

# Acceptance Testing with Marathon

**Santosh S**  
**Subex Systems Ltd.**

# Workshop Goals

**Learn how to automate Acceptance Tests using  
Marathon**

# Agenda



- Introduction
- Marathon User Interface
- Marathon Terminology
- Workshop – Swing Application
- Workshop – Web Application
- Resources

# Introduction

# Acceptance Testing @ Subex



- Products – Client/Server Architecture
- Custom Protocol
- Earlier Mechanisms
  - Spread Sheet
  - Dumper – Packet Sniffer
- Problems
  - Maintainability
  - Ease of Use
  - Duplication

# Acceptance Testing @ Subex



- Marathon
- Over 3000+ Test Cases across 3 Products
- Advantages
  - Zero Delay Delivery
  - Major Architecture Change
  - No Duplication
  - Tests all the functionality through UI
  - Test and Data together

- GUI Testing Tool for Java Swing
- Extended for Web (Windows only)
- Record and Playback
- Jython Scripts (Java Python)
  - Readable by testers, developers, analysts
- Current Version – 0.90a
- License – GNU LGPL
- Home – <http://marathonman.sourceforge.net>

- Usage
  - Smoke Testing
  - Regression Testing
  - Formal Acceptance Testing
- Advantages
  - Ease of Use
  - Non Intrusive(No changes in user code)
  - Position agnostic
  - Simple readable scripts
  - Extendibility – Resolvers
  - Semantic Actions (no keypress/mouse events)



- Prerequisites
  - Java version 1.4 and above
  - GNU tar and gzip or Winzip
- Copy marathon-0.90a.tar.gz and support-<version>.tar.gz
- Untar marathon-0.90a.tar.gz. Creates marathon-0.90a directory
- Untar support-<version>.tar.gz into the marathon-0.90a directory
- Set the marathon-0.90a in the system path
- Invoke using marathon.bat or marathon(Linux)



- Application Under Test
  - Java Swing application to be tested
- Fixtures
  - Similar to xUnit Fixtures
  - Mechanism to execute tests in the same environment
  - Provides setup() and teardown()
  - setup() – Test – teardown()
- Capture Script
  - Reusable piece of recorded code
  - Analogous to Functions

- Marathon Project File
  - Property file defining the structure for a Marathon Project
  - Includes classpath, main class, resolvers for the AUT
  - Provides the testcase, capture script and fixture directory
- Test Cases
  - Recorded script that tests the AUT
  - Jython script with 'test' as the function name

# Workshop – Swing Application



- Application – Sample App
- Creating a Marathon Project File
- Introducing Jython
- Test Cases
  - Simple Test Case
  - Capture Script Based Test Cases
  - Assertions
  - Resolvers
- Batch Mode Run

# Workshop – Web Application



- Application – Sample Web App
- Marathon Web vocabulary
- Test Cases
  - Simple Test Case
  - Capture Script Based Test Cases
  - Assertions
- Batch Mode Run

# Resources



- Marathon – <http://marathonman.sf.net>
- Junit – <http://www.junit.org>
- Jython – <http://www.jython.org>

**Thank you**

**Questions**