

# Original Software – Software testing tools

Ovum's Software Team

**February 2010**

# Original Software – Software testing tools

## At a glance

### Developer

The Original Software Group, Basingstoke, UK.

### Base data

- Suite of functional test automation and test management tools.
- Tests GUI and web front-ends on PCs.
- Tests IBM iSeries systems.
- Specific capabilities for Oracle, SQL Server and DB2 (on iSeries) databases.

### Strengths and weaknesses

- Accessible to non-technical testers including users in acceptance testing.
- Intelligent handling of changes in Application Under Test.
- Covers both client side interfaces (GUI and web) and server side databases.
- More emphasis than other tools on test data preparation and verification.
- Innovative approaches to manual testing and migration to automated testing.
- Interfaces to HP Quality Center software.
- Limited performance measurement features.
- Project-oriented platform.

## The Ovum verdict

### Distinctive in several ways

Original Software's suite of functional testing tools has some truly distinctive features.

Like most functional toolsets, there is a test execution driver, TestDrive, which operates via the visual interface of an application under test, and a test management program, Qualify, for planning and keeping track of your tests – nothing unusual there. Nearly all the leading test automation suites possess these elements.

From similar core functionality, other toolsets in the market have expanded in two typical directions: load/performance testing, and more recently, Quality

Management and Application Lifecycle Management. The leading vendors have followed the money, which is only natural, but is not always in the interests of improving test automation.

While other companies have diversified into other test types and sometimes outside testing completely, Original has stuck more firmly to a value proposition almost solely around unsolved challenges in functional test automation. It has filled out some yawning gaps in functional test automation and attempted to make test automation more accessible to non-technical testers.

Among the distinctive capabilities of Original's tools are:

- Extensive features for the preparation, use and checking of test data
- Code-free visual test scripting
- Guided manual test execution, including automatic migration of manual to automated test scripts.

### **Unique for data handling and manual test conversion**

Not all of that is unique to Original. Other vendors use visual scripting and support guided manual test execution, however, Original does both extremely well. Where Original is unique is in comprehensively tackling the issues around data in testing.

The test automation industry's lack of attention to test data is startling when you consider that data is at least as important as functional behaviour for most information systems. No other vendor has both the extent of functionality for managing test data, coupled with integration into test planning and test execution activities.

Original's ability to migrate a manual test to an automated test automatically is also unique.

In its totality, Original's suite is appropriate for automation specialists, non-technical testers and even business users for acceptance testing. It fits in with classic waterfall processes, and it can also be used in agile processes more easily than other functional test automation suites.

As a smaller company with comparatively more modest R&D resources than giants like HP or IBM, Original cannot cover all the bases. Although the toolset works with the most common ERP/CRM packages, Original's tools are not as tuned as some of the competition to application package's custom technologies. Also, although Original's test planning and management solution can be implemented on a corporate-wide basis, it lacks certain key features that make a truly enterprise-wide solution in our view. Finally, you will need another toolset to perform load/performance testing beyond very basic response time measurement, Original Software recommends its partner Facilita

## The verdict

If you are looking for a project-oriented functional testing tool and you test applications built using mainstream technologies, you may succeed more easily and get better value out of the Original tools than some of the market leaders.

## Product description

### Product overview

Original Software's testing tools are typical in function and structure to most functional test suites. However, they have some key capabilities in the areas of manual testing, database verification and data preparation that set them apart. The suite consists of:

- TestDrive – test execution driver allows you to create and run 'code free' automated tests
- TestDrive-Assist – automated assistance for running manual tests
- Qualify – quality management platform for monitoring, control and reporting of test runs, test results and defects
- TestBench – database tool with a range of capabilities for test results and test data.
- TestSmart – creates test data using the 'pair-wise' algorithm for test case generation.

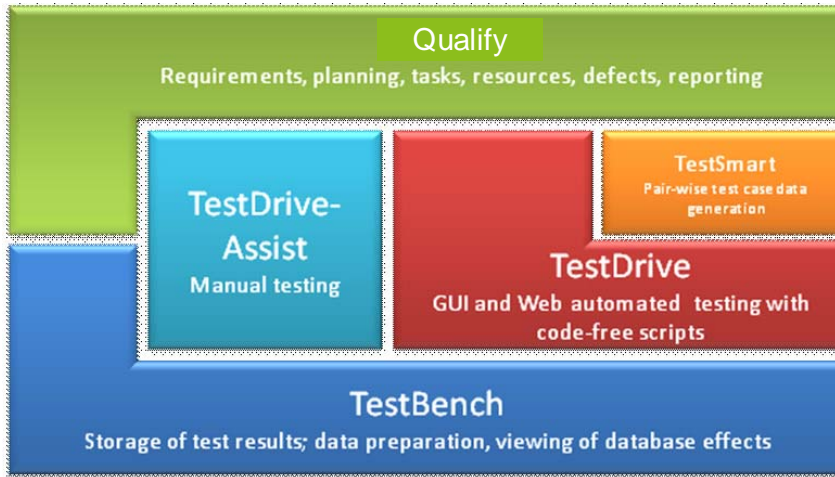
TestBench performs a number of diverse functions:

- analysis of database structures to support creation of test databases
- creation of test data inputs and expected results
- undo, rollback and reset database state to ensure test repeatability
- viewing and interrogation of database effects of tests
- storage of results of test executions (screens)
- scrambling (anonymisation) of test data to ensure data privacy.

