloses Zertifizierungskit für alle wichtigen Sicherheitsstandards

Unterstützte Testtypen:

- Black-Box-Tests / White-Box-Tests
- positive / negative Tests
- Anforderungstests / Robustheitstests
- einzelne / große Eingangsdatensätze
- funktions- / objektorientierte Tests
- Tests von Code / Header
- Aufrufsimulation / -steuerung (Wrapping)
- Isolations- / Integrationstests
- Host- / Targetausführung
- Anforderungstests / Baselinetests

Cantata arbeitet mit Ihrer Umgebung

Cantata wird auf Windows®- und Linux®-Host-Betriebssystemen mit einer Built-on-Eclipse®-IDE oder als Eclipse®-fähiges Plug-in-Set installiert. Es unterstützt GCC- und Microsoft Visual Studio® Compiler. Ergänzt wird Cantata durch eine Reihe eingebetteter Entwicklungs-Toolchains:

- ✓ IDEs / RTOSs
- ✓ GCC, Microsoft, Cross-Compiler
- ✓ Debugger
- ✓ Build-Integration/ Continuous Integration
- ✓ Software-Konfigurationsmanagement
- ✓ Anforderungsmanagement

Build-System Einstellungen von Nicht-Eclipse-Toolchains werden automatisch in Cantata-Testprojekte importiert. Um zu erfahren, wie Ihre Tools und Plattformen unterstützt werden, kontaktieren Sie bitte QA Systems.

Tests uneingeschränkt auf Zielumgebung ausführbar!

Für Cantata gibt es keine Lizenzbeschränkungen auf der Zielumgebung. Dafür sorgt der eingebaute Assistent. Die Bibliotheken und Konfigurationsoptionen sind mit Ihrem Code kompatibel. Sie sind zudem getestet und zertifiziert für sicherheitskritische Projekte. Der Austausch von Deployments macht es einfach eine Gruppe von Tests auf verschiedenen Targets für verschiedene Produktvarianten zu verwenden.

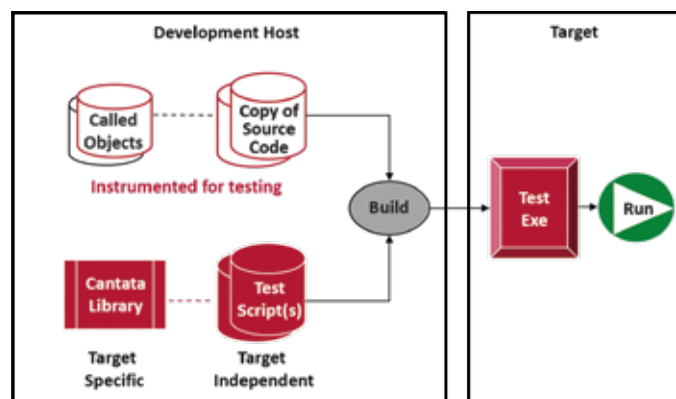
Tests einfach und flexibel. Auch auf Ihrer Zielplattform!

Tests mit Cantata sind als ausführbare C/C++-Dateien aufgebaut. Sie laufen auf einer Zielplattform genauso wie Ihr eigener Code auf einem

- ✓ Simulator
- ✓ Emulator
- ✓ physischen Target-Board

Zur Diagnose und für das Reporting werden Funktions- & Abdeckungsergebnisse zum Host zurückgeleitet. Dieser Prozess verläuft vollautomatisch und verwendet Cantata Makefiles. Zusammen mit Testskripten und der individuellen Targetanbindung wird einfach und flexibel von der GUI oder der Kommandozeile aus auf der Zielplattform getestet.

Ihr produktiver Code wird zum Testen nicht verändert.



Flexibler Testrahmen

Ein flexibler Testrahmen (Testskripts und eine unterstützende Bibliothek) ermöglicht jede beliebige Kombination von Testarten – sowohl für Unit-Tests, als auch für skalierbare Integrationstests. Tests können in einer GUI oder direkt als C/C++-Code bearbeitet werden. Sie laufen auf Host- wie auch auf Zielplattformen.

