

White Paper:
Why Pay For A Unit Test Tool?



White Paper

Why Pay For A Unit Test Tool?

QA Systems GmbH
Schwieberdinger Straße 56
70435 Stuttgart
Germany

Tel +49 (0)711 / 13 81 83 - 0
Fax +49 (0)711 / 13 81 83 - 10
E-Mail info@qa-systems.de
www.qa-systems.de



QA Systems Ltd
2 Palace Yard Mews
Bath BA1 2NH
United Kingdom

Tel +44 (0) 1225 321 888
Fax +44 (0) 1225 581 150
E-mail: info@qa-systems.com
www.qa-systems.com

Contents

Why Pay For A Unit Test Tool?	1
1 Introduction	3
2 Opportunity Costs of Open Source Tools	4
2.1 LACK OF PROCUREMENT OVERHEAD	4
2.2 EVALUATION	5
2.3 EASE OF USE	6
2.4 CONFIGURABILITY AND INTEGRATION	7
2.5 TECHNICAL SUPPORT AND SECURITY	8
2.6 TOOL UPDATES AND IMPROVEMENTS	9
2.7 BEYOND THE BASICS	10
2.8 STANDARDS COMPLIANCE	11
3 Conclusion	12

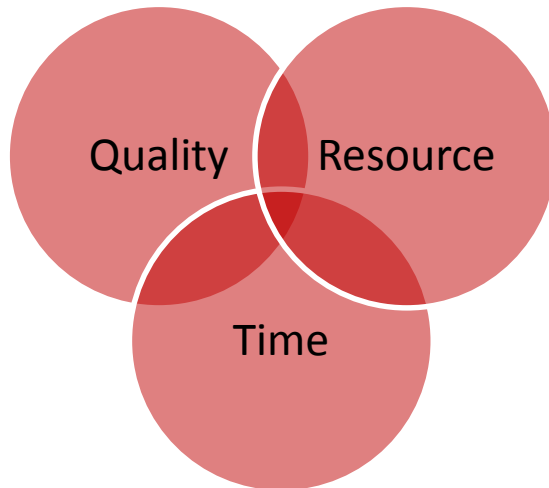
Copyright Notice

Subject to any existing rights of other parties, QA Systems GmbH is the owner of the copyright of this document. No part of this document may be copied, reproduced, stored in a retrieval system, disclosed to a third party or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of QA Systems GmbH.

© QA Systems GmbH 2014

1 Introduction

Business decisions are characterized by a triangular trade-off between the following.



Companies try to balance the need for device standards compliance with ever-shortening time to market and constant cost pressures. These considerations are equally important for software development tools as they are to the software development process.

This paper does not address the rationale for doing unit testing or integration testing as part of the software development lifecycle, because the compelling case for that is well documented elsewhere. Having made a decision to do unit / integration testing, device developers consider whether to go the open source or commercial vendor route for tools. This paper addresses the question “Why pay for a unit test tool?” by considering the Quality / Resource / Time opportunity costs of choosing an open source tool, and sets out the compelling case for investing in commercial tools.

There are numerous open source tools currently available, the most widely used of which for C/C++, are CppUnit, CxxTest and NUnit. As these tools are very similar, this paper does not distinguish between them.

There are a handful of vendors which supply commercial unit testing tools for C/C++. This paper addresses issues common to commercial unit test tools and adds specific details for QA Systems' Cantata.

2 Opportunity Costs of Open Source Tools

The opportunity cost of a decision is the value of the best alternative foregone. This section of the paper examines the Quality / Resource / Time opportunity cost of choosing an open source C/C++ unit test tool over a commercial tool, and addresses the key arguments made by open source advocates.

2.1 Lack of Procurement Overhead

Because an open source tool can be easily downloaded from the web and does not need to be purchased, the acquiring user does not have to deal with procurement overheads. Such overheads for commercial tools typically include staff resource effort to assess of and negotiate on price and license use terms and conditions and, to a limited extent, time - dependent on the responsiveness of the tool vendor.

The advantages of bypassing a procurement process may seem efficient at first. However, the process with commercial tool vendors serves three useful business functions which impact both resource costs and the quality of device software in the short term, and time in the longer term.

