

## Maps between the 2008 CSQE BOK and the 2002 BOK

The Certified Software Quality Engineer (CSQE) body of knowledge (BOK) has recently undergone an update so that the most current knowledge areas are tested in the examinations. If you would like more information on the process of updating a BOK, the process is explained on page 4 in the [Certification Handbook](#) (pdf 228 KB).

Part of the updating process is to conduct a job analysis survey to determine whether the current topics in the BOK are still relevant to the job role of software quality engineers and to identify any new topics that have emerged since the BOK was last updated. The results of the CSQE job analysis survey showed that all of the topics that were in the 2002 BOK are still relevant to the job roles of software quality engineers in 2008. However, a few new topics were added to the 2008 BOK because of changing industry needs. These new topics are highlighted on the following pages.

This document contains two BOK map presentations:

- Table 1 follows the new 2008 BOK order (outline and subtext) and the 1<sup>st</sup> column shows where the material was covered in the previous (2002) BOK. In addition, the table contains a 3<sup>rd</sup> column that highlights the new elements being introduced in the 2008 BOK.
- Table 2 follows the old 2002 BOK outline (it does not include the subtext) in the 1<sup>st</sup> and 2<sup>nd</sup> columns and the 3<sup>rd</sup> and 4<sup>th</sup> columns provide the matching 2008 BOK topic heading.

**The 2008 BOK mapped to the 2002 BOK**

The 2008 BOK will first be used in the December 6, 2008 administration. Both BOKs will be available online until after the June 7, 2008 exam is administered, at which time the 2002 BOK will be removed.

<b>TABLE 1: 2008 BOK &amp; NEW TOPICS</b>		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b>I. General Knowledge (16 questions)</b>	
	<b>A. Quality principles</b>	
I.A.1	<b>1. Benefits of software quality</b> Describe the benefits that software quality engineering can have at the organizational level. (Understand)	
I.A.3	<b>2. Organizational and process benchmarking</b> Use benchmarking at the organizational, process, and project levels to identify and implement best practices. (Apply)	
	<b>B. Ethical and Legal Compliance</b>	
I.D.1	<b>1. ASQ Code of Ethics</b> Determine appropriate behavior in situations requiring ethical decisions, including identifying conflicts of interest, recognizing and resolving ethical issues, etc. (Evaluate)	
I.D.2	<b>2. Legal and regulatory issues</b> Define and describe the impact that issues such as copyright, intellectual property rights, product liability, data privacy, the Sarbanes-Oxley Act, etc., can have on software development. (Understand)	Copyright and intellectual property rights, data privacy, and the Sarbanes-Oxley Act.
I.B.	<b>C. Standards and models</b> Define and describe the following standards and assessment models: ISO 9000 standards, IEEE software standards, and the SEI Capability Maturity Model Integrated (CMMI). (Understand)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b><i>D. Leadership skills</i></b>	
I.C.1 II.B.2	<b>1. Organizational leadership</b> Use leadership tools and techniques, such as organizational change management, knowledge-transfer, motivation, mentoring and coaching, recognition, etc. (Apply)	
I.C.4	<b>2. Facilitation skills</b> Use various approaches to manage and resolve conflict. Use negotiation techniques and identify possible outcomes. Use meeting management tools to maximize performance. (Apply)	
I.C.5	<b>3. Communication skills</b> Use various communication elements (e.g., interviewing and listening skills) in oral, written, and presentation formats. Use various techniques for working in multi-cultural environments, and identify and describe the impact that culture and communications can have on quality. (Apply)	Using techniques for working in multi-cultural environments and describing the impact that culture and communications can have on quality.
	<b><i>E. Team Skills</i></b>	
I.C.2	<b>1. Team management</b> Use various team management skills, including assigning roles and responsibilities, identifying the classic stages of team development (forming, storming, norming, performing, adjourning), monitoring and responding to group dynamics, and working with diverse groups and in distributed work environments. (Apply)	Distributed work environments
I.C.3	<b>2. Team tools</b> Use decision-making and creativity tools, such as brainstorming, nominal group technique (NGT), multi-voting, etc. (Apply)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b>II. Software Quality Management (26 questions)</b>	
	<b>A. Quality Management System</b>	
II.A.1 II.A.4	<b>1. Quality goals and objectives</b> Design quality goals and objectives for programs, projects, and products that are consistent with business objectives. Develop and use documents and processes necessary to support software quality management systems. (Create)	
II.A.5	<b>2. Customers and other stakeholders</b> Describe and distinguish between various stakeholder groups, and analyze the effect their requirements can have on software projects and products. (Analyze)	
II.A.3	<b>3. Planning</b> Design program plans that will support software quality goals and objectives. (Evaluate)	
II.A.2	<b>4. Outsourcing</b> Determine the impact that acquisitions, multi-supplier partnerships, outsourced services, and other external drivers can have on organizational goals and objectives, and design appropriate criteria for evaluating suppliers and subcontractors. (Analyze)	Designing appropriate criteria for evaluating suppliers and subcontractors
	<b>B. Methodologies</b>	
II.B.3 IV.B.2	<b>1. Cost of quality (COQ)</b> Analyze COQ categories (prevention, appraisal, internal failure, external failure) and their impact on products and processes. (Analyze)	
II.B.6	<b>2. Process improvement models</b> Define and describe elements of lean tools and the six sigma methodology, and use the plan-do-check-act (PDCA) model for process improvement. (Apply)	Basic elements of lean tools and the six sigma methodology
II.B.5	<b>3. Corrective action procedures</b> Evaluate corrective action procedures related to software defects, process nonconformances, and other quality system deficiencies. (Evaluate)	
I.A.2 I.D.3	<b>4. Defect Prevention</b> Design and use defect prevention processes such as technical reviews, software tools and technology, special training, etc. (Evaluate)	Special training will be framed to focus on training for defect prevention rather than general training for software quality engineers.

