

Développez, conformément aux normes de sûreté, vos logiciels C & C++ avec la solution Cantata pour les tests unitaires et d'intégration



En pointe sur la qualité des logiciels embarqués



# Pourquoi les leaders industriels utilisent Cantata?

# CANTATA

## Abaissier les coûts de certification



Cantata répond aux exigences de tests dynamiques exprimées par les différentes normes industrielles de sûreté des logiciels. C'est une solution tout intégrée pour les tests unitaires et d'intégration à la fois sur les plateformes hôtes et les cibles embarquées. Certifier un atelier logiciel, afin de démontrer sa conformité avec une norme de sûreté, peut avoir un coût très significatif pour une équipe de développement logiciel. Cantata est certifié de manière indépendante par l'organisme SGS-TÜV SAAR GmbH, et QA Systems fournit un kit de certification complet et pré-packagé, disponible gratuitement pour nos clients.

TÜV Certifié pour:

- › ISO 26262:2018 (Automobil)
- › EN 50128:2011 (Ferroviaire)
- › IEC 62304:2006 (Médical)
- › IEC 61508:2010 (Systèmes Industriels)
- › IEC 60880:2006 (Nucléaire)

Qualifiable pour:

- › DO-178B
- › DO-178C / DO-330
- › Autres normes sur demande

Comprendre les différentes méthodes et techniques à mettre en oeuvre afin que les logiciels soient conformes aux exigences des normes de sûreté, est un processus complexe et couteux en temps. Des indications détaillées sont fournies gratuitement sur la manière d'utiliser à bon escient les capacités avancées de Cantata, et ce afin de répondre à chaque exigence de test dynamique exprimée pour une norme donnée. La combinaison de ces indications et des fonctionnalités de Cantata vous permet d'aboutir à la conformité du logiciel vis-à-vis des normes de sûreté plus rapidement et à moindre coût.

## Contrôler et réduire les risques de défaillance logicielle

Les rappels de produits déployés sur le marché et l'impact en termes d'image et de réputation de l'entreprise peuvent excéder de loin les coûts de développement des applicatifs logiciels. Les tests unitaires constituent la technique de vérification la plus approfondie pour votre code, afin d'éviter d'embarquer des défauts dans les systèmes utilisés par vos clients. Les dépassements de planning de vos projets peuvent être compensés en déplaçant les efforts de vérification des logiciels très en amont des cycles de développement. Cette approche permet de réduire les risques de retards accumulés dans les phases ultimes de tests, car les modules testés de manière unitaire sont plus robustes et plus simples à intégrer. Les contentieux liés à la conformité quant à l'usage prévu constituent un risque croissant pour les industriels. Lorsque des sociétés ne font pas usage de pratiques reconnues pour la vérification de leurs logiciels, telles que les fonctions de tests unitaires proposées par Cantata, elles ne peuvent utiliser l'argument du respect de "l'état de l'art" lors de leur défense.

## Réduire le coût de vos tests

Tester au plus tôt pendant le cycle de vie du logiciel permet de réduire vos coûts en minorant les efforts de correction de votre code, qui seraient plus chronophages pour un problème découvert tardivement au cours du développement. Les développeurs peuvent identifier et corriger les défauts du code avec Cantata dès que leurs composants logiciels sont disponibles en phase de tests unitaires ou d'intégration.

Le coût élevé des tests unitaires et d'intégration, effectués dans le cadre de la conformité avec des normes industrielles de sûreté, peut être significativement réduit grâce à leur automatisation. Les exigences des normes de sûreté pour les tests dynamiques peuvent être couvertes plus rapidement et à moindre coût grâce à l'automatisation proposée par Cantata pour :

- › Les projets et harnais de tests
- › L'exécution sur machine hôte ou cible embarquée
- › La génération des cas de test
- › Tests de non-régression dans le cadre de l'intégration Continue
- › Harnais de non-régression pour vos anciens codes
- › La production des résultats et rapports de tests

L'intégration d'un outil de test dans un atelier logiciel complet peut entraîner des coûts cachés. L'intégration fine de Cantata avec les environnements de développement croisé ainsi que son interface utilisateur intuitive sous Eclipse® pour les tests C/C++, rendent facile son utilisation avec diverses chaînes d'outils. La courbe d'apprentissage de Cantata s'en trouve réduite, ce qui est la garantie d'une meilleure adoption par les équipes de développement.

