Software Quality Assurance

Early and Continuous Throughout the Lifecycle

Justifiable evidence and high confidence that your system performs as expected, when expected, is safe, and is secure.
Outline: Where Are We?

- Perspective, Challenges, Goals
  - Why Software Quality Assurance
  - Problem, Solution, Result
  - Software Quality Tools and Life Cycle
  - Independent Software Quality Assessment (ISQA)
  - Wrap-up
  - Glossary
Perspectives Influence Software Quality Goals

**Perspectives**

**DOD**
- Warfighter
- Tax Payer

**Corporations**
- Time To Market
- Reduced Expense
- Increased Profit
- Increased Market Share

**Academia**
- Payee sets goals
- Theory in practice
- Learning
- Research

**Quality Goals**
- Safety
- Security
- Performance
- Portability
- Reliability
- Maintainability
- Availability
- Interoperability
- Robust
- Adaptability
- Usability
- Etc.

CIO Executive Council™ Poll - 2006
Department of Defense

SSTC 2009 – SW Qual Assurance in Lifecycle (20-Apr-2009).ppt
Challenges to Attain Software Quality Goals

Challenges
- Defects
- Politics
- Process
- People
- Money
- Complacency
- Ignorance
- Poor planning
- Data Rights
- Training
- Motivation
- Criteria
- Tools
- Schedule
- SLOC
- Etc.

Quality Goals
- Safety
- Security
- Performance
- Portability
- Reliability
- Maintainability
- Availability
- Interoperability
- Robust
- Adaptability
- Usability
- Etc.

CIO Executive Council™ Poll - 2006
Department of Defense
Outline: Where Are We?

- Perspective, Challenges, Goals
- **Why Software Quality Assurance**
  - Problem, Solution, Result
  - Software Quality Tools and Life Cycle
  - Independent Software Quality Assessment (ISQA)
- Wrap-up
- Glossary
Why Software Quality Assurance?

- Increasing amount & complexity of software-only capabilities
- Growing complexity in COTS, GOTS, and OSS integration
- Example: Service Oriented Architecture (SOA)

Trend: “Hardened” Infrastructure; add more Software!
Outline: Where Are We?

• Perspective, Challenges, Goals
• Why Software Quality Assurance

• Problem, Solution, Result
  • Software Quality Tools and Life Cycle
  • Independent Software Quality Assessment (ISQA)
  • Wrap-up
  • Glossary
Problem: “Software vulnerabilities, malicious code, and software that does not function as promised pose a substantial risk to the Nation’s software-intensive critical infrastructure that provides essential information and services to citizens.” (DHS – Software Assurance in Acquisition: Mitigating Risks to the Enterprise, Oct. 2008)

Solution: Attain justifiable evidence throughout life cycle for your quality goals

Result: Higher confidence that system performs as intended and is not exploitable.
Outline: Where Are We?

- Perspective, Challenges, Goals
- Why Software Quality Assurance
- Problem, Solution, Result
- **Software Quality Tools and Life Cycle**
  - Independent Software Quality Assessment (ISQA)
  - Wrap-up
  - Glossary
Software Quality Assurance Tools
Where to look for “justifiable evidence”!

Contract Verbiage ➔
- Government Data Rights
- Defects – Forecasted and Actual
- Visibility at Government’s Discretion
- Payment Incentives for Defect Reduction
- Improved Configuration Management
- Supplier Credentials - clearance, pedigree, etc.
- Supplier’s evidence of their own assurance claims
- Independent Software Quality Assessment (iSQA)

Ask and You Shall Receive!
Software Quality Assurance Tools
Where to look for “justifiable evidence”!

Supplier’s Processes ➔ CMMI, ISO, Certifications, etc.

Your Own Processes ➔
- CMMI, ISO, etc.
- Defense Acquisition Guidebook (Chapter 4, Sys Eng.)
- DoD IA C&A Process (DIACAP) – (DoDI 8510.01)
- “Software Quality” DCSQ-1 (DoDI 8500.2)
- Secure Coding Requirements (IAW DoDD 8500.1)
- Open Source Software Requirements (AR25-2)
- Army Networthiness (AR25-1)
- COTS Security patch process
- Business Best Practices
- Trained Resources

Look at what is already available and required!
Software Quality Assurance Tools
Where to look for “justifiable evidence”!