1. It allows prospective tool users to understand (and by negotiation control) the prices of tool licenses and any configuration / customisation costs as well as levels of technical support, future tool fixes / upgrades, training and other services from the vendor. This clarity helps the business understand the total cost of ownership (TCO) of commercial tools before a selection decision is made. By contrast the TCO of open source tools is very hard to identify.
2. Because procurement requires authorisation, it allows management oversight and approval of tool selection decisions, which will affect a range of otherwise unrecognised costs of tool adoption. These can include tool set-up, end user learning curve and productivity, on-going tool availability and support, fixes and upgrades, as well as license compliance.
3. Through vendor qualification, prospective tool users develop confidence in the tool vendor as a supplier of both tools and related services, in the immediate acquisition period and for the long term supplier relationship.



The pricing and licensing for Cantata and related services are simple and straightforward. Cantata is a single tool with a comprehensive all inclusive set of capabilities with no hidden extras. This enables business management to understand the tool TCO during a procurement process. QA Systems, founded in 1996, are pleased to demonstrate in any vendor qualification, our enviable reputation as *The Software Quality Company*, and our long term commitment to working with customers to accelerate and improve their software development.

2.2 Evaluation

As acquisition of open source tools is mistakenly considered to not incur costs, several important considerations may be overlooked, and therefore not accounted for, during their evaluation when compared to commercial tools. The opportunity cost in terms of time saved and resources expended during an evaluation of commercial tools can include:

1. The tool technology for unit / integration testing has moved on very significantly from the simple black-box pure unit isolation testing over the last decade or so. While open source tools provide the basic functionality (often by combining several open source tools together), commercial tools differentiate themselves by offering more features, functionality and richer user experiences. Commercial vendors will also seek to inform prospective users of these advantages during an evaluation. The opportunity costs for the business of missing out of these considering applicability of these capabilities in their tool requirements checklists, can have major time, resource and quality impacts during eventual tool use.
2. Open source tools have limited user documentation and what is available is often scattered over many locations. Prospective users may spend significant time exploring open source tools to see whether they match the intended use cases. By contrast commercial vendors provide extensive tool documentation, supporting materials such as video and of course technical support to assist in evaluation of their tools.
3. While most open source tools can be configured to operate in the prospective users technical build environment, such configuration activity is undertaken by the business themselves, often at considerable resource and time cost. Engineers can spend valuable time to bundle together a set of open source tools, with limited documentation spread over various websites and blog posts just for an evaluation. By contract commercial vendors will have pre-configured their tool for evaluation.
4. In a similar way to procurement overheads, where engineers may believe they can by-pass the purchasing process by selecting an open source tool, there are opportunity costs of valuable management oversight and approval of tool selection decisions based on business requirements (rather than developer preferences), which will affect a range of otherwise unrecognised costs of tool adoption. These can include tool set-up, end user learning curve and productivity, on-going tool availability and support, fixes and upgrades, as well as license compliance.



Cantata can be evaluated on standard supported host Windows and Linux or configured in the users own environment, free of charge. Cantata's powerful capabilities for more intelligent testing are clearly defined in the comprehensive documentation and supporting materials. QA Systems appreciate your valuable time and provide full technical support during an evaluation phase. We are also pleased to offer the benefit of our extensive experience in organising a successful evaluation phase and subsequent tool adoption plan. A free evaluation of Cantata can be requested from sales@qa-systems.com.

2.3 Ease of Use

Open Source Tools, by definition, are not in the market to generate a profit. As such they are constructed more in line with the wishes of the volunteer developers designing it, rather than those of safety and business critical end users. Less attention is therefore paid to improving the user-interface, making open source tools less user-friendly and harder to use, especially in comparison with commercial tools.

