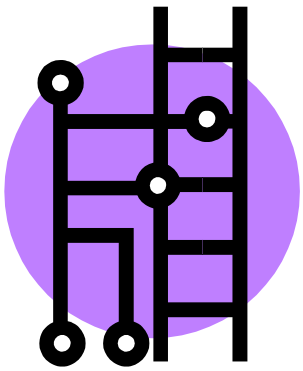
A Competency and Skills Framework for the Assessment of Software Engineering in the Royal Air Force

David Gill

January 2005



A dissertation submitted in partial fulfilment of the requirements for the degree of Master of Science in Software Engineering

Kellogg College
University of Oxford

Abstract

The idea for this dissertation was borne out of a study into military aviation design support services that made numerous recommendations, one of which was that a full review of software maintenance core competencies should be undertaken. However, what I initially thought would be a simple enough task soon revealed itself to be quite a complex and labour-intensive beast.

In order to conduct a review of core competencies, I first had to revisit the accepted academic definitions of Software Engineering (SE), dissecting and examining them in the context of military usage. From this work, I was able to arrive at a new definition that I subsequently used to create a SE Activity Model that encompassed both the development and support environments.

I then developed a strategy for the creation of a SE Competency and Skills (C&S) Assessment Framework, and subsequently created the framework itself. This was by far the longest phase of the dissertation project.

Once I had completed the C&S framework, I then trialled it on a military software maintenance team. The results of this trial were encouraging and enabled me to make recommendations for improving the framework, potential extensions to its use and suggestions for future work in other areas.

Acknowledgements

This dissertation would not have been possible were it not for the patience and support of my friends and colleagues. In particular I would like to thank the following for their specific contributions:

> The staff of the Software Engineering Programme, especially my project supervisor, Dr Jim Davies, for his wisdom and guidance.
>
> Scott Weekes, Jamie Brooks and Ian Grose for their encouragement, constructive criticism and challenging insight. Special thanks should be given to Jamie for his readiness to thoroughly absorb and discuss my ideas and Ian for taking the time to edit my work and save me from potential embarrassment!
>
> Mike Place and the personnel of HSMU who took the time to take part in the framework evaluation, giving me valuable feedback in the process.

However, the greatest appreciation by far must go to my family and Mary for giving me the strength and impetus to finish this task.

The author confirms that this dissertation does not contain material previously submitted for another degree or academic award and the work presented here is the author's own, except where otherwise stated.

Any opinions expressed in this dissertation are solely those of the author and can not be attributed to the Royal Air Force or Ministry of Defence.

Contents

Front Matter

Main Matter

Back Matter

Tables

Figures

Introduction

1.1 Background

The Royal Air Force (RAF) is no stranger to software; indeed, when compared to the British Army and Royal Navy, RAF defence capability has relied in part upon software-intensive systems since the early 1970s with the introduction of the Nimrod R1. It should therefore come as no surprise to the reader that there currently exists a number of RAF software maintenance teams, collectively referred to as Software Support Teams (SSTs). By and large, these SSTs each maintain a real-time application, effectively complimenting a platform's[1] Design Authority[2] (DA), and are staffed almost exclusively by military personnel giving them the prefix of *in-Service* SSTs.

The exact rationale behind the creation of each SST, however, is not altogether clear but can be surmised from their common tasking activities. Essentially, the existence of a platform SST affords the RAF a measured degree of control when considering certain software maintenance activities that may directly impinge upon the platform's operational capability and/or availability. In reality, this means that the RAF is relatively free to dictate the level and extent of some platform software maintenance activities undertaken in order to meet its wider operational commitment as long as it is willing to accept the risk. As such, most SST modifications are therefore released as Service Engineered Modifications (SEMs).

This tasking freedom is a feature that the RAF frequently exploits as it allows for the production of mission-critical software independent of the respective platform's DA. Without the need for this design contractor approval, the RAF can introduce new functionality, correct faults and enhance a system's capability all with typical military expediency. In essence the RAF has, to a certain extent, taken responsibility for mission-critical software maintenance in the operation/maintenance phase of a system's life-cycle.

However, the individual nature of each SST has essentially meant that there has been little by way of standardisation between them. The reader might be forgiven for thinking that this in itself should not present too much of a problem. After all, as long as there is a dedicated team established to support each platform then what should it matter if they all do things slightly differently? Indeed, this is analogous to most similar-product civilian manufacturing organisations (for example the motor manufacturing industry). This argument may well have held true when the SSTs were first formed, when funds and personnel were aplenty and when platforms were introduced by multifarious project teams.

[1] Generally speaking, supported equipment or aircraft.
[2] Normally the original design contractor for the platform.

However, problems arise when one considers the in-house nature of SSTs. As they are effectively owned and staffed almost exclusively by the RAF, consideration has to be given to the resource provisioning implications of bespoke formation – high initial set up costs, low opportunities for skills cross-pollination, difficulties in career progression. Today, with an increasingly diminishing defence budget, the RAF can no longer afford the luxury of uniqueness when considering the support of software intensive systems. Likewise, as software becomes the all pervasive deliverer of functionality, it can not afford to ignore it either. The key to support flexibility and mission availability now rests on careful resource management, and in particular one of the most valuable resources an organisation can ever have – personnel.

1.2 Aim

The primary aim of this dissertation is the enabling of a Software Engineering (SE) competency evaluation of RAF SSTs. Of secondary importance is the enabling of the identification and definition of RAF SST manning requirements. It is intended to achieve both of these aims through the creation and application of a SE Competency and Skills (C&S) Assessment Framework.

1.3 Project Outline

This dissertation will comprise a thesis with supporting project work. An outline of the project itself is as follows:

Defining Software Engineering

Software maintenance and SE can have significantly diverse meanings dependent on the task that they are required to describe. Indeed, even within the academic community itself, there exists a disparity of understanding over the terms themselves, and to make the assertion that all RAF SSTs undertake one over the other is not as straightforward as you may think. In order to address this problem, however, it is necessary to begin somewhere so I have chosen to analyse SE with the intention of arriving at a common definition and adapting it to the functional requirements of the RAF. Once this is achieved, however, it is then a relatively straightforward case of identifying and defining SE Functions and Activities.

Creation of a Competency & Skills Assessment Framework

Once the SE Functions and Activities have been defined, the next stage is to produce competency statements for each. This process requires the formation of a C&S creation strategy based on established sources of information, and then the application of this strategy in order to populate a C&S framework. It is not possible, however, to create statements for all of the competencies listed as this requires a degree of application domain knowledge that I do not possess. In order to overcome

this, I proceed on the basis that I can use the application phase of the framework itself to address these elements, drawing on the knowledge of interviewees as appropriate.

Framework Assessment

It would be preferable to trial the framework on as many SSTs as possible before publishing the final product. However, as dissertation size and available work time constraints do not allow for this to happen, I trial it on one SST only. I do not believe that this poses too great a handicap, as I am still able to effectively appraise the shortfalls of the developed framework assessment process itself.

Critical Review

Once the practical application of the framework is complete, I report the results and conduct a critical appraisal of the dissertation. From this I make recommendations for changes to the framework and possible extensions to its use.

1.4 Academic Relevance

Before I embark on the project work itself, however, it is well worth mentioning how this dissertation relates to the Software Engineering Programme (SEP). While conducting research and developing my thesis, I have attempted to utilise some of the taught principles of SE management throughout this dissertation. Specifically, I found the concept of ethnomethodological analysis (as taught in Requirements Engineering) and the notion of total integration and control of all relevant elements (as taught on Managing Risk and Quality) particularly useful. In addition, I was also able to apply some of the high level principles of Object Oriented Design in Section 3 in order to better illustrate my point. However, there is a more general connection to be made: that between the skills discussed here and the curriculum and general intent of a programme of education in SE.

This dissertation is focussed on the needs of the customer; it approaches the subject of SE from an owner's perspective, and makes bold recommendations that will ultimately require a shift in how SE is perceived, understood, applied and taught in both the business and academic communities. This raises questions regarding the skill sets of software engineers and customers alike. What education and training is required to produce an "intelligent customer"? What information does an intelligent customer ultimately need? Should managers be educated in the principles of product support? I hope that this work goes at least part way to answering some, if not all, of these questions.

