



ENHANCING SOFTWARE QUALITY

**ASSESS WITH YOUR BLINKERS OFF  
- WIDEN YOUR SCOPE FOR TOOL  
EVALUATION**

PREPARED BY : JAYA MURUGAN

**TABLE OF CONTENT**

<b>1. ABSTRACT .....</b>	<b>2</b>
<b>2. INTRODUCTION .....</b>	<b>2</b>
<b>3. GROWING TRENDS IN TESTING / COMPUTING .....</b>	<b>2</b>
<b>4. ARE WE DOING IT RIGHT? .....</b>	<b>3</b>
<b>5. GAPS.....</b>	<b>3</b>
<b>6. REFINED TOOL EVALUATION.....</b>	<b>4</b>
<b>7. CASE STUDY.....</b>	<b>5</b>
7.1 CASE STUDY 1:.....	5
7.2 CASE STUDY 2:.....	8
<b>8. CONCLUSION.....</b>	<b>10</b>

### 1. Abstract

We have come a long way in the field of technology. What was just a chalk piece has now become a stylus and what started as an abacus is now a computer. Maybe Charles Babbage had this vision of the future so why don't we visualize ours?

Personal computers / Smart phones have now become more of a necessity than a luxury. When we see this happening we can be sure more luxurious gadgets are going to invade the market. As testers, are we ready for this invasion? We don't want to be left behind too far away on Cloud Services and Devices.

It is known that testing is very crucial given any product. Manual testing grows with the product but are we too ready? This paper focuses on the importance of an automation tool selection. A study says 60% of automation projects fail and out of the 40% that are a success, only 8% have ROI. This causes hesitation in the testing world to go in for automation. Though there is no arguing on this study, getting to the root cause of it, it goes back to selecting the right tool.

Just as how critical diagnosis is for a surgery, finding the right tool is critical for automation as this lays the foundation for everything that lies ahead for a successful automation project. So why take a risk? This paper helps pull you away from the traditional parameters of tool assessment and encourages you to think beyond your blinkers.

### 2. Introduction

For reasons well known, automation is important in today's world of testing. A study says 60% of automation projects fail and out of the 40% that are a success, only 8% have ROI. This causes hesitation in the testing world to go in for automation. Though there is no arguing on this study, getting to the root cause of it, it goes back to selecting the right tool, choosing the right set of procedures and best practices which will reduce automation time, cost and energy.

There is no stressing enough that the focus should be on a reliable automation solution that can last the entire life span of a product. But with changing trends and technology it is not advisable to stop there. Once the tool is decided upon, the frame is built and half way through scripting if there is a discovery of an unsupported feature / need, rolling back is going to be close to impossible considering the time, effort and cost spent on this activity. Buffer for re-scripting can be made part of a project plan but re-evaluating a tool is not something anyone would enjoy.

### 3. Growing Trends in Testing / Computing

Closely following the trends in application development, some of the major areas that are growing really fast are around mobiles, services oriented architecture, etc. This paper attempts to elaborate test strategies used to automate applications with mobile support and applications with several service layers. In a single line, this paper talks about how tool evaluation should be done, gaps when it comes to applications specific to mobiles and service layers and process tweaks required to ensure test coverage.

To start off, testing can be tricky in these areas. Some of the challenges in mobile testing can be various operating systems, varieties of smart phone models, etc. Similarly when we look at Services Testing - How sure are we that GUI testing covers the Service layers? Services, message queues, database abstraction layers, and other GUI-less entity helps provide important business logic.

### 4. Are we doing it right?

Diving deeper into the subject of automation tool evaluation, here is a snapshot of how traditionally a tool is evaluated. In today's automation world, the following are a few common parameters that seem to play a vital role in evaluating a tool:

- Market share
- License cost
- Availability of skilled resources
- Supported technologies via add-ons and browsers
- Application/ Interface compatibility
- Data driven/ Parameterization feature support
- Object Recognition
- Script Maintainability
- Lifecycle integration
- Vendor support

The above appears to be a complete list. This may be true provided this list does not mislead the tester to lean on an easy way out given a situation with several constraints. For example, Capture and Playback is a way to tempt users saying scripting knowledge is not required to use the tool. Another scenario would be, let us say for example, there is Tool A which is an open source and Tool B which has a license cost attached but it supports more of the application. Weigh this carefully. Saving on the tool, you may end up spending on rework. Similarly, don't hesitate to pick a new tool but remember to include a few hours for learning it in your plan. There might always be a tradeoff but that is where experience comes in.