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b>C. Audits</b>	
II.C.2	<b>1. Audit types</b> Define and distinguish between various audit types, including process, compliance, supplier, system, etc. (Understand)	
II.C.1	<b>2. Audit roles and responsibilities</b> Identify roles and responsibilities for audit participants: client, lead auditor, audit team members and auditee. (Understand)	
II.C.2 II.C.3	<b>3. Audit Process</b> Define and describe the steps in conducting an audit, developing and delivering an audit report, and determining appropriate follow-up activities. (Apply)	
	<b>III. Systems and Software Engineering Processes (27 questions)</b>	
III.A.1 VI.D.1 I.C.3	<b>A. Life cycles and process models</b> Evaluate various software development lifecycles (iterative, waterfall, etc.) and process models (V-model, Feature Driven Development, Test Driven Development, etc.) and identify their benefits and when they should be used. (Evaluate)	Feature Driven Development and Test Driven Development
III.A.2	<b>B. Systems architecture</b> Identify and describe various architectures, including embedded systems, client-server, n-tier, web, wireless, messaging, collaboration platforms, etc., and analyze their impact on quality. (Analyze)	
	<b>C. Requirements engineering</b>	
III.C.1 III.C.4	<b>1. Requirement types</b> Define and describe various types of requirements including feature, function, system, quality, security, safety, regulatory, etc. (Understand)	
III.C.2 I.C.3	<b>2. Requirements elicitation</b> Describe and use various elicitation methods, including customer needs analysis, use cases, human factors studies, usability prototypes, joint application development (JAD), storyboards, etc. (Apply)	
III.C.3	<b>3. Requirements analysis</b> Identify and use tools such as data flow diagrams (DFDs), entity relationship diagrams (ERDs), etc., to analyze requirements. (Apply)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b><i>D. Requirements management</i></b>	
III.C.2	<b>1. Participants</b> Identify various participants who have a role in requirements planning, including customers, developers, testers, the quality function, management, etc. (Understand)	In addition to testing the role of customers in requirements elicitation, the roles of developers, testers, quality, and management will also be tested.
III.B.1	<b>2. Requirements evaluation</b> Assess the completeness, consistency, correctness and testability of requirements, and determine their priority. (Evaluate)	
III.B.2	<b>3. Requirements change management</b> Assess the impact that changes to requirements will have on software development processes for all types of lifecycle models. (Evaluate)	
III.B.3 VII.C.5	<b>4. Bidirectional traceability</b> Use various tools and techniques to ensure bidirectional traceability from requirements elicitation and analysis through design and testing. (Apply)	
	<b><i>E. Software analysis, design, and development</i></b>	
III.D.1	<b>1. Design methods</b> Identify the steps used in software design and their functions, and define and distinguish between software design methods such as object-oriented analysis and design (OOAD), structured analysis and design (SAD), and patterns. (Understand)	
V.B.3	<b>2. Quality attributes on design</b> Analyze the impact that quality-related elements (safety, security, reliability, usability, reusability, maintainability, etc.) can have on software design. (Analyze)	
III.D.2	<b>3. Software reuse</b> Define and distinguish between software reuse, reengineering, and reverse engineering, and describe the impact these methods can have on software quality. (Understand)	
III.D.4	<b>4. Software development tools</b> Select the appropriate development tools to use for modeling, code analysis, etc., and analyze the impact they can have on requirements management and documentation. (Analyze)	
III.D.3	<b>5. Software development methods</b> Define and describe principles such as pair programming, extreme programming, cleanroom, formal methods, etc., and their impact on software quality. (Understand)	Pair programming, extreme programming, and other new development methods will be tested.