Test Driven Development (TDD)

Mit Cantata können Tests geschrieben werden, sobald Funktionsprototypen in Header-Dateien erstellt wurden. Auf diese Weise kann der Testrahmen erstellt werden, bevor der Hauptteil der zu testenden Software vollständig implementiert ist. Cantata für TDD erweitert die traditionellen Black-Box-TDD-Techniken, indem es den Zugriff auf umfassende White-Box-Tests für gekapselte Code Interna wie private oder statische Daten und Funktionen ermöglicht.

Black-Box/White-Box

Die hohe Automatisierung im Tool unterstützt beim Erstellen umfassender, präziser Black-Box und effizienter White-Box Tests. Gründliche Black-Box Tests vollziehen sich durch Testvektoren mit vordefinierten und vom Nutzer erstellten Massen- und Randdaten. Der kombinatorische Effekt wird ständig mitberechnet und angezeigt, und auch ein CSV Export und Import großer Datenmengen ist möglich.

Die granulare Instrumentierungsoptionen in Cantata ermöglichen den Zugriff auf gekapselte Daten und Code. Dies geschieht ohne bedingte Kompilation, und sowohl statische als auch private Funktionen und Variablen werden unmittelbar vom Testskript aus zugänglich. Der Nutzer kann durch Instrumentierung flexibel Code zu Testzwecken hinzufügen, ohne den Originalcode zu verändern.

Robustheitstests

Robustheitstests sind einfach durchführbar. Dafür sorgen Cantata Regelsätze mit vorher festgelegten Werten für die Grunddatentypen. Sämtliche zugänglichen Daten werden automatisch auf versehentliche Änderungen geprüft.

Objektorientierte Tests

Cantata objektorientierte Test-Styles sind mit einer automatisierten Testfallwiederverwendung über parallele Vererbungshierarchien ausgestattet. Dazu verfügt Cantata über eine automatische Testklassenvererbungsstruktur für vererbte Klassen. Abstrakte Basisklassen (ABCs) oder rein virtuelle Methoden können automatisch implementiert werden.

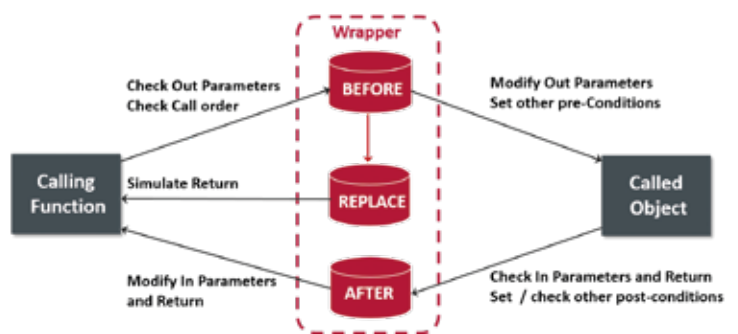
Objektorientierte Tests von Cantata werden als Klassen für Testmethoden, Templates oder Klassencluster implementiert. Damit es einfacher wird, C++-Tests isoliert durchzuführen, muss die Kette der Klassenabhängigkeiten unterbrochen werden. Das macht Cantata automatisch, indem es Klassenabhängigkeiten auf nicht festgelegte Referenzen automatisch lokalisiert und aufhebt.

