



# QA/Test Automation using QTP

## Extensive Workflow-based Software Systems

### Executive summary

The client firm provides web-based environmental, quality, health & safety, and business-performance management solutions, offering a library of over 65 unique software systems and modules, all of which strategically designed to address distinct specific business needs, to work either as stand-alone solutions or as seamlessly integrated combined solutions. Their software approach has high configurability, which allows the customers to build the perfect solution for their specific needs. How do they make sure it all works? The complexity implies tremendous QA/Testing efforts and respective/associated costs. A small change (any major/minor release to any software system/module) affects a considerable magnitude of workflows/scenarios. This gets worse if you consider the pure manual testing efforts. Last year, Mindfire started its relationship with the client firm, with a mission to automate software systems along with associated dependencies/modules. Starting with a single QA resource, the significant value addition resulted in a setup of three-member QA team within the 1<sup>st</sup> month. The team has already automated testing for first three assignments.

### About our client

<b>Client</b>	North America's leading provider of Environmental, Quality, Health & Safety, and Business Performance Management Systems
<b>Location</b>	Toronto, ON, Canada
<b>Industry</b>	Training, Document-Control, Environmental, Safety, Quality, Audits Management Systems

### Business situation

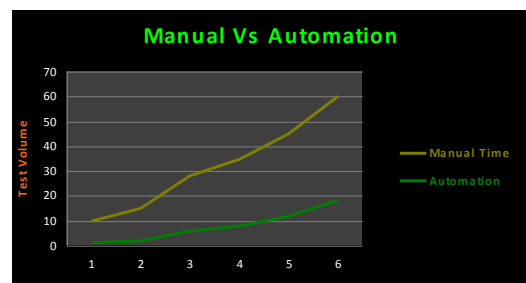
The client firm was looking for an expert and reliable partner to help them with QA automation of various software systems and their interdependencies. Mindfire took the opportunity and established a virtual team of QA resources adept at learning and understanding systems, associated workflows, creating manual and automated test cases as per requirements. An initial assessment period saw automation of existing set of manual test cases for a single module, resulting in the delivery of QTP automated scripts. The first selected system/module was as much a preview of complexities on offer, as it was a test of the QA skills/expertise. It was a less user-friendly application, with unstable layouts and difficult test cases. A tight timeline ensured a good test of delivery under pressure.

### Solution details

#### The Mindfire Solution

The first assignment underwent a swift execution, ensuring a successful on-time delivery. The second assignment saw QA automation of another module having about 100 test cases within a couple of weeks. A disciplined execution again ensured a successful delivery of scripts in time, with an automation-coverage of 92% of the manual test cases. The next assignment was a tough one, which required creation of 2300+ manual test cases before their migration to automated test cases. The QA team faced the following challenges to begin with, which, in fact, are likely to continue with future assignments:

- Missing functional specification document
- Missing manual test cases
- Automation unfriendly application





- Working with dynamic objects, complicated parameterization process, reading external files, writing custom functions, running reusable actions, synchronization of different pages and objects (especially, unpredictably time-taking pop-ups) etc.

Our QA automation team dealt with these challenges, carefully executing/implementing key steps:

- **Knowledge transfer** - Several Web-sharing sessions, discussions & conferences were carried out with the client technical team to understand the system and its workflow.
- **Starting on building test cases** – Preliminary manual testing started to get to a set of smoke test cases (which incidentally resulted in 50+ functional bugs – appropriately logged into the defect tracking system). This led to building of 2300+ test cases in around a month time.
- **Planning for effective automation process and quick production**
  - The team continues to perform test case analysis for automation, using a simple approach: [Plan → Inspect → Automate → Enhance → Run -> Debug → Analyze]
  - Planned and prepared the *Hybrid Automation Framework* (Combination of *Keyword*, *Data Driven*, *Modular*, and *Test Library* frameworks).
  - Built multiple *Shared Object Repositories* of objects;
  - Used *Data-Table*, *Environment*, and *Test-Action* parameters for *Parameterization*;
  - Using *VBScript* to write several *Function Libraries* for custom functions like converting custom date formats, calculating different equations, comparing Strings, calculating date difference, status checking and several other repetitive activities;
  - Used combinations of *Sync*, *Wait*, *Exist*, and *custom Function Library* methods to resolve the problems in synchronization;
  - Used *Descriptive Programming* to handle the unpredictable scenarios, dynamic objects, and checking non-existence of any button/object;
  - *Virtual Objects* are being used for objects unrecognizable by QTP;
  - *Recovery Scenario Manager* used for handling unpredictable alerts;
- **Delivery** - Among other deliverables, all required resources to run the automation scripts are included (such as Function Libraries, Shared Object Repositories etc.). The Automation Success Matrix, Automation Percentage (percentage of automated test cases), Manual vs. Automation Time-Consumption details are also sent.