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b><i>F. Maintenance management</i></b>	
III.E.1	<b>1. Maintenance types</b> Describe the characteristics of corrective, adaptive, and preventive maintenance types. (Understand)	
III.E.2	<b>2. Maintenance strategy</b> Describe various factors affecting the strategy for software maintenance, including service-level agreements (SLAs), short- and long-term costs, maintenance releases, product discontinuance, etc., and their impact on software quality. (Understand)	
	<b>IV. Project Management (24 questions)</b>	
	<b><i>A. Planning, scheduling, and deployment</i></b>	
IV.A.1 V.B.1	<b>1. Project planning</b> Use forecasts, resources, schedules, task and cost estimates, etc., to develop project plans. (Apply)	
IV.A.3	<b>2. Project scheduling</b> Use PERT charts, critical path method (CPM), work breakdown structure (WBS), Scrum, burn-down charts, and other tools to schedule and monitor projects. (Apply)	Scrum and burn-down charts
IV.A.2	<b>3. Project deployment</b> Use various tools, including milestones, objectives achieved, task duration, etc., to set goals and deploy the project. (Apply)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b><i>B. Tracking and controlling</i></b>	
IV.B.1	<b>1. Phase transition control</b> Use phase transition control tools and techniques such as entry/exit criteria, quality gates, Gantt charts, integrated master schedules, etc. (Apply)	
IV.A.4 IV.B.3	<b>2. Tracking methods</b> Calculate project-related costs, including earned value, deliverables, productivity, etc., and track the results against project baselines. (Apply)	
IV.B.4	<b>3. Project reviews</b> Use various types of project reviews such as phase-end, management, and retrospectives or post-project reviews to assess project performance and status, to review issues and risks, and to discover and capture lessons learned from the project. (Apply)	
VI.A.2	<b>4. Program reviews</b> Define and describe various methods for reviewing and assessing programs in terms of their performance, technical accomplishments, resource utilization, etc. (Understand)	Program reviews were tested in relation to V&V programs in the past and will now be tested in a broader context, not just V&V.
	<b><i>C. Risk management</i></b>	
IV.C.1 IV.C.2 VI.D.6	<b>1. Risk management methods</b> Use risk management techniques (assess, prevent, mitigate, transfer) to evaluate project risks. (Evaluate)	
IV.C.4	<b>2. Software security risks</b> Evaluate risks specific to software security, including deliberate attacks (hacking, sabotage, etc.), inherent defects that allow unauthorized access to data, and other security breaches, and determine appropriate responses to minimize their impact. (Evaluate)	
IV.C.4	<b>3. Safety and hazard analysis</b> Evaluate safety risks and hazards related to software development and implementation and determine appropriate steps to minimize their impact. (Evaluate)	



TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b>V. Software Metrics and Analysis (24 questions)</b>	
	<b>A. Metrics and measurement theory</b>	
V.A.1	<b>1. Terminology</b> Define and describe metrics and measurement terms including reliability, internal and external validity, explicit and derived measures, etc. (Understand)	
V.A.2	<b>2. Basic measurement theory and statistics</b> Define the central limit theorem, and describe and use mean, median, mode, standard deviation, variance, and range. Apply appropriate measurement scales (nominal, ordinal, ratio, interval) in various situations. (Apply)	
V.A.3	<b>3. Psychology of metrics</b> Describe how metrics and measuring affect the people whose work is being measured and how people affect the ways in which metrics are used and data are gathered. (Understand)	
	<b>B. Process and product measurement</b>	
V.B.2	<b>1. Software metrics</b> Use metrics to assess various software attributes such as size, complexity, number of defects, the amount of test coverage needed, requirements volatility, and overall system performance. (Apply)	
V.B.1 V.B.2 V.B.4 V.B.5 VI.B.4	<b>2. Process metrics</b> Measure the effectiveness and efficiency of software using functional verification tests (FVT), cost, yield, customer impact, defect detection, defect containment, total defect containment effectiveness (TDCE), defect removal efficiency (DRE), process capability and efficiency, etc. (Apply)	
	<b>3. Metrics reporting tools</b> Use various metric representation tools, including dashboards, stoplight charts, etc., to report results efficiently. (Apply)	Metrics reporting tools will be tested in this section, not with Communications skills. (I.D.3.)