### Component relationships

Figure 1 below illustrates the relationships of the components schematically.

Figure 1 Original Software test suite components



Source: Ovum

### Product profile

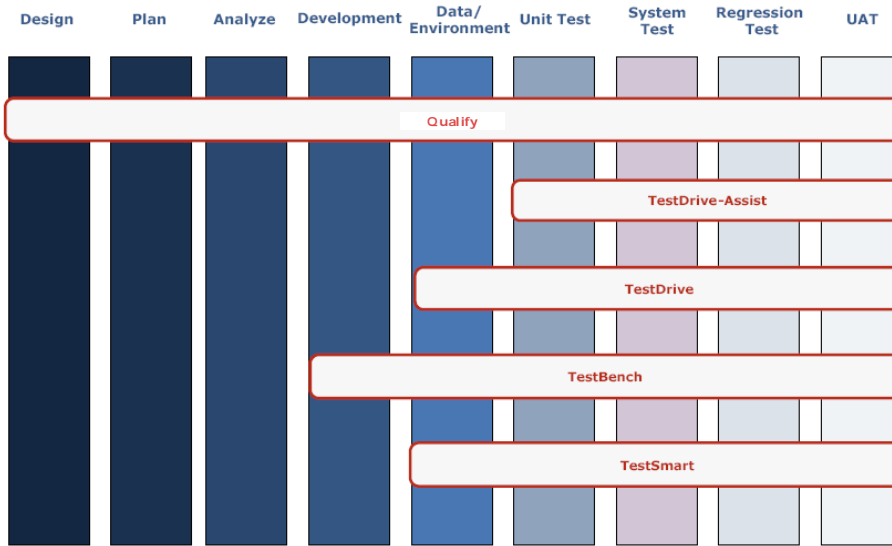
What it is	An integrated solution suite for application quality management incorporating functional test automation, manual testing and database testing
New or enhancement	Latest versions of a product line in existence for many years; Ovum has not reported on it since 2002, so we are covering the whole suite, not just the changes
Version	All products, version 6
What problems it aims to solve	<p>Automation of functional testing</p> <p>'TotalTesting': improve scope of testing to include not just the client interface, but also the database server</p> <p>Creation of coherent test data from production data while maintaining data privacy</p> <p>Lowering the skill level needed for automation through code free and self-healing scripts</p> <p>Automated support for manual testing</p> <p>Structured migration path from manual to automated testing</p> <p>Facilitate use of automation in high-change (e.g. agile) processes</p>
Typical user roles	Test manager, QA Tester, Business Tester, User Acceptance Tester, Technical Tester
Typical organisation type	Project-level testing organisation, Business Unit performing User Acceptance Test
Key capabilities	A holistic view of the software quality lifecycle, covering requirements management, resource and task planning, scheduling, automated test

	<p>execution, manual test monitoring, defect and issue management, and reporting</p> <p>Definition of test environments</p> <p>Workflow engine for setting decision paths</p> <p>Alert function for notification of key milestones</p> <p>Flexible filtering and reporting on project progress and results</p> <p>Synchronised testing of front- and back-ends of applications</p> <p>Visual scripting of tests (code-free)</p> <p>Record and playback of tests to create visual scripts</p> <p>Data-driven test capability</p> <p>Busy sense to determine when to capture results for validation</p> <p>Automatic spell checking and link checking</p> <p>Visual review of results shows screen appearance at each step</p> <p>Captures and stores annotation of results</p> <p>Versioning of scripts and results</p> <p>Support for VBA to perform enhanced validations</p> <p>Database stored rules for verification</p> <p>Action Maps to modularise scripts (keyword testing-like approach)</p> <p>Self-healing capability enables scripts to tolerate changes in the interface</p> <p>Guided execution of manual tests with support for prompting testers actions, capturing and saving test results (screen and database), recording tester's pass/fail, raising of defects with supporting documents from the test</p> <p>Auto-analysis of database relationships for test data extraction</p> <p>Scrambling of personal data</p> <p>Automated rollback of database to any chosen position, e.g. known start position</p> <p>Email notification of defect with supporting evidence</p> <p>Integration with HP Quality Center Software</p>
Environments	<p>HTML, .NET, Java, JSP, AJAX, IBM iSeries and zSeries green screen</p> <p>PHP, Python, Perl, VB, Powerbuilder</p> <p>DB2 (on iSeries), SQL Server, Oracle</p>

## Usage

You can get the flavour, in Figure 2, of how the various products are deployed throughout the stages of a test project.