Understanding Software Engineering

There are many different definitions of the term Software Engineering (SE) in use today leading to varying degrees of confusion from both within and without the SE community. Indeed, this disparity of understanding can have consequences both for the developer and customer especially if each party has different expectations of the other. An extreme example could be the eventual non-delivery of an entire project with the blame for this failure being laid firmly at each other's doorstep; a less serious illustration could be the omission of an expected deliverable. In any case, there clearly exists a need for this potential confusion to be eradicated, and, likewise, before any examination of RAF SE competencies can be undertaken it is first essential to establish a common understanding of SE between the author and reader.

2.1 A Definitive Problem

The Oxford Dictionary of English [Oxford, 2003] does not have a singular definition for SE. Instead it defines "software" as:

> "*the programs and other operating information used by a computer"*

and defines "engineering" as:

> *"the branch of science and technology concerned with the design, building, and use of engines, machines and structures"*

In the author's opinion, these definitions are best described as generalisations. Indeed, it would be somewhat difficult to arrive at a consensus over the meaning of "programs" and "operating information", apart from the notion that software is in some way related to the production of an instruction set required to execute a particular operation or control function within a computer. The engineering element is a little more specific, however, and although it is apparently only concerned with hardware (*engines, machines and structures*), it does manage to allude to the requirement for a life-cycle (*design, building and use*). However, there is precious little detail in this definition; a complete absence of any reference to any supporting processes such as project management, quality assurance, configuration management, measurement or the production of documentation. And, if we take a holistic view of a life-cycle from inception through to retirement, there is no reference to the requirement for any support activities, either. If we are ever to achieve a common understanding of SE then these definitions clearly will not do.

In 1968 Fritz Bauer [Naur, 1969] at a NATO conference in Garmisch, Germany, attempted to address the SE definition problem by proposing the following:

> *"[Software engineering is] the establishment and use of sound engineering principles in order to obtain, economically, software that is reliable and works efficiently on real machines"*

Whilst including the engineering function associated with software production, Bauer's definition also refers to the economical production of reliable and efficient software. To quote [McDermid, 1994], a "consideration of reducing the cost of production, [the use of] reliability metrics to monitor and control…development, and the critical issue of [software development] efficiency" are SE attributes that fall into the category of project management responsibilities. Therefore what Bauer effectively did was to introduce the concept of project management into the definition of SE, and spectacularly fail to include any detail as to how this was to be achieved. And, again, there is no reference to the requirement for any supporting activities either. However, there is no doubt that this definition is better than those published in [Oxford, 2003].

For inspiration then, one may turn to the IEEE Computer Society [IEEE, 1999]. The definition:

> *"Software engineering: (1) The application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software; that is, the application of engineering to software. (2) The study of approaches as in (1)"*

manages to include project management and even goes some way to identifying the associated activities of methodical planning (*systematic*), control (*disciplined*) and measurement (*quantifiable*) as well. However, we must ask ourselves is this detail enough? Could we be sure that we were doing everything that we need to do in order to produce engineered software? Perhaps not. There is a positive aspect to this definition, however, in the reference to the operation and maintenance phases of software. At least here the IEEE are seeking to acknowledge that software does indeed have a life outside of the development phase. However, it is the author's opinion that more detail is required if we are ever to be able to make any progress towards creating a C&S framework.

Having already quoted from John McDermid, maybe it is to he who we should turn to for a more comprehensive definition of SE. His definition [McDermid, 1994]:

> *"Software engineering is the science and art of specifying, designing, implementing and evolving – with economy, timeliness and elegance – programs, documentation and operating procedures whereby computers can be made useful to man"*

again, however, appears to be incomplete. Whilst he has striven to identify requirements gathering, designing, building and (the somewhat vaguely referenced)

evolving phase, he has still managed to omit any reference to project management. In effect, this is a more high-level and less descriptive definition of SE, which, unfortunately, is not what we are seeking.

What is abundantly clear from these different definitions is that they all appear to be lacking in one way or another. This can be partly attributed to which viewpoint, either developer or customer, that you adopt. The developer is concerned with the business aspect of SE, getting the software product to market as soon as possible and returning a profit. In order to achieve these goals, the developer will predominately be concerned with producing software and rationalising this production process as far as possible in order to reduce time to market. The customer, on the other hand, not only wants their software product as soon as possible but is also concerned with using and living with it after delivery. In effect, it is the support of the software product that takes precedence in the eyes of the customer once they have taken delivery of it. Unfortunately, regardless of which viewpoint you adopt, what is patently unclear is that there is no acknowledged activity set; no agreed list of what needs to be undertaken, both by developer and customer, in order to produce "correctly engineered software".

2.2 A Process Definition

Let us then approach the problem of defining SE from a different perspective. There are many texts that purport to give a definitive list of SE activities. If we take the curriculum point of view, for example, the Open University's Software Engineering syllabus [Open, 1997], we find a number of SE topics being used as a basis for teaching (see Appendix A, Table A1). This list not only serves to provide the student with a general all-round understanding of the SE discipline, but also serves as a fundamental foundation for establishing a set of SE "must do's". If we take a high level view of this syllabus, we can see what immediately appears to be an embryonic model of SE activities, thus:

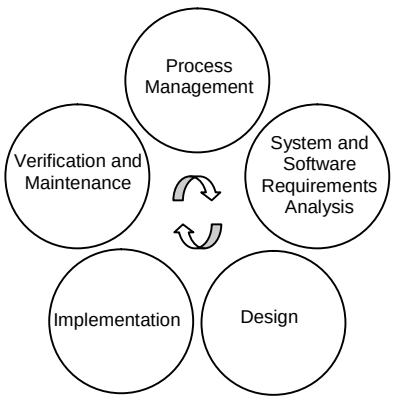


Figure 2.1 – Academic SE Topics

At first glance this appears to be an adequate basis for a SE model. Indeed, to make this assumption is forgivable given the limited detail presented. However, on closer inspection it appears to be nothing more than a broad list of tools and techniques that can be utilised for the production of engineered software. Nevertheless, it does effectively serve to transcribe a SE life-cycle whilst at the same time factoring in a requirement for some kind of overall management activity. However, if we examine this list again, there is one aspect that immediately appears to extend beyond the recognized bounds of SE – that of *System* and Software Requirements Analysis.

This reference to the system in which the software will reside represents an academic acknowledgement that SE cannot be viewed as simply being concerned with the production of software. We can extend this notion further by taking a contextual view of the SE process where we not only have the project life-cycle that produces our software product, but also the product's life-cycle after project development finishes. Effectively it is the software's relationship with the system as a whole that will ultimately drive out our SE definition.

A Customer Perspective

The RAF has attempted to acknowledge the relationship between software and system. Termed "software support" it is an approach to SE which attempts to cater not only for the physical system within which the software resides but also the operational system within which it is utilised [JAP, 2004]. This approach, however, is currently under review as previous attempts at producing a software support model have either failed to include any reference to users or have tried to simultaneously model both the functional and physical aspects of a software support system creating much confusion and debate in the process. I have therefore adapted an idea by [Brooks, 2004] (see Figure 2.2 – Software Support Model) that effectively only concerns itself with the functional requirements of a software support system.