## Accélérer la mise sur le marché de vos systèmes

Les grands acteurs industriels reconnaissent le besoin d'être réactif pour sortir de nouvelles gammes de produits sans pour autant mettre en péril leur qualité de fabrication. A ce titre, Cantata offre deux atouts majeurs aux responsables d'équipes de développement logiciel:

- › L'efficacité et la collaboration des équipes sont améliorées via la production de tests structurés et cohérents, et de rapports détaillés qui participent aux preuves exigées par les processus de certification.
- › Les cycles d'intégration logicielle sont réduits et plus prédictibles grâce des composants fiabilisés au préalable par les tests unitaires.

## Cantata highlights

- › Génération automatique des harnais et cas de tests
- › Support étendu et intégration des plateformes et environnements de développement
- › Interface Utilisateur Eclipse® intuitive gérant directement les scripts de tests en langage C/C++
- › Associations bidirectionnelles des tests avec vos exigences fonctionnelles
- › Contrôle des interfaces du code pour simuler/intercepter les appels
- › Injection modulable de code défini par l'utilisateur
- › Support du Développement Piloté par les Tests (TDD)
- › Analyse intégrée de la couverture de code
- › Automatisation des tests de non-régression
- › Maintenance des tests en fonction des modifications du code
- › Kit de certification gratuit pour la plupart des normes de sûreté

## Types de tests supportés par Cantata:

Boite Noire / Boite Blanche  
Classes Valides / Invalides  
Fonctionnel / Robustesse  
Unitaire / Charge  
Procédural / Orienté-Objet  
À partir du code source / des fichiers headers  
Bouchonnage / Interception des appels  
Isolation / Intégration  
Hôte / Cible Embarquée  
Création / Non-Régression

## Cantata est adapté à votre environnement

Cantata s'installe sous Windows® et Linux® sous la forme d'une interface utilisateur autonome développée sous Eclipse®, ou en tant que plug-in Eclipse® indépendant pour un environnement tierce-partie. Cantata supporte les compilateurs natifs GCC et Microsoft Developer Studio®, et peut également être intégré à de nombreux outils de développement logiciel croisés :

- |                        |                            |
|------------------------|----------------------------|
| ✓ OS temps-réel        | ✓ Intégration continue     |
| ✓ Compilateurs croisés | ✓ Gestion de configuration |
| ✓ Outils de débogage   | ✓ Gestion des exigences    |

Pour valider le support de vos sélections d'outils, veuillez contacter l'équipe QA Systems.

## Utilisation illimitée avec vos cibles embarquées

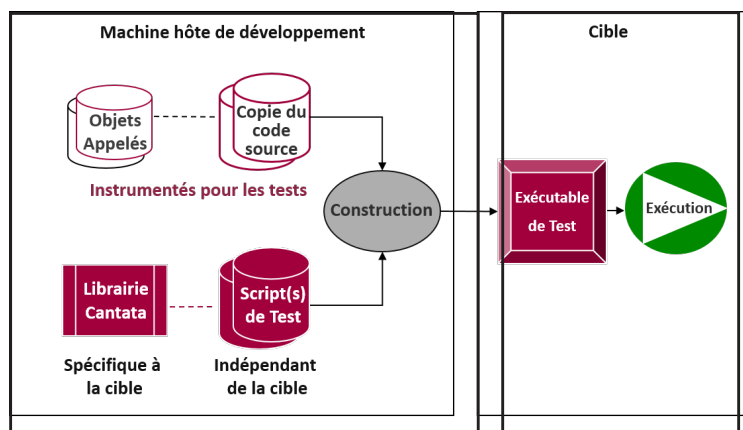
Pour déployer les tests sur vos cibles embarquées, un assistant intégré à Cantata fournit les moyens de configurer l'outil pour des déploiements sur cible illimités sans restriction de licence. Le set de déploiement Cantata consiste en une librairie run-time et des options de configuration compatibles avec votre code embarqué. Les déploiements Cantata sont testés et vérifiés pour un usage conforme dans le cadre de projets de sûreté soumis à la certification.