Keep in mind the following and don't let them impact your tool evaluation:

- Personal influence of peers – Remember two experiences are never the same. Tool Evaluation should not be biased.
- Resources' skill set – Picking the wrong tool may end up more expensive than training a resource.
- Closed mind – Be open and up to date on what is available in the market.

### 5. Gaps

Going back to the two major developing areas in testing – Mobile and Services – here are a few areas which we may fail to look at:

Mobile application testing is quite different from traditional desktop and web based testing. When it comes to an automated test, mobile applications can be tested using emulators and real devices. Real device testing is most preferred because while testing on a real device, one can measure the mobiles' and application's performance based on the CPU usage, memory usage, network, battery usage, etc.

Real device testing will offer most realistic test results when compared to emulators. Most of the open source tools do not support real device testing. If only licensed tools that may be expensive support this why not just test mobiles manually? With mobile application automation becomes mandatory to accommodate rapid application development cycles. Some companies offer (mobile cloud) labs with real (remote) devices for testing. But these tests still rely on system generated reports. To complicate the situation these days there are applications that are usually on Desktop, Web and Mobile. It is not advisable to pick a separate automation tool for each one of these. This will slow down the progress from simple reasons like reusability of functions may not be possible. Also think of scenarios where a value

# Assess with your blinkers off **2014**

## - Widen your scope for tool evaluation

passes from the Desktop application to a Mobile application. These will be marked as Non-automatable if different tools have been chosen.

When it comes to Services testing we see dynamic business needs changing every second that a tester needs to cope up with. Also regular GUI testing is will not suffice while testing service based applications. Service testing for the applications leans on services linked with business logic layers. Traditional black box testing approaches and tools would not help when we testing service based applications. We need a grey box kind of testing tool that should support both UI and the UI less layers.

Other areas discussed above such as RIAs which binds a data service to UI components, changing the UI content based on user events, calling a service based on user events use a lot Ajax, Flex and Flash controls as we are living in the 4G (LTE) era. Tools available in the market have limited support for these kinds of controls.

### 6. Refined Tool Evaluation

Bring the focus back, tool identification and evaluation plays a major role in any automation process with close control on making sure every phase of the project's life cycle is a success. With changing trends and understanding where we lack, here are a few pointers that will help assess with your blinkers off.

Before you even start with the tool evaluation here are small tweaks that can help improve the traditional process. The diagram below has the tweaks to the traditional approach in blue. There may not be many changes to the main phases but additions to sub tasks will go miles.

