

Quality Assurance and Testing

Case studies



Table of contents

Abstract	3
1. Case #1: Testing native windows application using Rational Robot	4
1.1. Background.....	4
1.2. Scope of QA services.....	4
1.3. Tools used.....	4
1.4. Acceptance	4
1.5. Team structure	4
1.6. Infrastructure	4
1.7. Acceptance and handover.....	4
1.8. Challenges.....	5
1.9. Summary	5
2. Case #2: Automating web based application testing	5
2.1. Tested application.....	5
2.2. Scope of QA services.....	5
2.3. Tools used.....	5
2.4. Team structure	5
2.5. Infrastructure	5
2.6. Challenges.....	6
2.7. Summary	6
3. Case study #3: Mediation solution testing	6
3.1. Tested application.....	6
3.2. Scope of QA services.....	6
3.3. Tools used.....	6
3.4. Acceptance	6
3.5. Infrastructure	6
3.6. Challenges.....	7
3.7. Results.....	7
4. Summary	7

Abstract

Verax Systems is a software house, consultancy provider and systems integrator specializing mainly in advanced and innovative IT solutions for the telecommunications, finance and enterprise markets. Since its incorporation, Verax Systems has successfully provided services and delivered numerous projects on five continents.

Verax Systems provides a wide range of **Quality Assurance Services (QAS)** including:

- Analysis and test planning based on software specifications or usage cases.
- Black-box automation of system test cases using industry-standard tools.
- Technical consulting in the SQA area.

This paper presents three case studies illustrating critical aspects of successful quality assurance projects performed by Verax Systems, summarized in the table below:

	TESTING NATIVE WINDOWS APPLICATION USING RATIONAL ROBOT	AUTOMATING WEB BASED APPLICATION TESTING	MEDIATION SOLUTION TESTING
Project duration	5 months	3 months	6 weeks
Staff involved	2-3 engineers	2 engineers	3 engineers
Tools used	IBM Rational Robot family of tools	Canoo WebTest, System ANT build scripts and the Eclipse environment	Simulators developed in UNIX scripted languages Scripts developed to process logs and databases
Infrastructure	Laptops with VPN access to the tested system	VPN tunnel to the tested system, source code repository and test databases Windows machines with the Eclipse environment	Tests were performed over the VPN at operator premises

Table 1: Case study summary.

Software systems that fail to provide full functionality, performance or to otherwise meet user needs can reduce profit and productivity. Optimized Software Quality Assurance practices result in cost-effective and high-quality software. Verax Systems defines and performs software quality tests using proven techniques tailored for the life cycle model provided by the Customer.

For more information on Verax Systems' professional service portfolio please visit us at <http://www.veraxsystems.com/en/services>.

1. Case #1: Testing native windows application using Rational Robot

1.1. Background

The quality assurance services commissioned by the Customer embraced front-end application testing. The applications were both web based as well as native GUIs, running in the Microsoft Windows environment. Server side testing has not been in the scope of the project.

Main objectives for the Customer in the project were:

- Keeping own engineering team focused on software development and project delivery, rather than quality aspects.
- Having an independent QA team without knowledge of system implementation details, which allowed “black box” testing.
- Adding additional resources to the project in order to allow interleaving implementation of testing, which resulted in improved risk management and eventually led to timely project delivery.

1.2. Scope of QA services

The services provided by Verax Systems included:

- Analysis of functional requirements, preparation of corresponding test suites and implementation of fully automated test cases (some test cases had been provided initially by the Customer). System requirements specifications and the test system were provided by the Customer. The testing was entirely **black box**, as the source code was not available to Verax Systems.
- Conducting tests and reporting problems to the Customer’s engineering team responsible for implementing the system.
- Delivery of test reports and test cases source code to the Customer.
- Providing documentation of traceability matrix: test cases to requirements.

1.3. Tools used

All the test cases have been fully automated using the IBM Rational Robot family of tools. Some extensions to Rational Robot have been implemented by Verax Systems, based on frameworks provided by the Customer.

1.4. Acceptance

Project acceptance was based on partial acceptances of individual test cases. The project was run as an agile project, with short cycles (a few days) to deliver and run priority test cases using a time and material billing model. Such an approach allowed the Customer’s software development team to fix problems and bugs as soon as they were detected.

1.5. Team structure

Initially, two engineers were deployed to Customer premises for training. The initial training included workshops on the system to be tested, as well as the Customer’s Rational Robot extensions.

1.6. Infrastructure

The network infrastructure used in the project was straightforward - the QA engineers used laptops with VPN access to the tested system (300 km away). The laptops had to be preinstalled at Customer premises with VPN keys.

1.7. Acceptance and handover

All project deliverables were checked into the Customer’s project repository (IBM Rational Clear Case) as the project was run. Partial reviews were conducted for each delivery, which usually constituted two to three days of work.

1.8. Challenges

Key challenges in the project included:

- Establishing direct communication between the Customer's engineering team and Verax Systems' QA engineers in order to assure high levels of communication and joint problem solving.
- Coping with extending the scope of the project and working in agile project mode.