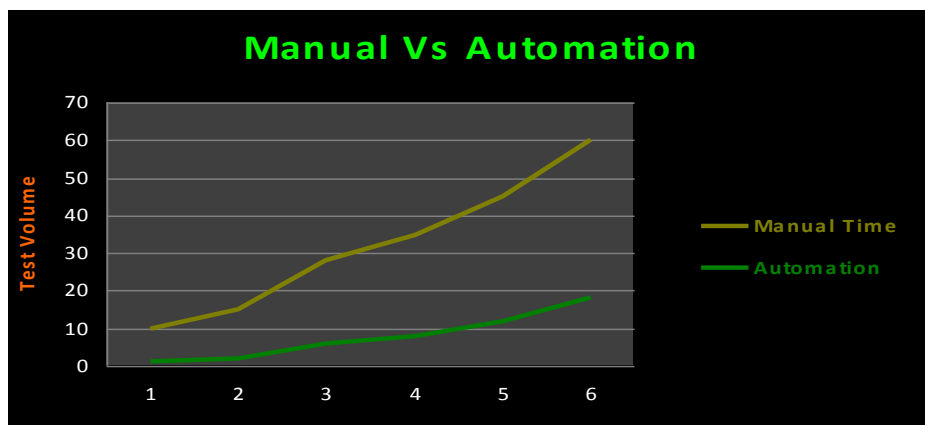


Figure 1 - A sample Manual vs. Automated Test Case Time-Consumption report

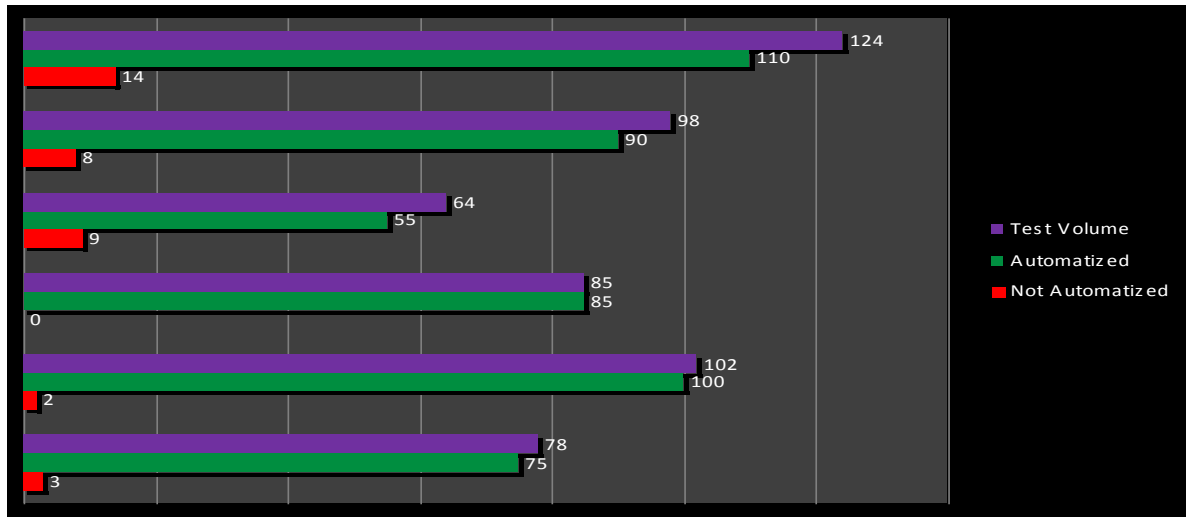


Figure 1 - A sample Automation Percentage report

### Achievements

- Mindfire team has successfully worked out solutions against all challenges faced so far, learning a few things alongside as well.
- Consistently gaining knowledge on system and its workflow, the team has managed to reduce the initial Script Automation time by 40%, using shared Object Repositories, reusable modules, and the Test Modularity framework.
- The team’s increased understanding of custom scripting using Descriptive Programming for the dynamic or random objects has helped achieve better and quicker results.

### Technologies

- The system to test is built using DotNet, and has over 65 independent modules.
- The team started with QTP 9.2, and is now migrating to QTP 10.0.

### Final results

#### Software system

After delivering the two initial, relatively small, modules, Mindfire team is working to deliver for the third module. There were about 23 scripts planned for this section, 19 of which have been finished & delivered so far.

#### Customer benefits

The client firm will greatly benefit with the automation cycle; in fact, automated testing for the first set of modules, in different customized environments, is already proving its worth. Automated scripts have replaced manual testing efforts spent on these modules, reducing the time involvement by 70%, and effectively eliminating active involvement of multiple human resources running the same manual suite across different platforms, OS, and customized environments.

#### Future relationship

We are actively working with the client firm since the last 6 months with three resources in the team, and we surely look forward to finishing the complete automation cycle, covering the rest of the systems/modules.