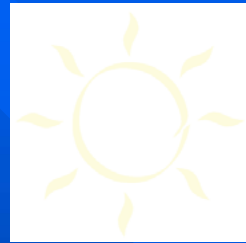


Selecting a Test Automation Tool



Presented by:

Sam Mathi

First Contact Software Consultants Inc.

www.firstcontactconsulting.com

sammathi@firstcontactconsulting.com

(416) 225-7951

Abstract

- There are many test automation tools available in the market. Each vendor offers a variety of features. How do one choose the right tool to make sure you are addressing the testing needs of your organization.
- This presentation offers practical advice for selecting the right test automation tool and for implementing automated testing practices within your organization.

Overview

- What is Software Test Automation?
- Why Is Software Test Automation Important?
- Software Test Automation Process
- Needs, Issues, and Challenges
- Classification of Software Test Tools
- Test Automation and Tools for Software

What is Test Automation?

- The activities and efforts to automate engineering tasks and operations in a software test process using well-defined strategies and systematic solutions.

Objectives of Test Automation

- Free engineers from tedious and redundant manual testing operations
- Speed up the software testing process and reduce software testing costs during a software life cycle
- Increase the effectiveness of the testing process by achieving pre-defined test criteria in a limited schedule

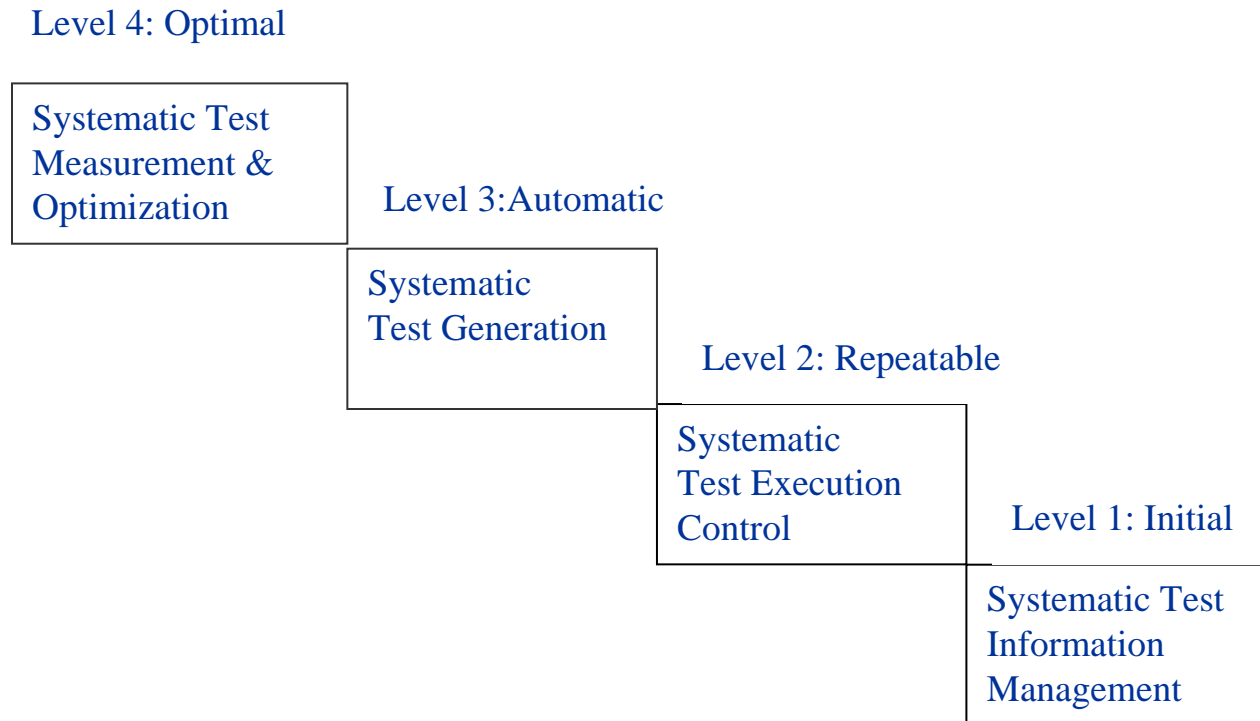
Key to Successful Test Automation

- Use a systematic testing solution to achieve better testing coverage, allowing you to reduce manual and redundant testing activities

Test Automation Scope

- *Enterprise-oriented test automation* - focus of automation effort is to automate an enterprise-oriented test process so that it can be used and reused with different product lines and projects within the organization
- *Product-oriented test automation* - automation activities are focused on a specific software product line to support its related testing activities
- *Project-oriented test automation* - automation effort is aimed at a specific project and its test process.

