

# 单元集成测试工具 Cantata

## 为什么产业领导者使用 Cantata?



### 降低遵从标准的成本

Cantata 满足软件安全标准的动态测试要求。它是主机和目标平台上的动态单元和集成测试为一体的解决方案。

开发工具的认证可能是一项沉重的成本负担。Cantata 已经通过了 SGS-TÜV SAAR 有限公司的独立认证，并提供了一个免费、开箱即用的认证工具套件。

了解如何遵守软件安全标准是复杂和耗时的。Cantata 免费提供全面的指导，使用强大的功能集，满足特定标准下每个动态测试的要求。使用 Cantata 这种指导和功能相结合的工具，能够更快、更廉价地符合示范标准。



TÜV 认证:

- › ISO26262:2011(汽车)
- › EN50128:2011(铁路)
- › IEC62304:2006(医疗)
- › IEC61508:2010(通用工业)
- › IEC 60880:2006(核电)

符合:

- › DO-178B
- › DO-178C/DO-330
- › 其他标准

### 降低软件故障的风险

产品召回和广泛的品牌、企业声誉的影响，可以远远超过个别应用程序的开发成本。单元测试是测试应用程序代码和防止传输设备中错误的最彻底方法。

通过将验证工作转移到软件开发生命周期的最早阶段，可以缓和项目超负荷运行。这样降低了在之后测试阶段中的延迟风险，因为已测试单元的组件更容易、更可预测地集成。

现在对公司和个人的针对性诉讼面临越来越大的风险。如果公司没有采用公认的行业惯例，例如通过 Cantata 进行彻底的单元测试，他们就不能使用“最先进的”法律辩护这样的诉讼。

### 降低测试成本

早期测试通过在开发生命周期中最小化代码重来降低成本。开发人员可以在每个组件可用时，立即识别 Cantata 在单元和集成测试中检测出的软件缺陷。

通过自动化可以显著降低标准，兼容单元和集成标准测试的高成本。Cantata 安全标准的动态测试要求更快、更廉价，通过自动化：

- › 测试框架生成
- › 测试用例生成
- › 主机或目标机上的测试执行
- › 结果诊断和报告生成

将工具集成到工具链中可能会增加隐藏的测试成本。Cantata 与交叉编译环境的紧密集成以及在 Eclipse®GUI 或代码编辑器中直观的 C / C ++代码测试，可以轻松插入任何工具链。这些集成降低了学习工具的时间并加快了测试活动，降低了整体测试成本。

### 缩短上市时间

产业领导者认识到在不危及质量的前提下，需要更快地发布产品。Cantata 测试为开发经理提供了两个关键的时间优势：

- › 通过结构一致的测试和认证准备报告，提高团队协作和效率。
- › 当集成经过验证的软件组件时，集成时间更短，更利于预测。

## Cantata 兼容您的工作环境

### Cantata 的亮点

- › 自动化测试工具和测试用例创建。
- › 广泛的平台支持和工具链集成。
- › 易于使用的Eclipse®GUI 和用C/C++编写的测试用例。
- › 双向需求可追溯性。
- › 独特的接口调用控制来模拟和拦截调用。
- › 集成代码覆盖分析。
- › 自动回归测试。
- › 所有主要安全标准的免费工具认证套件。

Cantata 可以安装在 Windows®和 Linux®主机操作系统上，兼容 Eclipse®IDE 或作为 Eclipse-Ready®插件集。它支持 GCC 和 MicrosoftDeveloperStudio®编译器，并且还集成了大量的嵌入式开发工具链：

- |               |           |
|---------------|-----------|
| ✓ IDEs /RTOSs | ✓ 构建/连续集成 |
| ✓ 交叉编译器       | ✓ 软件配置管理  |
| ✓ 调试器         | ✓ 需求管理    |

### 无限制的嵌入式目标使用

对于目标环境，内置向导提供无限制的部署和使用，无需任何许可限制。Cantata 配置包括与代码兼容的库，配置选项和二进制文件。这些都是经过测试和控制的，用于安全相关项目的认证使用。

### 在目标上简单灵活的测试

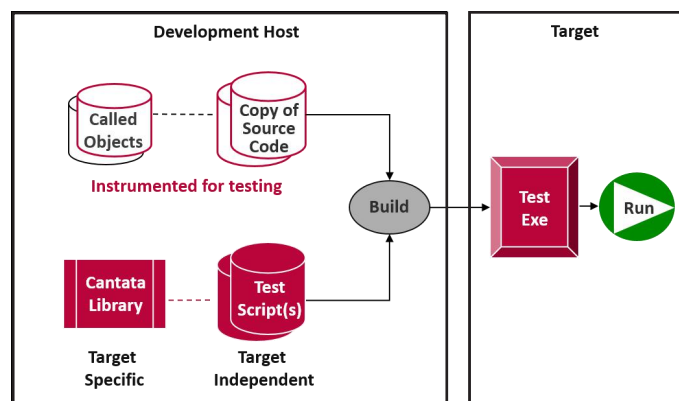
Cantata 测试（平台无关测试脚本在 C/C++和平台特定部署）构建为 C/C++可执行文件，在目标平台上加载运行，就如同在以下位置运行自己的代码一样：

