

What Project Teams Need to Know about Successful Test Automation

February 25th, 2016

Jim Trentadue – Software Quality Consulting Director

jtrentadue@origsoft.com



Original Software

Agenda

1 Test Automation background

2 Breakout #1

3 Contributions by Functional Area

Break

4 Automation Scale & Impressions

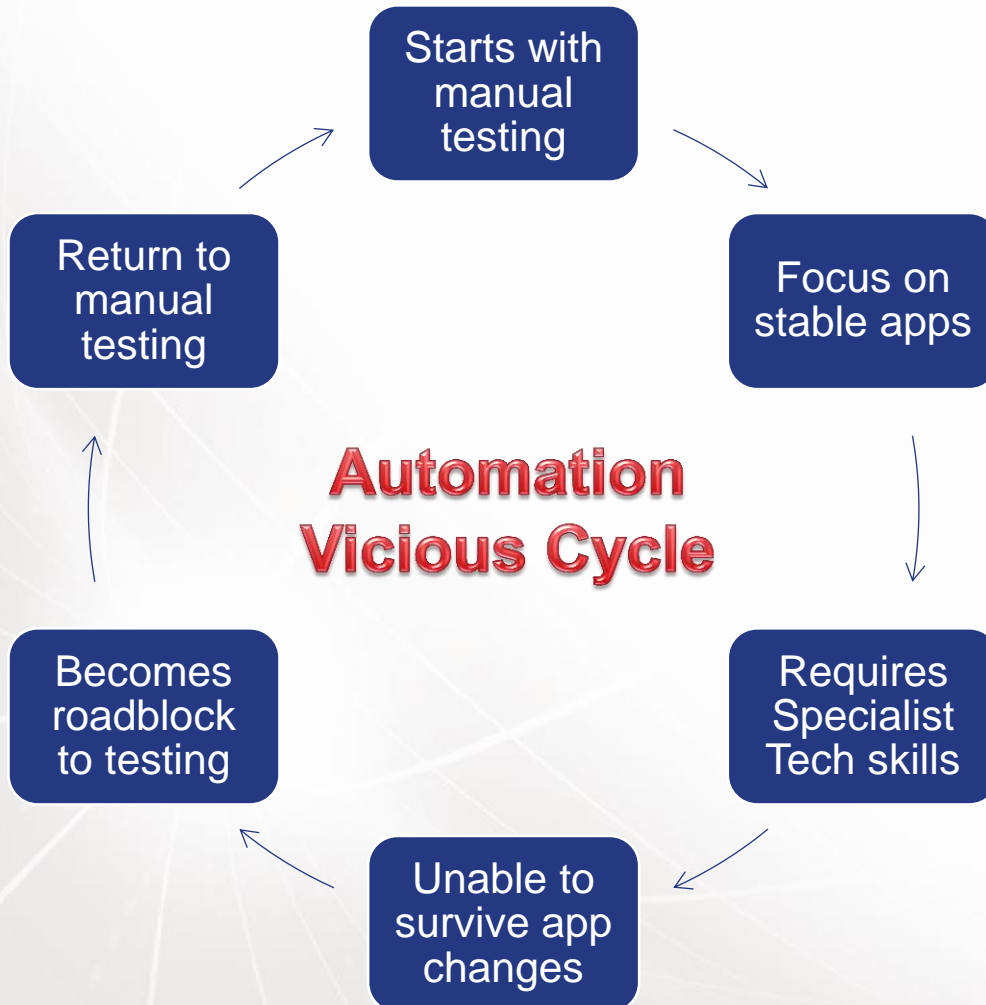
5 Breakout #2

6 Presentation Recap



Automation Antidote

Automation can fix the bottleneck, or be the bottleneck...



Approximately 40 years of expensive SHELFWARE or the LURE of freeware



Key points to know about Test Automation

Differentiation from manual testing

Test Application
Interaction

Accessibility

MyApplication

- Login ID
- Password



ObjectName	TextBox1
ObjectID	25
InnerText	LoginID

Test Case
Development

Modularity

Traditional manual methods:

TC1: Create Work Order
TC2: Update Work Order
TC3: Delete Work Order

Automation methods:

TC1: Create Work Order
TC1.1: Login
TC1.2: Work Order data entry
TC1.3: Confirmation screen
TC1.4: Logout

Test Case
Execution

Reliability

ISSUE

Application Under Test?