Test Automation Maturity Levels



Test Automation Maturity Levels

➤ Level 1: Initial

- Provides engineers with systematic solutions and tools to create, update, and manage all types of test information, including test requirements, test cases, test data, test procedures, test results, test scripts, and problem reports. No systematic solutions or tools are available to support engineers' test design, test generation, and test execution.

➤ Level 2: Repeatable

- Provides engineers with tools to manage diverse testing information, but also provides solutions to execute software tests in a systematic manner. These solutions allow engineers to use a systematic approach to run tests and validate test results. However, no systematic solutions and tools are available to assist test engineers in test design, test generation, and test coverage measurement.

➤ Level 3: Automatic

- Provides engineers with solutions to generate software tests using systematic methods. They could be useful to generate black box or white-box software tests. However, no systematic solutions are available to measure the test coverage of a test process.

Test Automation Maturity Levels

➤ Level 4: Optimal

- Systematic solutions are available to manage test information, execute tests, generate tests, and measure test coverage. The primary benefit of this level is to help engineers understand the current coverage of a test process, and identify the test coverage issues.

Essential Needs of Test Automation

- Dedicated work force for test automation
- Commitment from senior managers and engineers
- Dedicated budget and project schedule
- Well-defined plan and strategy
- Talent engineers and cost-effective testing tools
- Maintenance of automated tests and tools