## Tests sur cible intuitifs et flexibles

Les tests Cantata (constitués de scripts C/C++ et de déploiements spécifiques à l'environnement cible) sont générés en tant qu'exécutables croisés C/C++ et chargés en mémoire sur votre cible de la même manière que vos propres applications. Ils peuvent donc s'exécuter avec votre :

- ✓ Simulateur    ✓ Emulateur    ✓ Carte électronique

Les résultats des tests fonctionnels et la couverture de code associée sont rapatriés depuis la mémoire de la cible vers la machine hôte afin d'y effectuer la génération des rapports et diagnostics. Ce processus est complètement automatisé par l'outil Cantata (en mode IHM ou en ligne de commande) via ses Makefiles, harnais de test, librairie de déploiement, et ce afin vous offrir des moyens simples et adaptables de vérification de vos logiciels embarqués.



Le code sous test est stimulé par les scripts de tests portables associés à la librairie run-time spécifique de la cible, le tout étant construit sous la forme d'un exécutable déployable dans l'environnement compatible de votre choix. L'instrumentation de code est utilisée pour les tests de type boîte blanche et pour la couverture de code. L'instrumentation Cantata est toujours effectuée sur une copie du code de production, qui n'est jamais modifié pendant les phases de tests.



## Un environnement de test flexible

L'environnement de test Cantata (constitué des scripts et de la librairie run-time associée) s'adapte à toute combinaison et style de test unitaire et d'intégration. La flexibilité est donnée aux développeurs de les éditer directement dans une vue C/C++ de l'interface graphique, ainsi que de les exécuter à la fois sur une machine hôte ou sur plusieurs cibles embarquées.

## Le Développement Piloté par les Tests

La fonction TDD peut être utilisée pour générer des cas de tests à partir du moment où les prototypes des fonctions sont disponibles dans des fichiers de déclarations. Cela permet de générer des cas de tests exécutables même si le corps du code source à tester n'est pas encore implémenté. Utiliser Cantata pour un processus TDD améliore le test unitaire de type boîte noire, tout en offrant l'accès aux fonctions de tests de type boîte blanche, car Cantata peut directement accéder à des éléments enfouis du code, tels que les données/fonctions privées et statiques.

## Les tests de type boîte noire et boîte blanche

L'environnement hautement automatisé de création des tests offre des moyens fins de génération de tests de type boîte noire et boîte blanche. Les tests boîte noire peuvent être enrichis avec des jeux de valeurs prédéfinies par l'utilisateur. Des effets combinatoires sont obtenus simplement via des boucles de tests alimentées par l'import/export CSV de vos vecteurs de tests. Des tests précis de type boîte blanche sont générés via l'instrumentation du logiciel afin d'accéder à des portions du code normalement inaccessibles par le script de test (fonctions et variables de type statiques et privées).

## Les tests de robustesse

La gestion des tests de robustesse peut s'effectuer via des jeux de valeurs prédéfinies pour les différents types de données de votre langage de programmation. Dans une optique de stratégie de test défensive, toutes les données globales sont automatiquement vérifiées lors de l'exécution des tests afin de détecter des modifications accidentelles.

## Le test orienté objet

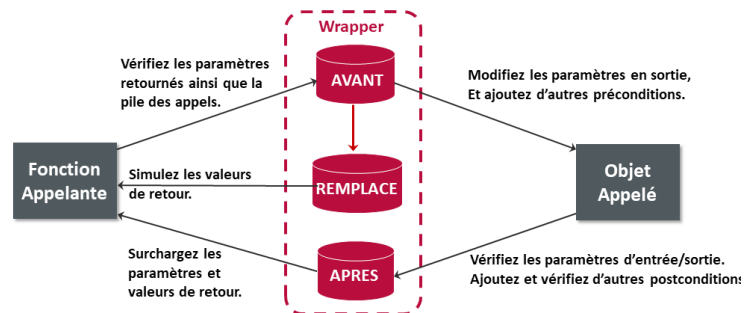
Les tests OO de Cantata sont implémentés sous la forme de classes pour vérifier vos méthodes, templates, et clusters de classes. L'automatisation couvre :