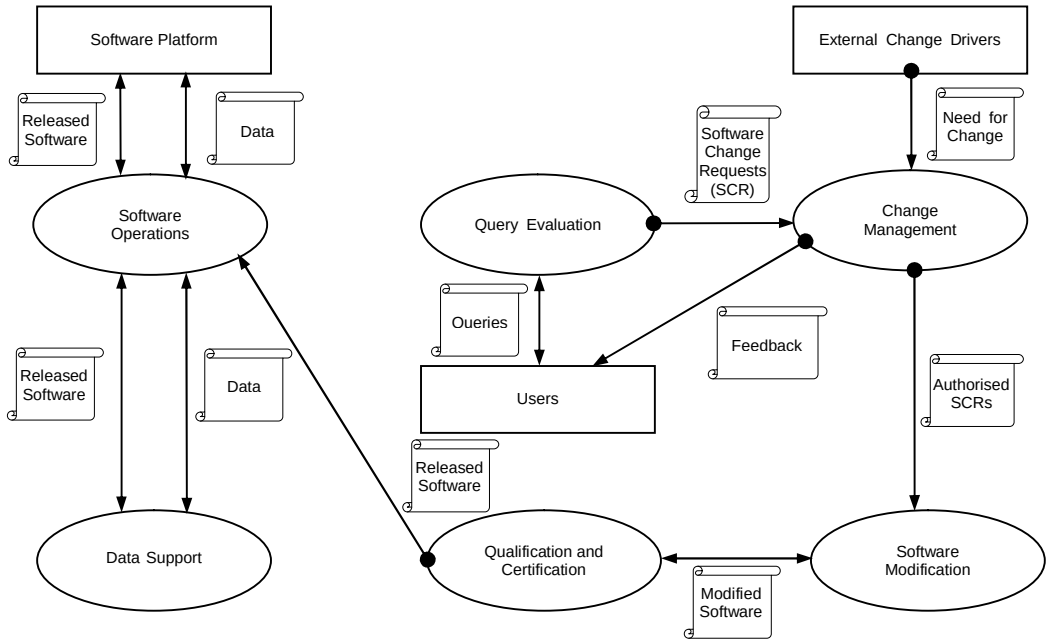


Figure 2.2 – Software Support Model, based on an idea by [Brooks, 2004]

Looking back at the SE definitions thus far encountered, we see that it is only the RAF that appears to have made a concerted effort to cater for the life of the software outside of its development environment. Known as the "evolutionary phase", during which maintenance activities are undertaken both on and with the software product in question, it has meant taking a systems approach to software development that serves to cater for the needs of the user organisation. This has been necessitated by the fact that almost all software-intensive systems procured by the RAF represent a considerable long-term financial investment.

Getting the Scope Right

In a bid, therefore, to arrive at an acceptable definition of SE we have identified several areas that appear to either fall outside the acknowledged boundaries of SE, or have been overlooked entirely. Consider then once again the SE definitions we have encountered. All of them, as one would expect, refer to the production of engineered software. Some of them refer to the support of that software once it has been produced. None of them refer to the system within which the software will ultimately reside. In order to dispel any confusion, a SE definition is required that will take us beyond the software development project boundary, essentially an all encompassing definition that takes into account the product's journey from cradle to grave.

For that reason we need to look beyond our preconceived boundaries of SE in order to find an appropriate definition that we can utilise. Software and its production is

essentially one element of a system that in turn requires global process management, and we need to identify all the constituent elements of this process. In his book, System Engineering Management, Benjamin Blanchard proposes this definition for SysE:

> *"The application of scientific and engineering efforts to (1) transform an operational need into a description of system performance parameters and a system of configuration through the use of an iterative process of definition, synthesis, analysis, design, test and evaluation, and validation; (2) integrate related technical parameters and ensure the compatibility of all physical, functional, and program interfaces in a manner that optimises the total definition and design; and (3) integrate reliability, maintainability, usability (human), safety, producibility, supportability (serviceability), disposability, and other such factors into the total engineering effort to meet cost, schedule, and technical performance objectives"* [Blanchard, 1998]

This truly comprehensive definition is Blanchard's identification of all engineering activities required to meet the customer's needs. This not only concerns itself with the project that creates the software product, but also with the product's life outside the project environment, and it is the author's proposition that this definition be used for SE, and subsequently as the basis for the creation of a SE Activity Model.

Before we can proceed any further, however, we first need to partition Blanchard's definition to allow subsequent consideration and modelling. I have achieved this by identifying specific functional areas for consideration thus:

a) *"…transform an operational need into a description of system performance parameters…"*. Here, Blanchard introduces the notion of system creation through the application of a transformation process to the customer's needs. The acknowledgement that **system** performance parameters are the output of this process will form the basis for the first functional area.

b) *"…through the use of an iterative process of definition, synthesis, analysis, design, test and evaluation, and validation"*. Here, Blanchard introduces the life-cycle into the engineering process definition. If we are to adopt this for SE, however, it needs to be adapted to our requirements. This can initially be achieved by making the life-cycle element software-specific; however, this may then be perceived as precluding its application to hardware within a system. Approaching this from a different perspective, we could identify both hardware and software as separate functional areas within any model, each with their own development life-cycles. However, as we are primarily concerned with the production of a SE model and in order to limit any confusion, my focus shall be on the **software** itself. It is evident,

however, that the life-cycle process is applicable to both hardware and software, and should therefore be borne in mind by the reader.

c) *"…integrate related technical parameters and ensure the compatibility of all physical, functional, and program interfaces in a manner that optimises the total definition and design"*. Although Blanchard is referring to physical, functional and program interfaces, it is the notion of integration that concerns us more here. For any project to be successful, there are certain essential activities that are required to be undertaken in order to achieve total system integration (e.g. project management, quality assurance and configuration management, to name but three). These "controlling" activities can be grouped together into the general functional area of **supporting activities** that are applicable to all areas of the SE process.

d) *"…reliability, maintainability, usability (human), safety, producibility, supportability (serviceability), disposability, and other such factors"*. These characteristics can all be attributed to the system's **supporting infrastructure**, whether applied to the product itself or to its operating and maintenance environments. They shall be used to identify areas that require management in order to facilitate SE.

SE Definition Conclusions

There currently does not exist any suitable definition of SE that completely meets the needs of the RAFs unique approach to the support of software. In order to address this, I have proposed the use of a SysE definition by [Blanchard, 1998] and subsequently partitioned it into four main areas for further consideration: Software, System, Supporting Infrastructure and Supporting Activities. These areas will now be used to form the basis of a SE Activity Model.

Creating a Model of Software Engineering Activities

Having therefore arrived at a suitable definition for SE, the next stage is to add detail; in effect, to unambiguously model what should be undertaken in order to achieve engineered software. This model will then form the basis for the conducting of a SE competency examination of RAF SSTs.

It should be reiterated that the secondary purpose of this dissertation is to enable the establishment of manning requirements for SSTs. The current Engineering Technician Avionics (Eng Tech Av) manpower bias presently employed by the RAF on SSTs is currently under review by Strike Command (STC), with a view to creating a new "supporting" trade group (Communications Information Systems). Consequently, it has been postulated that there are certain functions within RAF SSTs that do not necessarily require the skills and aircraft engineering background (i.e. operating domain knowledge) of a trained Eng Tech Av. This will undoubtedly have an effect on the way SSTs are structured and perform their tasking functions in the future, and it is intended, where possible, to identify in this dissertation functions and activities that do not necessarily require this operating domain knowledge.

Let us then return to the RAFs Software Support Model (Figure 2.2). Six activities are identified within this model, namely:

> *"Query Evaluation,*
> *Change Management,*
> *Software Modification,*
> *Qualification and Certification,*
> *Data Support, and,*
> *Software Operations."*

These activities are typical of those currently being undertaken by most RAF SSTs and can initially be functionally represented as follows:

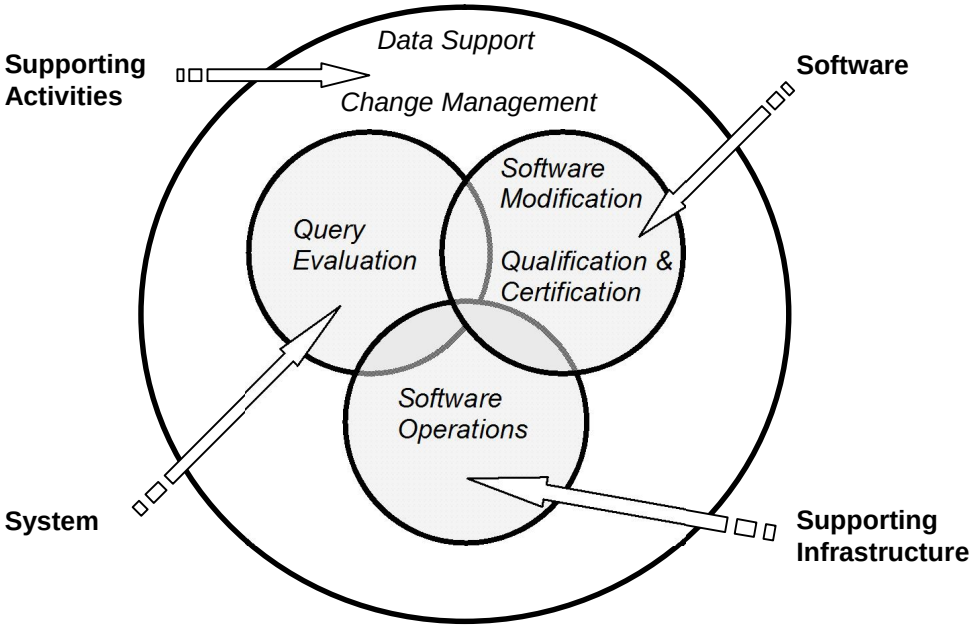


Figure 3.1 – RAF Software Support Team Functional Areas

The four functional areas in Figure 3.1 (Software, System, Supporting Infrastructure and Supporting Activities) and Software Support Model activities will henceforth form the basis from which to build a SE Activity Model. Taking this one stage further, it is also possible to initially perceive the following basic SST manning structure, reproduced here with reference to the proposed STC Communications Information Systems trade group and Figure 3.1:

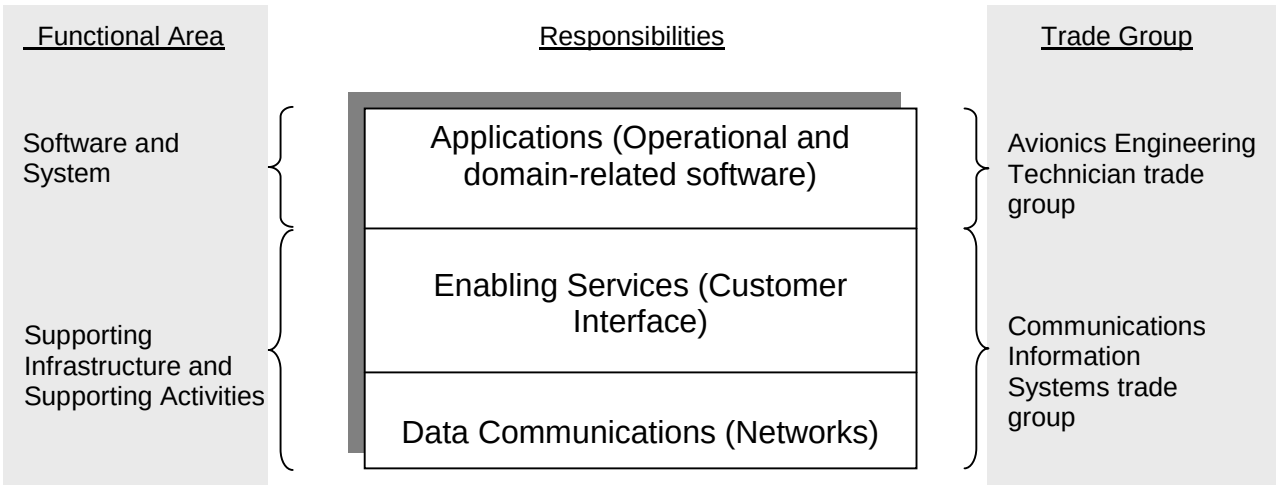


Figure 3.2 – Basic RAF SST Manning Model[3]

[3] Based on a proposition by the Communications Information Systems Manpower Strategy Team, RAF STC.

I do expect, however, that as this dissertation proceeds my initial perceptions on functional boundaries and SST manning (represented by Figures 3.1 and 3.2) may require some degree of modification. That fact established, let us now proceed with the evaluation of each Functional Area of Figure 3.1, beginning with the Software Function.

3.1 The Software Function

Let us then first concentrate on the Software functional area of Figure 3.1, in particular the Software Modification (SM) activity. SM is defined by the [100D-10, 1998] as:

> *"…the development and implementation of a design change to an in-Service software item…[with] 4 distinct types of software change:*
>
> *a. Corrective. A corrective change modifies a software item to remove a software fault.*
>
> *b. Adaptive. An adaptive change modifies a software item to enable it to continue to meet its specification in a changed environment.*
>
> *c. Perfective. A perfective change modifies a software item to enable it to meet its existing specification in an improved fashion.*
>
> *d. Enhancement. An enhancement change modifies a software item to add additional functionality to the system."*

At first glance it appears that what we are dealing with here is in fact a post-development activity, perhaps more accurately described as software maintenance, and if we were to produce a model based on this we could be forgiven for thinking that we may potentially overlook some relevant aspect of software development. However, the principles required to modify a software product are the same as the ones required to create that software product in the first place. This can be best illustrated by considering the difference between software and hardware modification. A corrective change is a modification activity peculiar to software in that its aim is to alter the software in some way or other in order for it to meet its original requirements. A corrective change to hardware (also known as a repair) on the other hand, is intended to return the hardware to its original state in order for it to meet its original requirements. The SM activity, therefore, will always result in the creation of a "new" software product as its output, regardless of whether the change was to correct a fault or to improve the product. This effectively means that software development and post-development activities can both be considered as SM.

However, this still does not identify what should be done in order to perform SM. The Software Engineering Body of Knowledge [SWEBOK, 2001] has attempted to address this issue by establishing ten Knowledge Areas (KA)[4] identified by it as residing within the IEEE Computer Society's definition of SE (as reproduced in this dissertation in Section 2.1). These are:

Software Requirements	SM
Software Design	SM
Software Construction	SM
Software Testing	SM
Software Maintenance	SM
Software Configuration Management	SA
Software Engineering Management	SA
Software Engineering Process	SA
Software Engineering Tools and Methods	SA
Software Quality	SA

Key: ^{SM} – Software Modification, ^{SA} – Supporting Activities

Table 3.1 – SE Knowledge Areas, after [SWEBOK, 2001]

Five of the KAs follow the classic waterfall life-cycle (identified in Table 3.1 by ^{SM}) and chime well with Blanchard's definition of SysE (the other five are identified by ^{SA} and will be discussed in the next section).

We can utilise the five ^{SM} KA activities to model SM and in doing so create the initial beginnings of an engineering life-cycle. However, there is also the RAF SST activity of Qualification and Certification to consider. [JAP, 2004] defines this as:

> *"During Qualification and Certification, the DA* [Design Authority] *demonstrates that the requirements have been met and the IPT* [Integrated Project Team] *(usually with independent safety audit, assessment or advice) is to satisfy itself that the software is acceptably safe for Release to Service, as detailed in JSP* [Joint Service Publication] *553. A final evaluation of user workload and performance envelope issues is also to be made, and updated user documentation is to be approved and issued"*

It can be argued that this is nothing more than a stringent scope for software testing. For comparison, though, we shall examine the [SWEBOK, 2001] definition of Software Testing:

[4] Knowledge Areas – The aims of [SWEBOK, 2001] were to promote a consistent worldwide view of SE, organise material recognised as belonging to the SE discipline into KAs, characterise the contents of SE, provide a topical access to the SWEBOK and provide a foundation for curriculum development.

"Software testing consists of the dynamic verification of the behaviour of a program on a finite set of test cases, suitably selected from the usually infinite executions domain, against the specified expected behaviour"

What is important to note here is the term "suitably selected" and what this actually implies. There are many test techniques available that may or may not be suitable for a given situation. What this means for the tester is that they must be aware of the different selection criteria and the implications of suitable test selection, especially in safety-related systems. This is a complex practice that requires the application of risk analysis techniques in order to meet the objectives of the testing phase. The following is some (but not all) of the testing topics identified in the [SWEBOK, 2001] decomposition of the testing phase:

a) **Testing and certification** – "…it is informative to consider testing from the point of view of…certifiers."

b) **Acceptance/qualification testing** – "Acceptance testing checks the system behaviour against the customer's requirements."

c) **Conformance testing/ Functional testing/ Correctness testing** – "Conformance testing is aimed at verifying whether the observed behaviour of the tested system conforms to its specification."

d) **Performance testing** – "This is specifically aimed at verifying that the system meets the specified performance requirements, e.g. capacity and response time."

e) **Usability testing** – "It evaluates the ease of using and learning the system (and system user documentation) by end users, as well as the effectiveness of system functioning in supporting user tasks…"

f) **Operational profile** – "In testing for reliability evaluation, the test environment must reproduce as closely as possible the product use in operation."

g) **Testing of real-time systems** – "…for some kinds of applications some additional know-how is required for test derivation [of]…"specialised" testing fields…"

These testing topics effectively demonstrate that the [JAP, 2004] Qualification and Certification function can in fact be considered as stringent Software Testing and therefore part of the SM process activity. This fact established, and as the software functional area is only concerned with SM, it would seem more appropriate to refer

to it as the SM Function (with corresponding classic waterfall life-cycle activities) when considered in the context of a SE Activity Model.