Commercial Tools are specifically designed to address the safety and business critical technical needs of end users, whilst providing ease of use and powerful diagnostics and reporting to make unit / integration testing as efficient as possible. If the commercial tool was not offering significant functional and productivity advantages over open source tools, with ease of use at their core, then they would not be able to demonstrate the return on investment required to justify their price. The ease of use opportunity costs of selecting open source tools include:

1. Enhanced end user productivity in the resource hungry unit and integration testing phases of device development, can have substantial impacts on testing time and budget.
2. Because commercial vendors want their tools to remain in use (not a consideration for open source tool developers), they want their tools to be easy to use. Foregoing commercial vendor commitment to ease of use, removes leverage businesses have over vendors continually improving their tool and support for it.
3. Any new tool adoption process has the potential to fail to deliver the expected business benefits, where there is resistance to change. Ease of use for a unit testing tool is critical to developers' acceptance of the activity.
4. Ease of use relates not only to the user interface but includes how well the tool is integrated into the end users own environment, the user documentation / supporting materials, and technical support / training available. Where open source tools are selected, much of what can make unit test tool easier to use has to be hunted for on the internet, or provided in-house taking time and resource away from the core business and at uncertain quality.



Ease of use is the core principle behind Cantata. It guides all customer interaction and the Cantata product roadmap. Cantata's ease of use and intelligent testing capabilities, enable end users to dynamically prove their safety or business critical code in the most efficient manner possible. Cantata provides a complete test development and analysis environment built on Eclipse. It easily integrates with the developer desktop in their normal build environment, lowering learning curve and providing rich ecosystem plug-and-play compatibility.

Cantata comes with comprehensive user documentation including: Quick Start Guide, Installation Guide, User Guide, Deployment Guide, Safety Manual, Tool Manual (fully hyperlinked), Context Sensitive Help (Within the Tool), Testing Standards mapping briefings, Certification and Qualification Kits, 'how-to' videos, technical notes, and is supported with a range of training and specialist consultancy all designed with ease of use and resulting end user productivity in mind.

2.4 Configurability and Integration

A common argument used in favour of open source tools, is that they can be easily modified for a company's individual requirements. This is because end users have access to the tool source code and can therefore, if so inclined, modify and re-build it from the ground up.

This flexibility to configure an open source tool for a specific environment, modify functionality or integrate one open source tool with others in the tool chain (e.g. for stubbing, mocking, code coverage etc), however also has opportunity costs compared to commercial tools.

1. The obvious cost is in time and resources to roll-your-own variant of an open source tool. By contrast it is the core business of commercial tool vendors to create and supply tools where they undertake configuration and integration of their tools. End users of commercial tools benefit from the economies of scale where the cost of such configurations and integrations are spread over many customers. When end users configure and integrate open source tools, themselves they bear the whole cost themselves.
2. Where open source tools or tool chains are constructed by end users, the behaviour and quality of the tool(s) may not be correct. While some open source tools have test suites, these are far from exhaustive and for businesses to have assurance that open source tools are set up correctly for safety or business critical testing can be prohibitively expensive, so quality is usually sacrificed. By contrast commercial vendors have an incentive to test their configurations and integrations, so not impair the reputation of their tool.
3. Modifications to open source tools by individual end users can jeopardise what technical support is available from open forums technical support. Commercial vendors will support what they sell.
4. Where end users do attempt integrations of open source tools, these are not always with other open source tools in the development tool chain. There is an opportunity costs in time and resources for end users interaction with other vendor that is borne by commercial unit test tool vendors.



Cantata is highly configurable and integrated with a wide range of compilers, IDEs, RTOS's, build and other tools supporting the development lifecycle. Built on the open Eclipse platform, Cantata integrations provide a rich ecosystem plug-and-play compatibility. Cantata includes a self-configuration capability (Cantata Builder) to deploy to and automatically verify embedded target deployments. Many end users also configure Cantata to integrate with their own unique build environment of tool chain, and QA Systems are pleased to offer support and consultancy for these. For more information on Cantata's flexibility and configurability, please see our customer case studies.