Independent Software Quality Assessment (iSQA)

- Code-level forensics
- Static and Runtime assessments
- Automated tools reduce time to “find” defects
- Targeted, actionable recommendations to improve
- Subject Matter Experts provide “operational” perspective
- Motivates software developers to do better
- Repeatable measure of software quality

“In general, third-party testing and evaluation provide a significantly greater basis for customer confidence than many other assurance techniques.”

Outline: Where Are We?

- Perspective, Challenges, Goals
- Why Software Quality Assurance
- Problem, Solution, Result
- Software Quality Tools and Life Cycle
  - ISQA
- Wrap-up
- Glossary
ISQA Capabilities
Perspectives to let the code speak.

- Error Detection
- Open Source Software Assessment
- Networthiness Assessment
- Service Oriented Architecture (SOA)
- Test Coverage

- Software Threat Detection
- Quality Software

- Performance Tuning
- Memory Leak Analysis
- Unit Inline
- 2nd Order Analysis
- Custom...

- Quality Assessment And Audit
Typical ISQA Customer Profiles
Your profile drives your perspective and ISQA needs.

1. “Code Red” Project
2. Rapid Prototyping
   (Creativity & Speed, not quality)
3. Legacy System
   (Reduce Cost, Reuse Fresh Coat of Paint)
4. System Integrator Syndrome
5. Conformance
   (Industry/Customer Standards)
6. Schedule Compression
7. Assessment for Confidence
8. Security Posture and Networthiness
## ISQA Return On Investment
### Composite Example – 4 Actual Projects

#### Industry Accepted SW Metrics

- $10,000/bug to Find & Fix a Defect
- Finding Bugs = 80% of Cost ($8,000 per)

<table>
<thead>
<tr>
<th>Traditional Defect Cost</th>
<th>&quot;find&quot; $$ Avoided</th>
<th>ISQA Cost</th>
<th>Net $$ Avoided</th>
<th>ROI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 335 Defects x $8,000 =</td>
<td>$2,680,000</td>
<td>$545,000</td>
<td>$2,135,000</td>
<td>4.9</td>
</tr>
<tr>
<td>2 219 Defects x $8,000 =</td>
<td>$1,608,000</td>
<td>$219,000</td>
<td>$1,389,000</td>
<td>7.3</td>
</tr>
<tr>
<td>3 1895 Defects x $8,000 =</td>
<td>$15,160,000</td>
<td>$1,214,000</td>
<td>$13,946,000</td>
<td>12.5</td>
</tr>
<tr>
<td>4 70 Defects x $8,000 =</td>
<td>$560,000</td>
<td>$140,000</td>
<td>$420,000</td>
<td>4.0</td>
</tr>
<tr>
<td>2519 Defects x $8,000 =</td>
<td>$20,008,000</td>
<td>$2,118,000</td>
<td>$17,890,000</td>
<td>9.5</td>
</tr>
</tbody>
</table>
ISQA Artifacts
What Justifiable Evidence Should You Expect?

- **Scorecard Summary**
  - Quick assimilation of data (e.g. graphics)
  - Highlight areas for improvement
  - Management / Executive audience

- **Detailed Technical Report**
  - Description of findings
  - Qualitative description of coverage
  - Short, Medium, and Long Term actionable recommendations

- **Raw Data – per defect**
  - Module, LOC, severity, problem, actionable recommendation
  - Formatted for ease of use (e.g. Common separated values, Excel spreadsheet, links from defect to actual line of code, etc.)
### Scorecard Example
Against DISA Application Security and Development STIG

#### CODE INSPECTION RESULTS

<table>
<thead>
<tr>
<th>APP No.</th>
<th>Defects</th>
<th>Instances</th>
<th>CAT I</th>
<th>CAT II</th>
<th>Minor</th>
<th>Bad Style</th>
<th>No Defect</th>
<th>Informational</th>
<th>% Assessed</th>
</tr>
</thead>
<tbody>
<tr>
<td>3050</td>
<td>Dead or Dormant Code</td>
<td>388</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3100</td>
<td>Apparent Undisclosed Stream</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3120</td>
<td>Exception Handling Attributes: Error Handling</td>
<td>233</td>
<td>0</td>
<td>25</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3120</td>
<td>Exception Handling Attributes: The program can potentially dereference a null pointer, thereby causing a segmentation fault</td>
<td>2300</td>
<td>0</td>
<td>115</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3110</td>
<td>Unreleased Resource</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3120</td>
<td>Exception Handling Attributes: Localizing Return Value Of Symbol</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