- |       |       |         |
|-------|-------|---------|
| ✓ 模拟器 | ✓ 仿真器 | ✓ 物理目标板 |
|-------|-------|---------|

功能和代码覆盖测试结果直接返回给主机，得出诊断和报告。该过程使用 Cantata 完全自动创建 makefiles、测试脚本、平台定制，方便灵活地从 GUI 或 CLI 完成目标测试。

### 支持的测试类型：

- 黑盒/白盒
- 正面/负面
- 需求/健壮性
- 单一/大量输入数据集
- 过程/面向对象
- 调用模拟/拦截
- 隔离/集成
- 主机/目标机执行
- 新的/回归



用户代码由具有目标兼容库的便携式测试脚本驱动，并且构建为单个测试可执行文件，在多个主机或目标平台上运行。插桩用于白盒访问和代码覆盖，因此无需为测试修改产品代码。

## 灵活的测试框架

灵活的测试框架（测试脚本和支持库）支持单元和可扩展的集成测试的任何测试样式组合。测试可以在 GUI 中直接编辑为 C / C++ 代码，在嵌入式目标平台和主机平台上作为可执行文件运行。

## 黑盒和白盒测试

高度自动化测试用例的创建，提供了黑盒测试的功效和精度，以及更有效的全面的白盒测试。通过用户选择或预定义的参数化循环测试，针对大型数据集的组合效果计算器和 CSV 导入/导出，可以实现强大的黑盒测试。通过 Cantata 插桩的白盒测试，直接从测试脚本自动访问封装代码，无需条件编译，还可以对静态、私有函数以及数据进行控制。

## 鲁棒性测试

鲁棒性测试能够很容易地使用 Cantata 规则集对基本数据类型值进行预定义，以循环测试用例的方式进行。自动检查所有可访问的全局数据是否在无意中发生更改。

## 面向对象测试

Cantata 面向对象测试由测试方法、模板或类的集群来实现。它们具有自动化：

- › 通过并行继承层次重用测试用例
- › 针对继承类的测试类继承结构
- › 具体实现抽象基类（ABCs）或纯虚方法（PVMs）

为了打破类依赖链并使 C++ 隔离测试变得容易，Cantata 自动发现并解析未被测试软件直接调用的未定义引用的依赖关系。

## 自动测试

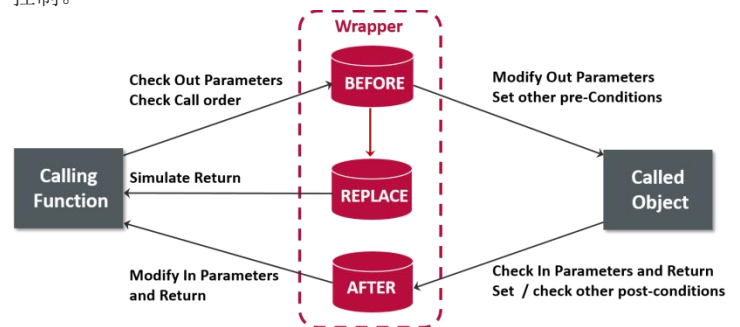
Cantata 自动测试生成完整的、可通过的单元测试脚本。测试用例向量可以达到 100% 的代码覆盖，同时还检查数据、参数和调用顺序。自动测试创建回归单元测试的安全网，以减少对高额系统测试的依赖，并自动关闭代码覆盖的缺口。

## 独有的调用控制

Cantata 自动生成测试控件，模拟（桩）和拦截（封装）来自被测软件的所有函数调用，提供：

- › 可选的自动检查参数和数据
- › 针对不同调用行为的多个实例
- › 每个测试用例中灵活的调用顺序验证
- › 接口错误检测和错误注入
- › 控制耦合测试

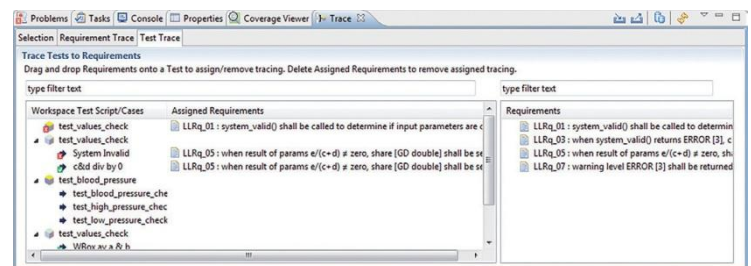
封装器拦截调用以验证被调用对象的实际或模拟行为，而不是假定行为。在模拟不可能或不可取（集成的内部调用，操作系统调用，硬件接口等）时，封装器提供了强大的调用控制。