Software Conclusions

In conclusion, we can state that SM is fundamentally different from hardware modification. Likewise, it is also true that SM always results in the production of a new software product. As a result one can attribute SM to both development and post-developmental processes, as Software Development (SD) inevitably results in the production of a new software product. I have chosen to refer to the software area as SM as this process is effectively its prime concern. For the purpose of identifying fundamental SM activities, I have utilised the classic waterfall life-cycle, asserted by [SWEBOK, 2001], whilst acknowledging that the stringency of the testing activity will vary according to when, where and why it is being conducted. However, SM is only one element of a comprehensive SE Activity Model. We now have to look further than this area, in particular the controlling processes required to aid the production of software. For this reason, we shall now examine the Supporting Activities functional area.

3.2 The Supporting Activities Function

Referring back to the KA in Table 3.1, we see that it is not only a list of five [SWEBOK, 2001] acknowledged SE life-cycle activities (identified by ^{SM}) but five Supporting Activities (SA) as well (identified by ^{SA}). In order to examine each of the SAs, I have ordered them as follows:

a) Software Quality

b) Software Configuration Management

c) Software Engineering Management, Software Engineering Process and Software Engineering Tools and Methods

It should be borne in mind that, although they appear to be software-specific the processes they describe may well be applicable to areas other than software products (for example, the configuration management of software-specific training). My reasons for grouping Software Engineering Management, Software Engineering Process and Software Engineering Tools and Methods will become evident in the forthcoming discussions.

Software Quality

Taking Software Quality as an activity, it would initially appear that it would be more appropriately referred to as Software Quality Assurance (SQA) as this is a *"…process* [that] *provides assurance that the software products and processes in*

the project life-cycle conform to their specified requirements" [SWEBOK, 2001], in effect a management function [Stebbing, 1986]. However, one can ask whether it would be better referred to purely as Quality Assurance (QA), as it is arguably a global activity within a SE model and not necessarily confined solely to the qualitative and quantitative assessment of software.

Indeed, approaching this from another perspective, one could also question the requirement for a software development background in order to be able to fulfil the requirements of this activity. After all, 3rd party ISO 9001:2000 process audits of the Quality Management System (QMS) are often conducted by agencies with little or no software background, essentially assessing the system with regard to the achievement of customer satisfaction. Conversely, this argument does not hold true, however, for TickIT certified organisations where ISO 9001:2000 is applied to the software QMS. In this case audits must only be conducted by personnel with *"…at least four years' software development experience"* [TickIT, 2001]. One could then argue, for instance, that auditing would be better facilitated if the personnel involved were familiar with the system's operational domain.

This dichotomy of opinion does indeed pose a problem when we consider the secondary aim of this dissertation to be the enabling of the identification and definition of RAF SST manning requirements (especially considering the hybrid trade group of Communications Information Systems). However, if we concern ourselves purely with a QMS, then little or no software development background is necessary. If, as is the case here, we are concerned with a software QMS, then a software development background is essential.

The requirement for a software background to, in effect, manage a software QMS, therefore, does raise the question of scope – is software testing essentially QA in a different guise? One could answer yes, if the view is taken that one of the auditor's tasks is to assess whether or not the customer's requirements have been realised in the delivered product, in the same way that the tester checks that the product meets the requirements behind it. However, as Figure 3.3 demonstrates, QA is not simply just about checking product conformity; the management of resources and analysis & improvement of the QMS also form a significant part of the quality auditor's responsibilities.

It is my view, therefore, that the quality activity is fundamentally relevant to the whole software development process. As such, it should be treated as a separate activity and be referred to as SQA.

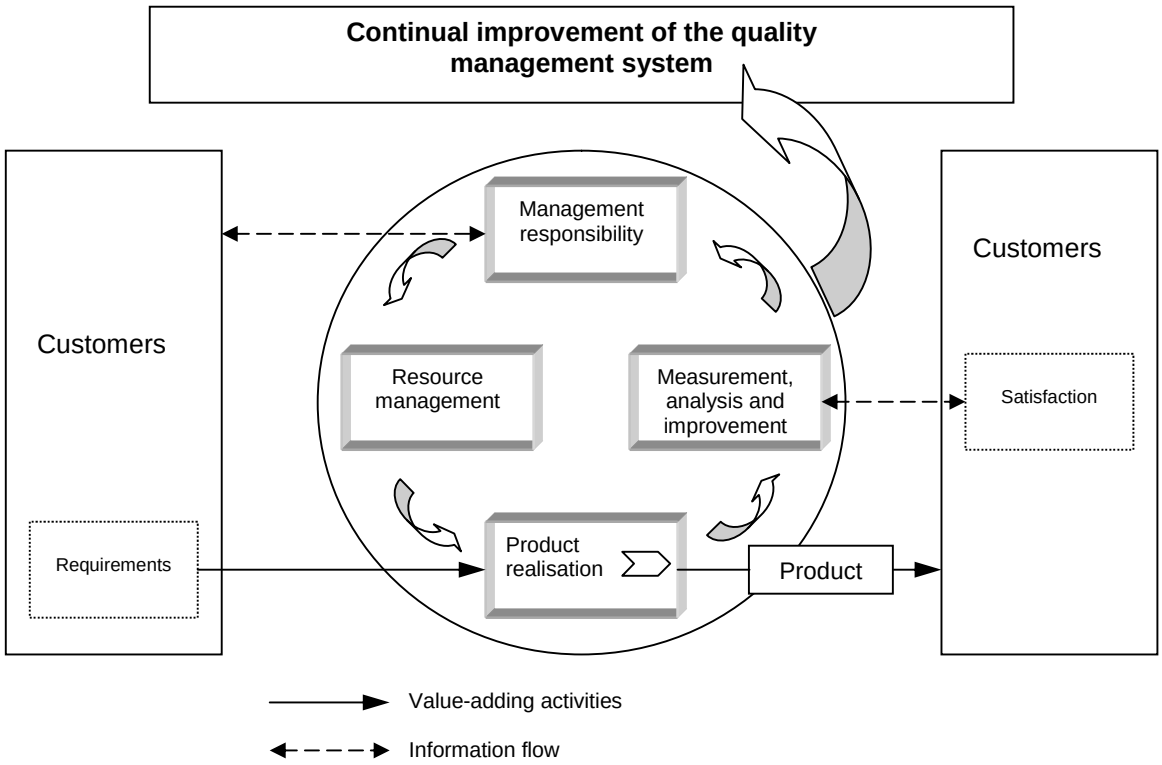


Figure 3.3 – Model of a Process-Based QMS [ISO 9001, 2000]

Software Configuration Management

Looking then to Software Configuration Management, we can refer to the [JAP, 2004] change management requirement for *"…a single* [Software Configuration Management Board] *SCMB* [to deal] *with user-initiated software change requests…*[reconciling] *these demands for change with the overall goals, constraints and strategies imposed* [upon] *it…"*. In this context we can simply subsume software configuration management into the Change Management activity (from Figure 3.1), separately identifying it as a sub-task where appropriate.

Software Engineering Management, Software Engineering Process and Software Engineering Tools and Methods

These three activities do require further examination. Software Engineering Tools and Methods is defined in [SWEBOK, 2001] as including:

> *Software Development Environments – computer-based tools that are intended to assist the software development process,* and,

> *Development Methods that impose structure on the software development activity with the goal of making the activity systematic and ultimately more likely to be successful.*

However, can Software Engineering Tools and Methods be referred to as an activity? In the above description it is more like a description of "things" that are at the software developer's disposal. That said, however, these "things" still need to be managed (procured, installed, supported, monitored, controlled), so it is my view that it would be more appropriate to introduce some form of management to this activity, referring to it as Software Engineering Tools and Methods Management.