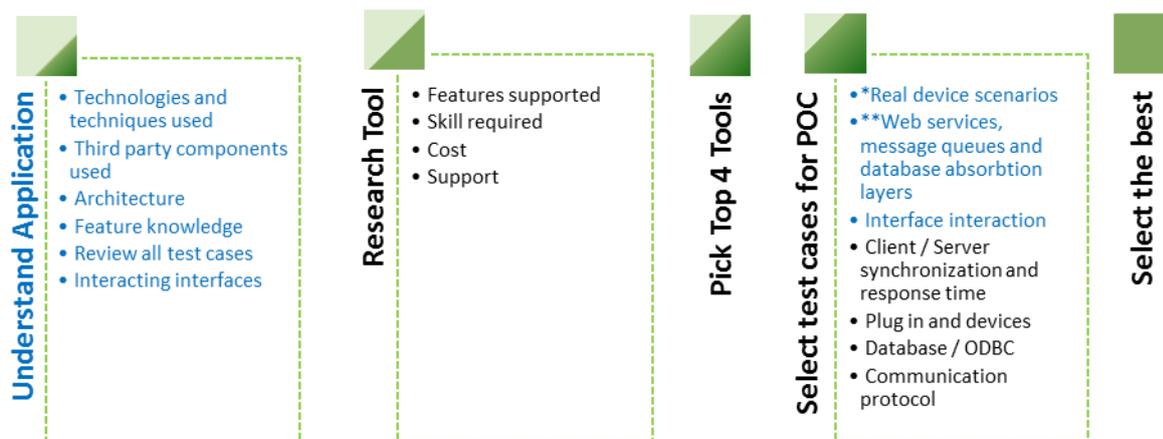


Figure (i) –Phases of Refined Approach

Taking a closer look at the introduced tweaks, understanding the application is the most important. Once you understand the technology, techniques and architecture, this will help you pick the right parameters for your tool evaluation. Knowing the boundaries of both the application and the tool, we can be sure of what can be covered and what cannot be. This will help avoid last minute surprises in the automation test coverage.

This phase will also help gain client confidence as several discussions can be entertained in this period of understanding the application's features. Third party components and other hardware interfaces can be tricky. Most of the time learning the architecture of these will pay off. The key to remember in this phase is not to worry about the time spent here. This is the most crucial period where we lay the foundation even before we get to the tool evaluation. Make this officially as part of your automation project plan and don't hesitate to take solid time to do this. The key to remember here in this phase is - do not derail. Keep your focus on the application.

# Assess with your blinkers off | 2014

## - Widen your scope for tool evaluation

---

Moving on to the other addition in this process in Phase 4 where we pick test cases for POC, with all that has been discussed in the previous sections about the importance of mobile and service testing, in addition to the traditional parameters, be sure to remember the following:

- Expanding on picking good parameters for automating mobile tests:
  1. Pan
  2. Pinch
  3. Zoom
  4. Scroll
  5. Support iOS and Android devices
  6. General mobile and application performance levers
  
- Expanding on picking good parameters for automating service tests:
  1. Good debugging interface
  2. Support transfer protocols like HTTP, SOAP etc.
  3. Support to technologies like REST, AMF, and JMS etc.
  4. Data driven testing – support to external data sources
  5. Support to message content assertion

Spoken thus far about the advantages of the tweaks in this process, remember every phase in tool evaluation is equally important. To do justice to each of these, the preceding phase should feed value in to the next. Just like in any project plan be sure to have entry and exit criteria adding value to the upcoming phase. The effort and learning obtained during tool evaluation does not stop there, it will help not just during development, execution and maintenance but will see you through successfully till the end of the project's lifespan.

## 7. Case Study

### 7.1 Case Study 1:

**Client** – Leading windshield provider in US

**Domain** – Retail POS

**Application Overview** – An application designed for glass retailers. The organization includes Purchasing and the Inventory allows you to manage all aspects of supply. The application is designed to maximize front-line customer service representative's productivity across all retail branches. Move jobs to other stores, schedule work for a neighboring branch or one that is across the country, or see inventory availability across the organization and at local vendors.

**Platforms:**

- Desktop – VB / SQL
- Web – Application in development stage (Firefox, Chrome and IE)
- Mobile–Sync data between mobile and desktop (Android)

**Challenges:**

- Selecting a tool that supports all the three platforms to ensure:
  - Coverage of device integrated test cases
  - Reduce cost
- Some of the licensed tools had limited or no support to the browsers like Chrome and Firefox

# Assess with your blinkers off | 2014

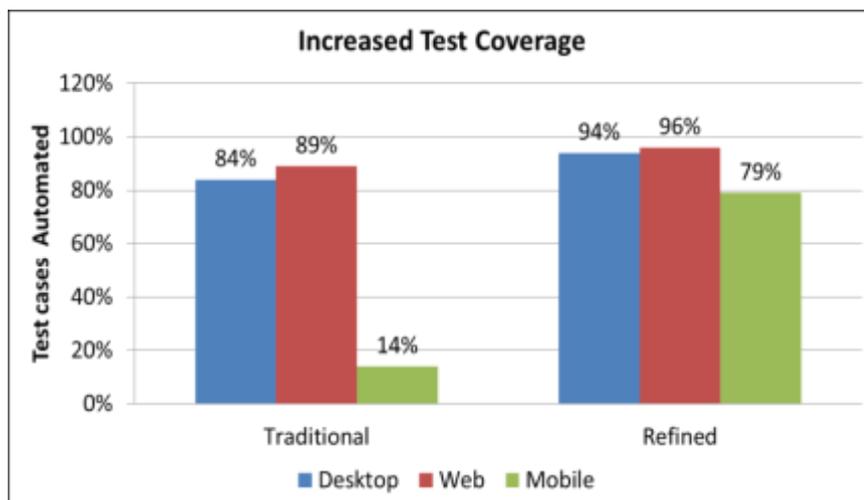
## - Widen your scope for tool evaluation