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b>C. Analytical techniques</b>	
V.C.2	<b>1. Sampling</b> Define and distinguish between sampling methods (e.g., random, stratified, cluster) as used in auditing, testing, product acceptance, etc. (Understand)	
II.B.4 V.C.1 VI.B.4	<b>2. Data collection and integrity</b> Describe the importance of data integrity from planning through collection and analysis, and apply various techniques to ensure its quality, accuracy, completeness, and timeliness. (Apply)	
V.C.3 V.C.4 V.C.5	<b>3. Quality analysis tools</b> Describe and use classic quality tools (flowcharts, Pareto charts, cause and effect diagrams, control charts, histograms, etc.) and problem-solving tools (affinity and tree diagrams, matrix and activity network diagrams, root cause analysis, etc.) in a variety of situations. (Apply)	
	<b>VI. Software Verification and Validation (V&amp;V) (27 questions)</b>	
	<b>A. Theory</b>	
VI.A.1	<b>1. V&amp;V methods</b> Select and use V&V methods, including static analysis, structural analysis, mathematical proof, simulation, etc., and analyze which tasks should be iterated as a result of modifications. (Analyze)	
VI.A.3	<b>2. Software product evaluation</b> Use various evaluation methods on documentation, source code, test results, etc., to determine whether user needs and project objectives have been satisfied. (Analyze)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b><i>B. Test Planning and Design</i></b>	
II.B.1 VI.C.3	<b>1. Test strategies</b> Select and analyze test strategies (test-driven design, good-enough, risk-based, time-box, top-down, bottom-up, black-box, white-box, simulation, automation, etc.) for various situations. (Analyze)	
VI.C.8 VI.D.1	<b>2. Test plans</b> Develop and evaluate test plans and procedures, including system, acceptance, validation, etc., to determine whether project objectives are being met. (Create)	
VI.C.4	<b>3. Test designs</b> Select and evaluate various test designs, including fault insertion, fault-error handling, equivalence class partitioning, boundary value, etc. (Evaluate)	
VI.C.1 VI.D.1	<b>4. Software tests</b> Identify and use various tests, including unit, functional, performance, integration, regression, usability, acceptance, certification, environmental load, stress, worst-case, perfective, exploratory, system, etc. (Apply)	
VI.C.7	<b>5. Tests of supplier components and products</b> Determine appropriate levels of testing for integrating third-party components and products. (Apply)	
VI.A.4 VI.C.5	<b>6. Test coverage specifications</b> Evaluate the adequacy of specifications such as functions, states, data and time domains, interfaces, security, and configurations that include internationalization and platform variances. (Evaluate)	
VI.D.4	<b>7. Code coverage techniques</b> Identify and use techniques such as branch-to-branch, condition, domain, McCabe's cyclomatic complexity, boundary, etc. (Apply)	
VI.C.6	<b>8. Test environments</b> Select and use simulations, test libraries, drivers, stubs, harnesses, etc., and identify parameters to establish a controlled test environment in various situations. (Analyze)	
VI.C.2	<b>9. Test tools</b> Identify and use utilities, diagnostics, and test management tools. (Apply)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
II.B.1 VI.B.1 VI.B.2 VI.B.3 VI.D.3	<b>C. Reviews and inspections</b> Identify and use desk-checks, peer reviews, walk-throughs, Fagan and Gilb inspections, etc. (Apply)	
VI.D.2	<b>D. Test execution documentation</b> Review and evaluate documents such as defect reporting and tracking records, test completion metrics, trouble reports, input/output specifications, etc. (Evaluate)	
I.D.3 VI.D.5 VII.F.2	<b>E. Customer deliverables</b> Assess the completeness of customer deliverables, including packaged and hosted or downloadable products, license keys and user documentation, marketing and training materials, etc. (Evaluate)	Marketing and training materials of customer deliverables
<b>VII. Software Configuration Management (16 questions)</b>		
<b>A. Configuration infrastructure</b>		
VII.A.1	<b>1. Configuration management team</b> Describe the roles and responsibilities of a configuration management group. (Understand) [NOTE: The roles and responsibilities of the configuration control board (CCB) are covered in area VII.C.2.]	
VII.A.3 VII.C.7	<b>2. Configuration management tools</b> Describe these tools as they are used for managing libraries, build systems, defect tracking systems, etc. (Understand)	
VII.A.2	<b>3. Library processes</b> Describe dynamic, static, and controlled processes used in library systems and related procedures, such as check-in/check-out, merge changes, etc. (Understand)	
<b>B. Configuration identification</b>		
VII.B.1 VII.B.2 VII.B.3	<b>1. Configuration items</b> Describe configuration items (documentation, software code, equipment, etc.), identification methods (naming conventions, versioning schemes, etc.), and when baselines are created and used. (Understand)	
VII.B.4	<b>2. Software builds</b> Describe the relationship between software builds and configuration management functions, and describe methods for controlling builds (automation, new versions, etc.). (Understand)	

TABLE 1: 2008 BOK & NEW TOPICS		
2002 BOK area	2008 BOK Topic Heading	New Elements in 2008 BOK
	<b>C. Configuration control and status accounting</b>	
VII.C.1 VII.C.2 VII.C.6 VII.D.2 VII.D.3	<b>1. Item, baseline, and version control</b> Describe processes for documentation control, tracking item changes, version control, etc., that are used to manage various configurations, and describe processes used to manage configuration item dependencies in software builds and versioning. (Understand)	
VII.C.3	<b>2. Configuration control board (CCB)</b> Describe the roles and responsibilities of the CCB and its members and the procedures they use. (Understand) [NOTE: The roles and responsibilities of the configuration management team are covered in area VII.A.1.]	
VII.C.4	<b>3. Concurrent development</b> Describe the use of configuration management control principles in concurrent development processes. (Understand)	
VII.D.1	<b>4. Status accounting</b> Discuss various processes for establishing, maintaining, and reporting the status of configuration items. (Understand)	
VII.E.1 VII.E.2	<b>D. Configuration audits</b> Define and distinguish between functional and physical configuration audits and how they are used in relation to product specifications. (Understand)	
	<b>E. Product release and distribution</b>	
IV.C.3 VII.F.1	<b>1. Product release</b> Review product release processes (planning, scheduling, defining hardware and software dependencies, etc.) and assess their effectiveness. (Evaluate)	
VII.D.3	<b>2. Archival processes</b> Review the source and release archival processes (backup planning and scheduling, data retrieval, archival processes for build environments, retention of historical records, offsite storage, etc.) and assess their effectiveness. (Evaluate)	

## The 2002 BOK mapped to the 2008 BOK

The following table provides the basic outline of the 2002 BOK in the first column. The second column provides the simple numbering system of the old BOK. The third and fourth columns provide the map to where the topic is now classified in the 2008 BOK.