Software Engineering Process is, however, slightly different in that [SWEBOK, 2001] approaches it from the context of two levels. The first of these levels (Level 1) encompasses the technical and managerial activities within the software engineering process that are performed during software acquisition, development, maintenance and retirement; activities that can essentially be referred to as holistic life-cycle management. Level 2 (termed *software process engineering*) is described as being at the meta-level and concerned with the definition, implementation, measurement, change and improvement of the software engineering process itself.

These two levels present us with a slight problem, however. It is clear that both levels, whilst significant in their own right, do in fact belong in different areas when considered in the context of producing a SE Activity Model. Level 1 is essentially a managerial activity that is more aligned to being an activity of Software Engineering Management; likewise, Level 2 is concerned with ensuring the quality of the software engineering process and it is my view that it should be aligned with the SQA activity.

Have we then covered all Supporting Activities? From the point of view of [SWEBOK, 2001], yes. However, we have not yet examined the RAF Software Support Team Functional Areas (Figure 3.1), and in particular the activity of Data Support. [JAP, 2004] has this definition of Data Support:

> *"Mission data will be created and modified in order to support aircraft and/or equipment software when operating in changing operational, training or trials environment. Users or relevant external agencies may generate mission data requests. The organization responsible for implementing the requirements is to ensure that suitable testing of the data has been carried our prior to its issue. It is assumed that mission data does not affect the functionality of the system and therefore will not have to undergo qualification and certification."*

Let us then examine this definition in greater detail. The first thing to note is that we are specifically referring to "data" as opposed to "software". Indeed, there is no reference to software within the definition itself and this presents us with a problem;

this dissertation explicitly concerns itself with SE and this is reflected in the adopted definition from Blanchard. The only definition of software thus far encountered is [Oxford, 2003]:

> *"the programs and other operating information used by a computer"*

and it has been established that this definition is a little too unspecific for the task at hand. There are other definitions for software, notably:

> *"Software consists of programs, documentation and operating procedures by which computers can be made useful to man"* [McDermid, 1994]

and:

> *"Software is all things which are not hardware in the system"* [Gilb, 1990]

but both of these definitions can hardly be described as being clear-cut. A better definition:

> *"Computer programs, procedures and associated documentation and data pertaining to the operation of a computer system"* [00-55, 1997]

serves to remove the apparent distinction between "software" and "data", effectively stating that data is but an element that constitutes software. However, in order to be able to do this we must understand the relevant differences between the two. As far as this dissertation is concerned, software will be considered as consisting of:

> *Functional programs and procedures (e.g. applications and/or operating systems),*
>
> *Associated data required to be manipulated by these functional programs and procedures (e.g. weapon ballistic look-up tables, dynamic link libraries, electronic warfare parametric look-up tables),* and
>
> *Associated documentation (e.g. descriptive output from each stage of the life-cycle).*

Although the Software Support Model (Figure 2.2) acknowledges two-way data traffic between a software platform, software operations and a data support function, for simplicity, we will specifically deal with data created by a support system for use by a functional program in an operational capacity, and not data created by way of a functional program executing (e.g. engine health and usage data or a Word document).

It is reasonable, therefore, to assert that this data will require a holistic life-cycle management process such as the classic waterfall life-cycle identified for SM. In effect, this means that we can treat the life-cycle processes performed on functional programs and those performed on associated data in the same manner. This assumption is further borne out by the [JAP, 2004] reference to *"Users or relevant external agencies may generate mission data requests"* and *"The organization responsible for implementing the requirements is to ensure that suitable testing of the data has been carried our prior to its issue"*, effectively implying that data modification will be initiated by some type of requirement and will require some form of testing after implementation. The Design phase may be nothing more than a simple apportionment of the requirements into personnel work-orders, or a more formal system representation. Contextually, though, we are stating that data, its creation and modification will follow a life-cycle process similar to SM, a representation of which is at Figure 3.4.

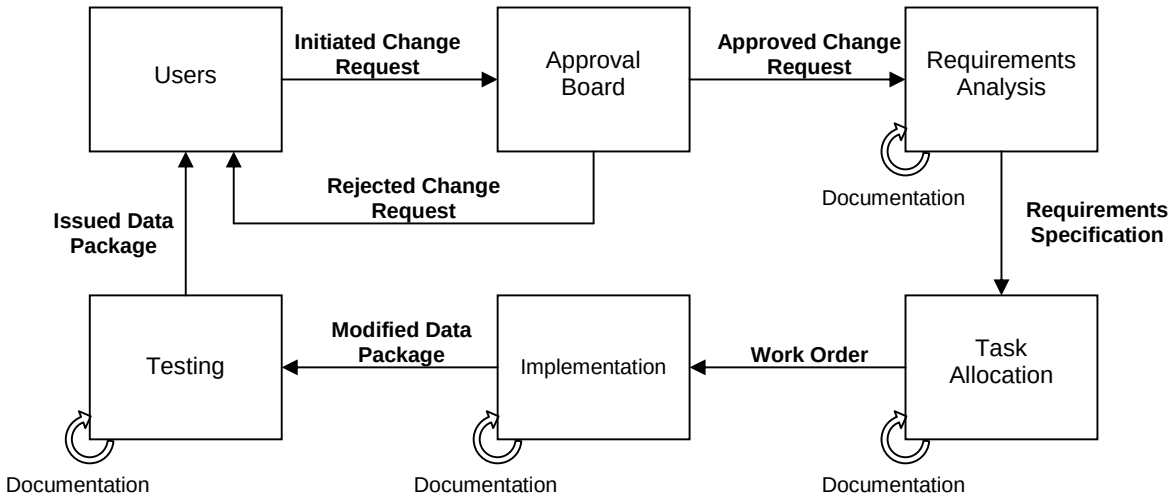


Figure 3.4 – Proposed Data Life-Cycle

Finally, the [JAP, 2004] assumption that *"…mission data* [will] *not affect the functionality of the system and therefore will not have to undergo qualification and certification"* is an example of a general definition that is unable to cover all eventualities of usage. For the purposes of this dissertation we will acknowledge the limitations of this definition and work within it, but it should be accepted that there may well be occasions where data modification may alter the actions undertaken by functional programs and procedures (e.g. electronic warfare data), or have sufficient impact on the functioning of the system as to warrant further qualification and certification activities.

Supporting Activities Conclusions

In conclusion, we have identified the following points with regard to SA. Software quality is fundamentally relevant to the whole software development process and should therefore be referred to as SQA, being undertaken by personnel with a software development- but not necessarily an operational domain-background. Software configuration management should fall within the remit of a global Change Management activity, and Software Engineering Tools and Methods requires some form of management process associated with it. SE process should be spread across both SE Management and SQA, and data support is to be viewed as a sub-activity of SM.

Effectively then, the Supporting Activities function comprises four separate global SE activities (SQA, Change Management, SE Tools and Methods Management and SE Management) that interact with, and influence each of, the other SE functions. However, so far we have only concentrated on the controlled production of software. We now need to take a wider view, essentially examining the system associated with the software itself.

3.3 The System Function

Let us then again revisit Figure 3.1 (RAF Software Support Team Functional Areas), concentrating this time on the system functional area. The RAFs Software Support Model has already given us the activity of Query Evaluation, and I initially designated this as a System-related activity. But how did I arrive at this decision? The provision of a query helpdesk effectively has a two-fold function. Its primary function is to resolve user-related problems with the supported software (and in some cases its associated system), but its secondary role is the gathering and categorization of new system requirements. In effect, the very existence of the Query Evaluation activity predicates the requirement for SysE activities, a notion borne out by the fact that RAF SSTs have to deal not only with software-related issues but system-related ones as well. Therefore, just as Blanchard acknowledges that:

> *"System engineers can no longer make hardware design decisions without considering the software implications. At the same time, software engineers must develop their software in the context of the requirements for the overall system and not as an independent entity"* [Blanchard, 1998]

we find a dimension resident within the SE Activity Model that needs to be considered when establishing its boundaries. This postulation is further substantiated by the [SWEBOK, 2001] identification of seven related disciplines that have a common intersection with its interpretation of SE:

Systems Engineering
Cognitive Sciences and Human Factors
Computer Engineering
Computer Science
Management and Management Science
Mathematics
Project Management

Table 3.2 – Related Disciplines [SWEBOK, 2001]

Table 3.2 identifies SysE as being one of several related disciplines that require consideration. This is because one of the objectives of the SWEBOK was to *"clarify the place - and set the boundary - of software engineering with respect to other disciplines…"* [SWEBOK, 2001]. However, although the SWEBOK maintains that SysE should remain a separate concern from SE, my proposition is that, where RAF SST functions are concerned, it should not be.