- › La réutilisation des cas de tests via les hiérarchies d'héritage
- › La réplique de la structure de test pour les classes dérivées héritées de celles des classes parents
- › L'implémentation concrète pour les classes abstraites ou les méthodes virtuelles
- › résolution des dépendances ou des références non définies, qui ne sont pas directement invoquées par le code

## Un système unique de contrôle des interfaces

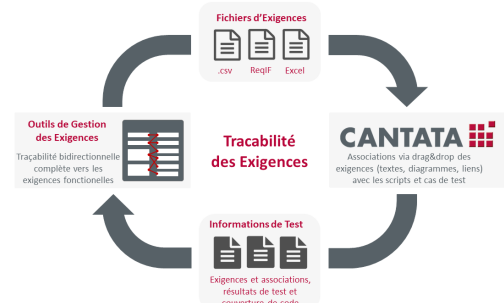
Tous les appels de votre code peuvent être simulés par des bouchons ou interceptés via les « Wrappers » générés par Cantata. Le contrôle des interfaces géré par Cantata offre :

- › Une vérification automatique des paramètres et valeurs de retour transitant à travers les interfaces
- › La création de différentes instances de comportement des wrappers en fonctions des cas de tests
- › Une vérification flexible des séquences d'appels pour chaque cas de test
- › Un mécanisme permettant d'injecter des fautes logicielles et de lever des exceptions
- › La gestion des tests de couplage des données/contrôle



Les Wrappers Cantata interceptent les appels pour vérifier le comportement réel (et non simulé) des objets appelés. Lorsque la simulation n'est pas possible ou souhaitable (appels systèmes, interfaces des OS et périphériques, etc.), les Wrappers constituent un moyen sophistiqué d'interception des appels avec les objets et librairies intégrés au harnais de test.

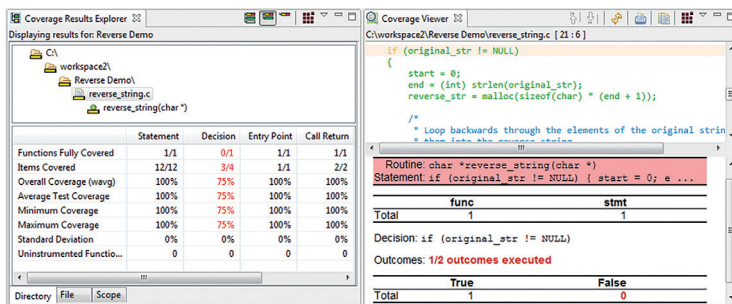
## L'association des tests avec les exigences fonctionnelles



La traçabilité bidirectionnelle des exigences logicielles vers les plans et cas de tests est requise par toutes les normes industrielles de sûreté. Grâce à Cantata Trace, vos exigences ou plans de tests sont importés via le serveur Cantata (à partir de fichiers CSV, Microsoft Excel®, ou ReqIf), puis associés avec les scripts et cas de tests à travers une interface graphique intuitive intégrée à Eclipse®. Après l'exécution des tests, les résultats générés par Cantata sont réexportés vers votre solution de gestion des exigences (e.g. Excel®, IBM® Rational® DOORS®, PTC Integrity®, Polaron® REQUIREMENTS™, Intland codeBemer®).

## La couverture de code

L'analyse de la couverture de code est une mesure objective de l'impact des tests sur l'exécution du code source (sous la supervision de Cantata ou d'une autre source d'exécution des tests). Des ensembles prédéfinis de règles de couverture, pour les différentes normes industrielles de sûreté, permettent en toute simplicité à l'utilisateur de Cantata : d'automatiser l'instrumentation de son code, de récupérer les traces de couvertures, et enfin de vérifier l'atteinte des niveaux de couvertures prescrits pour le projet.



