

Reusable Automation Framework with Excellent Support for Isomorphic Web Controls



STAG develops a robust, user-friendly automation framework that is reusable across different projects and platforms, and handles isomorphic web controls for consistent and successful execution of an end-to-end automation suite, to run on web-based metadata management software for a global leader in data integration software and services.



Domain/Category -
Enterprise Data Integration



Technology - Web isomorphic controls, Oracle Automation Tools - SilkTest, PERL, Java, Exam-XML, VBScript

CUSTOMER AND PRODUCT BACKGROUND

The customer is an independent data integration leader providing data integration software and services to organizations, thereby empowering them to access, integrate, and trust all their information assets, and giving them a competitive advantage in terms of growing revenues, improving profitability, and increasing customer loyalty.

The product is a web-based metadata management tool to browse and analyze metadata from disparate metadata repositories. The tool helps in understanding and managing information, processes, and the inter-relationship usage. The product has been developed with Isomorphic technology, a SmartClient product that helps developers get results that are more logically structured and simplified and provides a maintainable approach to web front-end development.

PROBLEM STATEMENT

The product underwent a major change with regard to its GUI; additionally, the QA team had to test multiple branches of the product. As a result of the GUI-related change, the existing automation scripts no longer worked. In this scenario, the customer wanted a test automation solution that could adapt their feature needs and also speed up the QA regression process.

The entire application was developed using Ajax, a programming technology that was not being completely recognized, in terms of the tree view and list view objects, by the SilkTest automation tool. The customer was looking for a framework that would provide the flexibility to support:

- Addition of new test cases with different data sets.
- Reusing of the general low-level library for adding business functions for all new features and also for the changed GUI.
- Comparison of MS Excel spreadsheets containing application data in MBs (manual testing of such test cases takes a lot of time).

- Fixing of existing automation scripts and adding verification calls for all the existing test scripts/test cases.
- Automating the new features of the product release and adding i18N/Localization support for all the test cases.

To resolve all these technical issues, the customer sought the assistance of an automation expert.

SOLUTION

The STAG team interacted with the customer's development team to understand the application setup, product features, customer problems, and existing test assets. The team found that approximately 15% of test cases identified for automation were redundant, and that no test data was available for most of the test cases.⁷



Test cases modified: 385



Test cases automated: 266

The STAG team first created a framework that was compatible with different browsers (IE6 and IE7) and operating systems (Windows 2003 and XP), and developed low-level libraries to interact with all the identified objects using a mix of Java scripts and x-Path concepts to find most of the workarounds.

The team then created business functions that could be reused not just in individual modules but also across modules. These functions were based on the concept of x-Path to ensure function portability across multiple browsers.

The following are the highlights of the workarounds developed by the STAG team and their implementations:

- A customized method in Java Script to resolve the issue of tree view objects not being recognized by the SilkTest tool
- A function to enable searchable list view objects
- A function to refresh the entire page and clear off a dialog box from the screen once the box is closed
- A utility function to convert the spreadsheet file (MS Excel) to the .csv file format and then shorten it, enabling the data coming out in consecutive runs to be in the same order and allowing comparison of the contents of the MS Excel sheets
- A utility for node-by-node comparison of XML files
- A functionality for cell-by-cell comparison of large MS Excel files

The team strictly followed all the specified guidelines and coding conventions in developing the scripting language to write the business functions. It created test data for each test case and added it to the SQL server.

For the identified browsers and OS, the team tested the functionality first at the unit level and then at the suite level. For i18N support, it converted the localization file from Java Script format to Properties file format to ensure easy format recognition by the tool. Finally, the team tested the script against both Japanese and English locales and automated two versions of the product.

OUTCOME AND VALUE ADDITIONS

The STAG team made quite a few significant contributions, including:

- Developing a robust and consistent solution to interact with the different types of isomorphic objects
- Designing business functions so they could be reused to create new test cases
- Maintaining and extending scripts in a consistent format to enable new team members to learn, understand, and implement them quickly
- Implementing extensive HTML Reporting for the entire suite to run results verification

The code coverage was as per the Acceptance criteria, and the solution was found to be very effective for regression cycles. The result was a significant reduction in execution time, from 4 weeks to 32 hours, which gave developers time to conduct a quick check on the regression issues.

Visit: www.stagsoftware.com | E-mail: marketing@stagsoftware.com

Bangalore: +91 80 28495574 / 75