

# Choosing What To Automate

Brian Le Suer  
Zeenyx Software, Inc.

## Overview

- Why automate?
- How to choose automation candidates
- Automation examples

## Why Automate?

- Productivity - reduces labor-some task of manual testing
- Repeatability - same tests get executed each time
- Efficiency – shortens test cycles
- Effectiveness – increases test coverage
- Flexibility - late changes can be accepted

## Why Automate?

- Stability – simulate usage over long period of time
- Complacency – over time groups tend to short-cut test cycles
- Environment – provides a means of testing when a component outside application changes
- Accountability – can produce a record of executed test procedures for audit
- Preservation – tester expertise is captured in the automated tests

## Efficiency: Cost Benefit

- **Costs of automation**
  - Test development
  - Maintenance
  - Frequency of change
  - Types of changes
- **Costs of not automating**
  - Manual test time
  - Frequency of change
  - Staff turnover/application expertise

## Efficiency

- **Compare**
  - Length of test cycle with and without automation
  - Automated tests run 24/7

## Effectiveness

- Attention to detail / Accuracy
  - An automated test will fail if a single character is incorrect
- Repeatability
  - Same test every time
- Entire regression can be re-run for each cycle or when a late change must be made

## Prioritizing What To Automate

- |  |                                |
|--|--------------------------------|
| ■ Good candidates  | ■ Poor candidates              |
| – Short or simple transactions                               | – Long or complex transactions |
| – Universal tests  | – One-offs                     |
| – Many data combinations                                     | – Unstable application         |
| – Expected results are stable or easy to generate at runtime | – Difficult to predict results |
| – Tasks that are difficult to do manually                    |                                |
| – Tests that are executed regularly                          |                                |
| – Highest priority features                                  |                                |

## Short or Simple Tests

- Tests that can be executed with minimal navigation
  - Data type validations
  - Data content validations
    - Invalid password
    - record not found
  - Missing required fields

## Example:

- POS Credit Card Application
  - Single page
  - Lots of data validation
    - Field formats
      - Dates
      - Phone numbers
      - Social Security numbers
    - Invalid character edits
    - Missing required fields
    - Confirmation fields
  - More than 1000 combinations

## Universal Tests

- Tests that can be run against each window or page in the application under test
  - Standards
    - Labels must include a colon
    - Dialogboxes must include Closebox
    - Radio group can not have more than 3 choices
  - Z order
  - Object size
    - Min and max button size

## Many Data Combinations

- Path through the application is shared by many tests
- Lots of different combinations of data need to be tested
- Write a reusable data-driven test
  - Separate test steps from test data
  - Relatively little work for a lot of coverage
  - Could pay for themselves in first run

## Example:

- Automate cashier functions at POS: sales, returns, exchanges
  - Definite path through the application
  - Easy to encapsulate GUI steps
  - Transactions executed with
    - Different tenders
    - Different items
    - In different states (taxes)
    - With different discounts

## Expected Results

- Static results – same result every time the test is run
- Predict results at runtime based on input parameters and look-ups
  - More difficult to create tests
  - but
  - Self-maintaining

## Example:

### ■ Retail web site order entry

- Products can be selected randomly or specifically
- Product attributes such as SKU, Description, Qty and Price can be captured as part of the test
- Same product attributes can be captured from the shopping cart, review order and submitted order pages
- All attributes can be written to a data store such as an .ini file or spreadsheet
- A universal function can be used to compare predicted and actual results

Predicted Results:  
[RegularSale1]  
Item3\_Style=528534  
Item3\_Price=29.99  
Item3\_Taxable=N  
Item3\_Dept=40  
TaxState=MA  
TaxRate=5.00  
SubTotal=29.99  
Total=31.49

Shopping Cart Capture:  
[RegularSale1]  
Item3\_Style=528534  
Item3\_Price=29.99  
Item3\_Taxable=N  
Item3\_Dept=40  
TaxState=MA  
TaxRate=5.00  
SubTotal=29.99  
Total=31.49

Review Order Capture  
[RegularSale1]  
Item3\_Style=528534  
Item3\_Price=29.99  
Item3\_Taxable=N  
Item3\_Dept=40  
TaxState=MA  
TaxRate=5.00  
SubTotal=29.99  
Total=31.49



## Difficult to Execute Manually

- Lots of detail to verify
- Need to execute repeated iterations
- Multiple concurrent users
  - Stress
  - Performance

## Example 1:

- Software Installation
  - Same test must be run on multiple environments/platforms
  - Testing manually is error prone because of the task is tedious on the one hand yet requires acute attention to detail
  - Lots of detail to verify
    - Data files installed
    - Registry settings made
    - Permissions set
    - Groups and shortcuts created
    - Application configurations