Automated Test Cases?



Breakout #1 – Project Team Contributions

Work in small groups

Select one key area for each function below that could be the most beneficial to test automation

Project Managers	Business Analysts
Developers	DBA's
System Testers	IT Management



Highlight of contributions

Key themes for each functional area

Project Managers

- Scope, Time, Cost

Business Analysts

- Technical Requirements for AUT's

Developers

- Application Standardization

DBA's

- Data & Database Integrity

System Testers

- Sets Testing direction

IT Management

- Initiative Champion



Project Managers

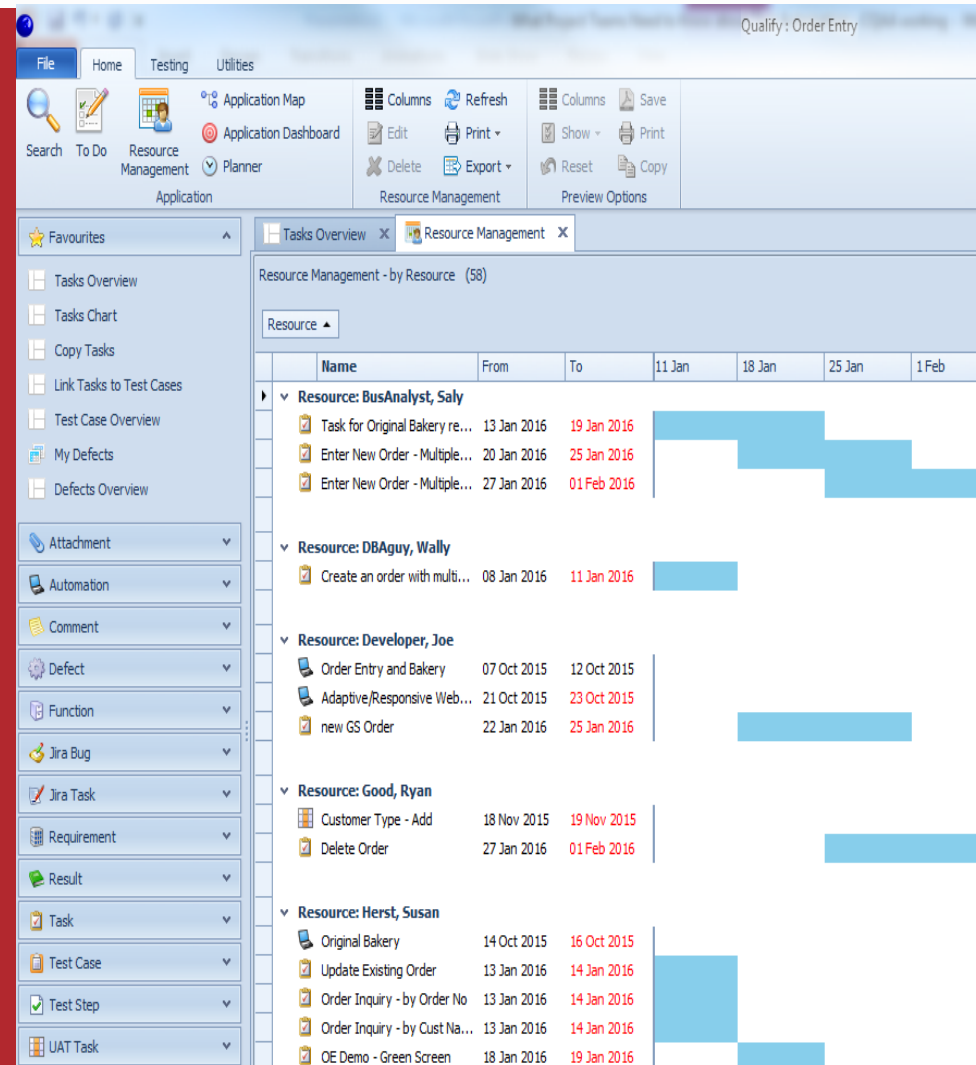
Scope, Time, Cost

Allocates Resources accordingly

Manages Schedules

Maintains Project Budget

Collaborates with Test Manager / Lead



Business Analysts

Technical Requirements: Application(s) Under Test

Leads Requirements Gathering Sessions

Documents Business Rules

Serves as Conduit to UAT

Knowledge of Integrated Systems / Data

The screenshot displays a software application interface with a ribbon menu at the top (File, Home, Testing, Utilities) and a sidebar on the left containing various task categories like Favourites, Attachment, Automation, Comment, Defect, Function, Jira Bug, Jira Task, Requirement, Result, Task, Test Case, Test Step, and UAT Task. The main workspace is divided into two panes. The top pane, titled 'Requirements List (23)', shows a tree view of requirements for 'Order Entry v1.0', including Functional and Non-Functional requirements, with columns for 'Req Name', 'Status', and 'ID'. The bottom pane, titled 'Linked Test Cases (21)', shows a table of test cases linked to the requirements.

Req Name	Status	ID
Order Entry v1.0	Draft	25
Functional Requirements	Draft	26
Creating Orders	Final	28
Customer Types and Customers	Draft	36
Deleting Orders	Final	33
Enhancements	Draft	41
Invoicing	Appr...	39
Maintain Supporting Tables	Final	35
Maintaining Sales Organization	Draft	34
Menu Structure	Final	27
Products and Suppliers	Final	37
Reporting	Final	40
Sales Organization	Draft	38
Updating Existing Orders	Draft	32
Non-Functional Requirements	Draft	42
Original Bakery Website	Draft	47

Description	Test Case Status	Test Type	Test Case ID
Enter New Order - Single Line Item	Final	In-House Appli...	2244
Branch - Add	Final	In-House Appli...	2251
Branch - Delete	Final	In-House Appli...	2252
Enter New Order - Multiple Line Item	Final	In-House Appli...	2245
Branch - Update	Final	In-House Appli...	2253
Salespeople - Add	Draft	In-House Appli...	2254
Delete Order	Final	In-House Appli...	2247
Salespeople - Delete	Draft	In-House Appli...	2255
Salespeople - Update	Draft	In-House Appli...	2256
Customer Type - Add	Draft	In-House Appli...	2257

