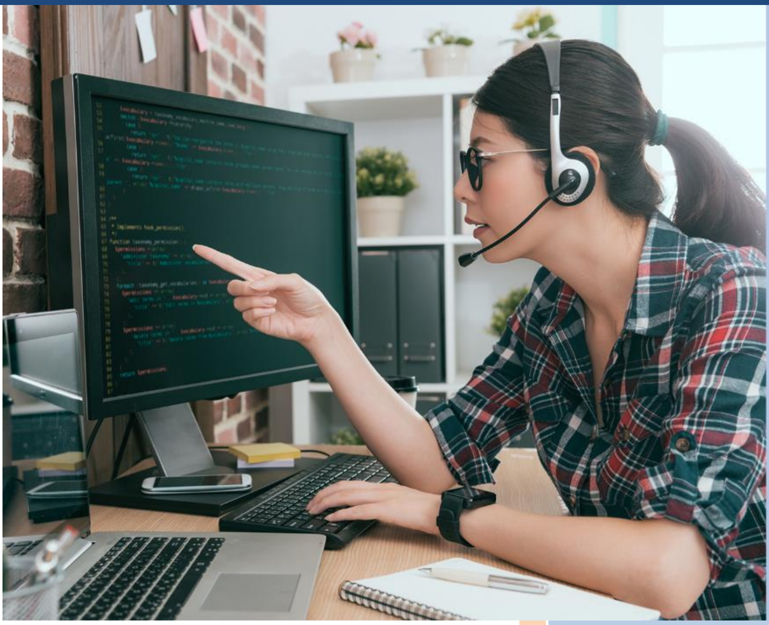
SFIA 7

Software Engineering Competencies



A collaboration with volunteers from the IEEE-CS Professional & Educational Activities Board (PEAB) Engineering Disciplines Committee and the SFIA 7 Software Engineering Working Group.

Software Engineering Competencies 2

SFIA and Software Engineering Competencies 2

SWEBOK Knowledge Areas 7

Software Requirements Knowledge Area 8

Software Design Knowledge Area 9

Software Construction Knowledge Area 10

Software Testing Knowledge Area 11

Software Maintenance Knowledge Area 12

Software Configuration Management Knowledge Area 13

Software Engineering Management Knowledge Area 14

Software Engineering Process Knowledge Area 15

Software Engineering Models Knowledge Area 16

Software Quality Knowledge Area 17

Software Engineering Professional Practice Knowledge Area 18

Software Engineering Economics Knowledge Area 19

Core Software Engineering Competencies 20

Professional skills meet generic attributes 20

Requirements definition and management REQM 21

Systems design DESN 23

Software design SWDN 24

Programming/software development PROG 26

Real-time/embedded systems development RESD 27

Methods and tools METL 29

Configuration management CFMG 30

Testing TEST 32

Systems integration and build SINT 34

Release and deployment RELM 36

Quality assurance QUAS 37

Measurement MEAS 39

Safety engineering SFEN 41

Application support ASUP 42

Software Engineering Management Competencies 43

Systems development management DLMG 43

Project management PRMG 44

Quality management QUMG 46

Conformance review CORE 48

Safety assessment SFAS 49

Organisational capability development OCDV 50

Levels of responsibility 51

Responsibility: Level 1 52

Responsibility: Level 2 53

Responsibility: Level 3 54

Responsibility: Level 4 55

Responsibility: Level 5 56

Responsibility: Level 6 57

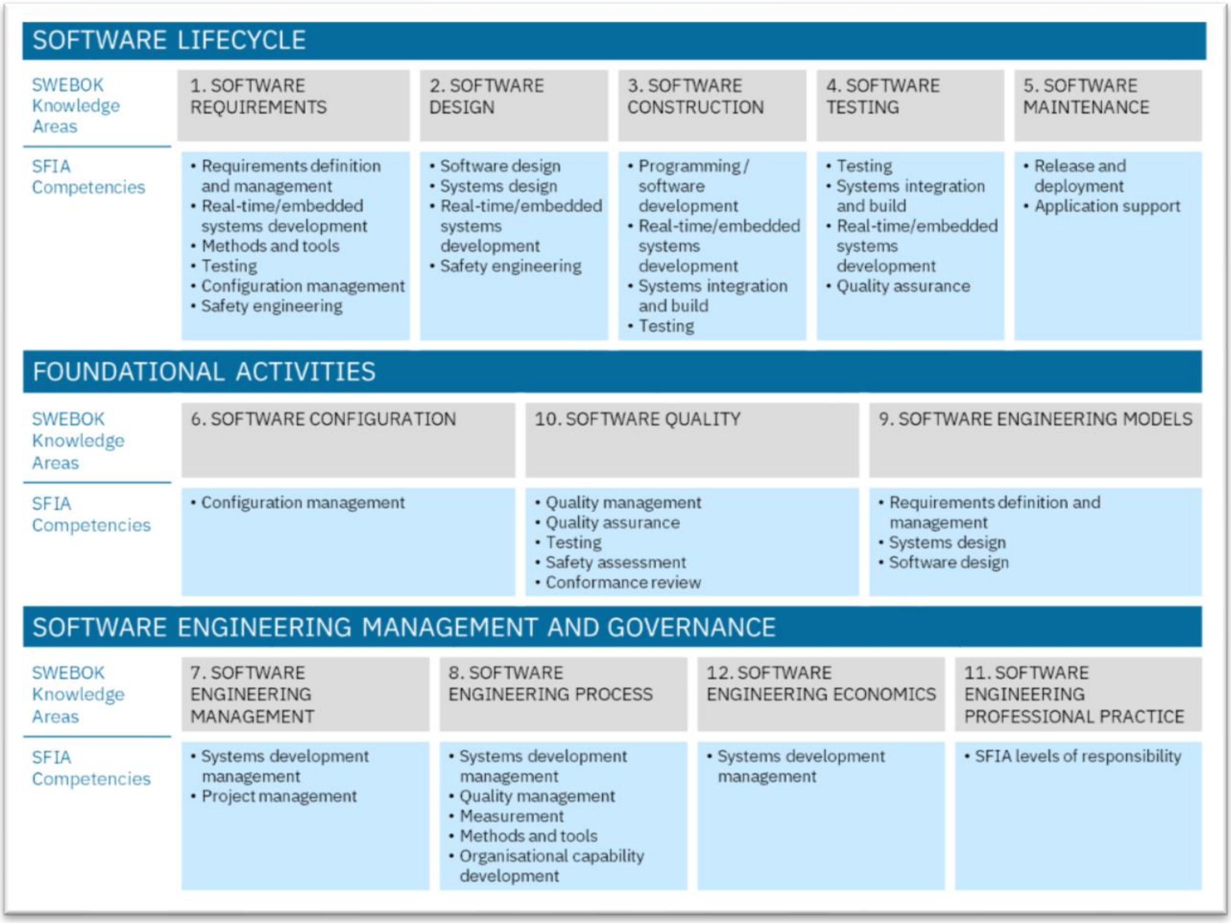
Responsibility: Level 7 58

Appendix A: Cross reference SWE competencies to relevant Knowledge areas in SWEBOK. 59

Appendix B: Changes made to SFIA 6 to fully incorporate Software Engineering in SFIA 7 60

Software Engineering Competencies

A view of the SFIA competencies which are applicable to Software Engineering practitioners and managers



SOFTWARE LIFECYCLE

SWEBOK Knowledge Areas	1. SOFTWARE REQUIREMENTS	2. SOFTWARE DESIGN	3. SOFTWARE CONSTRUCTION	4. SOFTWARE TESTING	5. SOFTWARE MAINTENANCE
SFIA Competencies	• Requirements definition and management • Real-time/embedded systems development • Methods and tools • Testing • Configuration management • Safety engineering	• Software design • Systems design • Real-time/embedded systems development • Safety engineering	• Programming / software development • Real-time/embedded systems development • Systems integration and build • Testing	• Testing • Systems integration and build • Real-time/embedded systems development • Quality assurance	• Release and deployment • Application support

FOUNDATIONAL ACTIVITIES

SWEBOK Knowledge Areas	6. SOFTWARE CONFIGURATION	10. SOFTWARE QUALITY	9. SOFTWARE ENGINEERING MODELS
SFIA Competencies	• Configuration management	• Quality management • Quality assurance • Testing • Safety assessment • Conformance review	• Requirements definition and management • Systems design • Software design

SOFTWARE ENGINEERING MANAGEMENT AND GOVERNANCE

SWEBOK Knowledge Areas	7. SOFTWARE ENGINEERING MANAGEMENT	8. SOFTWARE ENGINEERING PROCESS	12. SOFTWARE ENGINEERING ECONOMICS	11. SOFTWARE ENGINEERING PROFESSIONAL PRACTICE
SFIA Competencies	• Systems development management • Project management	• Systems development management • Quality management • Measurement • Methods and tools • Organisational capability development	• Systems development management	• SFIA levels of responsibility

SFIA and Software Engineering Competencies

Why use SFIA instead of creating a standalone Software Engineering competency framework?

SFIA is the globally accepted common language for the competencies and competency levels related to information and communication technologies, digital transformation and software engineering. It has been adopted by governments, corporates, academic institutions, professional bodies, training provider and individuals in almost 200 countries.

- It is regularly updated through a global, open and collaborative consultation process
- It has a 20+ year provenance and track record of successful use. There is a large amount of content which is immediately available in SFIA
- There is an established ecosystem and trusted infrastructure for creating, updating and publishing the content in multiple languages (these are costly and time-consuming activities for standalone frameworks)
- The SFIA Foundation works collaboratively with the major industry professional bodies in all disciplines to encourage an industry-wide global approach for skills and competencies

- A neutral approach – it is not aligned to specific technologies, vendors or professional bodies

So for the Software Engineering profession; SFIA describes the **specific competency needs of Software Engineering professionals** while also acknowledging the similarities and overlaps with other professional disciplines.

This enables greater transparency across the skills supply chain for all industry participants (employers, educational institutions, professionals, professional bodies, service providers and their client's benefit)

- Facilitating career paths into and out of software engineering roles
- Developing academic curricula and training programmes to develop the competencies needed by industry
- Ensuring individuals' skills are not discarded/discounted because the skill happens to have the "wrong" name or doesn't use a particular industry jargon
- Identifying ways to close skill gaps (for countries, industries, employers and individuals) by recognising the value of the existing skills of seasoned professionals as well as new entrants and mid-career professionals
- Differentiating between software engineering competencies - and the knowledge of software engineering principles, techniques, specific programming languages and tools.
- Enabling non-technical roles involved in talent management to understand the potential for reusable competencies and capabilities

Core Software Engineering Competencies

These are the competencies typically needed by Software Engineering practitioners.

These are listed below and full descriptions of the competency at each level are available on the website - follow the link from each skill name.

- Note that not all of the Competencies listed are required by all Software Engineers
- The set of Competencies required depends on the nature of the employing organisation and/or the specific roles and responsibilities of the Software Engineers they employ.

SWE Competencies	Competency Levels					
Requirements definition and management REQM		2	3	4	5	6
Systems design DESN		2	3	4	5	6
Software design SWDN		2	3	4	5	6
Programming/software development PROG		2	3	4	5	6
Real-time/embedded systems development RESD		2	3	4	5	6
Methods and tools METL			3	4	5	6
Configuration management CFMG		2	3	4	5	6
Testing TEST	1	2	3	4	5	6
Systems integration and build SINT		2	3	4	5	6
Release and deployment RELM			3	4	5	6
Quality assurance QUAS			3	4	5	6
Measurement MEAS			3	4	5	6
Safety engineering SFEN			3	4	5	6
Application support ASUP			3	4	5	

Software Engineering Management Competencies

- These are the competencies typically needed by managers of Software Engineering functions or teams.
- These are listed below and full descriptions of the competency at each level are available on the website - follow the link from each skill name.