To illustrate this further, a basic system life-cycle is described in Figure 3.5 showing two possible environments (development and operation) that a product may be in at any one time. This model serves to draw attention to the product's life outside of the project environment, essentially linking the operation and development phases together in a SysE life-cycle context. Therefore, one can take the view that SysE is effectively a key SE enabling function, rather than a separate concern, when considering the establishment of a SST SE Activity Model.

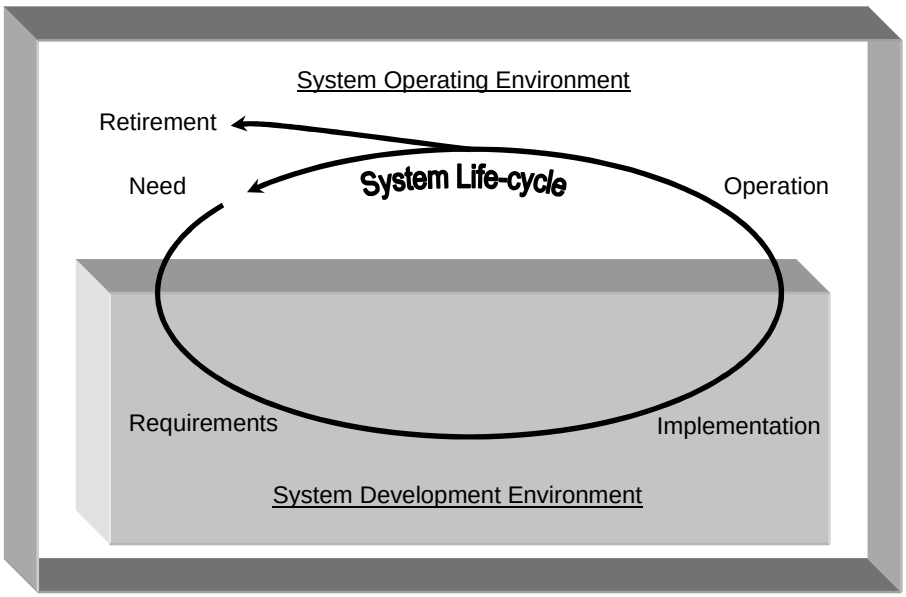


Figure 3.5 – System Life-Cycle

The six other SWEBOK disciplines (Cognitive Sciences and Human Factors, Computer Engineering, Computer Science, Management and Management Science, Mathematics and Project Management) do either fall outside the current scope of RAF SSTs and hence any SE competency evaluation, or are only normally encountered in RAF SSTs at the most general level (project management, for example). However it is acknowledged that, on occasion, personnel with specialist knowledge gained through personal study (e.g. a degree in mathematics) are posted into a SST. These personally developed skills can subsequently prove invaluable to a team[5]; however, they do not currently form part of the requisite skills deemed necessary in order to work within a SST[6]. Consequently, although the SWEBOK acknowledges that *"software engineers should of course know material from these fields"* they will not be explored in any great detail here but will be recognized as specialised fields outside that *"what is viewed as specific to software engineering"* [SWEBOK, 2001].

However, it is not merely enough to just identify SysE as a function resident within a SE model. A more detailed level of functional abstraction is required before any competency measurements can be made. [Blanchard, 1998] describes SysE as *"…a top-down, integrated, life-cycle approach to system design and development…"*. In particular, he has defined a system engineering process inherent within the overall system life-cycle. Critically, it is the iterative progression of assessment and validation or *feedback loop* which is integral to this system engineering process. Figure 3.6 (System Engineering Process) shows the basic activities that Blanchard states should be undertaken during SysE. He also states that these activities represent a process that should be followed each time there is a newly identified requirement, tailoring as necessary to suit the needs of the system. This is an important distinction to note as, essentially, it means that his SysE process can be applied both to development and post-development activities.

Example occasions of when a comprehensive process such as this may or may not need to be followed in full are:

[5] Case in point: HSMUs OFP release "HSMU5" comprised 6533 manhours and 7049 SLOC over the period July '93 to Feb '94, representing a higher SLOC/manhour ratio than that normally achieved.

[6] The training package provided for personnel employed upon Harrier GR7 software modification duties consists of Basic & Advanced Structured Programming, Real Time Applications and Verification & Validation courses conducted at the Defence IT Management Centre (DITMC), an Operational Flight Program (OFP) overview course conducted at RAF Wittering and AYK14, OFP Programmer and Software Testing courses conducted at Harrier Software Maintenance Unit (HSMU) [2237/01, 2003].

a) Full process - new developments (Eurofighter Typhoon[7], Joint Strike Fighter, Future Offensive Air System) or long term changes (planned version releases),

b) Tailored process - maintenance of legacy systems (Harrier GR7, Tornado GR4[8]), and,

c) Abbreviated process - fast track changes (Urgent Operational Requirement[9]).

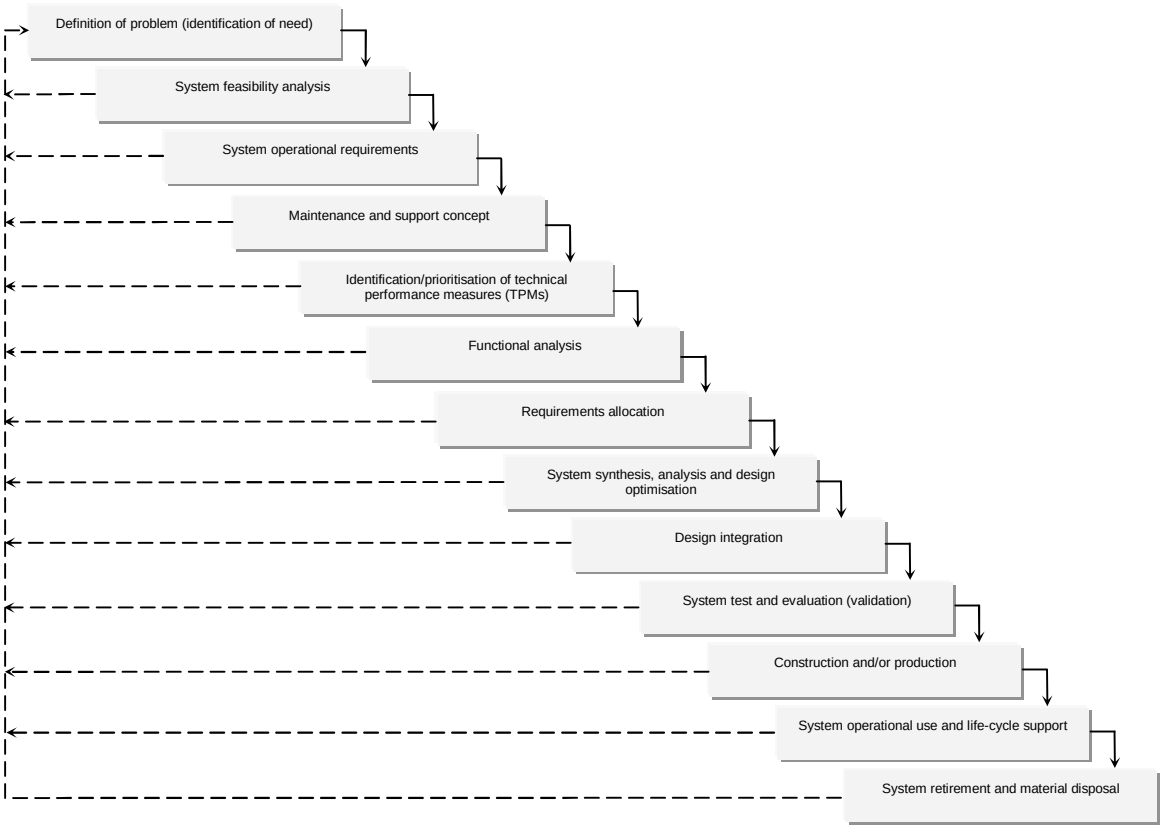


Figure 3.6 – System Engineering Process [Blanchard, 1998]

Tailoring of the SysE process in these cases will need to be performed by personnel with adequate experience, competency and skill. In turn, this competence will need to measured and tracked. In order to achieve this tailoring process, an examination

[7] Although Eurofighter Typhoon is considered by most to be rather "long-in-the-developmental-tooth", it can technically be classed as a new development as it is still yet to enter Service.
[8] Both the Harrier GR7 and Tornado GR4 employ assembler-based Operational Flight Programs.
[9] Military operations sometimes necessitate the support organisation to shortcut the "normal" development process in order to implement an Urgent Operational Requirement.

of each of Blanchard's SysE process activities is first required in order to identify relevant attributes.