Developers

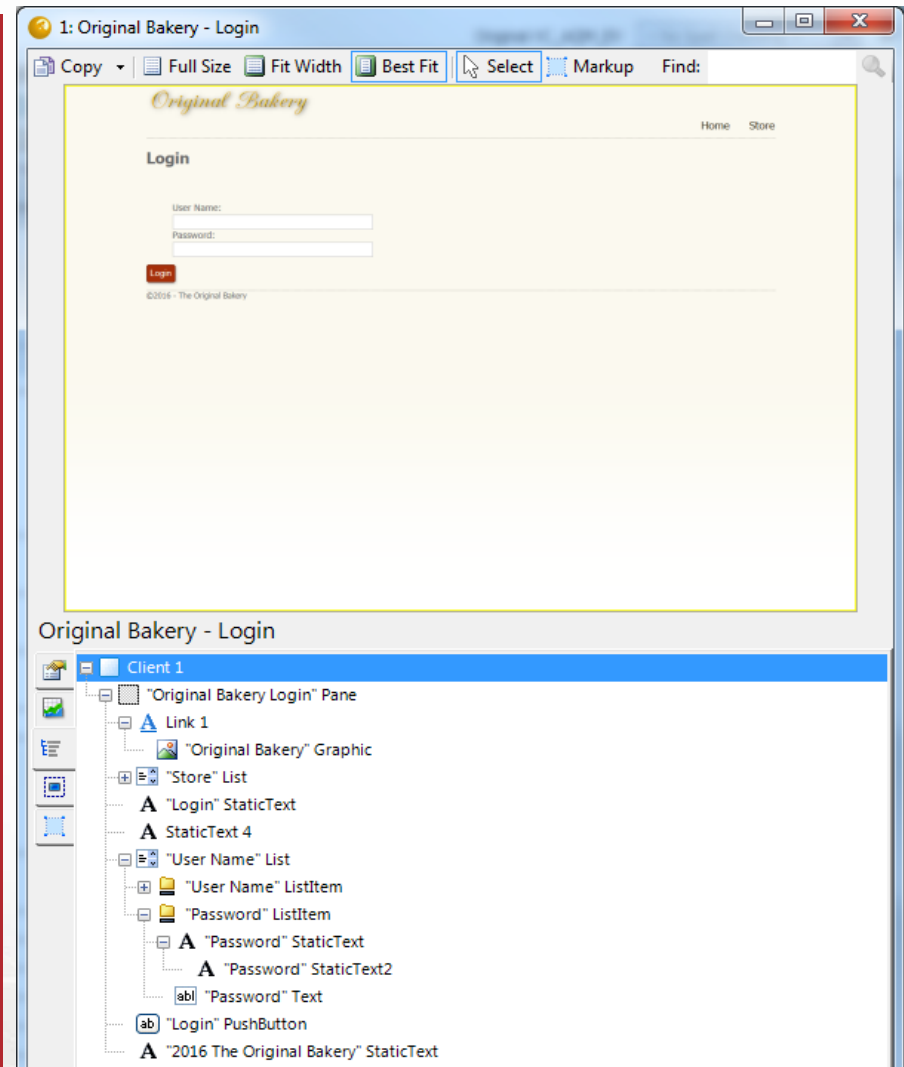
Application Standardization

Keeps Object Naming Conventions

Maintains Objects Accessibility

Debugs Test Cases

Diverts to Write Code Libraries



DBA's

Data & Database Integrity

Maps Database Effects from Tests

Maintains Test Environments

Enables Data Privacy

Compares Tables & Data across DB's

Action Started: 24 May 2011 12:04:44 Action Finished: 24 May 2011 12:04:44
Description: Database Effects created

Database Effects Run: 67

Table Name	Committed Transactions				Rolled Back	