SWE Management Competencies	Competency Levels				
Systems development management DELM			5	6	7
Project management PRMG		4	5	6	7
Quality management QUMG	3	4	5	6	7
Conformance review_CORE	3	4	5	6	
Safety assessment SFAS			5	6	7
Organisational capability development OCDV			5	6	7

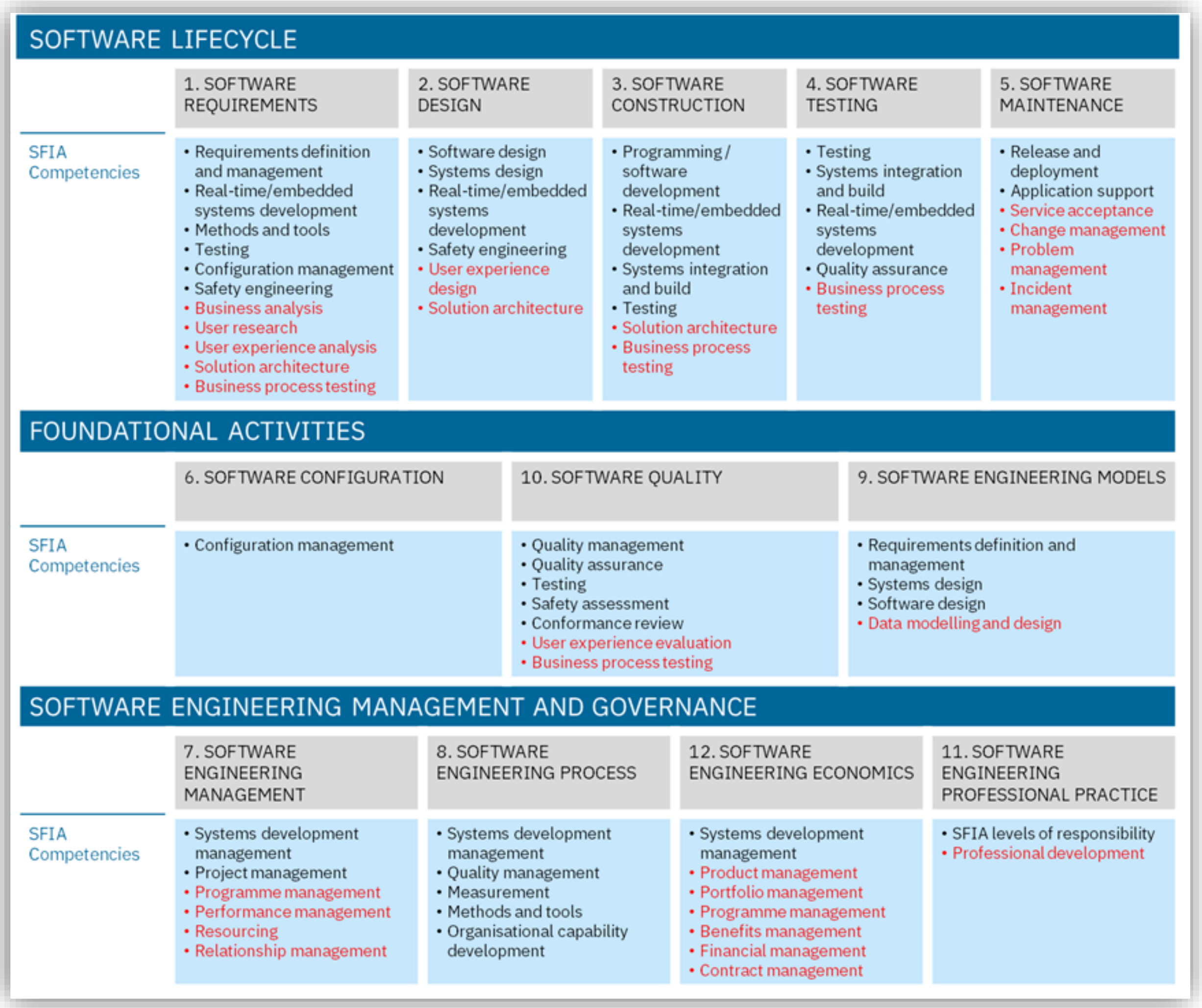
Related Enterprise IT Competencies in SFIA

These are the Competencies which are related to Enterprise functions and roles which typically support or interact with Software Engineering functions and team.
Many Software Engineering organisations and functions will find it useful to refer to and/or utilise some of these competencies.
This is particularly relevant in Enterprise IT organisations which employ Software Engineers.
These are listed below and full descriptions of the competency at each level are available on the website - follow the link from each skill name.

Extended view of the SFIA competencies for Software Engineering.

The diagram below extends the view of Software Engineering to include the Enterprise IT competencies (in red).

This is illustrative only and users are encouraged to review the full set of SFIA competencies to ensure a good fit to their organisational needs.



SOFTWARE LIFECYCLE

	1. SOFTWARE REQUIREMENTS	2. SOFTWARE DESIGN	3. SOFTWARE CONSTRUCTION	4. SOFTWARE TESTING	5. SOFTWARE MAINTENANCE
SFIA Competencies	• Requirements definition and management • Real-time/embedded systems development • Methods and tools • Testing • Configuration management • Safety engineering • Business analysis • User research • User experience analysis • Solution architecture • Business process testing	• Software design • Systems design • Real-time/embedded systems development • Safety engineering • User experience design • Solution architecture	• Programming / software development • Real-time/embedded systems development • Systems integration and build • Testing • Solution architecture • Business process testing	• Testing • Systems integration and build • Real-time/embedded systems development • Quality assurance • Business process testing	• Release and deployment • Application support • Service acceptance • Change management • Problem management • Incident management

FOUNDATIONAL ACTIVITIES

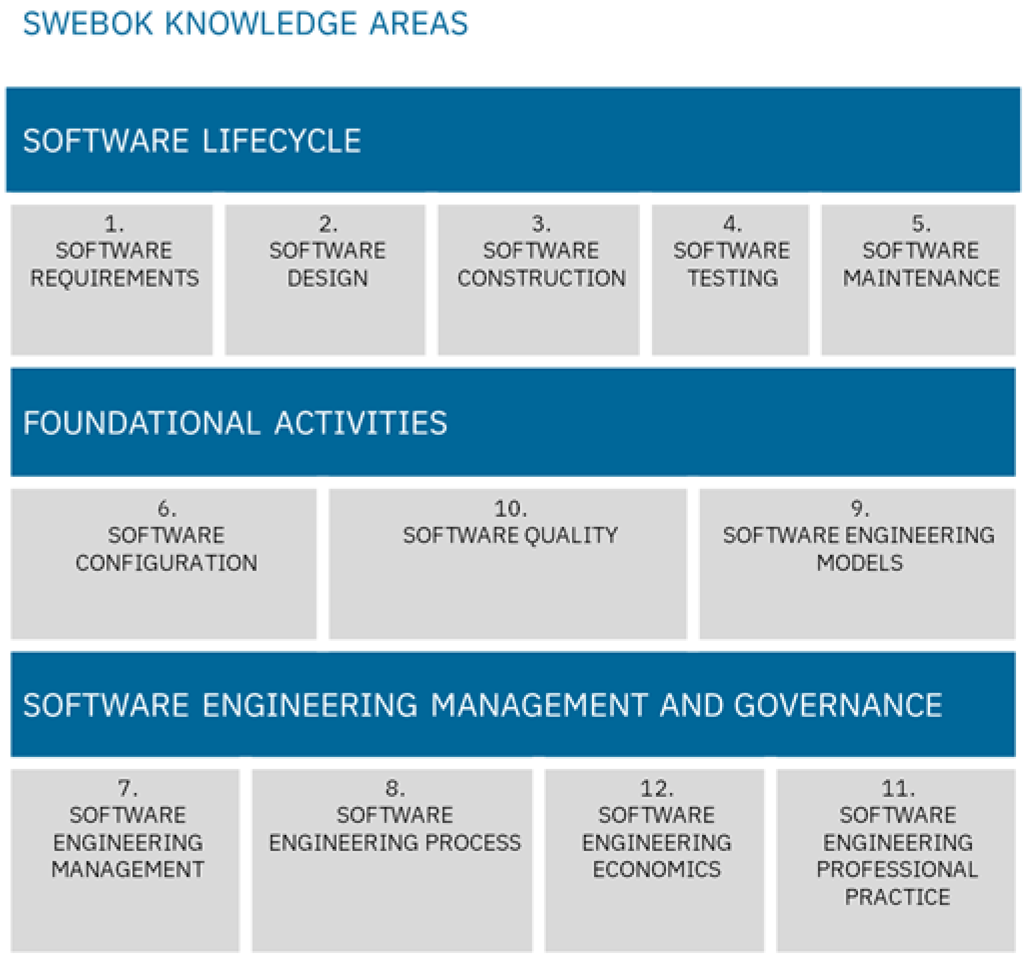
	6. SOFTWARE CONFIGURATION	10. SOFTWARE QUALITY	9. SOFTWARE ENGINEERING MODELS
SFIA Competencies	• Configuration management	• Quality management • Quality assurance • Testing • Safety assessment • Conformance review • User experience evaluation • Business process testing	• Requirements definition and management • Systems design • Software design • Data modelling and design

SOFTWARE ENGINEERING MANAGEMENT AND GOVERNANCE

	7. SOFTWARE ENGINEERING MANAGEMENT	8. SOFTWARE ENGINEERING PROCESS	12. SOFTWARE ENGINEERING ECONOMICS	11. SOFTWARE ENGINEERING PROFESSIONAL PRACTICE
SFIA Competencies	• Systems development management • Project management • Programme management • Performance management • Resourcing • Relationship management	• Systems development management • Quality management • Measurement • Methods and tools • Organisational capability development	• Systems development management • Product management • Portfolio management • Programme management • Benefits management • Financial management • Contract management	• SFIA levels of responsibility • Professional development

Related Enterprise IT Competencies	Competency Levels						
Business analysis BUAN			3	4	5	6	
User research URCH			3	4	5	6	
User experience analysis UNAN			3	4	5		
User experience design HCEV			3	4	5	6	
Solution architecture ARCH				4	5	6	
Data modelling and design DTAN		2	3	4	5		
Business process testing BPTS				4	5	6	
User experience evaluation USEV		2	3	4	5	6	
Service acceptance SEAC				4	5	6	
Change management CHMG		2	3	4	5	6	
Incident management USUP		2	3	4	5		
Problem management PBMG				3	4	5	
Portfolio management POMG					5	6	7
Programme management PGMG						6	7
Product management PROD			3	4	5	6	
Relationship management RLMT				4	5	6	7
Resourcing RESC				4	5	6	
Performance management PEMT				4	5	6	
Professional development PDSV				4	5	6	
Enterprise IT governance GOVN					5	6	7
Supplier management SUPP		2	3	4	5	6	7
Contract management ITCM				4	5	6	
Financial management FMIT				4	5	6	
Benefits management BENM					5	6	

SWEBOK Knowledge Areas



SWEBOK KNOWLEDGE AREAS

SOFTWARE LIFECYCLE

1. SOFTWARE REQUIREMENTS

2. SOFTWARE DESIGN

3. SOFTWARE CONSTRUCTION

4. SOFTWARE TESTING

5. SOFTWARE MAINTENANCE

FOUNDATIONAL ACTIVITIES

6. SOFTWARE CONFIGURATION

10. SOFTWARE QUALITY

9. SOFTWARE ENGINEERING MODELS

SOFTWARE ENGINEERING MANAGEMENT AND GOVERNANCE

7. SOFTWARE ENGINEERING MANAGEMENT

8. SOFTWARE ENGINEERING PROCESS

12. SOFTWARE ENGINEERING ECONOMICS

11. SOFTWARE ENGINEERING PROFESSIONAL PRACTICE

Each of the Knowledge Areas in the SWEBOK were analysed to determine the related Software Engineering (SWE) competencies. These competencies are grouped into