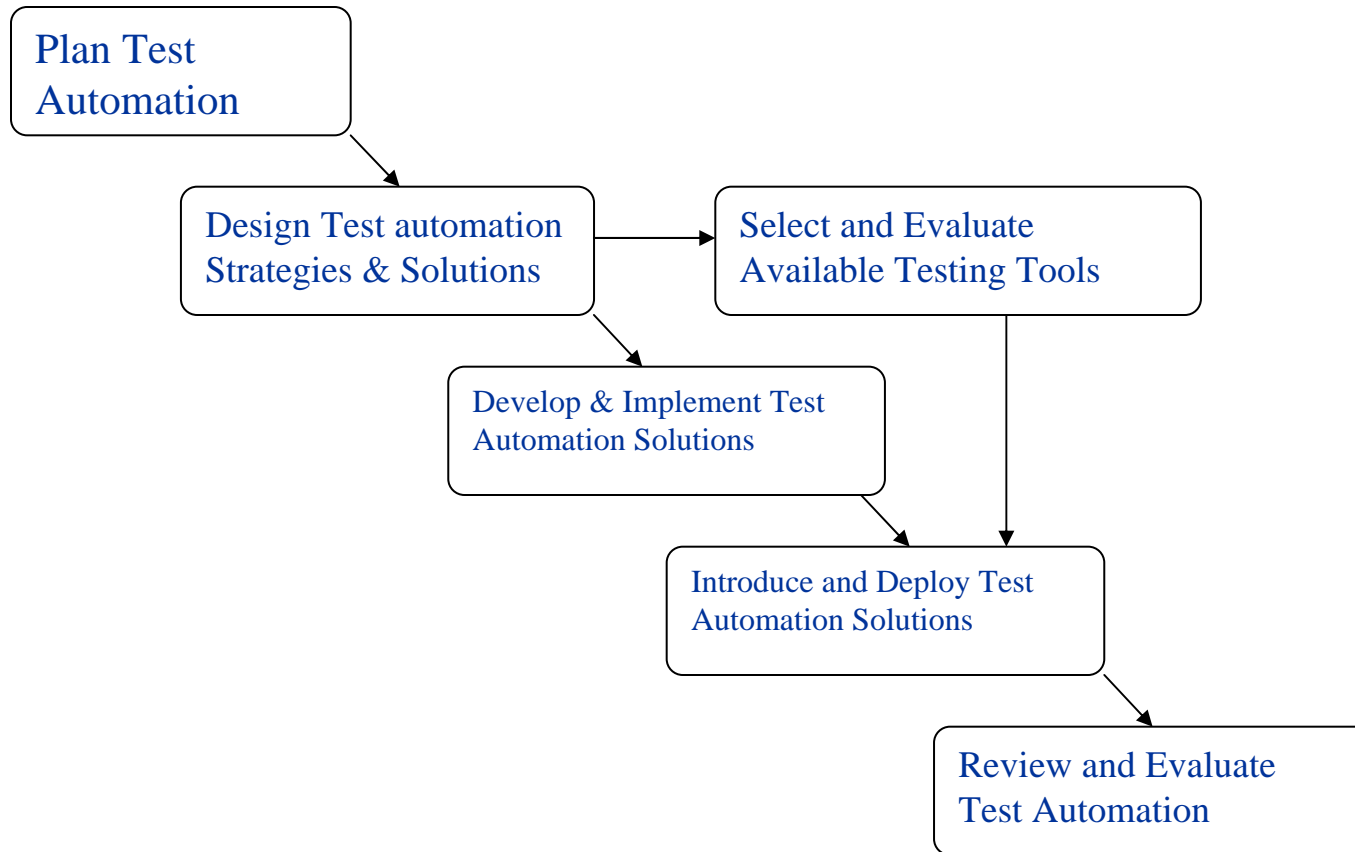
Basic Issues of Test Automation

- Poor manual test process
- Test automation is engaged late in the product life cycle
- Unrealistic goals and expectations from automated testing
- Organization issues
- Lack of understanding and experience of test automation

Benefits of Test Automation

- Reduce manual software testing operations and eliminate redundant testing efforts
- Repeatable tests and consistent testing results
- Execute much more software tests and achieve a better testing coverage in a very limited schedule

Test Automation Process



Test Automation Process

- **Step #1: Plan Test Automation**
 - Design a plan that specifies the identified test automation focus, objectives, strategies, requirements, schedule and budget
- **Step #2: Design Automation Strategy**
 - Design detailed test automation solutions to achieve the major objectives and meet the requirements in the test automation plan
- **Step #3: Develop Test Automation Solutions**
 - Develop and test the automation solutions. The key is to make sure that the developed tools are reliable and reusable, with detailed documentation

Test Automation Process

➤ **Step #4: Deploy Automation Solutions**

- Developed test tools and facilities are introduced into a project or onto a product line. At this step basic user training is essential, and proper user support is necessary

➤ **Step #5: Review and Evaluation**

- Whenever a new tool is deployed, a review should be conducted to identify its issues and limitations, and to evaluate its provided features. The review will provide valuable feedback to the test automation group for further improvements and enhancements

Test Automation Best Practices

- Focus on the Methodology, Not the Tool
- Choose Extensible Test Tools
- Separate Test Design and Test Automation
- Lower Costs
- Jumpstart with a Pre-Trained Team

Test Tool Selection Criteria

- The “Best” Testing Tool Depends On:
 - Type of Testing to be Done
 - Application/Environment to be Tested
 - Limitations of the Tool
 - Features of the Tool
 - Cost of the Tool
 - ☞ Includes cost of licenses, support, and maintenance
 - Ease of Use
 - ☞ Scripting language
 - ☞ Required user skills

Test Tool Classification

Test Tool Types	Basic Descriptions of Different Types of Test Tools
Test Information Management	Systematic solutions and tools to support test engineers and quality assurance people to create, update, and maintain diverse test information, including test cases, test scripts, test data, test results, and discovered problems.
Test Execution and Control	Systematic solutions and tools to help engineers set up and run tests, and collect and validate test results.
Test Generation	Systematic solutions and tools to generate program tests in an automated way.
Test Coverage Analysis	Systematic solutions and tools to analyze the test coverage during a test process based on selected test criteria.
Performance Testing and Measurement	Systematic solutions and tools to support program performance testing and performance measurement.
Software Simulators	Programs are developed to simulate the functions and behaviors of external systems, or dependent subsystems/components for an application under test.
Regression Testing	Test tools to support the automation performance of regression testing and activities, including test recording and re-playing.