	Total	Writes	Deletes	Updates	Total	Writes
OE_CORE.OEORDHP	2	2	0	0	0	0
OE_CORE.OEORDLP	2	2	0	0	0	0

Time	Column Name	Values Added
12:04:00		
12:04:37	OHCCDT	20110524
	OHCNUM	106
	OHCTYP	NENG
	OHDLDT	20110531
	OHLTMM	120437
	OHLTUP	20110524
	OHNUMB	255
	OHORDT	20110524
	OHPYDT	20110623
	OHSPCD	JP
	OHSPF1	
	OHSPF10	
	OHSPF20	
	OHSPF5	
	OHSTAT	ENT
	OHTOTL	176.00
	OHUNIT	1
	OHUSER	
	Row ID	41604877



System Testers

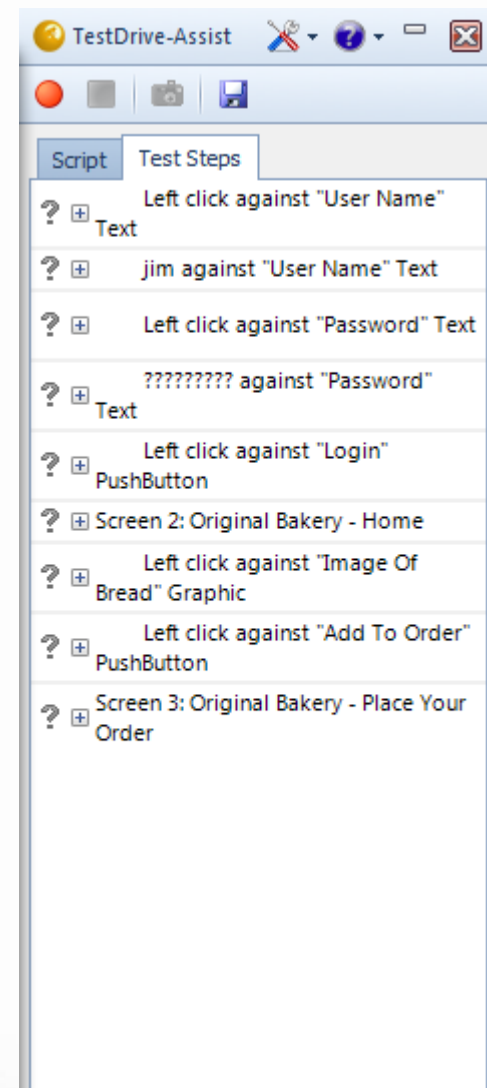
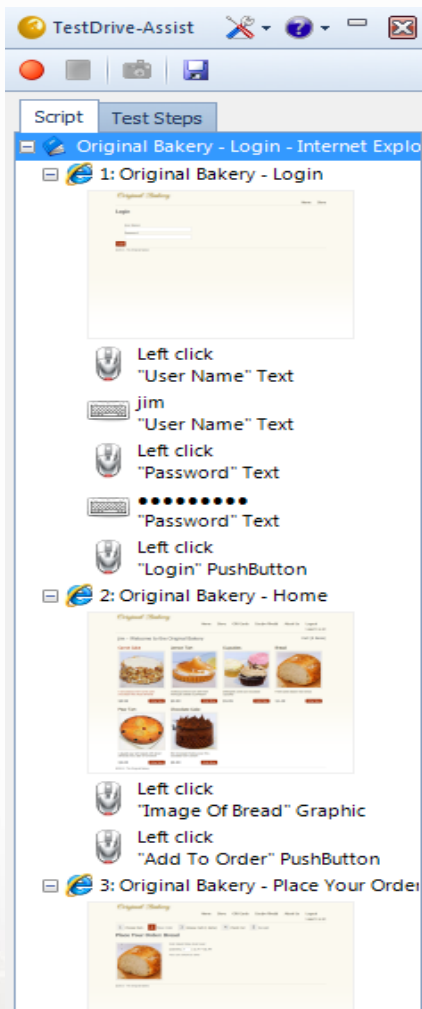
Sets Testing Direction