- Core Software Engineering competencies
- Software Engineering Management competencies
- Related Enterprise IT competencies

Software Requirements Knowledge Area

SWEBOK Chapter 1: Software Requirements

Software Requirements is concerned with the elicitation, analysis, specification, and validation of software requirements as well as the management of requirements during the whole life cycle of the software product.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Requirements definition and management
- Real-time/embedded systems development
- Methods and tools
- Testing
- Configuration management
- Safety engineering

SWE Management competencies

- Quality management

Related Enterprise IT Competencies in SFIA

- Business analysis
- User research
- User experience analysis
- User experience design
- Business process testing
- Solution architecture

Software Design Knowledge Area

SWEBOK Chapter 2: Software Design

Software design is the software engineering life cycle activity in which software requirements are analyzed in order to produce a description of the software's internal structure that will serve as the basis for its construction. A software design (the result) describes the software architecture—that is, how software is decomposed and organized into components—and the interfaces between those components. It should also describe the components at a level of detail that enables their construction.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Systems design
- Software design
- Real-time/embedded systems development
- Safety engineering

Related Enterprise IT Competencies in SFIA

- Solution architecture

Software Construction Knowledge Area

SWEBOK Chapter 3: Software Construction

Software construction refers to the detailed creation of working software through a combination of coding, verification, unit testing, integration testing, and debugging. The Software Construction knowledge area (KA) is linked to all the other KAs, but it is most strongly linked to Software Design and Software Testing because the software construction process involves significant software design and testing.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Programming/software development
- Real-time/embedded systems development
- Systems integration and build
- Testing

Related Enterprise IT Competencies in SFIA

- Solution architecture

Software Testing Knowledge Area

SWEBOK Chapter 4: Software Testing

Software testing consists of the ***dynamic*** *verification that a program provides* ***expected*** *behaviors on a* ***finite*** *set of test cases, suitably* ***selected*** *from the usually infinite execution domain. The words in bold correspond to key issues in describing the Software Testing knowledge area (KA):*

- ***Dynamic****: This term means that testing always implies executing the program on selected inputs. To be precise, the input value alone is not always sufficient to specify a test, since a complex, nondeterministic system might react to the same input with different behaviors, depending on the system state. ... Static techniques are different from and complementary to dynamic testing. Static techniques are covered in the Software Quality KA.*
- ***Finite****: Even in simple programs, so many test cases are theoretically possible that exhaustive testing could require months or years to execute. This is why, in practice, a complete set of tests can generally be considered infinite, and testing is conducted on a subset of all possible tests, which is determined by risk and prioritization criteria. Testing always implies a tradeoff between limited resources and schedules on the one hand and inherently unlimited test requirements on the other.*
- ***Selected****: The many proposed test techniques differ essentially in how the test set is selected, and software engineers must be aware that different selection criteria may yield vastly different degrees of effectiveness. How to identify the most suitable selection criterion under given conditions is a complex problem; in practice, risk analysis techniques and software engineering expertise are applied.*
- ***Expected****: It must be possible, although not always easy, to decide whether the observed outcomes of program testing are acceptable or not; otherwise, the testing effort is useless. The observed behavior may be checked against user needs (commonly referred to as testing for validation), against a specification (testing for verification), or, perhaps, against the anticipated behavior from implicit requirements or expectations.*
- *In recent years, the view of software testing has matured into a constructive one. Testing is no longer seen as an activity that starts only after the coding phase is complete with the limited purpose of detecting failures. Software testing is, or should be, pervasive throughout the entire development and maintenance life cycle.*

[Extract from SWEBOK v3.0].

Core SWE competencies

- Testing
- Quality assurance
- Real-time/embedded systems development

SWE Management competencies

- Quality management

Related Enterprise IT Competencies in SFIA

- Business process testing

Software Maintenance Knowledge Area

SWEBOK Chapter 5: Software Maintenance

Software development efforts result in the delivery of a software product that satisfies user requirements. Accordingly, the software product must change or evolve. Once in operation, defects are uncovered, operating environments change, and new user requirements surface. The maintenance phase of the life cycle begins following a warranty period or postimplementation support delivery, but maintenance activities occur much earlier.

Software maintenance is an integral part of a software life cycle. However, it has not received the same degree of attention that the other phases have. Historically, software development has had a much higher profile than software maintenance in most organizations. This is now changing, as organizations strive to squeeze the most out of their software development investment by keeping software operating as long as possible. The open source paradigm has brought further attention to the issue of maintaining software artifacts developed by others.

In the SWEBOK, software maintenance is defined as the totality of activities required to provide cost-effective support to software. Activities are performed during the predelivery stage as well as during the postdelivery stage. Predelivery activities include planning for postdelivery operations, maintainability, and logistics determination for transition activities. Postdelivery activities include software modification, training, and operating or interfacing to a help desk.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Release and deployment
- Application support

SWE Management competencies

- Systems development management

Related Enterprise IT Competencies in SFIA

- Change management
- Service acceptance
- Incident management
- Problem management

Software Configuration Management Knowledge Area

SWEBOK Chapter 6: Software Configuration Management

The configuration of a system is the functional and physical characteristics of hardware or software as set forth in technical documentation or achieved in a product ; it can also be thought of as a collection of specific versions of hardware, firmware, or software items combined according to specific build procedures to serve a particular purpose. Configuration management is the discipline of identifying the configuration of a system at distinct points in time for the purpose of systematically controlling changes to the configuration and maintaining the integrity and traceability of the configuration throughout the system life cycle.

Software configuration management (SCM) is a supporting-software life cycle process that benefits project management, development and maintenance activities, quality assurance activities, as well as the customers and users of the end product. The concepts of configuration management apply to all items to be controlled, although there are some differences in implementation between hardware CM and software CM.

The SCM activities are management and planning of the SCM process, software configuration identification, software configuration control, software configuration status accounting, software configuration auditing, and software release management and delivery.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Configuration management
- Systems integration and build
- Release and deployment
- Application support

Software Engineering Management Knowledge Area

SWEBOK Chapter 7: Software Engineering Management

Software engineering management can be defined as the application of management activities—planning, coordinating, measuring, monitoring, controlling, and reporting1—to ensure that software products and software engineering services are delivered efficiently, effectively, and to the benefit of stakeholders. Software engineering management activities occur at three levels: organizational and infrastructure management, project management, and management of the measurement program.

[Extract from SWEBOK v3.0].

SWE Management competencies

- Systems development management
- Project management
- Quality management
- Methods and tools

Related Enterprise IT Competencies in SFIA

- Programme management
- Resourcing
- Performance management
- Relationship management
- Enterprise IT governance
- Supplier management
- Contract management

Software Engineering Process Knowledge Area

SWEBOK Chapter 8: Software Engineering Process

An engineering process consists of a set of interrelated activities that transform one or more inputs into outputs while consuming resources to accomplish the transformation. Many of the processes of traditional engineering disciplines (e.g., electrical, mechanical, civil, chemical) are concerned with transforming energy and physical entities from one form into another, as in a hydroelectric dam that transforms potential energy into electrical energy or a petroleum refinery that uses chemical processes to transform crude oil into gasoline.

In this knowledge area (KA), software engineering processes are concerned with work activities accomplished by software engineers to develop, maintain, and operate software, such as requirements, design, construction, testing, configuration management, and other software engineering processes. Software processes are specified for a number of reasons: to facilitate human understanding, communication, and coordination; to aid management of software projects; to measure and improve the quality of software products in an efficient manner; to support process improvement; and to provide a basis for automated support of process execution.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Methods and tools
- Measurement

SWE Management competencies

- Systems development management
- Organisational capability development
- Quality management

Software Engineering Models Knowledge Area

SWEBOK Chapter 9: Software Engineering Models

Software engineering models and methods impose structure on software engineering with the goal of making that activity systematic, repeatable, and ultimately more success-oriented. Using models provides an approach to problem solving, a notation, and procedures for model construction and analysis. Methods provide an approach to the systematic specification, design, construction, test, and verification of the end-item software and associated work products. Software engineering models and methods vary widely in scope—from addressing a single software life cycle phase to covering the complete software life cycle.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Requirements definition and management
- Systems design
- Software design
- Methods and tools

Related Enterprise IT Competencies in SFIA

- Data modelling and design

Software Quality Knowledge Area

SWEBOK Chapter 10: Software Quality

Software quality may refer: to desirable characteristics of software products, to the extent to which a particular software product possess those characteristics, and to processes, tools, and techniques used to achieve those characteristics. Over the years, authors and organizations have defined the term quality differently. To Phil Crosby, it was "conformance to requirements" . Watts Humphrey refers to it as "achieving excellent levels of "fitness for use". Meanwhile, IBM coined the phrase "market-driven quality," where the "customer is the final arbiter".
More recently, software quality is defined as the "capability of software product to satisfy stated and implied needs under specified conditions" and as "the degree to which a software product meets established requirements; however, quality depends upon the degree to which those established requirements accurately represent stakeholder needs, wants, and expectations". Both definitions embrace the premise of conformance to requirements. Neither refers to types of requirements (e.g., functional, reliability, performance, dependability, or any other characteristic). Significantly, however, these definitions emphasize that quality is dependent upon requirements.
Software quality is achieved by conformance to all requirements regardless of what characteristic is specified or how requirements are grouped or named.
For all engineered products, the primary goal is delivering maximum stakeholder value, while balancing the constraints of development cost and schedule; this is sometimes characterized as "fitness for use." Stakeholder value is expressed in requirements. For software products, stakeholders could value price (what they pay for the product), lead time (how fast they get the product), and software quality.

[Extract from SWEBOK v3.0].

Core SWE competencies

- Quality assurance
- Testing

SWE Management competencies

- Quality management
- Safety assessment
- Conformance review

Related Enterprise IT Competencies in SFIA

- Business process testing
- User experience evaluation

Software Engineering Professional Practice Knowledge Area

SWEBOK Chapter 11: Software Engineering Professional Practice

The Software Engineering Professional Practice knowledge area (KA) is concerned with the knowledge, skills, and attitudes that software engineers must possess to practice software engineering in a professional, responsible, and ethical manner. Because of the widespread applications of software products in social and personal life, the quality of software products can have profound impact on our personal well-being and societal harmony. Software engineers must handle unique engineering problems, producing software with known characteristics and reliability. This requirement calls for software engineers who possess a proper set of knowledge, skills, training, and experience in professional practice.

A software engineer displays professionalism notably through adherence to codes of ethics and professional conduct and to standards and practices that are established by the engineer's professional community. The professional community is often represented by one or more professional societies; those societies publish codes of ethics and professional conduct as well as criteria for admittance to the community. Those criteria form the basis for accreditation and licensing activities and may be used as a measure to determine engineering competence or negligence.

[Extract from SWEBOK v3.0].

Related Enterprise IT Competencies in SFIA

- Professional development
- SFIA generic responsibilities

Software Engineering Economics Knowledge Area

SWEBOK Chapter 12: Software Engineering Economics

Software engineering economics is about making decisions related to software engineering in a business context. The success of a software product, service, and solution depends on good business management. Yet, in many companies and organizations, software business relationships to software development and engineering remain vague. This knowledge area (KA) provides an overview on software engineering economics. Economics is the study of value, costs, resources, and their relationship in a given context or situation. In the discipline of software engineering, activities have costs, but the resulting software itself has economic attributes as well. Software engineering economics provides a way to study the attributes of software and software processes in a systematic way that relates them to economic measures. These economic measures can be weighed and analyzed when making decisions that are within the scope of a software organization and those within the integrated scope of an entire producing or acquiring business. Software engineering economics is concerned with aligning software technical decisions with the business goals of the organization. In all types of organizations — be it "for-profit," "not-for- profit," or governmental — this translates into sustainably staying in business. In "for-profit" organizations this additionally relates to achieving a tangible return on the invested capital — both assets and capital employed.