Figure 2 **Deployment across the stages of a test project**



Source: Original Software Group

## Analysis

### Different in important ways

Original's functional test tools have several key features that differentiate them from the typical and ubiquitous script-based front-end oriented test execution tools. Original has attempted to tackle more of the core problems that afflict this class of test automation than any other vendor. They address four significant problems:

- The technical barriers to adoption of test automation, leaving the vast majority of functional testing to be done manually.
- The difficulties of using automation in high-change situations with the result that automation can only be cost-justified in regression testing.
- The neglect by other functional test suites of all aspects of preparing, retrieving and checking persistent test data, i.e. databases.
- Allowing the test management solution complete flexibility when it comes to mapping test processes

Original's solutions don't map directly to these four problems, but each solution touches on one or more of them. For a small company, they have perhaps the largest set of slogans and catchphrases we've encountered to describe their

approaches. Normally we ignore such marketing terminology, but we mention some of it here, because it helps characterise Original's distinctiveness.

## Focus on test data

Functional test tools generally concentrate on driving and verifying the application front-end interface. Remarkably they tend largely to ignore test data. Provision for data driven testing via spreadsheets of test inputs is almost universally available, but that's all. If you want a test script to access a database directly or check the database effect of a transaction, you must do it programmatically in a script, run SQL post-hoc the test or use the application under test to re-query the database after an update. These approaches are primitive, long-winded and surprising considering that data issues are at least as important as front-end behaviour in most information systems.

TotalTesting is Original's slogan for its capabilities for test data preparation before testing, test data initialisation at the start and end of tests and test data verification. This solution is implemented mostly in TestBench, which supports:

- analysis of your databases to understand the relationships for test data extraction
- extraction of test data including anonymisation (scrambling) of data to protect privacy
- creation, storage and execution of database resident validation rules
- checkpointing and rollback of databases to known start positions to ensure predictable running – and re-running – of test scripts.

TestDrive supports data-driven testing. The integration of TestBench's data capabilities with TestDrive and TestDrive-Assist also means that when you review your screen results, you also see the database changes associated with that transaction.

Finally, Original also provides TestSmart, which generates test cases based on the pair-wise algorithm for generating test case data.

Test data preparation, test case data generation, test run initialisation, inline viewing of database effects associated with front-end input: these are essential functional test activities over and above simple data-driven test capabilities. No other functional test suite that we are aware of covers test data management to this extent. It is Original's most powerful differentiator.

## Accessible automation

Original's central answer to overcoming the technical barriers to test automation is their 'code-free' model of scripting in the TestDrive product.

They are not unique in taking this approach. In Ovum's report, 'A new breed of software test automation', we categorised this approach as 'visual scripting'. The language of the script is expressed in a humanly meaningful vocabulary of

operations on visible screen objects (buttons, text fields) which have recognisable names, usually based on visible screen labels. The script text is accompanied by a screenshot so you can visualise the effect of every script command. The hope is that a code-free script language will not intimidate non-technical users and the visual link to the screen shot will make it even more accessible.

Our experience is that visual scripting is helpful and it can lower the barrier to test automation for most testers and some business users performing testing. However, despite being called code-free, the scripts are still code; they are just not code like C, VB or Java. Even so-called code-free scripting can put off non-technical users.

Other technology within TestDrive lowers the barriers in further ways. Automatic validations are a powerful and very useful capability. They save the user having to decide what content to extract for checking, when to insert a validation and how to cope with server response times.

## **Simplifying maintenance**

It is no secret that one of the biggest inhibitors to successful test automation is the effort to maintain scripts over time as an application under test is maintained. Original's solution to one aspect of this problem is something it calls 'self-healing' technology.

Self-healing technology uses an algorithm to spot when visible elements of an application have changed and to correct the script. It can deal with the following circumstances:

- objects on a page or screen have been added or removed
- objects on a page or screen have been moved
- missing screens or pages.

The user can incorporate the changes automatically into an updated version of the script or make their own changes instead.

Self-healing technology is not unique to Original, but it does add to the resilience of scripts over time. Be aware, however: although it is useful, it cannot perform miracles and adjust a script to compensate for radical changes to the underlying application under test.

## **Easing into automation**

Original encourages you to take up test automation progressively, in line with your capability to absorb new automated processes and technology, through a process of adoption they call 'Crawl, Walk, Run'.