### Benefit of Refined Process:

- Promised support for current platforms and upcoming platforms
- Complete automation test coverage
- Significant cost savings on license

**Readiness:** Keeping the future in mind, if the application expands to support iOS, we are still ready!

### Metrics and Analytics:



**Figure (ii) – Test Coverage**

**Inference:** With our refined process, the number of automated candidates increased therefore increasing automation test coverage. There was a drastic count difference in what could be automated in the mobile device.

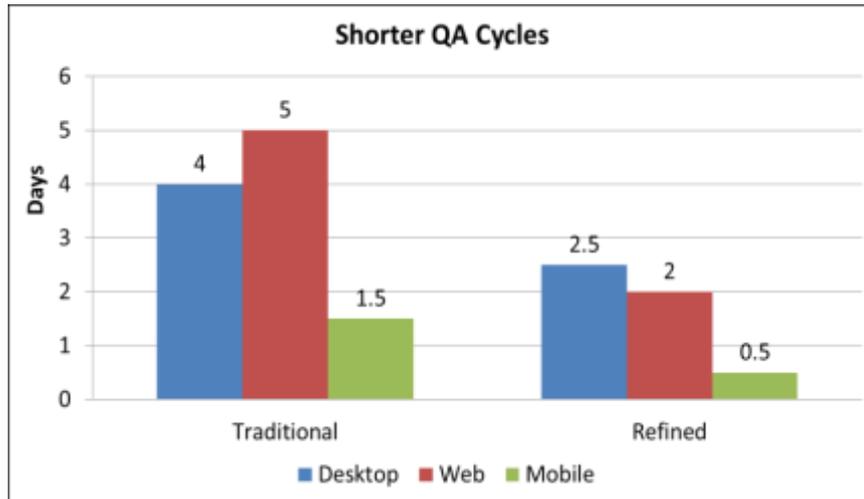


Figure (iii) – QA Cycle

**Inference:** This chart represents the complete time spent on QA (on an average) – Manual and Automation. Since the automation test coverage increased, it helped save time on the manual testing.

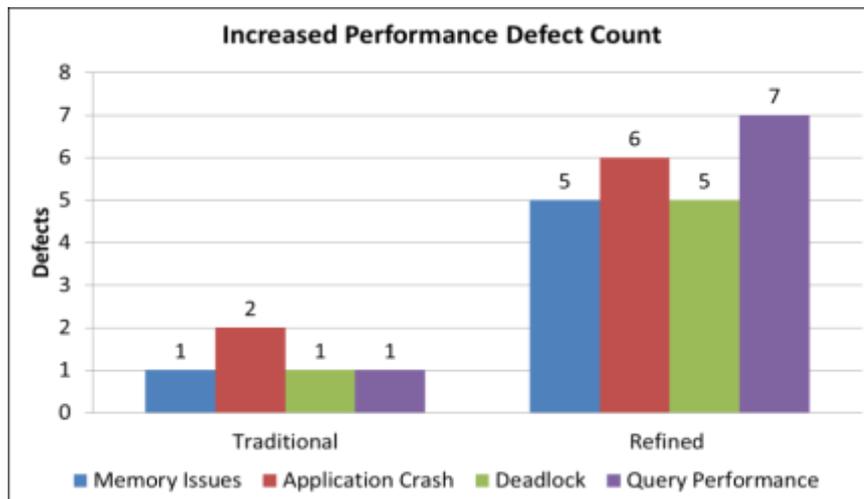


Figure (iv) – Performance Defect Count

**Inference:** The above chart a few areas such as Memory Leak, Application Crash, Deadlock and Query Performance were identified and with our refined approach the defects found in each of these have increased significantly.