[Extract from SWEBOK v3.0].

SWE Management competencies

- Systems development management

Related Enterprise IT Competencies in SFIA

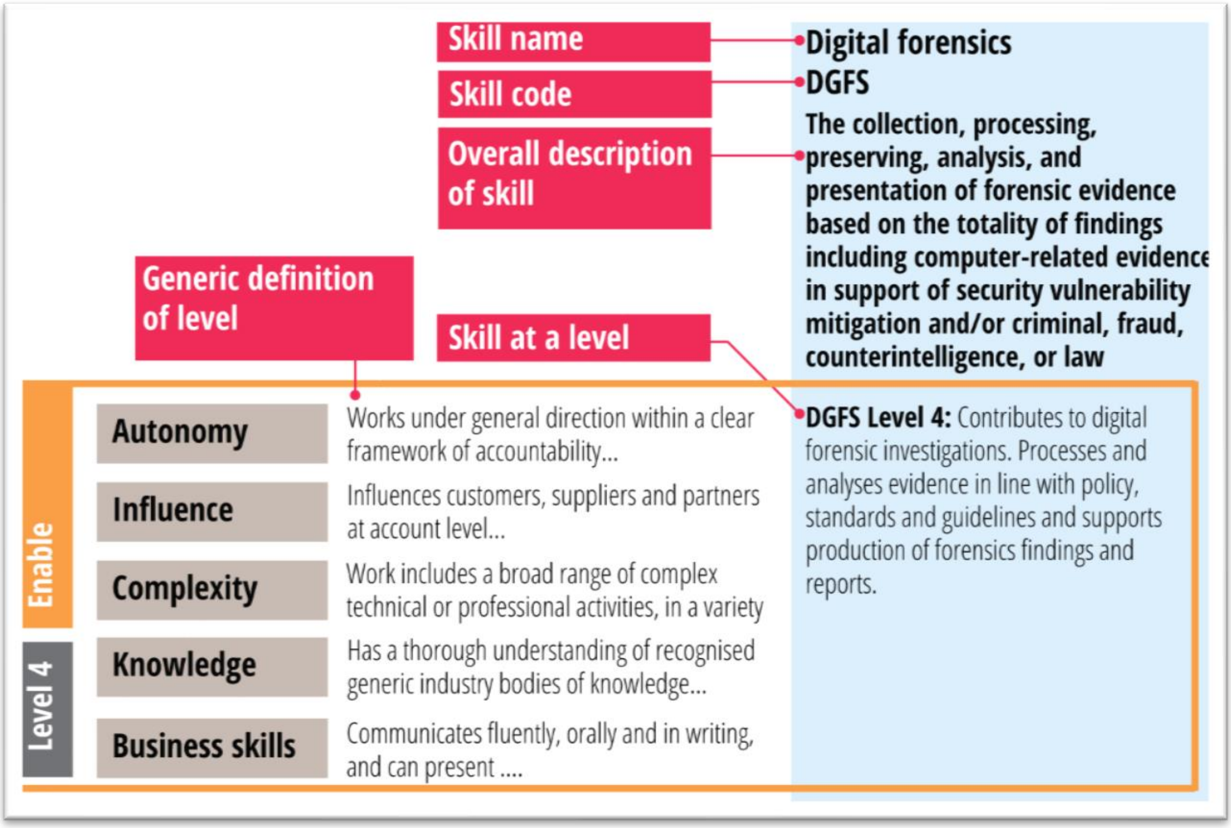
- Financial management
- Portfolio management
- Product management
- Programme management
- Benefits management
- Contract management

Core Software Engineering Competencies

Professional skills meet generic attributes

The levels of responsibility, and specifically their generic attributes, are used together with the professional skills to describe competence.

Each skill description comprises an overall definition of the skill and a description of the skill at each of up to seven levels at which the skill might be exercised. These descriptions provide a detailed definition what it means to practice the skill at each level of competency.



Skill name – Digital forensics

Skill code – DGFS

Overall description of skill – The collection, processing, preserving, analysis, and presentation of forensic evidence based on the totality of findings including computer-related evidence in support of security vulnerability mitigation and/or criminal, fraud, counterintelligence, or law

Generic definition of level

Skill at a level – **DGFS Level 4:** Contributes to digital forensic investigations. Processes and analyses evidence in line with policy, standards and guidelines and supports production of forensics findings and reports.

Level 4

Enable

Autonomy	Works under general direction within a clear framework of accountability...
Influence	Influences customers, suppliers and partners at account level...
Complexity	Work includes a broad range of complex technical or professional activities, in a variety
Knowledge	Has a thorough understanding of recognised generic industry bodies of knowledge...
Business skills	Communicates fluently, orally and in writing, and can present ....

The following pages provides extracts from the full SFIA framework for the SWE Core Competencies and the SWE Management Competencies.

See the SFIA website for additional competency definitions and further advice and guidance.

Requirements definition and management REQM

The elicitation, analysis, specification and validation of requirements and constraints to a level that enables effective development and operations of new or changed software, systems, processes, products and services. The management of requirements throughout the whole of the delivery and operational life cycle of the software, system, processes, products or services. The negotiation of trade-offs that are both acceptable to key stakeholders and within budgetary, technical, regulatory, and other constraints. The adoption and adaptation of requirements management lifecycle models based on the context of the work and selecting appropriately from plan-driven/predictive approaches or more adaptive (iterative and agile) approaches.

Requirements definition and management: Level 6

Develops organisational policies, standards, and guidelines for requirements definition and management. Raises awareness and champions the importance and value of requirements management principles and the selection of appropriate requirements management lifecycle models. Drives adoption of, and adherence to, policies and standards. Develops new methods and organisational capabilities. Plans and leads scoping, requirements definition and priority setting for complex, strategic programmes.

Requirements definition and management: Level 5

Plans and drives scoping, requirements definition and prioritisation activities for large, complex initiatives. Selects, adopts and adapts appropriate requirements definition and management methods, tools and techniques selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Obtains input from, and formal agreement to, requirements from a diverse range of stakeholders. Negotiates with stakeholders to manage competing priorities and conflicts. Establishes requirements baselines. Ensures changes to requirements are investigated and managed. Contributes to the development of organisational methods and standards.

Requirements definition and management: Level 4

Contributes to selection of the requirements approach for projects, selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Defines and manages scoping, requirements definition and prioritisation activities for initiatives of medium size and complexity. Facilitates input from stakeholders, provides constructive challenge and enables effective prioritisation of requirements. Reviews requirements for errors and omissions. Establishes the requirements base-lines, obtains formal agreement to requirements, and ensures traceability to source. Investigates, manages, and applies authorised requests for changes to base-lined requirements, in line with change management policy.

Requirements definition and management: Level 3

Defines and manages scoping, requirements definition and prioritisation activities for small-scale changes and assists with more complex change initiatives. Follows agreed standards, applying appropriate techniques to elicit and document detailed requirements. Provides constructive challenge to stakeholders as required. Prioritises requirements and documents traceability to source. Reviews requirements for errors and omissions. Provides input to the requirements base-line. Investigates, manages and applies authorised requests for changes to base-lined requirements, in line with change management policy.

Requirements definition and management: Level 2

Assists in the definition and management of requirements. Uses standard techniques to elicit, specify, and document requirements for simple subject areas with clearly-defined boundaries. Assists in the creation of a requirements baseline and in investigating and applying authorised requests for changes to base-lined requirements, in line with change management policy.

Systems design DESN

The design of systems to meet specified requirements, compatible with agreed systems architectures, adhering to corporate standards and within constraints of performance and feasibility. The identification of concepts and their translation into a design which forms the basis for systems construction and verification. The design or selection of components. The development of a complete set of detailed models, properties, and/or characteristics described in a form suitable for implementation. The adoption and adaptation of systems design lifecycle models based on the context of the work and selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches.

Systems design: Level 6

Develops organisational policies, standards, guidelines, and methods for systems design. Champions the importance and value of systems design principles and the selection of appropriate systems design lifecycle models; whether predictive (plan-driven) approaches or more adaptive (iterative/agile) approaches. Drives adoption of and adherence to relevant policies, standards, strategies and architectures. Leads systems design activities for strategic, large and complex systems development programmes. Develops effective implementation and procurement strategies, consistent with specified requirements, architectures and constraints of performance and feasibility. Develops systems designs requiring introduction of new technologies or new uses for existing technologies.

Systems design: Level 5

Adopts and adapts appropriate systems design methods, tools and techniques selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches, and ensures they are applied effectively. Designs large or complex systems. Undertakes impact analysis on major design options and trade-off. Makes recommendations and assesses and manages associated risks. Reviews others' systems designs to ensure selection of appropriate technology, efficient use of resources, and integration of multiple systems and technology. Ensures that the system design balances functional and non-functional requirements. Contributes to development of systems design policies and standards and selection of architecture components.

Systems design: Level 4

Designs components using appropriate modelling techniques following agreed architectures, design standards, patterns and methodology. Identifies and evaluates alternative design options and trade-offs. Creates multiple design views to address the concerns of the different stakeholders of the architecture and to handle both functional and non-functional requirements. Models, simulates or prototypes the behaviour of proposed systems components to enable approval by stakeholders. Produces detailed design specification to form the basis for construction of systems. Reviews, verifies and improves own designs against specifications.

Software design SWDN

The specification and design of software to meet defined requirements by following agreed design standards and principles. The definition of software, components, interfaces and related characteristics. The identification of concepts and patterns and the translation into a design which provides a basis for software construction and verification. The evaluation of alternative solutions and trade-offs. The facilitation of design decisions within the constraints of systems designs, design standards, quality, feasibility, extensibility and maintainability. The development and iteration of prototypes/simulations to enable informed decision-making. The adoption and adaptation of software design models, tools and techniques based on the context of the work and selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches.

Software design: Level 6

Leads the selection and development of appropriate software design methods, tools, techniques; whether predictive (plan-driven) approaches or more adaptive (iterative/agile) approaches. Develops organisational policies, standards, and guidelines for software design and software architectures. Ensures adherence to technical strategies and systems architectures (including security).

Software design: Level 5

Selects, adopts and adapts appropriate software design methods, tools and techniques; selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Specifies and designs large or complex software components. Undertakes impact analysis on major design options, makes recommendations and assesses and manages associated risks. Specifies prototypes/simulations to enable informed decision making. Evaluates the quality of others' systems designs to ensure adherence to standards and identifies corrective action, if needed. Ensures that the system design balances functional, quality, security and systems management requirements. Contributes to development of organisational software design and architecture policies and standards.

Software design: Level 4

Designs software components and modules using appropriate modelling techniques following agreed software design standards, patterns and methodology. Creates and communicates multiple design views to identify and balance the concerns of all stakeholders of the software design and to allow for both functional and non-functional requirements. Identifies and evaluates alternative design options and trade-offs. Recommends designs which take into account target environment, performance security requirements and existing systems. Reviews, verifies and improves own designs against specifications. Leads reviews of others' designs. Models, simulates or prototypes the behaviour of proposed software to enable approval by stakeholders, and effective construction of the software. Verifies software design by constructing and applying appropriate methods.

Software design: Level 3

Undertakes complete design of moderately complex software applications or components applying agreed standards, patterns and tools. Assists as part of a team in the design of components of larger software systems. Specifies user and/or system interfaces. Creates multiple design views to address the concerns of the different stakeholders of the design and to handle separately functional and non-functional requirements. Assists in the evaluation of options and trade-offs. Collaborates in reviews of work with others as appropriate.