The initial crawling stage involves test planning and management using Qualify and support for manual testing with TestDrive-Assist. Walking introduces test data management using TestBench. A step like this is not supported by any other

functional test suite – not in the comprehensive way Original supports it. Running brings you to full test automation using TestDrive.

Original's solution to manual testing is another approach to easing into full automation. Manual test support is part of a trend in the industry, where several tools now support guided manual test execution. Given that it is estimated that only 20% to 40% of test projects use automation, the vendors have decided, if you can't beat them, join them.

Original's TestDrive-Assist is an excellent example of guided manual test technology. Launch a manual test from Qualify and TestDrive-Assist will fire up the application and silently record your activity. You will be prompted to perform the actions in your manual test stored in Qualify and if you spot an error, you can immediately annotate the captured results and create a defect report.

At minimum, the logs from TestDrive-Assist can be saved to create a complete audit trail of test results. Moreover, automatically tracking and recording your test with screen shots means you can provide developers with a package that describes how to reproduce an error. Used with TestBench, you can retrospectively associate the test and defect with database activity.

TestDrive-Assist provides you with a rich environment for manual testing, but the bigger payback is how it facilitates adoption of full automation. Uniquely for any of the guided manual test execution products we know of, Original can automatically convert a manual TestDrive-Assist test into a fully automated TestDrive test that can be repeated.

Automated migration from manual test to automated test is a major benefit. Even if your intention is to automate from the outset, it makes sense to start in TestDrive-Assist (not TestDrive) and convert to a fully automated TestDrive script only after the test sequence is debugged. Overall, this is a low-risk, low-cost way of progressively adopting test automation.

## **Beyond regression testing**

Despite many efforts by all the test vendors to make functional test automation easy to implement, success and ROI is usually limited to regression testing. It is commonly observed that test automation generally has no payback until you repeat a test at least three times. Automation works in areas that are relatively stable and low risk.

One area where TestDrive-Assist can be surprisingly appropriate is in unstable, changing circumstances. The two most notable situations are the first release of an application and in agile processes.

By recording your manual test, you may not have the benefit of full automation, such as automatic data entry and validation. However, you do get an audit trail and clear information for reproducing bugs. The ability to automatically migrate your manual test to an automated one may pay off immediately, if you're lucky,

but in any case, by the end of your scrum or iteration, it will form the foundation of a future automated test.

## **The rest**

Having focused on the most notable capabilities of the Original suite, we need to reassure you that it is competent in the usual areas covered by functional test automation packages.

Qualify acts as a central test management tool for keeping track of your software quality lifecycle. Organisations gain a centralised view of all their quality processes and tasks from project inception through to release. The product is also configurable to a great extent, enabling organisations to fit the solution to their processes and workflow. Qualify supports a range of development methodologies including Agile, waterfall, and hybrid processes. The solution offers pre-built templates for Agile methodologies such as Scrum, DSDM, XP etc. Users can also create their own templates, or modify an existing template to suit their needs.

One particularly good feature supported by the Original tools is the ability to summon up the script and results from the current or any prior version of a test.

## **Limitations and drawbacks**

There are a few characteristics of the Original tools that may limit their applicability.

Without a web-based platform for Qualify, we consider the suite appropriate for projects, rather than as an enterprise-wide solution. Obviously you can standardise on it and deploy it throughout a company. However, an enterprise (versus corporate) solution for Ovum is deployed on a web-enabled, multi-tier architecture platform with unified, integrated administration, security and backup features.

Another slight drawback for us is that the tools are functionally oriented, not role oriented. TestBench in particular is a Swiss Army Knife of functionality to be used in various stages of the test process by various types of testers, data analysts and database administrators.

Finally, based on our experience of other visual scripting tools our suspicion is that Original's tools will be most successful with well-behaved applications. We advise you to check that the scripting works with your technology.

---

Ovum does not endorse companies or their products. Ovum operates under an Independence Charter. For full details please see [www.ovum.com/about/charter.asp](http://www.ovum.com/about/charter.asp).

For full details of Ovum's citation policy, see [www.ovum.com/media/citation.asp](http://www.ovum.com/media/citation.asp).

Whilst every care is taken to ensure the accuracy of the information contained in this material, the facts, estimates and opinions stated are based on information and sources which, while we believe them to be reliable, are not guaranteed. In particular, it should not be relied upon as the sole source of reference in relation to the subject matter. No liability can be accepted by Ovum Europe Limited, its directors or employees for any loss occasioned to any person or entity acting or failing to act as a result of anything contained in or omitted from the content of this material, or our conclusions as stated. The findings are Ovum's current opinions; they are subject to change without notice. Ovum has no obligation to update or amend the research or to let anyone know if our opinions change materially.