Einzigartig: Die Cantata Funktionsaufrufsteuerung - Wrapping

Cantata generiert automatisch Testelemente, um alle Funktionsaufrufe der getesteten Software sowohl zu simulieren (Stubs) als auch abzufangen (Wrapper), wobei Folgendes bereitgestellt wird:

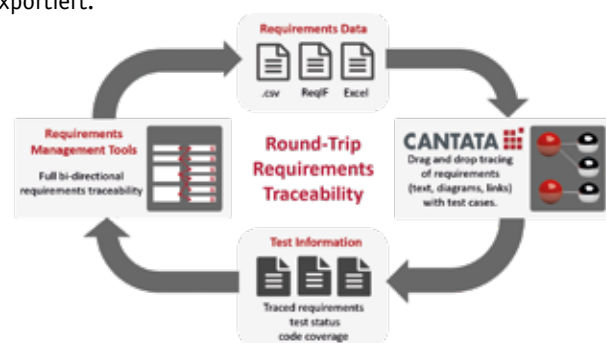
- > optionale automatische Prüfungen von Parametern und Daten
- > mehrere Instanzen für unterschiedliches Aufrufverhalten
- > flexible Überprüfung der Aufrufreihenfolge in jedem Testfall
- > Erkennung von Schnittstellenfehlern und Fehlersimulation
- > Control Coupling Testing

Wrapper dienen dazu, das Ist-Verhalten eines aufgerufenen Objekts zu überprüfen. Wenn eine Simulation nicht möglich ist oder nicht gewünscht wird (zum Beispiel interne Aufrufe bei der Integration, Betriebssystemaufrufe, Hardwareschnittstellen usw.), sorgen Wrapper für eine leistungsstarke Aufrufkontrolle, die nur Cantata bietet.



Rückverfolgung der Anforderungen

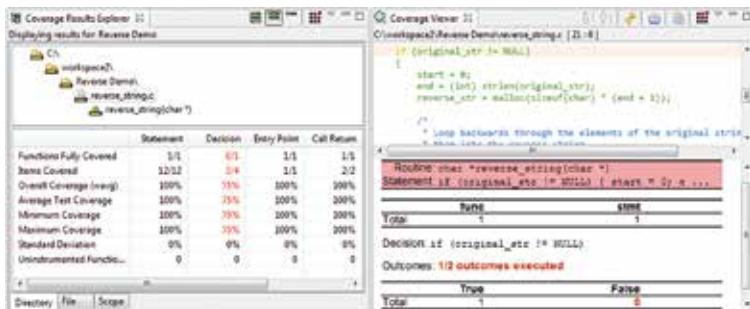
Cantata unterstützt Anforderungsverfolgung in beide Richtungen: von den Anforderungen zur Software und zurück. Testpläne oder Anforderungen werden in den Cantata Server importiert. Die Beziehungen zu Testfällen werden über eine Drag-and-Drop-Schnittstelle zugewiesen und dann zusammen mit den Testergebnissen und Informationen zur Codeabdeckung exportiert.



Sowohl der Import, als auch der Export können über CSV, Microsoft Excel® oder Requirements Interchange Format™ (ReqIF™) erfolgen – je nachdem, was Ihre Anforderungsmanagementlösung (z.B. IBM®, Rational®, DOORS®, PTC Integrity®, Polarion® Requirements™, Intland codeBeamer®) unterstützt.

Codeabdeckung

Cantata Codeabdeckung bietet die Möglichkeit, objektiv zu messen, wie gründlich die Tests den Quellcode ausgeführt haben (unabhängig von Cantata Testskripten). Die standard-spezifischen Cantata Coverage Regelsätze sind sehr bedienerfreundlich, weil die Datenberichterstattung und integrierte Prüfung der Codeabdeckungslevel hoch automatisiert sind.



Die Codeabdeckung wird anhand folgender Kennzahlen gemessen:

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

Cantata Build Variant Coverage sammelt Daten zum Quellcode, wenn Definitionen zum Erstellen von ausführbarem Code in verschiedenen Varianten verwendet werden, und bietet eine Analyse und ein zertifiziertes Reporting der Coverage, die über alle Varianten aggregiert wird.