Software design: Level 2

Creates and documents detailed designs for simple software applications or components applying agreed modelling techniques, standards, patterns and tools. Contributes to the design of components of larger software systems. Reviews own work.

Programming/software development PROG

The planning, designing, creation, amending, verification, testing and documentation of new and amended software components in order to deliver agreed value to stakeholders. The identification, creation and application of agreed software development and security standards and processes. Adopting and adapting software development lifecycle models based on the context of the work and selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches.

Programming/software development: Level 6

Develops organisational policies, standards, and guidelines for software construction and refactoring. Plans and leads software construction activities for strategic, large and complex development projects. Develops new methods and organisational capabilities and drives adoption of, and adherence to policies and standards.

Programming/software development: Level 5

Takes technical responsibility across all stages and iterations of software development. Plans and drives software construction activities. Adopts and adapts appropriate software development methods, tools and techniques selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Measures and monitors applications of project/team standards for software construction including software security. Contributes to the development of organisational policies, standards, and guidelines for software development.

Programming/software development: Level 4

Designs, codes, verifies, tests, documents, amends and refactors complex programs/scripts and integration software services. Contributes to selection of the software development approach for projects, selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Applies agreed standards and tools, to achieve well-engineered outcomes. Participates in reviews of own work and leads reviews of colleagues' work.

Programming/software development: Level 3

Designs, codes, verifies, tests, documents, amends and refactors moderately complex programs/scripts. Applies agreed standards and tools, to achieve a well-engineered result. Collaborates in reviews of work with others as appropriate.

Programming/software development: Level 2

Designs, codes, verifies, tests, documents, amends and refactors simple programs/scripts. Applies agreed standards and tools, to achieve a well-engineered result. Reviews own work.

Real-time/embedded systems development RESD

The architecture, design and development of reliable real time software, operating systems, tools and embedded systems. Embedding computer systems with a dedicated function within a larger mechanical or electronic system, often with real-time, safety, security, and reliability constraints. Typically includes interfacing with hardware, mechanical sensors and actuators for monitoring and control in applications such as industrial, automotive, aerospace and medical machinery, robots and equipment including IoT (Internet of Things) devices.

Real-time/embedded systems development: Level 6

Provides overall direction in the conception and design of real-time/embedded systems. Develops real-time/embedded software architectures in order to exploit new technologies or new uses for existing technologies. Develops effective implementation and procurement strategies, consistent with specified requirements, systems architectures and constraints of performance, cost and feasibility. Sets organisational policies and standards for, and leads on, the development of real-time/embedded systems including how critical non-functional requirements such as performance, safety, security, and reliability are achieved. Drives adoption of and adherence to relevant strategies, policies, standards.

Real-time/embedded systems development: Level 5

Develops real-time/embedded software architectures and designs to meet agreed systems specifications within resource constraints due to power, cost, physical space, response time and reliability. Selects programming languages, models, techniques, specialised tools and hardware to enable the design, development, debugging and validation of real-time/embedded software. Plans and manages real- time/embedded systems developments. Undertakes impact analysis on major design options and trade-offs between hardware and software, makes recommendations and assesses and manages associated risks. Validates and verifies other’s designs to ensure selection of appropriate components and efficient use of resources. Investigates the impact of software requirements with complementary hardware and other related disciplines such as electrics, electronics, mechanics, acoustics, physiology and optics.

Real-time/embedded systems development: Level 4

Designs and develops complex real-time/embedded systems components often incorporating fail-safe characteristics or graceful degradation. Develops and implements software to operate in embedded systems. Develops prototypes or simulations of real time/embedded systems to support decision-making. Designs physical layouts reflecting connections between the components of real-time/embedded systems to test and optimise performance. Contributes to validation and verification activities. Uses specialised tools and hardware (such as logic analysers, in-circuit emulators or digital storage oscilloscopes) for developing, testing, debugging and troubleshooting of embedded software to ensure high levels of integrity and reliability.

Real-time/embedded systems development: Level 3

Designs the interactions between medium-complexity embedded systems components with hardware and the physical world through sensors, actuators and I/O ports. Selects and uses appropriate programming languages (high and low-level) and scripting languages to develop medium complex real-time/embedded components as part of an overall systems design typically requiring high levels of reliability or integrity. Applies a range of approaches to perform extensive testing of real-time/embedded systems, using specialised tools such as logic analysers, in-circuit emulators or digital storage oscilloscopes to demonstrate that high levels of systems integrity and reliability are addressed.

Real-time/embedded systems development: Level 2

Designs the interactions between simple embedded systems components with hardware and the physical world, through sensors, actuators and I/O ports. Uses low level programming languages to develop simple real-time/embedded components as part of an overall systems design. Applies standard approaches to perform extensive testing of real-time/embedded systems, using specialised tools such as logic analysers, in-circuit emulators or digital storage oscilloscopes.

Methods and tools METL

The definition, tailoring, implementation, assessment, measurement, automation and improvement of methods and tools to support planning, development, testing, operation, management and maintenance of systems. Ensuring methods and tools are adopted and used effectively throughout the organisation.

Methods and tools: Level 6

Develops organisational policies, standards, and guidelines for methods and tools. Sets direction and leads in the introduction and use of techniques, methodologies and tools, to match overall business requirements, ensuring consistency across all user groups. Leads the development of organisational capabilities for methods and tools (including automation) to ensure adoption and adherence to policies and standards.

Methods and tools: Level 5

Provides advice, guidance and expertise to promote adoption of methods and tools and adherence to policies and standards. Evaluates and selects appropriate methods and tools in line with agreed policies and standards. Implements methods and tools at programme, project and team level including selection and tailoring in line with agreed standards. Manages reviews of the benefits and value of methods and tools. Identifies and recommends improvements. Contributes to organisational policies, standards, and guidelines for methods and tools.

Methods and tools: Level 4

Provides advice and guidance to support adoption of methods and tools and adherence to policies and standards. Tailors processes in line with agreed standards and evaluation of methods and tools. Reviews and improves usage and application of methods and tools.

Methods and tools: Level 3

Provides support on the use of existing methods and tools. Configures methods and tools within a known context. Creates and updates the documentation of methods and tools.

Configuration management CFMG

The planning, management, control and governance of organisational, project and service assets and artefacts. The identification, classification and specification of configuration items (CIs) and their inter-relationships. Identifying the configuration and version of source code, software, systems, documents and service dependent CIs at distinct points in time. Systematically controlling changes to the configuration and maintaining the integrity and traceability of the configuration throughout the project, system and/or service life cycle. Identifying and documenting the functional and physical characteristics of CIs, controlling changes to those characteristics, recording and reporting change processing and implementation status. Verifying and auditing CIs for data quality and compliance with specified internal and external requirements.

Configuration management: Level 6

Develops configuration management strategies, policies, standards, and guidelines. Champions the importance and value of configuration management and develops new methods and organisational capabilities (including automation) for configuration management. Provides resources to drive adoption of, and adherence to, policies and standards. Measures and monitors adherence to standards and ensures consistent execution of the process across the organisation.

Configuration management: Level 5

Agrees scope of configuration management processes and the configuration items (CIs) and related information to be controlled. Identifies, evaluates and manages the adoption of appropriate tools, techniques and processes (including automation) for configuration management to ensure information is complete, current and accurate. Plans the capture and management of CIs and related information. Contributes to development of configuration management strategies, policies, standards, and guidelines.

Configuration management: Level 4

Proposes and agrees the configuration items (CIs) to be uniquely identified with naming conventions. Ensures that operational processes are in place to maintain secure configuration, consistent classification and management of CIs, and for the verification and audit of configuration records. Develops, configures and maintains tools (including automation) to identify, track, log and maintain accurate, complete and current information. Reports on the status of configuration management. Identifies problems and issues and recommend corrective actions.

Configuration management: Level 3

Applies tools, techniques and processes to track, log and correct information related to configuration items. Verifies and approves changes ensuring protection of assets and components from unauthorised change, diversion and inappropriate use. Ensures that users comply with identification standards for object types, environments, processes, lifecycles, documentation, versions, formats, baselines, releases and templates. Performs audits to check the accuracy of information and undertakes any necessary corrective action under direction.

Configuration management: Level 2

Applies tools, techniques and processes to administer, track, log, report on and correct configuration items, components and changes. Assists with audits to check the accuracy of information and undertakes any necessary corrective action under direction.

Testing TEST

The planning, design, management, execution and reporting of tests, using appropriate testing tools and techniques and conforming to agreed process standards and industry specific regulations. The purpose of testing is to ensure that new and amended systems, configurations, packages, or services, together with any interfaces, perform as specified (including security requirements) , and that the risks associated with deployment are adequately understood and documented. Testing includes the process of engineering, using and maintaining testware (test cases, test scripts, test reports, test plans, etc) to measure and improve the quality of the software being tested.

Testing: Level 6

Determines testing policy, and owns the supporting processes including software security testing. Takes responsibility for the management of all testing activities within a development or integration project or programme. Manages all risks associated with the testing and takes preventative action when any risks become unacceptable. Assesses and advises on the practicality of testing process alternatives, including automated testing. Initiates improvements to test processes and directs their implementation. Assesses suppliers' development and testing capabilities. Determines project testing standards for all phases, influencing all parties to conform to those standards. Manages client relationships with respect to all testing matters.

Testing: Level 5

Coordinates and manages planning of the system and/or acceptance tests, including software security testing, within a development or integration project or programme. Takes responsibility for integrity of testing and acceptance activities and coordinates the execution of these activities. Provides authoritative advice and guidance on any aspect of test planning and execution. Defines and communicates the test strategy for the project. Manages all test processes, including test plans, resources, costs, timescales, test deliverables and traceability. Manages client relationships with respect to testing matters. Identifies process improvements, and contributes to corporate testing standards and definition of best practice.

Testing: Level 4

Accepts responsibility for creation of test cases using own in-depth technical analysis of both functional and non-functional specifications (such as reliability, efficiency, usability, maintainability and portability). Creates traceability records, from test cases back to requirements. Produces test scripts, materials and regression test packs to test new and amended software or services. Specifies requirements for environment, data, resources and tools. Interprets, executes and documents complex test scripts using agreed methods and standards. Records and analyses actions and results, and maintains a defect register. Reviews test results and modifies tests if necessary. Provides reports on progress, anomalies, risks and issues associated with the overall project. Reports on system quality and collects metrics on test cases. Provides specialist advice to support others.

Testing: Level 3

Reviews requirements and specifications, and defines test conditions. Designs test cases and test scripts under own direction, mapping back to pre-determined criteria, recording and reporting outcomes. Analyses and reports test activities and results. Identifies and reports issues and risks associated with own work.

Testing: Level 2

Defines test conditions for given requirements. Designs test cases and creates test scripts and supporting data, working to the specifications provided. Interprets, executes and records test cases in accordance with project test plans. Analyses and reports test activities and results. Identifies and reports issues and risks.

Testing: Level 1

Executes given test scripts under supervision. Records results and reports issues. Develops an understanding of the role of testing within system development, as a tool for design improvement as well as a validation process.

Systems integration and build SINT

The planning, implementation and control of activities to integrate/build components, subsystems and interfaces to create operational systems, products or services for delivery to customers, or for internal or interim purposes such as testing. The development of organisational capabilities for systems integration and build including automation and continuous integration.

Systems integration and build: Level 6

Develops organisational policies, standards, and guidelines for systems integration and build. Leads the development of organisational capabilities for systems integration and build including automation and continuous integration. Provides resources to ensure systems integration and build can operate effectively and ensure adoption and adherence to policies and standards.

Systems integration and build: Level 5

Identifies, evaluates and manages the adoption of appropriate tools, techniques and processes (including automation and continuous integration) to create a robust integration framework. Leads integration work in line with the agreed system and service design. Monitors and reports on the results of each integration and build. Designs and builds integration components and interfaces. Contributes to the overall design of the service and the definition of criteria for product and component selection. Contributes to development of systems integration policies, standards and tools.