## Example 2:

- Simulate multiple terminals to ensure proper handling of concurrent transactions
  - Difficult to synchronize timing
  - Requires lots of resources
  - Difficult to repeat when problems are found

## Relative Importance

- High priority tests
  - Core features
  - Data integrity
  - Performance
- Tests that are executed frequently
  - Repeatability
  - Accuracy

## Other Considerations

- Application stability
  - Does application crash frequently
- Control of the test environment
  - Can testers anticipate changes
- Frequency of change (Agile)
  - Automation approach must promote flexibility and test maintenance

## Other Uses Of Automation

- Creating test case data
- Performing routine tasks
- Keeping a secure machine from logging off
- Executing setup steps
- Cleaning up after manual or automated testing
- Capturing monitor data on system

Thank you!  
Questions?

Zeenyx Software



**Software Quality Group of New England**

SQGNE is made possible by the support of our sponsors:



Oracle  
and  
Sun Integrated Systems



consulting • training • auditing

Logo design: Sarah Cole Design

Slide 1

## Welcome to our 16<sup>th</sup> season!

- An all-volunteer group with no membership dues!
- Supported entirely by our sponsors...
- Over 700+ members
- Monthly meetings - Sept to July on 2<sup>nd</sup> Wed of month
- E-mail list - contact John Pustaver [pustaver@ieee.org](mailto:pustaver@ieee.org)

■ **NEW SQGNE Web site: [www.sqgne.org](http://www.sqgne.org)**



Slide 2

## Volunteers / Hosts / Mission

**Volunteers**


- John Pustaver - Founder and Director
- Steve Rakitin – Programs and web site
- Gene Freyberger – Annual Survey
- Dawn Wu – Greeter

**Our gracious Hosts:**

- Paul Ratty
- Tom Arakel
- Margaret Shinkle
- Jack Guilderson

**Mission**




- To promote use of engineering and management techniques that lead to delivery of high quality software
- To disseminate concepts and techniques related to software quality engineering and software engineering process
- To provide a forum for discussion of concepts and techniques related to software quality engineering and the software engineering process
- To provide networking opportunities for software quality professionals




Slide 3

## ASQ Software Division

- Software Quality Live - for ASQ SW Div members...
- Software Quality Professional Journal [www.asq.org/pub/sqp/](http://www.asq.org/pub/sqp/)
- CSQE Certification info at [www.asq.org/software/getcertified](http://www.asq.org/software/getcertified)
- SW Div info at [www.asq.org/software](http://www.asq.org/software)








Slide 4

## SQGNE 2009-10 Schedule

Speaker	Company/Affiliation	Date	Topic
Eric Lotter	Surgient	9/9/09	Using Virtualization to Accelerate Quality/Test Cycles
Steve Rakitin	Software Quality Consulting	10/14/09	Software Quality Assurance Turns 50 A Critical Look at the Profession
Howie Dow and Steve Rakitin		11/11/09	Interactive Requirements Exercise...
Michael Mah	QSM Associates	12/9/09	Rightsizing Your Project in a Down Economy
Robin Goldsmith	GoPro Management	1/13/10	I went to a Testing Conference and all they talked about was Requirements
Stan Wrobel	CSC	2/10/10	Requirements Tracing - Lessons from the Waterfall for the Agile and SCRUM teams
Billie Bell	Intuit	3/10/10	End-to-End Testing in a SaaS environment: Extending the Definition of Quality
Michael Mah	QSM Associates	4/14/10	Rightsizing Your Project in a Down Economy
Urvashi Tyagi	Microsoft	5/12/10	A day in the life of a tester at Microsoft...
Brian LeSuer	Star Quality	6/9/10	To be announced...
Everyone		7/14/10	Annual Hot Topics Night...



Slide 5

## Tonight's Speaker...


### A Day in the Life of a Tester at Microsoft

Urvashi Tyagi  
Software Test Manager, Microsoft

You will learn about what a software tester does at Microsoft, how the role interfaces with program managers and developers, what it takes to be successful in this role, various tools we use, and some specifics about Microsoft NERD (New England Research and Development Center)

**Bio:**

Urvashi currently works at Microsoft as a Senior Test Lead/Manager for Microsoft's Application Virtualization team based in Cambridge, MA. Prior to joining the team in September 2008, Urvashi was a Team Lead and Test Architect at IBM Software group working on Rational line of products. Prior to IBM, she gained software experience at NuGenesis, a technology startup and at Indian Institute of Management. She is a frequent speaker at sales and user conferences talking about product deployments, testing and debugging. Urvashi holds a M.S degree in Information Systems from Worcester Polytechnic Institute, M.B.A in Finance and Information Systems, and a Bachelor in Engineering in Mechanical and Computerized automation systems.



Slide 6