### 7.2 Case study 2:

**Client** – Leading Automobile Reseller

**Domain** – CRM

**Application Overview:** This project is a customized CRM application which manages authentication details of its downstream applications (our client has more than 5 different applications for various business needs). The CRM's robust account management system automatically tracks customer activities, authorization privileges across the different applications (like Application Access, Authorization, and Authentication). All the support activities (Cases, tickets, Queues, Claims...) take place in the CRM. All the dealers and the users are created and managed by CRM. This application is also used to generate reports for all the transactions that happen in its downstream applications.

**Platforms:**

- Web – via only IE
- Plug-in – via Outlook

**Challenges:**

- Selecting a tool that supports UI and UI less layers testing
- Testing all the service layers in the application like
  - Web Services
  - Metadata Services
  - Integration Services
  - Workflow Services

**Benefit of Refined Process:**

- 100 % Automation coverage and solution
- Reducing the risk of service layer failures
- Early defect detection
- Significant cost savings on license
- Increased test coverage and shorten testing cycles.
- Improving the overall quality by testing services and its consumers

### Metrics and Analytics:

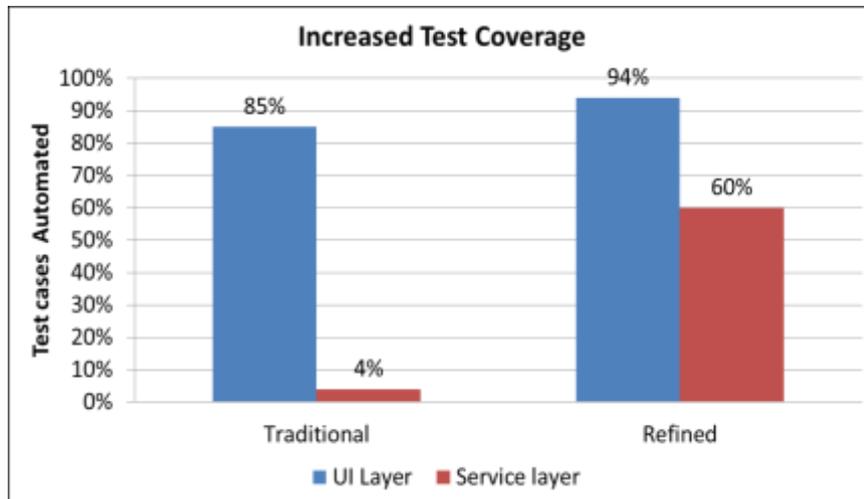


Figure (v) – Test Coverage

**Inference:** By including Service testing into the scope of automation, test coverage has increased. This in turn leads to a confident delivery since not just the UI has been tested but also the Service layers.

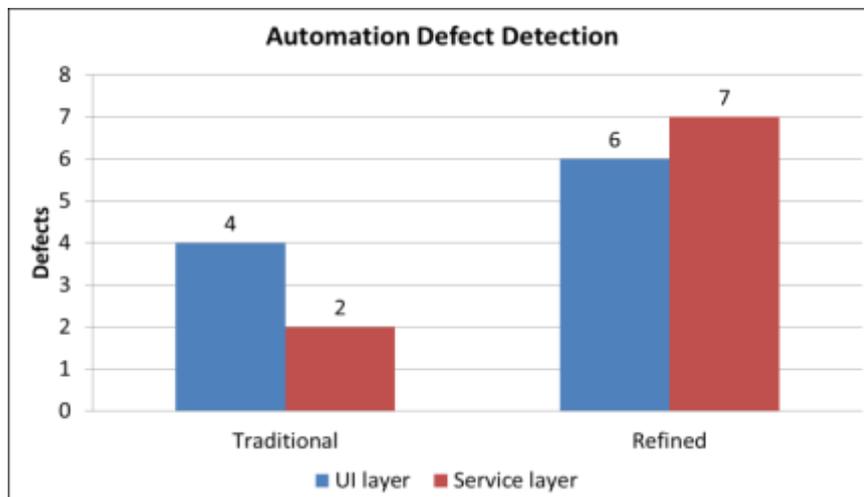


Figure (vi) – Automation Defect Detection

# Assess with your blinkers off **2014**

## - Widen your scope for tool evaluation

**Inference:** Defects identified in Service layer improved. This ensured overall quality of the application both from a front-end and a back-end perspective.