Test Tool Classification

- Three types of test information management systems and tools:
 - Test Information Management Tool
 - ☞ Allows engineers to create, update, and manage all types of test information such as test requirements, test cases, test data, procedures, and results. HP's *Quality Center* is an example.
 - Test Suite Management Tool
 - ☞ Enables engineers to create, update, and manage various software test scripts for test execution.
 - Problem Management Tool
 - ☞ Helps engineers with bookkeeping and managing the discovered problems during a test process.

Test Tool Classification

➤ Test Execution Tools

- Programs developed to control and execute program tests and test scripts automatically. Usually consist of the capability to set up the selected test scripts and test data, invoke and execute them, and validate the test results based on the expected testing outputs. HP's *QuickTest Professional* is a typical example.

➤ Test Generation Tools

- Programs that generate tests for an application under test using a systematic solution. There are two classes:
 - ☞ *White-box test generation tools* – Generate tests based on program source code and structures using program-based test models and methods.
 - ☞ *Black box test generation tools* – Generate tests based on program requirements using black box test methods, such as random testing and boundary value analysis.

Test Tool Classification

➤ Software Simulators

- Programs developed to simulate the functions and behaviors of external software/hardware entities, components, or subsystems.
- Useful for program integration and system testing.
- Three types of software simulator:
 - ☞ *Model-driven program simulator* – Simulates the behavior of a program process based on a specific model, such as a finite state machine.
 - ☞ *Data-driven program simulator* – Simulates the behavior of a functional program based on the given input and returns the pre-defined output.
 - ☞ *Message-driven program simulator* – Simulates the behavior of a communication process to generate the protocol-oriented outgoing messages, based on the pre-defined incoming messages.

Test Tool Classification

- Regression Test Tools:
 - *Software Change Analysis* – Systematic facility that identifies various types of software changes and discovers ripple effects and impacts.
 - *Software Test Selection* – Systematic facility that assists engineers in selecting reusable tests for regression testing based on software change information.
 - *Test Change Analysis* – Systematic solution that identifies the necessary test changes based on the given software changes. The major task of test change analysis is finding reusable tests and identifying obsolete tests so that new tests can be added and existing tests updated.
 - *Test Recorder And Playback* – Software tool that records the manual test steps and replays them for testing. HP's *QuickTest Professional* is a typical testing tool that is able to record and playback user-system interactive sequences using recorded test scripts in a variety of application environments.

Test Tool Classification

Types of Test Tools	Test Tool Vendors	Test Tools
<i>Problem Management Tools</i>	<i>IBM Rational Inc.</i>	ClearQuest, ClearDDTS
	<i>Synergex</i>	PVCS Tracker
<i>Test Information Management Tools</i>	<i>IBM Rational Inc.</i>	TestManager
	<i>HP</i>	Quality Center
<i>Test Suite Management Tools</i>	<i>eValid</i>	eV.Manager
	<i>SUN</i>	JavaTest Harness
	<i>McCabe & Associates</i>	McCabe IQ2
<i>White-Box Test Tools</i>	<i>ezTester</i>	VcSmith, VcTester

Test Tool Classification

Test Execution Tools	<i>Softbridge</i>	ATF/TestWright
	<i>IBM Rational Inc.</i>	Visual Test, Robot
	<i>HP</i>	QuickTest Professional
	<i>Compuware</i>	QARun
	<i>Borland Software</i>	SilkTest
Code Coverage Analysis Tools	<i>Case Consult Corp.</i>	CC Analyzer
	<i>IPL Software Product Group</i>	Cantata++, Klocwork
	<i>IBM Rational</i>	PureCoverage
	<i>Compuware</i>	TrueCoverage
	<i>Software Research</i>	TestWorks Coverage
	<i>SUN</i>	JavaScope
	<i>ParaSoft</i>	TCA
	<i>Software Automation Inc</i>	Panorama