1.9. Summary

The project lasted for about 45 months and involved 2-3 engineers – eventually all the functional requirements were tested, and the Customer was able to deliver its **project on schedule while maintaining high levels of quality.**

2. Case #2: Automating web based application testing

2.1. Tested application

The Quality Assurance Services provided by Verax Systems included preparation of automated test cases for a web based application. Input to testing was structured around "areas of weakness" and most frequently used parts of tested application, rather than functional specification. It was decided to test the application based on GUI usage cases, building a usage tree from the login box through all the options that users can use in the system. The approach was to test most frequently used scenarios and shallow paths first (i.e. requiring 3-4 clicks), and leave most nested parts of application (i.e. requiring many clicks) for later – this allowed elimination of the most common problems first, and improvement of the application's user experience in a sense that bugs most likely to occur were eliminated first.

2.2. Scope of QA services

Verax Systems' role was to automate initial test cases obtained from the Customer using Canoo, a web testing framework, and later to devise and implement new test cases. The tested application had typical three-tier architecture, based on Tomcat, Struts, HTML, Dojo and Ajax technologies, and used Oracle as the database system.

The tested system was only available through a VPN; no local copies or source code were transferred to Verax Systems.

2.3. Tools used

The most important tool used in this project was Canoo WebTest – an open source tool for automated testing of web applications. The framework was integrated with system ANT build scripts and the Eclipse environment. In addition, since Canoo can check only user interface behavior, Unix scripts have been developed to check database data consistency as individual actions were triggered from the user interface.

2.4. Team structure

Project acceptance was based on review and approval of delivered test cases. The project was a fixed price project, i.e. had a man hour budget in which the team QA was to deliver as many cases as possible (prioritized from the most important to the least important ones). The team had a local lead engineer and reported directly to the Customer's project manager.

2.5. Infrastructure

A permanent VPN tunnel to Customer premises where the tested system, source code repository and test databases have been established. All test case development took place locally on Windows machines using the Eclipse environment.

2.6. Challenges

Key challenges of the project included:

- Establishing good communication between project leader and local QA team – daily stand up calls were allowed to resolve most problems.
- Checking results in the database in integrating this functionality with the Canoo framework.
- Working with the product R&D team – as Canoo puts some requirements on tested HTML output (all elements must have valid, unique identifiers), the R&D team involvement was necessary and some important cases had to be postponed until required functionality was developed.

2.7. Summary

Involvement of Verax System's QA team, which allowed for quick detection and fixing of the most annoying user interface bugs. Since all the tests are repeatable, each release of the system can be tested – **investment is returned in a few releases, giving additional benefits in the form of good user perception of the software.**

3. Case study #3: Mediation solution testing

3.1. Tested application

The Quality Assurance Services described in this case study involved testing of a real-time billing mediation solution. The application consisted of a number of background processes and ran without any GUI. The focus on testing was thus solely on data processing.

3.2. Scope of QA services

The primary role of Verax Systems personnel was to:

- Design test plans based on the High Level Design (HLD) and Low Level Design (LLD) solution documents. The plans were written as Microsoft Word documents. Each case was described step by step to deliver precise test specifications.
- Check traceability to assure that test cases covered all requirements outlined in the HLD and LLD.
- Automate all tests in a specially developed simulator. About a hundred test cases have been developed. Each test case was very complex – running an entire test suite could take a few days.

3.3. Tools used

Tools used in this project included:

- Simulators developed in UNIX scripted languages as the system had an automated engine to run scripts.
- Scripts developed to process logs and databases.

3.4. Acceptance

The project was performed at a fixed price as the HLD and LLD documents were known up front. Project acceptance consisted of two stages: test plan document followed by test cases. Scripts and test data were required to be stored in a CVS repository.

3.5. Infrastructure

The project has been delivered entirely out of Verax Systems premises; the tests were performed over the VPN at operator premises.

3.6. Challenges

There were some challenges Verax Systems had to face. Key challenges embraced:

- Full automation of the testing process using command line tools and scripts.
- Long, overnight test runs.
- Processing of non-visual output data (e.g. databases, syslogs).

3.7. Results

The project lasted for about six weeks and involved three engineers – eventually all the LLD and HLD requirements were tested, and system testing has been completed **on time and at carrier-grade quality.**

4. Summary

Verax Systems' QA team uses proven quality assurance techniques in order to:

- Discover quality problems.
- Streamline applications.
- Determine appropriate project standards.
- Automate test cases.

Verax Systems' QA team has implemented a multitude of standards and tools for internal software development and project deliveries including:

- Formal project management methodology.
- Automated system and unit testing.
- Test coverage measurements.
- Regression testing.
- Continuous build integration.
- Software for creation of test plans and reports.

Members of our QA team have Software Testing Foundation (ISTQB) certifications.

To learn more please contact one of our offices or visit us on the Internet at:

<http://www.veraxsystems.com/en>.