ASQ’s 2002 CSQE Body of Knowledge

Quality Week 2002

Douglas Hoffman
Software Quality Methods, LLC.
24646 Heather Heights Place
Saratoga, California 95070-9710
Phone 408-741-4830
Fax 408-867-4550
doug.hoffman@acm.org

Copyright © 2002, Software Quality Methods, LLC. No part of these graphic overhead slides may be reproduced, or used in any form by any electronic or mechanical duplication, or stored in a computer system, without written permission of the author.

Douglas Hoffman Copyright © 2002, SQM, LLC. 1

“The Certified Software Quality Engineer…

… is a professional who has comprehensive understanding of software quality development and implementation; has a thorough understanding of software inspection, testing, verification, and validation; and can implement software development and maintenance processes and methods.”

ASQ CSQE Certification brochure, Revised 5-02. Copyright © 2002 American Society for Quality.

Douglas Hoffman Copyright © 2002, SQM, LLC. 2

(c) 2002, Software Quality Methods, LLC. 1
ASQ

• The American Society for Quality
• Not-for-profit professional society
• Leading quality improvement organization in US for more than 50 years
• More than 117,000 individual and
• 1,100 corporate sustaining members
• 247 local Sections
• 22 industry and topic-specific Divisions.

CSQE

• Certification Requirements
• The Subject Areas of the CSQE 2002 BOK
• Bloom’s Levels Of Cognition
• Example of Performance Skill Levels
• Example of Mapping of Performance Levels To Job Requirements
• Describing Individual Performance Levels
Certification Requirements

- Education and/or Experience
  - 8 years in quality field
  - up to 5 years credit for degrees
- Proof of Professionalism
- Examination
  - proctored, open book exam
  - 160 questions

Recertification Requirements

- Recertify every 3 years
- 18 points needed
  - Professional Development
  - Employment
  - Instructor/Student
  - Meetings
  - Committees
  - Certifications
  - Proctoring
  - Publishing
CSQE BOK Subject Areas

- General Knowledge, Conduct, and Ethics
- Software Quality Management
- Software Engineering Processes
- Program and Project Management
- Software Metrics, Measurement, and Analytical Methods
- Software Verification and Validation (V&V)
- Software Configuration Management

General Knowledge, Conduct, and Ethics

- Quality philosophy and principles
- Standards, specifications, and models
- Leadership tools and skills
- Ethical conduct and professional development
General Knowledge, Conduct, and Ethics

- Quality philosophy and principles
  - Benefits of software quality (C)
  - Prevention vs. detection (C)
  - Organizational and process benchmarking (An)

- Standards, specifications, and models (Ap)

---

General Knowledge, Conduct, and Ethics

- Leadership tools and skills
  - Organizational leadership (Ap)
  - Team management (Ap)
  - Team tools (Ap)
  - Facilitation skills (Ap)
  - Communication skills (Ap)
General Knowledge, Conduct, and Ethics

- Ethical conduct and professional development
  - ASQ Code of Ethics (B)
  - Software liability and safety issues (Ap)
  - Professional training and development (Ap)

Software Quality Management

- Goals and objectives
- Methodologies
- Audits
Software Quality Management

• Goals and objectives
  – Quality goals and objectives (E)
  – Outsourced services (E)
  – Planning (E)
  – Software quality management systems documentation (C)
  – Customer requirements (E)

Software Quality Management

• Methodologies
  – Reviews, inspections, and testing (E)
  – Change management methods (E)
  – Cost of quality (COQ) (Aa)
  – Quality data tracking (E)
  – Problem reporting and corrective actions (E)
  – Quality improvement process (E)
Software Quality Management

- Audits
  - Program development and administration (C)
  - Audit preparation and execution (C)
  - Audit reporting and follow up (Ap)

Software Engineering Processes

- Environmental conditions
- Requirements management
- Requirements engineering
- Analysis, design, and development methods and tools
- Maintenance management
Software Engineering Processes

• Environmental conditions
  – Life cycles (E)
  – Systems architecture (An)

• Requirements management
  – Requirements prioritization and evaluation (E)
  – Requirements change management (E)
  – Bi-directional requirements traceability (E)

Software Engineering Processes

• Requirements engineering
  – Requirements types (An)
  – Requirements elicitation (C)
  – Requirements analysis and modeling (An)
  – System and software requirements specifications (An)
Software Engineering Processes

- Analysis, design, and development methods and tools
  - Software design methods (Ap)
  - Types of software reuse (Ap)
  - Clean room and other formal methods (C)
  - Software development tools (Ap)

Software Engineering Processes

- Maintenance management
  - Maintenance types (C)
  - Operational maintenance (C)
Program and Project Management

• Planning
  – Project planning elements (Ap)
  – Goal-setting and deployment (Ap)
  – Project planning tools (Aa)
  – Cost and value data (Ap)

Program and Project Management

• Tracking and controlling
  – Phase transition control techniques (Aa)
  – Interpreting and reporting COQ data (E)
  – Tracking elements and methods (E)
  – Project reviews (Aa)
Program and Project Management

• Risk management
  – Risk management planning methods (S)
  – Risk probability (E)
  – Product release decisions (E)
  – Software security, safety, and hazard analysis issues (An)

Software Metrics, Measurement, and Analytical Methods

• Metrics and measurement theory
• Process and product measurement
• Analytical techniques
Software Metrics, Measurement, and Analytical Methods

- Metrics and measurement theory
  - Metrics and measurement theory \(^{(C)}\)
  - Basic measurement theory and techniques \(^{(Ap)}\)
  - Psychology of metrics \(^{(C)}\)