#### DATA SECURITY
- **3150.2** Cryptography: Standard pseudo-random number generators cannot withstand cryptographic attacks
- **3310** Password Management: Credential Management-Passwords Stored as Clear Text

#### INPUT VALIDATION
- **3570** Command Injection: Executing commands that include un-validated user input can cause an application to act on behalf of an attacker
- **3510** SQL Injection: SQL Injection User Input
- **3540.1** SQL Injection: SQL Injection User Input

#### PORTABILITY AND SECURITY
- **3600** Code Hacking Attributes: Canonical Representation Vulnerabilities
- **3630.3** Code Hacking Attributes: Deprecated Thread Functions

#### SUMMARY OF ISSUES FOUND

<table>
<thead>
<tr>
<th>Key Defects</th>
<th>ALL Defects</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>162</td>
</tr>
<tr>
<td>162</td>
<td>162</td>
</tr>
</tbody>
</table>

- **Category of Finding**
- **STIG Requirement Number**
- **Validate “Real and Actionable”**
- **Actionable Results Feed Into developer’s “Get Well Plan” for the system.**
Outline: Where Are We?

- Perspective, Challenges, Goals
- Why Software Quality Assurance
- Problem, Solution, Result
- Software Quality Tools and Life Cycle
- Independent Software Quality Assessment (ISQA)

- Wrap-up
- Glossary
DoD / Army Software Quality Assurance
Life Cycle Evidence for Confidence to Operate

Materiel Solution Analysis
Technology Development
Engineering & Manufacturing Development
Production & Deployment
Operations & Support

IA Management & Engineering
Contract Language
Networthiness
ISQA

Certificate Of Networthiness
Authorization To Operate
<table>
<thead>
<tr>
<th><strong>Glossary</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AR</strong> – Army Regulation (e.g. AR25-2)</td>
</tr>
<tr>
<td><strong>Assurance</strong> - a statement or indication that inspires confidence, a guarantee</td>
</tr>
<tr>
<td><strong>C&amp;A</strong> – Certification and Accreditation</td>
</tr>
<tr>
<td><strong>CON</strong> – Certificate of Networthiness for the Army</td>
</tr>
<tr>
<td><strong>COTS</strong> – Commercial Off the Shelf software</td>
</tr>
<tr>
<td><strong>DHS</strong> – Department of Homeland Securities</td>
</tr>
<tr>
<td><strong>DIACAP</strong> – Defense Information Assurance Certification and Accreditation Process</td>
</tr>
<tr>
<td><strong>DISA</strong> - Defense Information Systems Agency</td>
</tr>
<tr>
<td><strong>DoDD</strong> – Department of Defense Directive</td>
</tr>
<tr>
<td><strong>DoDI</strong> – Department of Defense Implementation</td>
</tr>
<tr>
<td><strong>GOTS</strong> – Government Off the Shelf software</td>
</tr>
<tr>
<td><strong>Life Cycle</strong> – all phases of a system’s life from concept through disposal</td>
</tr>
<tr>
<td><strong>OSS</strong> – Open Source Software</td>
</tr>
<tr>
<td><strong>Quality</strong> – an essential or distinctive characteristic, property, or attribute</td>
</tr>
<tr>
<td><strong>Software Assurance</strong> - “…the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidently inserted at any time during its life-cycle, and that it functions in the intended manner.” [CNSSI no 40090]</td>
</tr>
<tr>
<td><strong>STIG</strong> – Security Technical Implementation Guide</td>
</tr>
</tbody>
</table>
Presenter’s Credentials and Contact Information
About The Presenter

Credentials

Name: Bruce Weimer

Employer: US Army – CECOM LCMC Software Engineering Center, Software Assurance Division

Experience:

- 4+ years in Civilian Army – System’s Engineer
- 23 years in Industry – Pharma., Financial, Telecom, SW products
- Full Software Life-cycle Development
- Software Quality Assurance
- Information Assurance (IA)
- Process Improvement – CMMI, Lean Six Sigma, ISO
- Content/Document Management
- Workflow and Process Improvement
- Masters in Software Engineering

Contact: bruce.weimer@conus.army.mil, 732.532.5020 / DSN 992