Development-Driven Testing
Ensuring Testing Meets the Needs of Software Developers

tim.farley@glylity.com
Who’s your customer?
Focus on needs of software developers

- Understand Projects Constraints & Critical Developer Needs
- Assess SQA Capabilities
  - Interfaces
  - Information
  - Interactions
- Adapt to Meet Developer Needs

Read the paper for the full details.
Example Project: Print Engine
Project Overview

- Engine for print/copy/scan/fax system
- “Clean Sheet” project
- 5 year initial engagement for SQA
- 200 mechanical/electrical engineers
- 20 software engineers
- 5 SQA engineers
- Iterative concurrent HW/SW development
Project Constraints

High Daily Development Cost

Expensive Prototypes

Difficult Integration
Critical Development Needs

Component Qualification

Immediate Feedback

Multiple Builds
Assess SQA Capabilities

- Component Qualification
- Immediate Feedback
- Multiple Builds
Immediate Feedback Assessment

1. Test Development
   - Can tests be developed and maintained? Quickly enough?
   - When are the tests needed?
   - Who is involved in creating the tests?

2. Test Execution
   - Can tests be run frequently enough? How often?
   - When do the tests need to be run?
   - Who needs to run the tests?

3. Results Reporting
   - Are results clear? Can they be acted upon immediately?
   - Who needs the results?
   - When are results needed?
1. Test Development Assessment

• Current SQA Practice
  • SQA creates tests against traditional specs written by developers and system engineers
  • Tests are traceable to requirements
  • Changes communicated to SQA through defect/change management system

• Critical Development Needs & Project Constraints
  • Software development driven by iterative experiments
  • Developers won’t be writing specs
  • SQA still needs to demonstrate test coverage against requirements

• Assessment
  • SQA needs a new way of developing and maintaining tests
2. Test Execution Assessment

- **Current SQA Practice**
  - Testing cycles from 1 week (targeted) to 2 months (full regression)
  - Test on pre-production prototypes
  - Mix of manual and automated tests

- **Critical Development Needs & Project Constraints**
  - Full regression test needed every day
  - No budget for prototypes for daily full regression test
  - No budget for manual testers for daily full regression test

- **Assessment**
  - SQA needs a new way of executing tests
3. Results Reporting Assessment

- **Current SQA Practice**
  - Test progress and coverage summary reported by feature
  - No individual test results reported to program team
  - Test summary included in weekly status report to program team

- **Critical Development Needs & Project Constraints**
  - Daily test results
  - Targeted to software developers
  - Actionable

- **Assessment**
  - SQA needs a new way of reporting results
Multiple Builds Assessment

- **Current SQA Practice**
  - Only latest build tested
  - Latest build tested on latest customer-intent prototype
  - Testing targeted at validating HW/SW for external customer use

- **Critical Development Needs & Project Constraints**
  - Testing to sustain development effort and extend life of prototypes
  - Multiple, simultaneous builds need to be tested
  - One build for each prototype revision in use

- **Assessment**
  - SQA needs to test multiple, simultaneous builds
Component Qualification Assessment

- **Current SQA Practice**
  - SQA tests integrated systems only
  - Tests run on customer-intent hardware
  - Human testers interact with device the same way end users would

- **Critical Development Needs & Project Constraints**
  - Test print engine as stand-alone component
  - No budget for prototypes for software developers and SQA
  - No way to interact with device as an end user would

- **Assessment**
  - SQA needs to qualify component without prototype hardware
SQA Assessment Summary

- SQA needs a new way of developing and maintaining tests
- SQA needs a new way of executing tests
- SQA needs a new way of reporting results
- SQA needs to test multiple, simultaneous builds
- SQA needs to qualify component without prototype hardware
Adapt to Meet Developer Needs

Test Automation

Simulation Environment

Scalable Test Infrastructure
Immediate Feedback Adaptations

- **Test Development**
  - Specification by Example written by SQA
  - Parallel code and test development
  - Automated tests ready to run when code is ready to be tested

- **Test Execution**
  - Fully automated daily test and developer-initiated test
  - Tests require no human interaction and run in parallel
  - Simulators for 1/100\textsuperscript{th} the cost of a full prototype

- **Results Reporting**
  - Daily with automatic capture of results, differences, logs, etc...
  - Results acted upon immediately by Development & Management
  - SQA investigation, isolation and recommendation of fixes
Multiple Builds
Adaptations

- Development and SQA support for current prototype revisions plus 2 previous revisions
- Daily full regression test of software builds for 3 prototype revisions
  - Separate simulator racks for each
  - Separate results report for each
- Code for automated tests and test environment branched for each prototype revision
Component Qualification

Adaptations

- Daily fully automated test on simulators
  - Full regression test (Functional, Performance, Stress, ...)
  - System communications protocol tested
  - Fully automated build checking, device upgrading, ...

- Weekly 4-hour test on integrated system with latest prototype hardware

- Complete regression test on integrated system with latest prototype hardware prior to program milestones
  - Automated tests run on simulators and real devices
  - Write the test once and use in both environments
Impact

- NEVER caused system integration failure
- Saved TENS OF MILLIONS of dollars over manual testing on real hardware
- Reduced iteration and program development time
- MOST RELIABLE print engine in company history
Conclusion

SQA can do more than validate products for end users. By focusing on the needs of software developers, SQA can use testing to reduce product development costs, shorten product development cycles and improve product quality.