Software Metrics, Measurement, and Analytical Methods

- Process and product measurement
  - Process, product, and resource metrics \(^{(Ap)}\)
  - Commonly used metrics \(^{(Ap)}\)
  - Software quality attributes \(^{(C)}\)
  - Defect detection effectiveness measures \(^{(Ap)}\)
  - Program performance and process effectiveness \(^{(An)}\)
Software Metrics, Measurement, and Analytical Methods

- Analytical techniques
  - Data integrity (S)
  - Quality tools (An)
  - Sampling theory and techniques (An)

Software Verification and Validation (V&V)

- Theory
- Reviews and inspections
- Test planning and design
- Test execution and evaluation
Software Verification and Validation (V&V)

- Theory
  - V&V planning procedures and tasks (S)
  - V&V program (A)
  - Evaluating software products and processes (S)
  - Interfaces (C)

Software Verification and Validation (V&V)

- Reviews and inspections
  - Types (Ap)
  - Items (Ap)
  - Processes (Ap)
  - Data collection, reports, and summaries (Ap)
Software Verification and Validation (V&V)

- Test planning and design
  - Types of tests (S)
  - Test tools (C)
  - Test strategies (S)
  - Test design (Ap)
  - Test coverage of specifications (S)
  - Test environments (S)
  - Supplier components and products (Ap)
  - Test plans (Ap)

Software Verification and Validation (V&V)

- Test execution and evaluation
  - Test implementation (Ap)
  - Test documentation (Ap)
  - Test reviews (S)
  - Code coverage metrics (Ap)
  - Customer deliverables (S)
  - Severity of anomalies (S)
Software Configuration Management

- Configuration infrastructure
- Configuration identification
- Configuration control
- Configuration status accounting
- Configuration audits
- Release and distribution issues

Software Configuration Management

- Configuration infrastructure
  - Configuration management
  - Library/repository processes
  - Defect tracking and library tools
- Configuration identification
  - Configuration items
  - Baselines
  - Configuration identification methods
  - Software builds

Douglas Hoffman  Copyright © 2002, SQM, LLC.
Software Configuration Management

- Configuration control
  - Item and baseline control (Ap)
  - Proposed modifications (C)
  - Review and configuration control boards (CCBs) (Ap)
  - Concurrent development (Ap)
  - Traceability (Ap)
  - Version control (Ap)
  - Configuration item interfaces (Ap)

Software Configuration Management

- Configuration status accounting
  - Status reporting (C)
  - Changes to configuration items and baselines (C)
  - Documentation control (C)
Software Configuration Management

- Configuration audits
  - Functional configuration audit (C)
  - Physical configuration audit (C)

- Release and distribution issues
  - Product release process issues (C)
  - Packaging, production, and distribution (K)

Levels of Cognition

- Knowledge
- Comprehension
- Application
- Analysis
- Synthesis
- Evaluation

Levels of Cognition

<table>
<thead>
<tr>
<th>Level</th>
<th>Verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge</td>
<td>write, list, name, define, label, state</td>
</tr>
<tr>
<td>Comprehension</td>
<td>explain, describe, summarize, illustrate, paraphrase</td>
</tr>
<tr>
<td>Application</td>
<td>use, solve, apply, construct, demonstrate, compute</td>
</tr>
<tr>
<td>Analysis</td>
<td>analyze, compare, contrast, separate</td>
</tr>
<tr>
<td>Synthesis</td>
<td>create, design, invent, develop</td>
</tr>
<tr>
<td>Evaluation</td>
<td>judge, recommend, critique, justify</td>
</tr>
</tbody>
</table>

Example of Performance Skill Levels

<table>
<thead>
<tr>
<th>Area</th>
<th>Knowledge</th>
<th>Comprehension</th>
<th>Application</th>
<th>Analysis</th>
<th>Evaluation</th>
<th>Synthesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audits</td>
<td>answers questions during audits</td>
<td>understands purpose and underlying meaning of audit questions</td>
<td>able to participate as an auditor within an audit team</td>
<td>participates as Lead Auditor given the audit plan</td>
<td>creates the audit plan and is Lead Auditor</td>
<td>Teams Lead Auditors and participates in audit improvements</td>
</tr>
<tr>
<td>Leadership</td>
<td>able to participate in projects</td>
<td>acts independently of project</td>
<td>Leads creative projects</td>
<td>Plans and leads small projects</td>
<td>Plans and leads a large scale organizational project</td>
<td>Plans and manages changes in organizational culture</td>
</tr>
</tbody>
</table>

- Row for each Skill Area
- Column for each Level of Performance
- Cell describes behavior that demonstrates mastery
- Color code by columns
### Example Mapping Of Performance Levels To Job Requirements

<table>
<thead>
<tr>
<th>Area</th>
<th>Associate QE</th>
<th>Quality Engineer</th>
<th>Senior QE</th>
<th>Fellow</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Knowledge, Conduct, and Ethics</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Quality philosophy and principles</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Standards, specifications, and models</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Leadership tools and skills</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Ethical conduct and professional development</td>
<td>N/A</td>
<td></td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

- Column for each job title and grade
- Color indicates Performance Skill Level expected
- Text amplifies or explains

---

### Example Describing Individual Performance Levels

<table>
<thead>
<tr>
<th>Name</th>
<th>General Knowledge, Conduct, and Ethics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Quality Engineer</td>
</tr>
<tr>
<td></td>
<td>Associate QE</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>N/A</td>
</tr>
</tbody>
</table>

- Individual performance shown in second column
- Job Requirements column included for reference
- Remarks possibly color coded for strengths and weaknesses

---

(c) 2002, Software Quality Methods, LLC.
Summary

- CSQE covers a wide field
- CSQE includes level of cognition
- You can
  - decide what applies to you
  - define Performance Skill Levels
  - define Job Requirements
  - describe Individual Performance