<b>TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK</b>			
2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b>I. General Knowledge, Conduct, and Ethics</b>			
<i>A. Quality philosophy and principles</i>			
1. Benefits of software quality	I.A.1	I.A.1	Benefits of software quality
2. Prevention vs. detection	I.A.2	II.B.4	Defect prevention
3. Organizational and process benchmarking	I.A.3	I.A.2	Organizational and process benchmarking
<i>B. Standards, specifications, and models</i>	I.B.	I.C.	<i>Standards and models</i>
<i>C. Leadership tools and skills</i>			
1. Organizational leadership	I.C.1	I.D.1	Organizational leadership
2. Team management	I.C.2	I.E.1	Team management
3. Team tools	I.C.3	I.E.2 III.A. III.C.2	Team tools Life cycles and process models • <i>only the RAD portion of the old BOK will move to this new topic area</i> Requirements elicitation • <i>only the JAD portion of the old BOK will move to this new topic area</i>
4. Facilitation Skills	I.C.4	I.D.2	Facilitation skills
5. Communication skills	I.C.5	I.D.3 V.B.3	Communication skills Metric reporting tools

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b><i>D. Ethical conduct and professional development</i></b>			
1. ASQ Code of Ethics	I.D.1	I.B.1	ASQ Code of Ethics
2. Software liability and safety issues	I.D.2	I.B.2	Legal and regulatory issues
3. Professional training and development	I.D.3	II.B.4	Defect prevention <ul style="list-style-type: none"> <li>• <i>Training needs for software quality engineers have been tested in the past. The new BOK will test the training needs required in order to prevent defects (i.e., special training). The knowledge base is the same (needs analysis) but the overall purpose of the training will be framed differently (training for defect prevention versus basic training needs for software quality engineers).</i></li> </ul>
		VI.E.	Customer deliverables
<b>II. Software Quality Management</b>			
<b><i>A. Goals and objectives</i></b>			
1. Quality goals and objectives	II.A.1	II.A.1	Quality goals and objectives
2. Outsourced services	II.A.2	II.A.4	Outsourcing
3. Planning	II.A.3	II.A.3	Planning
4. Software quality management (SQM) systems documentation	II.A.4	II.A.1	Quality goals and objectives
5. Customer requirements	II.A.5	II.A.2	Customers and other stakeholders

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b><i>B. Methodologies</i></b>			
1. Review, inspection, and testing	II.B.1	VI.B.1 VI.C.	Test strategies Reviews and inspections
2. Change management methods	II.B.2	I.D.1	Organizational leadership
3. Cost of quality (COQ) NOTE: In the 2002 BOK, interpreting cost of quality was also covered in BOK area IV.B.2. In the 2008 BOK, the topic is only listed once, in area II.B.1	II.B.3	II.B.1	Cost of quality (COQ)
4. Quality data tracking	II.B.4	V.C.2	Data collection and integrity
5. Problem reporting and corrective action procedures	II.B.5	II.B.3	Corrective action procedures
6. Quality improvement processes	II.B.6	II.B.2	Process improvement models
<b><i>C. Audits</i></b>			
1. Program development and administration	II.C.1	II.C.2	Audit roles and responsibilities
2. Audit preparation and execution	II.C.2	II.C.1	Audit types
		II.C.3	Audit process
3. Audit reporting and follow up	II.C.3	II.C.3	Audit process
<b>III. Software Engineering Processes</b>			
<b><i>A. Environmental conditions</i></b>			
1. Life cycles	III.A.1	III.A.	<i>Lifecycles and process models</i>
2. Systems architecture	III.A.2	III.B.	<i>Systems architecture</i>