Systems integration and build: Level 4

Provides technical expertise to enable the configuration of software, other system components and equipment for systems testing. Collaborates with technical teams to develop and agree system integration plans and report on progress. Defines complex/new integration builds. Ensures that integration test environments are correctly configured. Designs, performs and reports results of tests of the integration build. Identifies and documents system integration components for recording in the configuration management system. Recommends and implements improvements to processes and tools.

Systems integration and build: Level 3

Defines the software modules needed for an integration build and produces a build definition for each generation of the software. Accepts completed software modules, ensuring that they meet defined criteria. Produces software builds from software source code for loading onto target hardware. Configures the hardware and software environment as required by the system being integrated. Produces integration test specifications, conducts tests and records and reports on outcomes. Diagnoses faults and records and reports on the results of tests. Produces system integration reports.

Systems integration and build: Level 2

Produces software builds from software source code. Conducts tests as defined in an integration test specification, records the details of any failures. Analyses and reports on integration test activities and results. Identifies and reports issues and risks.

Release and deployment RELM

The management of the processes, systems and functions to package, build, test and deploy changes and updates (which are bounded as “releases”) into a live environment, establishing or continuing the specified service, to enable controlled and effective handover to operational management and the user community. The application of automation to improve the efficiency and quality of releases.

Release and deployment: Level 6

Sets the release policy for the organisation in the context of both development and production/operations. Ensures that management processes, tools, techniques and resources (including automation) are in place to ensure that the transition of services, service components and packages are planned and compliant and that test and validation and configuration management are partnered in all release and deployment activities. Provides authorisation for critical release activity and point of escalation.

Release and deployment: Level 5

Leads the assessment, analysis, planning and design of release packages, including assessment of risk. Liaises with business and IT partners on release scheduling and communication of progress. Conducts post release reviews. Ensures release processes and procedures are applied and that releases can be rolled back as needed. Identifies, evaluates and manages the adoption of appropriate release and deployment tools, techniques and processes (including automation).

Release and deployment: Level 4

Assesses and analyses release components. Provides input to scheduling. Carries out the builds and tests in coordination with testers and component specialists maintaining and administering the tools and methods – manual or automatic - and ensuring, where possible, information exchange with configuration management. Ensures release processes and procedures are maintained.

Release and deployment: Level 3

Uses the tools and techniques for specific areas of release and deployment activities. Administers the recording of activities, logging of results and documents technical activity undertaken. May carry out early life support activities such as providing support advice to initial users.

Quality assurance QUAS

The process of ensuring, through independent assessment and review, that appropriate working practices, quality control activities, organisational processes and quality standards are in place and adhered to and that best practices are promoted throughout the organisation. Quality assurance provides confidence to internal management and external bodies, such as customers or regulators, that quality requirements will be fulfilled. Quality assurance may relate to any area where quality standards are applied, including products, data, services and business processes.

Quality assurance: Level 6

Leads, develops and is accountable for an organisational approach and commitment to quality assurance. Ensures that quality assurance processes and activities are robust and based on industry best practice. Considers the implications of emerging technology, approaches, trends, regulations and legislation. Plans and resources the organisational quality assurance activities. Monitors and reports on quality assurance activities, levels of compliance and both organisational and project risks. Reviews and analyses results from audit activities and identifies improvement opportunities for the organisation.

Quality assurance: Level 5

Plans, organises and conducts formal independent audits of complex projects, major programmes or functional areas. Evaluates, appraises and identifies non-compliances with organisational standards, and determines whether appropriate quality control has been applied. Prepares and reports audit findings and determines the risks associated with those findings and ensures that corrective actions are carried out. Reviews and analyses audit reports to identify common areas of non-compliance and identifies opportunities to improve the effectiveness and efficiency of the organisational control mechanisms. Performs audits throughout the supply chain. Plans and oversees the assurance activities of others.

Quality assurance: Level 4

Conducts formal audits or reviews to ensure compliance with organisational standards for activities, processes, data, products or services. For projects, development or support activities; plans, organises and conducts audits and determines whether appropriate quality control has been applied. Collates, collects and examines records, analyses the evidence and drafts all or part of formal compliance reports. Determines the risks associated with findings and non-compliance and proposes corrective actions. Provides advice and guidance in the use of organisational standards. Performs quality assurance reviews of suppliers and throughout the supply chain.

Quality assurance: Level 3

Contributes to the collection of evidence and the conduct of formal audits or reviews of activities, processes, data, products or services. Examines records for evidence that appropriate testing and other quality control activities have taken place and determines compliance with organisational directives, standards and procedures. Identifies non-compliances, non-conformances and abnormal occurrences.

Measurement MEAS

The development and operation of a measurement capability to support agreed organisational information needs. The planning, implementation, and control of activities to measure attributes of processes, products, and services in order to assess performance, progress, and provide indications and insights to actual or potential problems, issues, and risks. The identification of requirements, selecting measures and measurement scales, establishing data collection and analysis methods, setting target values and thresholds. Measurement can be applied to organizations, projects, processes, and work products.

Measurement: Level 6

Creates the measurement framework and aligns measurement objectives with business objectives. Develops organisational policies, standards, guidelines for measurement. Leads the development of organisational capabilities for measurement (including automation). Provides resources to ensure adoption and adherence to policies and standards.

Measurement: Level 5

Provides advice and guidance for effective use of measures and measurement. Establishes measurement objectives and the scope of measurement for functions, teams and projects. Plans and implements improvements to measurement capability. Selects measures appropriate to the context and organisational objectives. Contributes to organisational policies, standards, and guidelines for measurement. Reviews data collection and storage mechanisms (including automation) to support measurement.

Measurement: Level 4

Supports projects, functions or teams in the development of project and/or operational methods for measurement. Specifies base and derived measures which support agreed information needs. Identifies and prioritises appropriate measures, scales, and targets. Specifies how to collect and store the data for each required measure. Provides guidance on collection of data including automation. Designs reports and reporting formats.

Measurement: Level 3

Applies standard techniques to support the specification of measures and the collection and maintenance of data for measurement. Generates, produces and distributes reports. Uses measurement tools for routine analysis of data. Identifies and implements improvements to data collection methods.

Safety engineering SFEN

The application of appropriate methods to assure safety during all lifecycle phases of safety-related systems developments, including maintenance and re-use. These include safety hazard and risk analysis, safety requirements specification, safety-related system architectural design, formal method design, safety validation and verification, and safety case preparation.

Safety engineering: Level 6

Takes full responsibility for hazard analysis and risk assessment, safety-related system architectural design, safety assurance planning and compliance and safety case preparation on systems up to the highest safety integrity levels. Takes responsibility for the safety-related aspects of multiple complex or high safety integrity level projects, providing effective leadership to team members.

Safety engineering: Level 5

Identifies and analyses hazards and contributes to the identification and evaluation of risk reduction measures, ensuring these are adequately documented. Specifies safety-related systems architectures up to the highest safety integrity levels. Develops and maintains project safety assurance plans, monitors compliance and ensures that safety assurance evidence is gathered for safety case preparation.

Safety engineering: Level 4

Contributes to the identification, analysis and documentation of hazards, and to the capture, evaluation and specification of safety requirements. Analyses and documents safety validation results. Contributes to the development and maintenance of project safety assurance plans, and gathers safety assurance evidence for safety case preparation.

Safety engineering: Level 3

Assists with the collection of safety assurance evidence, undertaking all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools. Documents the results of hazard and risk analysis activities.

Application support ASUP

The provision of application maintenance and support services, either directly to users of the systems or to service delivery functions. Support typically includes investigation and resolution of issues and may also include performance monitoring. Issues may be resolved by providing advice or training to users, by devising corrections (permanent or temporary) for faults, making general or site-specific modifications, updating documentation, manipulating data, or defining enhancements Support often involves close collaboration with the system's developers and/or with colleagues specialising in different areas, such as Database administration or Network support.

Application support: Level 5

Drafts and maintains procedures and documentation for applications support. Manages application enhancements to improve business performance. Advises on application security, licensing, upgrades, backups, and disaster recovery needs. Ensures that all requests for support are dealt with according to set standards and procedures.

Application support: Level 4

Maintains application support processes, and checks that all requests for support are dealt with according to agreed procedures. Uses application management software and tools to investigate issues, collect performance statistics and create reports.

Application support: Level 3

Identifies and resolves issues with applications, following agreed procedures. Uses application management software and tools to collect agreed performance statistics. Carries out agreed applications maintenance tasks.

Application support: Level 2

Assists in the investigation and resolution of issues relating to applications. Assists with specified maintenance procedures.

Software Engineering Management Competencies

Systems development management DLMG

The planning, estimating and execution of programmes of systems development work to time, budget and quality targets. The identification of the resources needed for systems development and how this will be met with an effective supply capacity. The alignment of systems development activity and deliverables with agreed architectures and standards. The development of roadmaps to communicate future systems development plans. The adoption and adaptation of systems development lifecycle models based on the context of the work and selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches.

Systems development management: Level 7

Leads the definition, implementation and review of the organisation’s systems development management framework. Authorises the structure of systems development functions and platforms and is responsible for alignment with business strategy & objectives and with emerging IT and digital opportunities. Sets strategy for resource management within systems development, authorises the allocation of resources for systems development programmes, and maintains an overview of the contribution of such programmes to organisational success. Manages the quality and appropriateness of the work performed and delivers measurable business benefits.

Systems development management: Level 6

Sets policy and drives adherence to standards for systems development projects whether predictive (plan-driven) approaches or more adaptive (iterative/agile) approaches. Promotes the benefits of addressing all security issues during systems development. Identifies and manages the resources necessary for all stages (planning, estimation, execution) of systems development projects, ensuring that technical, financial and quality targets are met.

Systems development management: Level 5

Defines systems development projects which support the organisation's objectives and plans. Selects, adopts and adapts appropriate systems development methods, tools and techniques selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Ensures that senior management is both aware of and able to provide the required resources. Facilitates availability and optimum utilisation of resources. Monitors and reports on the progress of development projects, ensuring that projects are carried out in accordance with agreed architectures, standards, methods and procedures (including secure software development). Develops road maps to communicate future development activity.

Project management PRMG

The management of projects, typically (but not exclusively) involving the development and implementation of business processes to meet identified business needs, acquiring and utilising the necessary resources and skills, within agreed parameters of cost, timescales, and quality. The adoption and adaptation of project management methodologies based on the context of the project and selecting appropriately from predictive (plan-driven) approaches or adaptive (iterative/agile) approaches.

Project management: Level 7

Sets organisational strategy governing the direction and conduct of project management, including application of appropriate methodologies. Authorises the management of large scale projects. Leads project planning, scheduling, controlling and reporting activities for strategic, high impact, high risk projects. Manages risk and ensures that solutions to problems are implemented in line with change control processes.

Project management: Level 6

Takes full responsibility for the definition, documentation and successful completion of complex projects (typically with significant business, political, or high-profile impact, and high-risk dependencies). Adopts and adapts project management methods and tools, selecting appropriately from plan-driven/predictive approaches or more adaptive (iterative and agile) approaches. Ensures that effective project control, change control, risk management and testing processes are maintained. Monitors and controls resources, revenue and capital costs against the project budget and manages expectations of all project stakeholders.

Project management: Level 5

Takes full responsibility for the definition, approach, facilitation and satisfactory completion of medium-scale projects (typically with direct business impact and firm deadlines). Identifies, assesses and manages risks to the success of the project. Ensures that realistic project plans are maintained and ensures regular and accurate communication to stakeholders. Adopts appropriate project management methods and tools whether predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Ensures Quality reviews occur on schedule and according to procedure. Manages the change control procedure, and ensures that project deliverables are completed within agreed cost, timescale and resource budgets, and are signed off. Provides effective leadership to the project team, and takes appropriate action where team performance deviates from agreed tolerances.