Test Tool Classification

Load Test and Performance Tools	<i>IBM Rational Inc.</i>	Rational Suite PerformanceStudio
	<i>InterNetwork AG</i>	smartTest
	<i>Compuware</i>	QALoad
	<i>HP</i>	LoadRunner
	<i>SUN</i>	JavaLoad
	<i>Borland Software</i>	SilkPerformer
	<i>Quest Software</i>	Benchmark Factory
Regression Testing Tools	<i>IBM</i>	Regression Testing Tool (ARTT) Distillation Assistant
GUI Record/Replay	<i>Software Research</i>	eValid
	<i>HP</i>	QuickTest Professional

Functional Testing Tool

- Common features among all the testing tools
 - Record and playback of scripts
 - Object recognition abstraction to accommodate application changes
 - Recovery scenarios to handle unexpected circumstances
 - Data-driving capability to improve test reusability; iterations
 - Each product offers “advanced” reporting capabilities

Functional Testing Tool Specifics

- HP QuickTest Professional
 - Record and playback of scripts
 - Scripting language: VBScript
 - “Smart Object Identification” features
 - Data-driven testing capabilities
 - “Recovery Scenario” feature to respond to unexpected situations
 - “Active Screen” to allow test development while application is unavailable
 - Default data format: Excel
 - Extensive application/environment support

Functional Testing Tool Specifics

- Compuware TestPartner
 - Record and playback of scripts
 - Scripting language: VisualBasic for Applications (VBA)
 - Wizard interface to programming structures
 - Team collaboration features
 - Features to validate all controls on a page simultaneously
 - Data-driven testing capabilities
 - “Screen Preview” to allow test development while application is unavailable
 - Extensive application/environment support

Functional Testing Tool Specifics

- Borland/Segue SilkTest
 - Record and playback of scripts
 - 4Test scripting language: object-oriented, designed for testing
 - Ability to write 4Test functions in C
 - Ability to script tests in Java
 - Built-in support for distributed test execution (can install just a “run-time” to execute tests)
 - Recovery features enabled by default (eg/ respond to application crash)

Functional Testing Tool Specifics

- IBM/Rational Visual Test
 - Record and playback of scripts
 - Ability to define and automate keywords that can be reused in functional and manual tests
 - Choice of Visual Basic .Net or Java scripting language: standard, full-featured languages
 - Integration with Eclipse development environment; consistent interface if already familiar with Eclipse
 - Supports test script version control
 - Proxy SDK allows support for custom controls
 - Support for scripting language's native debugging tools: Eclipse for Java and Visual Studio for VB
 - Nearly full support for Linux (all features except test recording)

Functional Testing Tool Specifics

- IBM/Rational Robot
 - Scripting language SQABasic: integrated environment allowing users to view and edit the script while it is being recorded

Functional Testing Tool Specifics

- Seapine QA Wizard
 - Scripting language: Visual Basic-like
 - Grid view and Text view available
 - URL formatting rules to ease handling of dynamic application URLs
 - Object repository is stored in a central location so all testers have access to it (rather than “local object repositories”)
 - Support for storing test data outside of QA Wizard, including: Access, Excel, Oracle, SQLServer, text files
 - Built-in support for remote/batch script execution
 - Integrates with source control systems such as Surround SCM and Visual SourceSafe
 - Integrates with TestTrack TCM and TestTrack Pro
 - Supports Internet Explorer and Firefox

Functional Testing Tool Specifics

➤ Selenium

- Record and playback of test scripts
- Open Source (licensed under the Apache 2.0 License)
- Scripts are recorded in Firefox (most others require Internet Explorer) and can be played back in most browsers: Firefox, Internet Explorer, Safari, Opera
- Scripts can be generated in Ruby, PHP, Java, Perl, C#, Python: flexible choice of languages, and possibility to incorporate Selenium scripts into other tools/frameworks
- Ability to generate scripts in any language by writing custom code generators
- Supports Windows, Linux, Solaris, MacOS