**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b><i>B. Requirements management</i></b>			
1. Requirements prioritization and evaluation	III.B.1	III.D.2	Requirements evaluation
2. Requirements change management	III.B.2	III.D.3	Requirements change management
3. Bi-directional requirements traceability	III.B.3	III.D.4	Bidirectional traceability
<b><i>C. Requirements engineering</i></b>			
1. Requirement types	III.C.1	III.C.1	Requirement types
2. Requirements elicitation	III.C.2	III.C.2 III.D.1	Requirements elicitation Participants
3. Requirements analysis and modeling	III.C.3	III.C.3	Requirements analysis
4. System and software requirements specifications	III.C.4	III.C.1	Requirement types
<b><i>D. Analysis, design, and development methods and tools</i></b>			
1. Software design methods	III.D.1	III.E.1	Design methods
2. Types of software reuse	III.D.2	III.E.3	Software reuse
3. Cleanroom and other formal methods	III.D.3	III.E.5	Software development methods
4. Software development tools	III.D.4	III.E.4	Software development tools
<b><i>E. Maintenance management</i></b>			
1. Maintenance types	III.E.1	III.F.1	Maintenance types
2. Operational maintenance	III.E.2	III.F.2	Maintenance strategy

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b>IV. Program and Project Management</b>			
<i>A. Planning</i>			
1. Project planning elements	IV.A.1	IV.A.1	Project planning
2. Goal-setting and deployment	IV.A.2	IV.A.3	Project deployment
3. Project planning tools	IV.A.3	IV.A.2	Project scheduling
4. Cost and value data	IV.A.4	IV.B.2	Tracking methods
<i>B. Tracking and controlling</i>			
1. Phase transition control techniques	IV.B.1	IV.B.1	Phase transition control
2. Interpreting and reporting cost of quality (COQ) data	IV.B.2	II.B.1	Cost of quality (COQ)
3. Tracking elements and methods	IV.B.3	IV.B.2	Tracking methods
4. Project reviews	IV.B.4	IV.B.3	Project reviews
<i>C. Risk management</i>			
1. Risk management planning methods	IV.C.1	IV.C.1	Risk management methods
2. Risk probability	IV.C.2	IV.C.1	Risk management methods
3. Product release decisions	IV.C.3	VII.E.1	Product release
4. Software security, safety, and hazard analysis issues	IV.C.4	IV.C.2	Software security risks
		IV.C.3	Safety and hazard analysis

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b>V. Software Metrics, Measurement, and Analytical Methods</b>			
<i>A. Metrics and measurement theory</i>			
1. Definitions	V.A.1	V.A.1	Terminology
2. Basic measurement theory and techniques	V.A.2	V.A.2	Basic measurement theory and statistics
3. Psychology of metrics	V.A.3	V.A.3	Psychology of metrics
<i>B. Process and product measurement</i>			
1. Process, product, and resource metrics	V.B.1	V.B.2 IV.A.1	Process metrics <ul style="list-style-type: none"> <li>• <i>only the process and product metrics will move to this new topic area</i></li> </ul> Project planning <ul style="list-style-type: none"> <li>• <i>only the resource metrics will move to this new topic area</i></li> </ul>
2. Commonly used metrics	V.B.2	V.B.1 V.B.2	Software metrics Process metrics (There are new topics within this area)
3. Software quality attributes	V.B.3	III.E.2	Quality attributes and design
4. Defect detection effectiveness measures	V.B.4	V.B.2	Process metrics
5. Program performance and process effectiveness	V.B.5	V.B.2	Process metrics

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b><i>C. Analytical techniques</i></b>			
1. Data integrity	V.C.1	V.C.2	Data collection and integrity
2. Sampling theory and techniques	V.C.2	V.C.1	Sampling
3. Quality tools: Charts	V.C.3	V.C.3	Quality analysis tools
4. Quality tools: Diagrams	V.C.4		
5. Miscellaneous quality tools	V.C.5		
<b>VI. Software Verification and Validation (V&amp;V)</b>			
<b><i>A. Theory</i></b>			
1. V&V planning procedures and tasks	VI.A.1	VI.A.1	V&V methods
2. V&V program	VI.A.2	IV.B.4	Program reviews
3. Evaluating software products and processes	VI.A.3	VI.A.2	Software product evaluation
4. Interfaces	VI.A.4	VI.B.6	Test coverage specifications
<b><i>B. Reviews and inspections</i></b>			
1. Types	VI.B.1	VI.C.	<b><i>Reviews and inspections</i></b>
2. Items	VI.B.2		
3. Processes	VI.B.3		
4. Data collection, reports, and summaries	VI.B.4	V.B.2 V.C.2	Process metrics Data collection and integrity