Project management: Level 4

Defines, documents and carries out small projects or sub-projects (typically less than six months, with limited budget, limited interdependency with other projects, and no significant strategic impact), alone or with a small team, actively participating in all phases. Identifies, assesses and manages risks to the success of the project. Applies appropriate project management methods and tools whether predictive (plan-driven) approaches or adaptive (iterative/agile) approaches. Agrees project approach with stakeholders, and prepares realistic plans (including quality, risk and communications plans) and tracks activities against the project schedule, managing stakeholder involvement as appropriate. Monitors costs, timescales and resources used, and takes action where these deviate from agreed tolerances. Ensures that own projects are formally closed and, where appropriate, subsequently reviewed, and that lessons learned are recorded.

Quality management QUMG

Quality management establishes within an organisation a culture of quality and a system of processes and working practices to deliver the organisation's quality objectives. This involves the application of techniques for the monitoring and improvement of the quality of any aspect of a function, processes, products, services or data. The achievement of, and maintenance of compliance to, national and international standards, as appropriate, and to internal policies, including those relating to quality, service, sustainability and security.

Quality management: Level 7

Sets the quality strategy and policies for approval and adoption by organisational management and secures commitment to it from executive leadership. Determines the extent to which the quality policy meets the organisation's needs and objectives and reviews it as necessary. Establishes an organisational quality management system that delivers the quality strategy. Plans, resources and monitors the performance of the quality management system and an internal quality audit schedule. Defines and reviews quality systems. Ensures that adequate technology, procedures and resources are in place to support the quality system.

Quality management: Level 6

Prioritises areas for quality improvement by considering the strategy, wider business objectives and results from internal and external audits. Initiates the application of appropriate quality management techniques in these areas. Initiates improvements to processes by changing approaches and working practices, typically using recognised models. Achieves and maintains compliance against national and international standards, as appropriate. Identifies and plans systematic corrective action to reduce errors and improve the quality of the systems and services, by examination of the root causes of problems.

Quality management: Level 5

Advises on the application of appropriate quality management techniques and standards. Ensures that projects, teams and functions have appropriate practices in place and are meeting required organisational quality levels. Determines areas where existing processes should change from analysing audit findings. Takes responsibility for controlling updating and distributing organisational standards. Facilitates improvements to processes by changing approaches and working practices, typically using recognised models.

Quality management: Level 4

Assists projects, functions or teams in planning the quality management for their area of responsibility. Assists in the development of new or improved practices and organisational processes or standards. Facilitates localised improvements to the quality system or services.

Quality management: Level 3

Uses appropriate methods and a systematic approach in the development, maintenance, control and distribution of quality and environmental standards. Makes technical changes to and controls the updates and distribution of quality standards. Distributes new and revised standards.

Conformance review CORE

The independent assessment of the conformity of any activity, process, deliverable, product or service to the criteria of specified standards, best practice, or other documented requirements. May relate to, for example, asset management, network security tools, firewalls and internet security, sustainability, real-time systems, application design and specific certifications.

Conformance review: Level 6

Specifies organisational procedures for the internal or third-party assessment of an activity, process, product or service, against recognised criteria. Develops plans for review of management systems, including the review of implementation and use of standards and the effectiveness of operational and process controls. May manage the review, conduct the review or manage third party reviewers. Identifies areas of risk and specifies interrogation programs. Recommends improvements in processes and control procedures. Authorises the issue of formal reports to management on the extent of compliance of systems with standards, regulations and/or legislation.

Conformance review: Level 5

Plans formal reviews of activities, processes, products or services. Evaluates and independently appraises the internal control of processes, based on investigative evidence and assessments undertaken by self or team. Ensures that independent appraisals follow agreed procedure and advises others on the review process. Provides advice to management on ways of improving the effectiveness and efficiency of their control mechanisms. Identifies and evaluates associated risks and how they can be reduced.

Conformance review: Level 4

Conducts formal reviews of activities, processes, products or services. Collects, collates and examines records as part of specified testing strategies for evidence of compliance with management directives, or the identification of abnormal occurrences. Analyses evidence collated and drafts part or all of formal reports commenting on the conformance found to exist in the reviewed part of an information systems environment.

Conformance review: Level 3

Collects and collates evidence as part of a formally conducted and planned review of activities, processes, products or services. Examines records as part of specified testing strategies for evidence of compliance with management directives, or the identification of abnormal occurrences.

Safety assessment SFAS

The assessment of safety-related software systems to determine compliance with standards and required levels of safety integrity. This involves making professional judgements on software engineering approaches, including the suitability of design, testing, and validation and verification methods, as well as the identification and evaluation of risks and the means by which they can be reduced. The establishment, maintenance and management of an assessment framework and practices.

Safety assessment: Level 6

Leads assessments up to IEC 61508 Safety Integrity level 4 (or equivalent standard) or participates in any level of assessment. Determines assessment methods, techniques and tools that are to be used as appropriate to the integrity levels of the assessments undertaken.

Safety assessment: Level 5

Participates in assessments up to IEC 61508 Safety Integrity level 3 (or equivalent standard), and undertakes safety analyses on initial designs using HAZOPS, FMEA or similar methods.

Organisational capability development OCDV

The provision of leadership, advice and implementation support to assess organisational capabilities and to identify, prioritise and implement improvements. The selection, adoption and integration of appropriate industry frameworks and models to guide improvements. The systematic use of capability maturity assessments, metrics, process definition, process management, repeatability and the introduction of appropriate techniques, tools and enhanced skills. The delivery of an integrated people, process and technology solution to deliver improved organisational performance in line with organisation's strategic plans and objectives. The scope of improvement is organisational but may also be highly focussed as necessary for example software development, systems development, project delivery or service improvement.

Organisational capability development: Level 7

Represents and leads organisational capability improvement at the highest level. Determines the need for strategic organisation-level capability improvement to satisfy the strategic goals and long-term objectives of the organisation. Liaises with the organisation's functions to establish requirements and identifies, proposes, initiates and leads significant organisational capability improvement programmes. Manages the quality and appropriateness of the work performed and delivers measurable business benefits. Adopts and/or modifies existing capability improvement approaches as necessary.

Organisational capability development: Level 6

Leads substantial improvement programmes. Seeks out, identifies, proposes, and initiates capability improvement activities within the organisation typically driven by the need to enhance performance, satisfy new business opportunities or to respond to external drivers. Selects frameworks, approaches and techniques for use. Plans and manages the evaluation or assessment of organisational capabilities. Devises solutions and leads change initiatives including communication, transition and implementation activities. Takes action to exploit opportunities to deliver measurable, beneficial impacts upon operational effectiveness. Monitors international, national, and sector trends in order to establish the needed capability.

Organisational capability development: Level 5

Develops and maintains a detailed knowledge of capability improvement approaches and techniques and selects appropriate approaches for the organisation. Contributes effectively to identifying new areas of capability improvement within the organisation which may be enhancements to skills, technology or processes. Carries out capability improvement assignments, such as maturity or performance assessments to identify strengths and weaknesses. Selects and prioritises improvement opportunities, generates buy-in and plans improvement activities justified by measurable organisational benefits. Mentors and supports localised continuous improvement activities.

Levels of responsibility

This section describes the generic attributes that characterise SFIA's seven levels of responsibility and accountability. The underlying structure of the framework ensures that the definitions of professional skills are defined in a way that makes their different levels recognisably distinct and aligned to the levels of responsibility.

The power of the levels of responsibility

The SFIA seven Levels of Responsibility not only enable recognition of career progression but also provides a means by which other frameworks and corporate structures may map to the SFIA Framework. The nature of the generic attributes makes them suitable for use as the basis of core competencies, mappings and stages within a career path.

An organisation that already has a set of core competencies or values can use them in combination with SFIA's professional skills and benefit from the spacing that the SFIA levels provide.

An organisation, or a professional body or trade association for instance, that wishes to map its own established structure to SFIA can do so using the levels of responsibility characterised by the generic attributes as the basis of such a mapping.

Universal applicability

SFIA is intended as a framework for the digital, IT and software engineering community – the professional skills reflect this, although many are directly relevant outside of this area. Its universal applicability means that SFIA can be extended beyond these broad areas into any technical endeavour and probably beyond that too. These levels of responsibility allow for an integration of different professional work using the levels of responsibility as the foundation whether that be framework to framework or an organisation's structure to the SFIA Framework.

The full reference guide and the website provide advice and guidance on the adoption of SFIA.

Responsibility: Level 1

Autonomy

Works under supervision. Uses little discretion. Is expected to seek guidance in unexpected situations.

Influence

Minimal influence. May work alone, or interact with immediate colleagues.

Complexity

Performs routine activities in a structured environment. Requires assistance in resolving unexpected problems.

Knowledge

Has a basic generic knowledge appropriate to area of work. Applies newly acquired knowledge to develop new skills.

Business skills

Has sufficient communication skills for effective dialogue with others.

Demonstrates an organised approach to work.

Uses basic systems and tools, applications, and processes

Contributes to identifying own development opportunities.

Follows code of conduct, ethics and organisational standards. Is aware of health and safety issues.

Understands and applies basic personal security practice.

Responsibility: Level 2

Autonomy

Works under routine direction. Uses limited discretion in resolving issues or enquiries. Works without frequent reference to others.

Influence

Interacts with and may influence immediate colleagues. May have some external contact with customers, suppliers and partners. May have more influence in own domain. Aware of need to collaborate with team and represent users/customer needs.

Complexity

Performs a range of work activities in varied environments. May contribute to routine issue resolution.

Knowledge

Demonstrates application of essential generic knowledge typically found in industry bodies of knowledge. Has gained a basic domain knowledge. Absorbs new information when it is presented systematically and applies it effectively.

Business skills

Has sufficient communication skills for effective dialogue with customers, suppliers and partners.

Is able to work in a team. Is able to plan, schedule and monitor own work within short time horizons. Demonstrates a rational and organised approach to work.

Understands and uses appropriate methods, tools and applications.

Identifies and negotiates own development opportunities.

Is fully aware of and complies with essential organisational security practices expected of the individual.

Responsibility: Level 3

Autonomy

Works under general direction. Uses discretion in identifying and responding to complex issues and assignments. Receives specific direction, accepts guidance and has work reviewed at agreed milestones. Determines when issues should be escalated to a higher level.

Influence

Interacts with and influences colleagues. Has working level contact with customers, suppliers and partners. May supervise others or make decisions which impact the work assigned to individuals or phases of projects. Understands and collaborates on the analysis of user/customer needs and represents this in their work.

Complexity

Performs a range of work, sometimes complex and non-routine, in a variety of environments. Applies methodical approach to issue definition and resolution.

Knowledge

Has a sound generic, domain and specialist knowledge necessary to perform effectively in the organisation typically gained from recognised bodies of knowledge and organisational information. Demonstrates effective application of knowledge. Has an appreciation of the wider business context. Takes action to develop own knowledge.

Business skills

Demonstrates effective communication skills.

Plans, schedules and monitors own work (and that of others where applicable) competently within limited deadlines and according to relevant legislation, standards and procedures.

Contributes fully to the work of teams. Appreciates how own role relates to other roles and to the business of the employer or client.

Demonstrates an analytical and systematic approach to issue resolution.

Takes the initiative in identifying and negotiating appropriate personal development opportunities.

Understands how own role impacts security and demonstrates routine security practice and knowledge required for own work.

Responsibility: Level 4

Autonomy

Works under general direction within a clear framework of accountability. Exercises substantial personal responsibility and autonomy. Plans own work to meet given objectives and processes.

Influence

Influences customers, suppliers and partners at account level. May have some responsibility for the work of others and for the allocation of resources. Participates in external activities related to own specialism. Makes decisions which influence the success of projects and team objectives. Collaborates regularly with team members, users and customers. Engages to ensure that user needs are being met throughout.