Functional Testing Tool Specifics

- Automated QA TestComplete
 - Support for creating/executing manual tests
 - Built-in support for many scripting languages: VBScript, Jscript, C++Script, C#Script, DelphiScript
 - Supports load, stress, and scalability testing
 - Support for distributed testing to create multi-user tests
 - Support for unit testing frameworks: MSTest, Junit, Nunit, Dunit
 - Available API to control TestComplete from another application.
 - Integrates with source control systems: Visual SourceSafe, Team Coherence, CVS, and others
 - TestExecute component can be installed to use a machine for execution without having to install the full product
 - Ability to embed script recording capability into your application

Load Testing Tool Specifics

- Common features
 - Record and playback of scripts
 - Rich set of performance monitors to track performance
 - real-time view of collected monitor data
 - light-weight virtual users to allow high number of virtual users with minimal resource usage
 - Automatic correlation of dynamic data
 - Test data variation features (parameterizing data)
 - Scheduling capabilities to customize the load test
 - Capture of HTML pages/screenshot during record

Load Testing Tool Specifics

- Compuware QALoad
 - Scalable Testing
 - Streamlined Test Script Development
 - Comprehensive Analysis
 - Integrated View of System Resources
 - Supported Technologies

Load Testing Tool Specifics

- IBM/Rational Performance Tester
 - Performance testing on Windows, Linux, UNIX, and mainframe
 - Integrates with IBM Rational Software Delivery Platform: to ensure a shared user experience across team disciplines
 - Script language: Java
 - Script can be presented as a hierarchical sequence of web pages, so users can avoid needing to understand scripting
 - Root Cause Analysis tools can help identify the physical application tier and source code causing a bottleneck
 - Support for Windows and Linux for both test recording and execution
 - Support for Entrust TruePass Technology

Load Testing Tool Specifics

- Oracle/Empirix e-Load
 - Automates testing of the most complex Web applications and Web services with robust test scripts
 - Simulates hundreds to tens of thousands of users while minimizing the hardware required
 - Gathers critical performance metrics to identify bottlenecks
 - Simplifies accessibility with an intuitive Web based user interface
 - Allows distributed users to share testing results during live testing

Load Testing Tool Specifics

- Radview WebLoad
 - Performance testing functions test the response time of a given request while the system is under load.
 - Scalability testing functions determine the parameters for planning hardware procurement in the face of growing concurrent requests.
 - Reliability testing reveals potential problems arising from extended runs.
 - Specially designed to support testing of Web 2.0 internet applications, including multimedia and Adobe Flex based sites. WebLOAD Professional has built-in support for AJAX technologies, JSON data types and different flavors of SOAP and XML web services.

Load Testing Tool Specifics

- HP LoadRunner/Performance Center
 - Script language: ANSI C
 - Scripts can be executed from Windows or Linux (Load Generators)
 - Supports web, client/server and mainframe testing

Open Source Test Tools

- Open Source software does not always mean “free”
- Costs can be incurred through evaluation, implementation and maintenance
- Open source test tools can save hundreds and sometimes thousands of up front costs

Open Source Test Tools

- Open source test tools should be seriously considered, especially for smaller enterprises
- A cost effective alternative to commercial test tools
- Open source test tools are usually created with a specific environment or application in mind, commercial test tools are made to work with any number of environments

Open Source Functional Test Tools

Name	Summary	Limitations	Cost	Source	Training
Abbot	<ul style="list-style-type: none"> • automated event generation and validation • can be invoked from Java code or executed through the use of scripts • uses Costello Script Editor, which can record and play back user actions • operating system independent 	<ul style="list-style-type: none"> • for Java based GUI components • knowledge of Java events and listeners necessary 	<ul style="list-style-type: none"> • N/A • can contribute by writing user manuals or UI 	http://abbot.sourceforge.net	<ul style="list-style-type: none"> • No training courses • User guides and Tutorials
Anteater	<ul style="list-style-type: none"> • sends an HTTP/HTTPS request and checks the response meets required criteria • check for headers, response codes, parameters and contents • built-in web server suitable for testing web services 	<ul style="list-style-type: none"> • for Web and XML applications 	<ul style="list-style-type: none"> • N/A 	http://aft.sourceforge.net	<ul style="list-style-type: none"> • No training courses • User manuals