## 需求追踪

所有安全相关标准都要求需求或测试计划到测试用例的双向追踪。

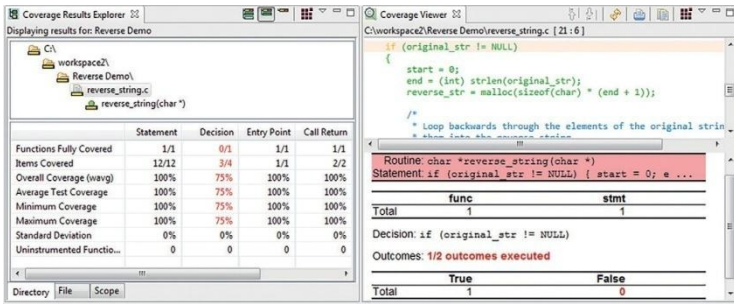
需求或测试计划集被导入到 Cantata 服务器，通过拖放界面分配关系，然后导出 Cantata 测试结果状态和代码覆盖信息。



可以通过 CSV、Microsoft Excel® 或需求交换格式（ReqIf）导入/导出，满足工作流和特定的需求管理工具版本（例如 IBM® Rational® DOORS®、PTC Integrity®、Polarion® REQUIREMENTS™）。Cantata 服务器也有一个完整的 REST API。

## 代码覆盖率

Cantata 代码覆盖提供了关于如何彻底执行源代码测试的客观测量（无论是否由 Cantata 进行测试）。自动插件，数据报告，以及对所需代码覆盖率级别的集成检查，可以易于使用特定标准的 Cantata 覆盖规则集。



代码覆盖率使用以下度量衡量：

- › 入口点
- › 调用返回
- › 语句
- › 基本块
- › 判定（分支）
- › 条件
- › MC / DC（Masking 和 Unique Cause 变体）
- › 循环
- › 关系运算符

针点型（Pin-point）诊断可以过滤或聚合完整项目代码树的覆盖范围，通过测试用例、测试运行、度量类型和代码前后关系（继承、线程、状态、数据耦合等），下拉到单个代码结构，进入每行代码中。自动测试用例优化有助于从大型数据集中选择测试用例向量，并减少回归测试消耗。

## 持续测试

Cantata 在测试创建时自动生成 Makefiles，以批处理的方式对测试套件进行编译、链接、配置、执行和获取结果。它们可使用现有的 Makefiles，易于与开源或商业持续集成工具集成。

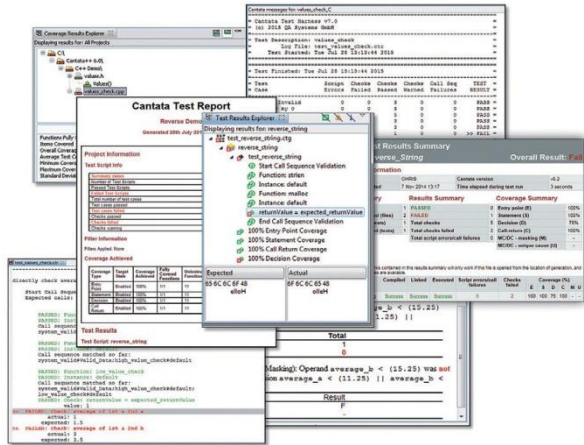
## 同步静态测试

Cantata 与 QA-C 和 QA-C++ 静态分析工具集成，保持动态和静态测试进程同步。



在安装了 Cantata 和 QA Framework 的情况下，Cantata 工作区中更改过的源代码可以自动触发 QA-C / QA-C++ 重新分析，可以立即知道这些更改是否符合静态分析要求。

## 诊断和报告



Cantata 在 Eclipse®GUI 中提供了强大的可过滤的功能，诊断测试和代码覆盖结果，灵活的用户可配置的 XML、HTML 格式报告和准备认证的 ASCII 文本。

## 可选插件：Team Reporting

Cantata Team Reporting 插件，使用客户端-服务器架构，Web 界面和 REST API，使分布式团队能够更有效地工作。





## 客户反馈

“我们对 Cantata 在航空电子设备，军事应用以及医学领域测试高完整性软件的良好记录印象深刻。这是我们很自然的选择。”

**UrsReidt, 研发总监**



**SAAB**

“几年来我已经使用 Cantata 验证和测试机载安全关键软件。这个工具是伟大的，化繁为简。”“Cantata 易于学习和使用。”

**Johnny Johansson, 检测工具经理**

“使用 Cantata 的系统使我们能够在成本方面具有最高的效率，具有最短的单元测试阶段。”

**Philippe Lomazzi, 软件开发主管**



“通过多年使用 Cantata 的各种平台和语言的单元测试经验...我们有信心向我们的客户建议 Cantata。”

**Padmakumar TV, 高级工程专家**

“保守地说，这个产品可能已经值回了双倍的花费！”

**John Duckett, 特别项目经理**



**Bosch EngineeringGmbH**

“由于软件越来越复杂，开发过程中的模块测试越来越重要。在实践中，Cantata 已经被证明成功地提高了单元测试效率。QA Systems 的 Cantata 为创建和执行单元或集成测试（包括覆盖分析）提供了理想的解决方案。”

**Matthias Schmidt, 测试经理, 检验和确认**