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b><i>C. Test planning and design</i></b>			
1. Types of tests	VI.C.1	VI.B.4	Software tests
2. Test tools	VI.C.2	VI.B.9	Test tools
3. Test strategies	VI.C.3	VI.B.1	Test strategies
4. Test design	VI.C.4	VI.B.3	Test designs
5. Test coverage of specifications	VI.C.5	VI.B.6	Test coverage specifications
6. Test environments	VI.C.6	VI.B.8	Test environments
7. Supplier components and products	VI.C.7	VI.B.5	Tests of supplier components and products
8. Test plans	VI.C.8	VI.B.2	Test plans
<b><i>D. Test execution and evaluation</i></b>			
1. Test implementation	VI.D.1	VI.B.2 III.A. VI.B.4	Test plans (implementation aspects) <b><i>Lifecycles and process model</i></b> (V-model) Software tests (acceptance testing)
2. Test documentation	VI.D.2	VI.D.	<b><i>Test execution documentation</i></b>
3. Test Reviews	VI.D.3	VI.C.	<b><i>Reviews and inspections</i></b>
4. Code coverage metrics	VI.D.4	VI.B.7	Code coverage techniques
5. Customer deliverables	VI.D.5	VI.E.	<b><i>Customer deliverables</i></b>
6. Severity of anomalies	VI.D.6	IV.C.1	Risk management methods

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b>VII. Software Configuration Management</b>			
<i>A. Configuration infrastructure</i>			
1. Configuration management	VII.A.1	VII.A.1	Configuration management team
2. Library/repository processes	VII.A.2	VII.A.3	Library processes
3. Defect tracking and library tools	VII.A.3	VII.A.2	Configuration management tools
<i>B. Configuration identification</i>			
1. Configuration items	VII.B.1	VII.B.1	Configuration items
2. Baselines	VII.B.2		
3. Configuration identification methods	VII.B.3		
4. Software builds	VII.B.4	VII.B.2	Software builds
<i>C. Configuration control</i>			
1. Item and baseline control	VII.C.1	VII.C.1	Item, baseline, and version control
2. Proposed modifications	VII.C.2		
3. Review and configuration control boards (CCBs)	VII.C.3	VII.C.2	Configuration control board (CCB)
4. Concurrent development	VII.C.4	VII.C.3	Concurrent development
5. Traceability	VII.C.5	III.D.4	Bi-directional traceability
6. Version control	VII.C.6	VII.C.1	Item, baseline, and version control
7. Configuration item interfaces	VII.C.7	VII.A.2	Configuration management tools

**TABLE 2: 2002 BOK MAPPED TO THE 2008 BOK**

2002 BOK Topic Heading	2002 BOK #	2008 BOK #	2008 BOK Topic Heading
<b><i>D. Configuration status accounting</i></b>			
1. Status reporting	VII.D.1	VII.C.4	Status accounting
2. Changes to configuration items and baselines	VII.D.2	VII.C.1	Item, baseline, and version control
3. Documentation control	VII.D.3	VII.C.1 VII.E.2	Item, baseline, and version control Archival processes
<b><i>E. Configuration audits</i></b>			
1. Functional configuration audit	VII.E.1	VII.D.	<b><i>Configuration audits</i></b>
2. Physical configuration audit	VII.E.2		
<b><i>F. Release and distribution issues</i></b>			
1. Product release process issues	VII.F.1	VII.E.1	Product release
2. Packaging, production, and distribution	VII.F.2	VI.E.	Customer deliverables