Open Source Functional Test Tools

Name	Summary	Limitations	Cost	Source	Training
AutoTestFlash	<ul style="list-style-type: none"> record and playback capable 	<ul style="list-style-type: none"> for Flash and Flex applications can record only one application at a time drag and drop not supported onMouseDown / Up /Over cannot be captured 	<ul style="list-style-type: none"> N/A 	http://tiago.webstartpoint.net/flash	<ul style="list-style-type: none"> No training courses Very little user guides
DejaGNU	<ul style="list-style-type: none"> testing applications written in TCL, C, C++, Java and network applications uses a “test harness”, which tries to provide a single front end for a wide range of applications 	<ul style="list-style-type: none"> written in Expect language 	<ul style="list-style-type: none"> N/A Can volunteer to add components or write documentation 	http://www.gnu.org/software/dejagnu	<ul style="list-style-type: none"> No training courses Online manual

Open Source Functional Test Tools

Name	Summary	Limitations	Cost	Source	Training
Jemmy	<ul style="list-style-type: none">• create automated tests for Java GUI• used for big, complicated and dynamic Java GUI applications	<ul style="list-style-type: none">• no recording	<ul style="list-style-type: none">• N/A	<p>http://jemmy.netbeans.org http://jemmy.dev.java.net</p>	<ul style="list-style-type: none">• No training courses• Online documentation
MaxQ	<ul style="list-style-type: none">• similar engine and paradigm as Quick Test, but at a fraction of the cost• record and playback capable	<ul style="list-style-type: none">• specific to web applications	<ul style="list-style-type: none">• N/A• Sign up for free membership	<p>http://maxq.tigris.org</p>	<ul style="list-style-type: none">• No training courses• Online guide

Open Source Functional Test Tools

Name	Summary	Limitations	Cost	Source	Training
STP (Scalable Test Platform)	<ul style="list-style-type: none"> record and playback capable used for benchmarking and regression testing on diverse hardware systems 	<ul style="list-style-type: none"> only for Linux 	<ul style="list-style-type: none"> N/A Sign up as an associate (free, contribute to the software) 	http://devresources.linux-foundation.org/dev/stp	<ul style="list-style-type: none"> No Training courses Online guide
SharpRobo	<ul style="list-style-type: none"> functional testing tool for WinForm applications written in C# supports all standard WinForm controls 	<ul style="list-style-type: none"> only for WinForm controls 	<ul style="list-style-type: none"> N/A 	http://gforge.public.thoughtworks.org/projects/sharp-robo	<ul style="list-style-type: none"> No Training courses Online doc manager

Open Source Performance Test Tools

Name	Summary	Limitations	Cost	Source	Training
Apache JMeter	<ul style="list-style-type: none"> • can test both static and dynamic resources (files, servlets, objects, databases, ftp servers) • simulates load on server, network or objects • supports Swing and Multithreading 	<ul style="list-style-type: none"> • 100% Java 	<ul style="list-style-type: none"> • N/A • Contribute to the software 	http://jakarta.apache.org/jmeter/	<ul style="list-style-type: none"> • No Training courses • Online Manual, Javadocs, FAQ, Tutorials
DOTS (Database Opensource Test Suite)	<ul style="list-style-type: none"> • Pre-defined set of test cases to stress test a database 	<ul style="list-style-type: none"> • pre-defined, not readily modifiable 	<ul style="list-style-type: none"> • N/A 	http://ltp.sourceforge.net/	<ul style="list-style-type: none"> • No Training courses • Online documentation

Open Source Performance Test Tools

Name	Summary	Limitations	Cost	Source	Training
DB Monster	<ul style="list-style-type: none"> • application to generate random data then inserts into a SQL database for testing under heavy load • OS independent 	<ul style="list-style-type: none"> • does not keep a collection of possible data to use in the db (ie. Set of data given by user) 	<ul style="list-style-type: none"> • N/A • Testing the application and writing dictionaries 	http://dbmonster.kearnelpanic.pl	<ul style="list-style-type: none"> • No Training courses • Online project documentation
DieselTest	<ul style="list-style-type: none"> • simulates hundreds or thousands of users hitting a website • shows real-time results as script executes • record and playback capable 	<ul style="list-style-type: none"> • slow execution 	<ul style="list-style-type: none"> • N/A 	http://dieseltest.com	<ul style="list-style-type: none"> • No Training courses • Online guides