2.5 Technical Support and Security

A perceived benefit of open source tools is the open source community peer level technical support provided free of charge. What seems a clear cost saving does have less obvious opportunity costs:

1. Clarity and accuracy of technical support answers from multiple open source forums and blog posts varies widely and can create confusion. Another unhelpful characteristic of such crowd sourced technical advice is that consistency exponentially decreases as the number of participants increases. For any one question, an end user may receive multiple answers that are misinformed or contradictory. By contrast commercial vendors have dedicated tool technical support services, and build up a clear body of knowledge for end users. Unclear or incorrect technical support can cause time delays and result in lower quality tests.
2. Response times from the open source community depend on availability of existing information or created when the community is active. Volunteers are not paid like commercial vendors to be responsive. Getting stuck waiting for answers for unit tests can create time delays for activities further down the lifecycle or result in gaps affecting quality.
3. Open source tools are characterized by their uncertain future, as they are developed by volunteers who often lose interest or involvement in the tool. Continuity of technical support service contracts, with planned tool support deprecations from commercial vendors, by contrast permits businesses continuity planning by end users.
4. The security of end user code / data cannot be protected by non-disclosure agreements as with commercial vendors, so there can be opportunity costs of abstracting code under test for open community support.
5. Finally, open source tool forums do not bear any commercial accountability or liability for clarity, accuracy, response time, continuity or security of the technical support. Relying on these through good faith, the opportunity costs of failures are borne by end users.



Technical support services for Cantata are clearly laid out in all commercial offers. QA Systems' dedicated technical support is provided over phone, email and webinar, by engineers actively involved in on-going development of the tool. Technical support enquiries are immediately responded to and allocated tracking numbers and typically dealt with within 24 hours. Cantata was initially released in 1992; no version of Cantata has been deprecated, and many customers have support contracts lasting for many years. Customers can supply their own code in confidence through NDAs.

QA Systems also provide on-site training courses and consultancy to ease tool adoption and kick-start end users productivity gains with Cantata.

QA Systems are proud of and committed to the Cantata technical support and training services we and our resellers provide. However do not just take our word for it, see our customer case studies or try it for yourselves.

2.6 Tool Updates and Improvements

Open Source supporters maintain that since there are numerous users working on the tool, it will be of higher quality, more secure and less prone to bugs than proprietary systems. This argument severely underestimates the issues that are bound to arise when too many ‘cooks’ are involved in the production of any one thing, and does not recognise that commercial vendors have a greater imperative – competition. The opportunity costs of relying on the open source community for tool updates and improvements include:

1. A lack of focus and direction arising from a collective volunteer approach and the requests if a majority or non-safety or business critical users for changes. By contrast commercial vendors actively seek to address these market demands (including technology and regulatory changes), and seek to differentiate themselves from both open source and other commercial tools.
2. Lack of commitment to open source tools happens over time as voluntary contributors reach a point where they are content with their work or, simply, lose interest and cease to propose new ideas or remain involved. Once development reaches this plateau, users may find themselves stuck with orphan software that is no longer improving. Commercial vendors on the other hand are invested in their technology, and have the incentive of future revenues from existing and prospective customers in a competitive environment to protect.
3. Open source tools cannot be held accountable for any product road map which they may publish, so end users are reliant on the volunteer project to decide when and what updates and improvements to the tools will be published. Commercial vendors can and some do commit to a product roadmap, addressing fixes, and improvements.



With every update of Cantata, QA Systems provides its customers with a ‘What’s New in version X’ document, accompanied with detailed release notes. In addition to bug fixes and enhancements, recent Cantata updates have included new features and capabilities that make the practice of unit testing more efficient. For example, the highlights of Cantata 6.2 Update A included:

- > **AutoTest for embedded code** generation engine added the leading EDG (Edison Design Group) C/C++ parser, enabling Cantata to automatically test a much wider range of embedded target code including support of non-standard language features.
- > **Out of the box embedded target configurations** were added, fully integrated with a wide range of commonly used cross-compiler IDEs, in-built pre-configured target libraries reduce Cantata deployment within these environments to just a few mouse-clicks.

A Cantata product roadmap can be requested from sales@qa-systems.com under an NDA.

2.7 Beyond the Basics

There are many open source tools available for C/C++, but all of them share a heritage in Kent Beck's JUnit framework. While JUnit is an excellent tool for unit testing Java, the C and C++ languages have much greater flexibility to program badly. The underlying flexibility of these languages and their diverse configuration in the heterogeneous environments of embedded systems commonly used in safety and business critical sectors, means the open source tools offer only very basic functionality. For the open source tools, this means being constrained by inefficient test approaches (constrained by TDD styles of unit test), or supplementing CxxTest, CppUnit, etc with other tools to deal with call interface simulation, code coverage etc, but usually both.