Key Levers	With Refined Approach	Value Additions
Service Layers Validation	Increased by 80%	<ul style="list-style-type: none"><li>Increasing Test Coverage</li><li>Reducing the risk of service layer failures</li></ul>
Test Execution	Faster by 25%	<ul style="list-style-type: none"><li>With more automated test cases, manual test time is saved</li></ul>
Early Defect Detection	Faster by 10%	<ul style="list-style-type: none"><li>Early defect detection of few UI defects due to service layers failures</li><li>Very easy root cause analysis of defects</li></ul>
Cost	Increased bottom line	<ul style="list-style-type: none"><li>No need to invest on a different tool for Services testing</li><li>Significant cost saving on license fees</li></ul>

**Table (i) – Benefits of Refined Approach**

## 8. Conclusion

We talk about continuous process improvement all the time. This has become the trend of the day. This is true in an automation project as well. Process tweaks need to be done more often than we would love to, to keep up with the ever changing times. To conclude here, with new tools mushrooming every day and technologies growing at a rapid speed you can be sure to beat the competition if you just open up to assess with your blinkers off!

# Assess with your blinkers off | 2014

## - Widen your scope for tool evaluation

---

### About Indium:

Indium Software is exclusively focused independent software testing services firm since 1999. Over the years, Indium mastered objective methods that minimize the risk of failure of applications and software products. With a global headcount of over 300 employees, Indium works for a mix of marquee Enterprise and ISV clients spread across the globe. Indium is aggressively pursuing the Social, Mobile & Cloud agenda to make these the core of our next wave of service specialization.

### Contact us:

#### USA - Indium Software Inc.

##### Corporate Office California

Suite 210, 1250 Oakmead Parkway  
Sunnyvale, CA - 94085.  
Phone: +1(408) 501-8844  
Fax: +1(408) 501-8808

##### Atlanta

Crown Office Suites,  
1870 The Exchange, Suite 100,  
Atlanta, Georgia 30339. USA  
Phone: +1(770) 989-7302,  
+1(678) 608-0809

##### Sales Inquiries

[americas.sales@indiumsoft.com](mailto:americas.sales@indiumsoft.com)  
[apac.sales@indiumsoft.com](mailto:apac.sales@indiumsoft.com)  
[emea.sales@indiumsoft.com](mailto:emea.sales@indiumsoft.com)  
[india.sales@indiumsoft.com](mailto:india.sales@indiumsoft.com)  
[sales@indiumsoft.com](mailto:sales@indiumsoft.com)

#### INDIA - Indium Software (India) Limited

##### Global Delivery Headquarters Chennai

VDS House  
41, 2<sup>nd</sup> Floor  
Cathedral Road  
Chennai - 600 086.  
Phone: +91-44-2811 6330  
Fax: +91-44-4210 4033

Prince Arcade Building  
22-A, 3<sup>rd</sup> floor  
Cathedral Road  
Chennai – 600 086.  
Phone: +91-44-4347 7200

Steeple Reach,  
Old No. 25, New No. 39,  
Cathedral Road, Chennai –  
600 086.  
Phone: +91-44-4347 7122

##### General Enquiries

[careers@indiumsoft.com](mailto:careers@indiumsoft.com)  
[info@indiumsoft.com](mailto:info@indiumsoft.com)

##### Bengaluru

Salarpuria Adonis, 2nd Floor  
Binnamangala,3/1,Old Madras Road,  
Kadranapalya, Indira Nagar  
Bengaluru – 560 038.  
Phone: +91-80-6784 7500

##### New Delhi

F-1/5 Okhla Industrial Area,  
Phase-I,  
New Delhi – 110 020.  
Phone:+91-11-6613 0400