Open Source Performance Test Tools

Name	Summary	Limitations	Cost	Source	Training
FunkLoad	<ul style="list-style-type: none">• performance testing of web applications• record and playback capable	<ul style="list-style-type: none">• monitoring is only for Linux	<ul style="list-style-type: none">• N/A• Must download GNU(free)	http://funkload.nuxeo.org	<ul style="list-style-type: none">• No Training courses• Online examples
Hammerhead 2	<ul style="list-style-type: none">• can create IP aliases and 256+ users at any given time	<ul style="list-style-type: none">• used for Linux and Solaris	<ul style="list-style-type: none">• N/A• Must download GNU(free)	http://hammerhead.sourceforge.net	<ul style="list-style-type: none">• No Training courses• Online docs

Open Source Performance Test Tools

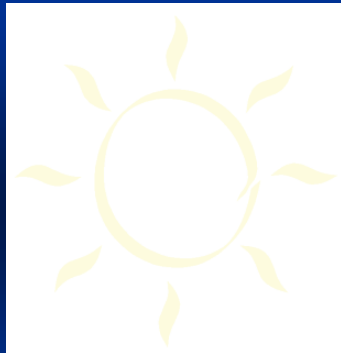
Name	Summary	Limitations	Cost	Source	Training
Seagull	<ul style="list-style-type: none"> • multi-protocol traffic generator • supports Diameter, XCAP over HTTP, TCAP (GSM Camel, MAP, WIN) • similar scenarios and action blocks as Load Runner • scenarios based on XML • allows addition of new protocols 	<ul style="list-style-type: none"> • reporting is not fancy, pure text output 	<ul style="list-style-type: none"> • Free (sponsored by HP Open Call software) 	http://gull.sourceforge.net	<ul style="list-style-type: none"> • No Training courses • Online manuals
TestMaker	<ul style="list-style-type: none"> • friendly GUI development environment • handles HTTP, HTTPS, SOAP, XML-RPC, SMTP, POP3, IMAP), protocols 	<ul style="list-style-type: none"> • uses Jython as its scripting language 	<ul style="list-style-type: none"> • N/A 	http://www.pushtotest.com	<ul style="list-style-type: none"> • No Training courses • Downloadable online manual

Open Source Management Test Tools

Name	Summary	Limitations	Cost	Source	Training
Bugzilla	<ul style="list-style-type: none">• Defect tracking system• reporting system – graphs and reports• enterprise support	<ul style="list-style-type: none">• only tracks defects	<ul style="list-style-type: none">• N/A	http://www.bugzilla.org	<ul style="list-style-type: none">• No Training courses• Online documentation and FAQs
Fitness	<ul style="list-style-type: none">• management of defects and documentation	<ul style="list-style-type: none">• not very fancy, very simple, lightweight	<ul style="list-style-type: none">• N/A	http://fitness.org/	<ul style="list-style-type: none">• No Training courses• Online User Guide

Open Source Management Test Tools

Name	Summary	Limitations	Cost	Source	Training
QATraq	<ul style="list-style-type: none">• incorporates test plan, test cases, results• many different types of reports can be generated	<ul style="list-style-type: none">• does not have defect tracking	<ul style="list-style-type: none">• N/A	http://www.testmanagement.com/	<ul style="list-style-type: none">• No Training courses• Online User Guide
TestMaster	<ul style="list-style-type: none">• covers entire test process, similar to Test Director• fully web based	<ul style="list-style-type: none">• uses apache server and postgresql as the database	<ul style="list-style-type: none">• N/A	http://testmaster.sourceforge.net/	<ul style="list-style-type: none">• No Training courses• Online documentation



Sam Mathi

First Contact Software Consultants

sammathi@firstcontactconsulting.com

www.firstcontactconsulting.com