Commercial C/C++ unit test tool vendors have all gone way beyond the basic functionality of these open source tools. The commercial tools need to offer more in all three areas of the business trade off because they also compete with open source tools:

- Quality – both in the tool itself, and in the levels of quality the software unit / integration testing can generate in the code being tested,
- Time – in tester productivity using the tools,
- Resource – in return on investment for end users.

All commercial tools offer different capabilities, and this paper does not seek to summarise or compare them.



The main areas where Cantata offers far more than the basics available with open source tools, what we refer to as "Intelligent Testing" are highlighted below.

- > 300+ static code metrics, to aid test planning.
- > Eclipse GUI and comprehensive IDE integrations
- > AutoTest generation of complete passing unit tests for C exercising up to 100% MC/DC coverage
- > Automatic programmable call interface control for simulation and interception
- > Automatic white-box test case generation for static and private functions/data
- > Parameterized test case construction, by user selection or data type
- > Automatic OO parallel test case re-use, ABC test construction and template instantiation
- > Integrated code coverage for all standard metrics plus others, with automatic test case optimisation
- > Automated regression test framework
- > Powerful drill-down test diagnostics and configurable reporting
- > Tool certification

For more information on Cantata is available from www.qa-systems.com/cantata or requested from sales@qa-systems.com.

2.8 Standards Compliance

Of all the opportunity costs associated with open source tools, the most compelling arises for end users planning to do unit testing to comply with regulatory standards. All the main safety related standards impose some means by which the quality of the test tools used is demonstrated to achieve certification credit for the device software.

Open source tools can be qualified or certified as suitable or useable in a safety related context. However the costs associated with such an exercise (defining requirements for the tool, developing and running tests against those and demonstrating the tool operates correctly in the end users environment), are prohibitive. It is for that reason that developers of safety critical devices just do not do it.

Most commercial unit test vendors have sought and achieved, some independent certification of their tools, or they support users qualification of their tools (where this is required e.g. by DO-178B/C).



Cantata has been certified by SGS-TÜV Saar GmbH, an independent third party certification body for functional safety, as “usable in development of safety related software”, up to the highest safety integrity levels, for all the main safety related standards:

- > IEC 61508:2010 (general industrial)
- > ISO 26262:2011 (automotive)
- > EN 50128:2011 (railways)
- > IEC 60880:2006 (nuclear power)
- > IEC 62304:2006 (medical devices)



Cantata also has a comprehensive tool qualification kit available under an NDA for DO-178B/C.

QA Systems believe as a vendor should ease your route to device certification, so available free of charge for Cantata is a comprehensive tool Certification or Qualification Kit including:

- > Detailed standard briefing – mapping the requirements to Cantata capabilities
- > Cantata Safety manual – providing guidance for how to use the Cantata tool in a safety related context, and to meet the requirements of the independent tool certification
- > SGS-TÜV Certificate
- > All necessary Tool Qualification evidence and process guidance

Please contact QA Systems or your supplier for more details.

3 Conclusion

This paper has set out to demonstrate a compelling set of reasons why the opportunity costs of selecting an open source C/C++ unit testing tool outweigh the total cost of ownership of commercial C/C++ unit and integration test tools.

Many companies first try out unit testing with open source tools, before upgrading to a commercial tool. Where they have already invested significant time and resource into tests using such tools, they should explore with commercial vendors whether they offer a migration path. That is easily achieved by embedding these tests inside a Cantata test script, and enhancing them with Cantata's intelligent testing capabilities such as:

- > Call Interface Control – stubs and isolates to simulate or wrappers to intercept calls
- > Integrated code coverage analysis and checks on achieving coverage target
- > Checking data (including white-box checks on data declared static to the file or private)

This paper is not a detailed description of QA Systems as a commercial tool vendor, nor is it a detailed description of the capabilities of Cantata. For more information on Cantata or a free trial, so that you can determine why to pay for it as a unit test tool, please contact sales@qa-systems.com or visit the website at www.qa-systems.com/cantata.