Provides Manual Test Cases to Automate

Identifies Validation Points in AUT

Captures Step-By-Step details

Documents End-To-End Test Scenarios



IT Management

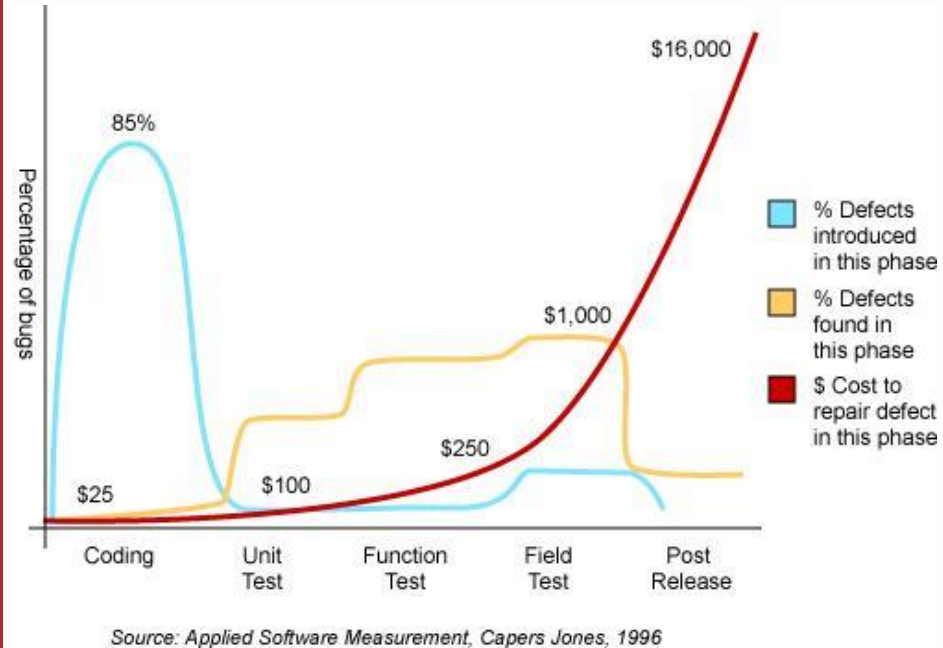
Initiative Champion

Promotes Initiative thru IT Organization

Secures Capital Funding

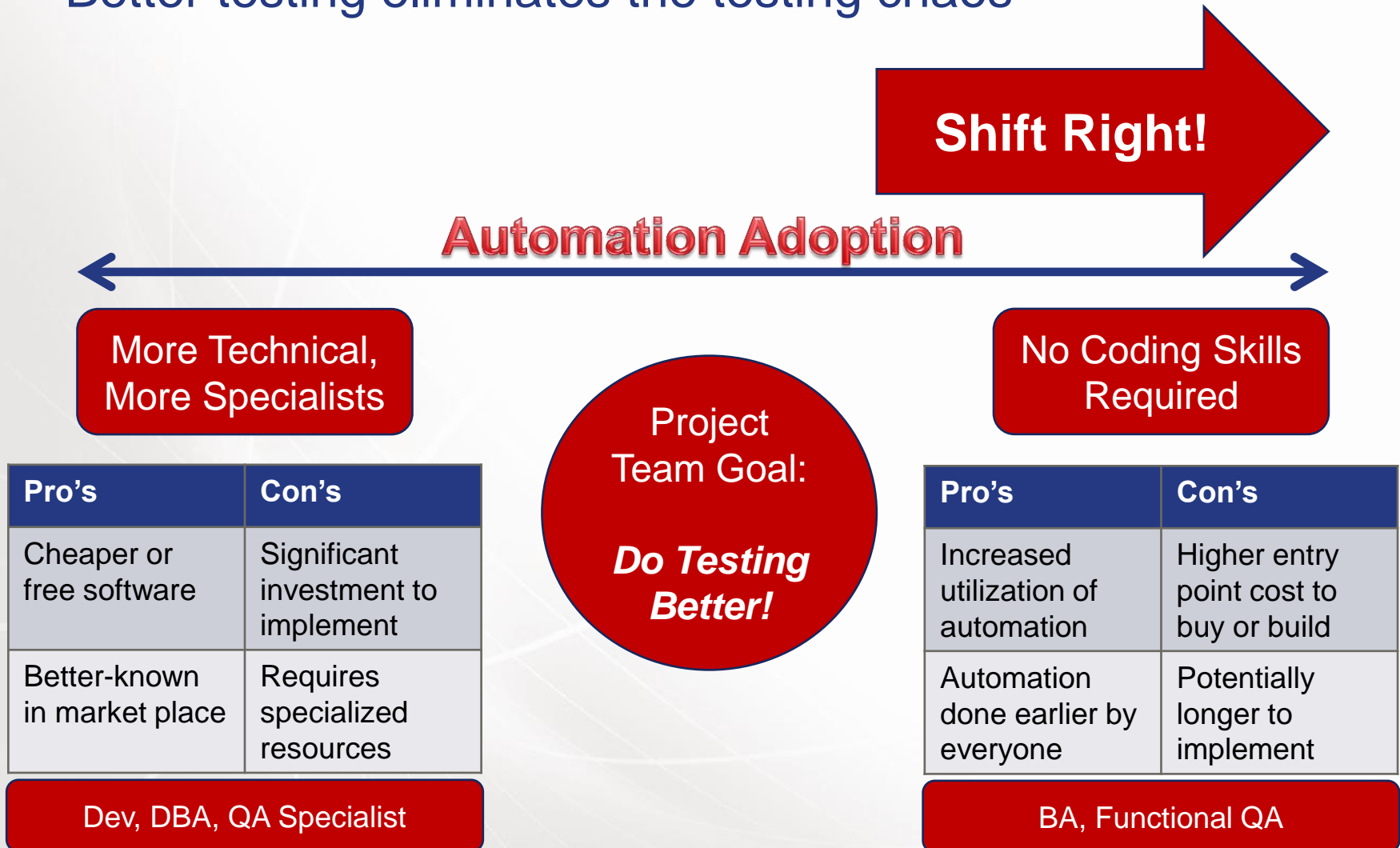
Understands related ROI & KPI's

Sets Multi-Year Vision Plan



Improving testing by phase

Better testing eliminates the testing chaos



Favorable impressions?

What reactions can we expect to with management?

System Testing
Component Testing

- Predictable
- Dependable

Integration Testing
Story Testing

- Thoroughness
- Regularity

Acceptance Testing
Release Testing

- Accountability
- Confidence



Breakout #2 – Individual notes submitted

Ranking Test Phases

Please rank the testing phase that could benefit the most from test automation, with 4 being the highest

Unit Testing

System Testing

Integration Testing (E2E)

User Acceptance Testing

Please rank your top 3 Goonies?

	1	2	3	4	5	6	7
Brand	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Andy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Stef	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mikey	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Data	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mouth	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Chunk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



Presentation Recap

Three key points to take away

- Test Automation can clear the source of the problem or can be the source
- Test Automation demands a wide buy-in to ensure success
- The effort invested can be distributed, saving time and money in the end



Thank you!

Jim Trentadue – Software Quality Consulting Director

jtrentadue@origsoft.com