La couverture de code peut se mesurer en utilisant les métriques suivantes :

- › Points d'entrée
- › Retour des appels all
- › Instructions
- › Blocs
- › Décisions (branches)
- › Conditions
- › MC/DC (Masking et Cause Unique) C/DC
- › Boucles
- › Opérateurs relationnels

La fonction de Couverture de Code pour les Variantes de Configuration Logicielle supporte l'agrégation des données de couverture collectées à travers plusieurs processus de build, fournissant des résultats et rapports certifiables pour vos variantes de code source.

Les arbres de couverture de code dans les projets Cantata donnent une vue synthétique de la complétude des différents niveaux de couverture, en liaison avec les cas de tests et leurs contextes d'exécution (tâches, états, héritage, etc.). L'optimisation automatique du script de test permet de désactiver les cas de test ne contribuant pas à une augmentation du niveau de couverture. Pour de gros volumes de données, cette fonction permet d'optimiser le temps alloué aux tests de non-régression.

## Le test et l'intégration continue

Les Makefiles de Cantata sont générés automatiquement à la suite des scripts de tests, afin de pouvoir compiler, linker, déployer, exécuter et récupérer les résultats finaux en mode ligne de commande. Ils peuvent donc être utilisés avec vos propres Makefiles, et peuvent simplement s'interfacer avec les solutions open-source ou commerciales pour l'intégration continue.

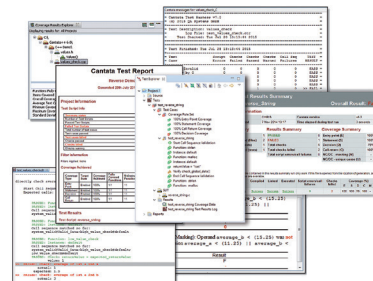
## La fonction AutoTest

La fonction AutoTest de Cantata permet de générer automatiquement un ensemble de scripts de tests unitaires s'exécutant sans erreurs. Les vecteurs calculés par AutoTest ont pour objectif d'atteindre 100% de couverture de code pour le niveau sélectionné par l'utilisateur, tout en vérifiant les données, paramètres et séquences d'appel des primitives testées. AutoTest peut s'utiliser dans une optique de non régression pour d'anciens codes dont les exigences et/ou tests ont disparus, et également comme un moyen efficace d'augmenter la couverture de code obtenue à l'issue des tests fonctionnels (y compris pour les niveaux contraignants tels que les MC/DC).

## Analyse des Changements du Code

L'analyse des changements du code permet de réduire les coûts de maintenance des tests logiciels. Les modifications du code impactant les tests existants sont identifiées et font l'objet de suggestions de mises à jour automatiques des scripts de tests.

## Les diagnostics et rapports



Cantata produit des résultats de tests et de couverture détaillés qui peuvent être filtrés et configurés par l'utilisateur au travers de l'interface graphique sous Eclipse®. Ils sont configurables aux formats XML, HTML, et ASCII pour les contextes de certification.

## L'extension optionnelle: Team Reporting

L'extension Team Reporting pour Cantata, basée sur une architecture client-serveur, offre à la fois une interface web et REST, afin de permettre aux équipes distribuées de travailler de manière plus étroite et efficace. Référez-vous à la fiche technique de Team Reporting pour découvrir cette extension plus en détail.



## Les Témoignages de nos clients

"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**



**SAAB**

"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**

"Speaking conservatively, this product has probably paid for itself twice over already!"

**John Duckett, Special Projects Manager**



## 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**



### Obtenez une démonstration

Contactez-nous pour organiser une réunion sur site ou via internet pour démontrer l'adéquation de Cantata à vos exigences de test.



### Démarrez une évaluation gratuite

Évaluez gratuitement Cantata avec votre propre code. Essayez le produit complet, y compris le service de support technique, durant votre phase d'évaluation.



### Découvrez encore plus

Visitez le site internet de Cantata pour obtenir plus d'informations.

[www.qa-systems.com/tools/cantata](http://www.qa-systems.com/tools/cantata)