Die präzise Diagnose kann die Abdeckung für komplette Projekt-Codebäume filtern oder zusammenfassen, und nach Testfall, Testdurchlauf und Codekontext auf einzelne Codekonstrukte innerhalb der einzelnen Codezeilen herunterbrechen (Vererbung, Threads, Zustände, Datenkopplung, usw.). Die automatische Testfalloptimierung hilft bei der Auswahl von Testfallvektoren aus großen Datensätzen und reduziert den Aufwand bei Regressionstests.

Continuous Tests



Cantata Makefiles werden bei der Testerstellung automatisch generiert, um Ergebnisse für Testreihen im Batch-Modus zu kompilieren, zu verknüpfen, bereitzustellen, auszuführen und abzurufen. Sie können mit bestehenden Makefiles verwendet werden und sie lassen sich für Regressionstests leicht mit Open Source oder kommerziellen Continuous Integration Lösungen, wie Jenkins und Bamboo, integrieren.

AutoTest

Die automatische Generierung von Testfallvektoren kann eine 100%ige Codeabdeckung erreichen, während Daten, Parameter und Aufrufreihenfolge überprüft werden. Auf diese Weise können sowohl Lücken in der Codeabdeckung geschlossen, als auch vollständige Unit-Tests erstellt werden, welche einfach mit Anforderungen verknüpft werden können. Somit werden die Abhängigkeiten von manuell generierten Unit-Tests und teuren Systemtests verringert.

Änderungsbasiertes Testen

Die Cantata Code Change Analyse automatisiert die Testpflege. Änderungen am getesteten Code, die Einfluss auf einen vorhandenen Test haben, werden identifiziert und Maßnahmen vorgeschlagen. Auf Wunsch werden die Testskripte automatisch an die Änderung angepasst.

Diagnose und Berichte



Cantata bietet eine leistungsstarke filterbare Diagnose von Testergebnissen und Codeabdeckungsergebnissen innerhalb der Eclipse GUI und flexible, vom Anwender konfigurierbare Berichte in XML und zertifizierungsfähigem ASCII-Text sowie HTML.

Team Reporting - Das Manager-Add-on

Cantata Team Reporting ist mit einer Client-Server-Architektur, einer Webschnittstelle und REST API ausgestattet. Das Tool zeigt den aktuellen Teststatus und die historischen Trends über mehrere Codeversionen hinweg.

Sie möchten mehr darüber wissen? Fragen Sie nach dem aktuellen Team Reporting Datenblatt!



Customer testimonials

"We were highly impressed with Cantata's track record in testing high integrity software in avionics, military applications and, of course, in medicine. It was the natural choice for us." **Urs Reidt, Research and Development Director**



"I've used Cantata for several years now to validate and test airborne safety-critical software. The tool has been great and quite simple in its full complexity."
"Cantata is easy to learn and use."

Johnny Johansson, Validation & Verification Tools Manager

"The systematic use of Cantata has enabled us to have the shortest unit test phase possible with great efficiency in terms of cost."

Philippe Lomazzi, Head of Software Development



"Through years of experience in unit testing using Cantata for various platforms and languages... we are confident in suggesting Cantata to our clients."

Padmakumar TV, Senior Engineering Specialist

"I was impressed with the capabilities in Cantata since it was able to support many testing levels (unit, integration and system testing)"

Pham Hong Son, Manager for Consulting Services



Bosch Engineering GmbH

"Module testing early during development is becoming more important due to the increasing complexity of software. In practice Cantata has proved successful and increased unit testing efficiency. Cantata from QA Systems offers an ideal solution for the creation and execution of unit and integration tests, including coverage analysis."

Matthias Schmidt, Testmanagement, Verification and Validation



Get a demo

Vereinbaren Sie einen Termin für Ihre persönliche Cantata Präsentation



Start free trial

Testen Sie Cantata mit einer kostenlosen Testversion. Alle Funktionen sowie technischer Support stehen Ihnen während der Testphase zur Verfügung.



Learn more

Für weitere Informationen besuchen Sie unsere Cantata Website.

www.qa-systems.de/cantata