Complexity

Work includes a broad range of complex technical or professional activities, in a variety of contexts. Investigates, defines and resolves complex issues.

Knowledge

Has a thorough understanding of recognised generic industry bodies of knowledge and specialist bodies of knowledge as necessary. Has gained a thorough knowledge of the domain of the organisation. Is able to apply the knowledge effectively in unfamiliar situations and actively maintains own knowledge and contributes to the development of others. Rapidly absorbs new information and applies it effectively. Maintains an awareness of developing practices and their application and takes responsibility for driving own development.

Business skills

Communicates fluently, orally and in writing, and can present complex information to both technical and non-technical audiences.

Plans, schedules and monitors work to meet time and quality targets.

Facilitates collaboration between stakeholders who share common objectives.

Selects appropriately from applicable standards, methods, tools and applications.

Fully understands the importance of security to own work and the operation of the organisation. Seeks specialist security knowledge or advice when required to support own work or work of immediate colleagues.

Responsibility: Level 5

Autonomy

Works under broad direction. Work is often self-initiated. Is fully responsible for meeting allocated technical and/or project/supervisory objectives. Establishes milestones and has a significant role in the assignment of tasks and/or responsibilities.

Influence

Influences organisation, customers, suppliers, partners and peers on the contribution of own specialism. Builds appropriate and effective business relationships. Makes decisions which impact the success of assigned work, i.e. results, deadlines and budget. Has significant influence over the allocation and management of resources appropriate to given assignments. Leads on user/customer collaboration throughout all stages of work. Ensures users' needs are met consistently through each work stage.

Complexity

Performs an extensive range and variety of complex technical and/or professional work activities. Undertakes work which requires the application of fundamental principles in a wide and often unpredictable range of contexts. Understands the relationship between own specialism and wider customer/organisational requirements.

Knowledge

Is fully familiar with recognised industry bodies of knowledge both generic and specific. Actively seeks out new knowledge for own personal development and the mentoring or coaching of others. Develops a wider breadth of knowledge across the industry or business. Applies knowledge to help to define the standards which others will apply.

Business skills

Demonstrates leadership. Communicates effectively, both formally and informally.

Facilitates collaboration between stakeholders who have diverse objectives.

Analyses, designs, plans, executes and evaluates work to time, cost and quality targets. Analyses requirements and advises on scope and options for continuous operational improvement. Takes all requirements into account when making proposals. Demonstrates creativity, innovation and ethical thinking in applying solutions for the benefit of the customer/stakeholder

Advises on the available standards, methods, tools and applications relevant to own specialism and can make appropriate choices from alternatives.

Maintains an awareness of developments in the industry. Takes initiative to keep skills up to date. Mentors colleagues.

Assesses and evaluates risk.

Proactively ensures security is appropriately addressed within their area by self and others. Engages or works with security specialists as necessary. Contributes to the security culture of the organisation.

Responsibility: Level 6

Autonomy

Has defined authority and accountability for actions and decisions within a significant area of work, including technical, financial and quality aspects. Establishes organisational objectives and assigns responsibilities.

Influence

Influences policy and strategy formation. Initiates influential relationships with internal and external customers, suppliers and partners at senior management level, including industry leaders. Makes decisions which impact the work of employing organisations, achievement of organisational objectives and financial performance.

Complexity

Has a broad business understanding and deep understanding of own specialism(s). Performs highly complex work activities covering technical, financial and quality aspects. Contributes to the implementation of policy and strategy. Creatively applies a wide range of technical and/or management principles.

Knowledge

Promotes the application of generic and specific bodies of knowledge in own organisation. Has developed business knowledge of the activities and practices of own organisation and those of suppliers, partners, competitors and clients.

Business skills

Demonstrates clear leadership. Communicates effectively at all levels to both technical and non-technical audiences.

Understands the implications of new technologies. Understands and communicates industry developments, and the role and impact of technology in the employing organisation. Absorbs complex information.

Promotes compliance with relevant legislation and the need for services, products and working practices to provide equal access and equal opportunity to people with diverse abilities.

Takes the initiative to keep both own and colleagues' skills up to date.

Manages and mitigates risk.

Takes a leading role in promoting security throughout own area of responsibilities and collectively in the organisations.

Responsibility: Level 7

Autonomy

At the highest organisational level, has authority over all aspects of a significant area of work, including policy formation and application. Is fully accountable for actions taken and decisions made, both by self and others to whom responsibilities have been assigned.

Influence

Makes decisions critical to organisational success. Inspires the organisation, and influences developments within the industry at the highest levels. Advances the knowledge and/or exploitation of technology within one or more organisations. Develops long-term strategic relationships with customers, partners, industry leaders and government.

Complexity

Leads on the formulation and implementation of strategy. Applies the highest level of leadership skills. Has a deep understanding of the industry and the implications of emerging technologies for the wider business environment.

Knowledge

Has established a broad and deep business knowledge including the activities and practices of own organisation and a broad knowledge of those of suppliers, partners, competitors and clients. Fosters a culture to encourage the strategic application of generic and specific bodies of knowledge within own area of influence.

Business skills

Has a full range of strategic management and leadership skills.

Communicates the potential impact of emerging practices and technologies on organisations and individuals and assesses the risks of using or not using such practices and technologies.

Understands, explains and presents complex ideas to audiences at all levels in a persuasive and convincing manner.

Assesses the impact of legislation and actively promotes compliance and inclusivity.

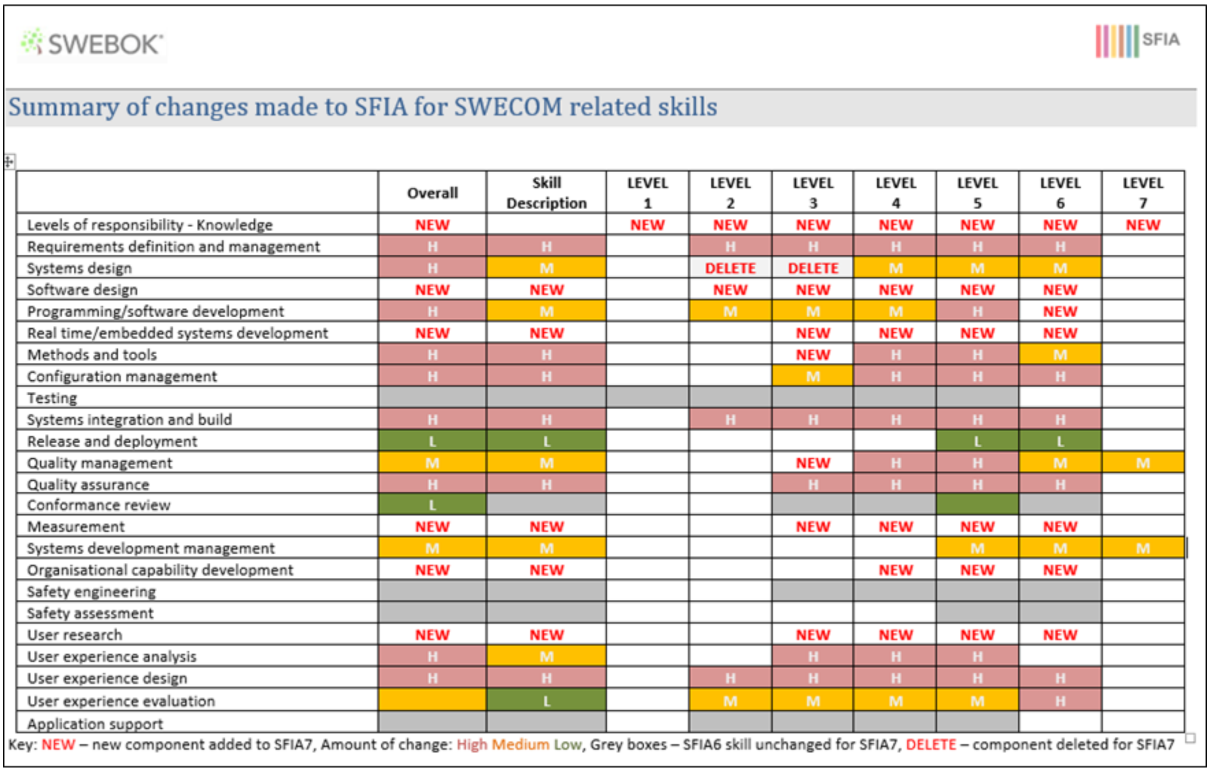
Ensures that the organisation develops and mobilises the full range of required skills and capabilities.

Champions security within own area of work and throughout the organisation.

Appendix A: Cross reference SWE competencies to relevant Knowledge areas in SWEBOK.

SWE Competencies	SWECOM Knowledge Areas
Requirements definition and management REQM	Software Requirements Software Engineering Models
Systems design DESN	Software Design Software Engineering Models
Software design SWDN	Software Design Software Engineering Models
Programming/software development PROG	Software Construction
Real-time/embedded systems development RESD	Software Requirements Software Design Software Construction Software Testing
Methods and tools METL	Software Requirements Software Engineering Process
Configuration management CFMG	Software Requirements Software Configuration
Testing TEST	Software Requirements Software Construction Software Testing Software Quality
Systems integration and build SINT	Software Construction Software Testing
Release and deployment RELM	Software Maintenance
Quality assurance QUAS	Software Testing Software Quality
Measurement MEAS	Software Engineering Process
Safety engineering SFEN	Software Requirements Software Design
Application support ASUP	Software Maintenance
Systems development management DELM	Software Engineering Management Software Engineering Process Software Engineering Economics
Project management PRMG	Software Engineering Management
Quality management QUMG	Software Quality Software Engineering Process
Conformance review CORE	Software Quality
Safety assessment SFAS	Software Quality
Organisational capability development OCDV	Software Engineering Process

Appendix B: Changes made to SFIA 6 to fully incorporate Software Engineering in SFIA 7



Summary of changes made to SFIA for SWECOM related skills

	Overall	Skill Description	LEVEL 1	LEVEL 2	LEVEL 3	LEVEL 4	LEVEL 5	LEVEL 6	LEVEL 7
Levels of responsibility - Knowledge	NEW		NEW	NEW	NEW	NEW	NEW	NEW	NEW
Requirements definition and management	H	H		H	H	H	H	H	
Systems design	H	M		DELETE	DELETE	M	M	M	
Software design	NEW	NEW		NEW	NEW	NEW	NEW	NEW	
Programming/software development	H	M		M	M	M	H	NEW	
Real time/embedded systems development	NEW	NEW			NEW	NEW	NEW	NEW	
Methods and tools	H	H			NEW	H	H	M	
Configuration management	H	H			M	H	H	H	
Testing	▨	▨	▨	▨	▨	▨	▨		
Systems integration and build	H	H		H	H	H	H	H	
Release and deployment	L	L					L	L	
Quality management	M	M			NEW	H	H	M	M
Quality assurance	H	H			H	H	H	H	
Conformance review	L	▨			▨	▨	🟩	▨	
Measurement	NEW	NEW			NEW	NEW	NEW	NEW	
Systems development management	M	M					M	M	M
Organisational capability development	NEW	NEW				NEW	NEW	NEW	
Safety engineering	▨	▨			▨	▨	▨	▨	
Safety assessment	▨	▨					▨	▨	
User research	NEW	NEW			NEW	NEW	NEW	NEW	
User experience analysis	H	M			H	H	H		
User experience design	H	H		H	H	H	H	H	
User experience evaluation	🟨	L		M	M	M	M	H	
Application support	▨	▨		▨	▨	▨	▨		

Key: NEW – new component added to SFIA7, Amount of change: High Medium Low, Grey boxes – SFIA6 skill unchanged for SFIA7, DELETE – component deleted for SFIA7