Definition of Problem (Identification of Need)

In the development phase this activity concerns itself with the identification and specification of the primary and secondary functions of the system. It involves the developer, system customer and system user (if different) with the intention of identifying "WHAT" is required rather than "HOW" it is to be implemented. However, in the post-development phase this activity then concerns itself with the provision of assistance to Users. In SSTs this assistance is achieved through the provision of helpdesk facilities and Query Evaluation where a mutual understanding of "WHAT" is required to be delivered (information or modified product) is established.

System Feasibility Analysis

"The most rigorous functions are selected as the basis for defining system-level design requirements. All various possible design approaches are identified, evaluated and a preferred approach recommended." Blanchard's use of the word rigorous is somewhat undefined, as it is unclear what actually defines a rigorous function. However, this activity would require the developer's expertise and application domain knowledge in order to design and evaluate potential solutions.

System Operational Requirements

The what, when, where and how of the operational concept is defined, including the following information:

a) Operational distribution or deployment – where the system is to be utilised, especially critical when considering dispersed military operations (e.g. home and foreign airfields, sea-based operations).

b) Mission profile or scenario – what mission objective (or objectives) is the system to accomplish and how is it to accomplish it. This could be as simple as "find target and explode" (missile), or "conduct multiple role operations in hostile and friendly airspace" (fighter aircraft).

c) Performance and related parameters – critical system performance parameters necessary to accomplish the mission are defined, working closely within the system's operating design envelope. Examples are number of weapons able to be released in a particular time-frame (essentially defining time required to be spent in a hostile environment), or real-time algorithmic computation speed (possibly defining navigational data accuracy).

d) Utilisation requirements – anticipated usage of the system by its operator and the environment. This is usually easy to define through task analysis as the requirement is typically produced in response to some operational need.

e) Effectiveness requirements – efficiency parameters such as cost, operational availability, dependability, reliability, failure rate, readiness rate, maintenance downtime, facility utilisation and personnel competency and skill levels are identified and defined. Whilst this dissertation will ultimately provide a competency and skill reference framework, the additional application of Logistics Support Analysis (LSA) will assist in this process as supportability factors are established, defined and, critically, allowed to influence the design of the system.

f) Operational life-cycle – the anticipated time that the system will be in operational use by the User. This may be based on comparative analysis of existing systems, or may require an educated guess, in which case the system design should be flexible enough to cope with changing demand.

g) Environment – definition of the system's operating environment. This will need to be established for new system operations, and will then be used to provide maintenance boundaries in post-development activities.

Maintenance and Support Concept

Again, through the application of LSA, the following information is considered for establishing the supportability requirements in system design:

a) Levels of maintenance – organisational (1st and 2nd line), intermediate (3rd line) or supplier maintenance (4th line) levels[10] are defined, assessed for suitability and recommended depending on the nature and mission of the system.

b) Repair policies – Terms usually associated with hardware (non repairable, partially repairable or fully repairable) have limited application when considering software. However, one could consider bespoke software as being repairable (or modifiable) in this context by SSTs. Defining examples of non-repairable software may be Commercial-Off The Shelf (COTS) or one of its variants (e.g. Military-, Government-), but a true analogy may not be achievable between non-reparable hardware (i.e. uneconomic to repair) and software (i.e. unable to repair) due to their inherent dissimilarities.

[10] 1st, 2nd, 3rd and 4th line levels of maintenance are military descriptors and are discussed in detail under the Users heading of section 3.4.

c) Organisational responsibilities – the definition of who is ultimately responsible for the accomplishment of maintenance. There may be more than one organisation involved in the maintenance process, each performing separate assigned tasks. The application of LSA will help define these responsibilities, but ultimately, these separate activities will need to be controlled by one responsible body.

d) Maintenance support elements – *"criteria pertaining to supply support, test and support equipment, personnel and training, transportation and handling equipment, facilities, data and computer resources will need to be defined and maintained"*. The *"criteria"* that Blanchard refers to is not clear; however, the application of LSA will help to identify and define these elements in the development phase. In the post-development phase, these elements will then require managing, adjusting as necessary in order to adapt to changing system requirements.

e) Effectiveness requirements – factors associated with the support capability, such as turn round time, skill levels, error rates, training rates and errors per mission segment or line of code, Change Traffic (i.e. the rate at which SM activities are required) and Expansion Capability (i.e. available memory, processor performance, mass storage capacity and input/output bandwidth) [00-60-3, 1998] will again need to be defined. Although these elements should form a support system requirements set preferably defined by the customer, more often than not the customer will not even know that they require a support system, let alone what performance they want from it. Indeed, the customer may not even appreciate just how these support factors may ultimately influence the design of the system itself. The support system is an important area that will require the skill, judgement and the application of LSA on the part of the developer and customer to define, create and manage.

f) Environment – definition of the environment as it pertains to maintenance and support. Again, this can be an important factor that is often overlooked. Operating locations and maintenance units may be widely dispersed and this may well have a significant impact on supportability especially if time is a critical mission availability factor.

Identification and Prioritisation of Technical Performance Measures (TPMs)

During the development of system operational requirements and maintenance concept, it is necessary to assign each a measure of their relative degree of importance. In real-time avionic systems, for example, processing speed may well be considered to have a higher degree of importance than system capacity, yet a (relatively) lower level of importance when considered against reliability. In order

to measure whether or not these requirements have been met it is necessary to identify a number of measurable goals or TPMs for each. This is achieved (normally by a committee composed of designers and customer representatives) by identifying functional attributes of each requirement and then weighting them based on degree of importance. This then ensures that each functional attribute is assigned an appropriate amount of emphasis in the design process, and their presence in the final system can be subsequently determined. This activity will effectively provide customers with oversight and reassurance that their needs have been met in the delivered product. Although primarily a development activity, there is no reason why this cannot be applied to post-development activities, essentially serving to provide assurance of the presence of critical parameters in a modified product.

Functional Analysis

This is the iterative top-level process of breaking requirements down from the system level, to the sub-system, and as far down the hierarchical structure as necessary to identify input design criteria and/or constraints for the various elements of the system. This process can be achieved through the use of functional flow block diagrams which arrange the system into a basic structural organisation with functional interfaces. The main aim, however, is to identify "WHAT" is required as opposed to "HOW" it is to be achieved, that is, this is still a process that is as distinct from the design stage as the problem definition stage is.

Requirements Allocation

After the functional analysis has been conducted, the next stage is to break the system down into components through partitioning. This involves the decomposition of the system into subsystems and lower-level elements, identifying, associating and grouping closely-related functions into packages. At this stage it is fair to say that this is akin to system decomposition from an object oriented perspective, as one of Blanchard's aims is to design the *"…system packages* [to be] *as independent as possible with a minimum of interaction effects with other packages."* This will ultimately help to achieve the *"…design objective* [of being] *able to remove and replace a given package without having to remove and replace other packages, or requiring an extensive amount of alignment and adjustment in the process"*, thus aiding future supportability. In effect, this can be considered to be a high level design activity.

System Synthesis, Analysis and Design Optimisation

"Synthesis refers to the combining and structuring of components in such a way as to represent a feasible system configuration." Effectively, this is the detailed design phase of the system engineering process and is used to define "HOW" things will be achieved in response to "WHAT" is required, usually leading to the definition of several possible alternative design approaches (in a similar fashion to system

feasibility analysis). These alternatives are then subject to further analysis, evaluation, refinement and optimisation as follows:

a) Definition of analysis goals – the identification of all possible design candidates in order to eliminate the unfeasible. This may be practically impossible to achieve in a large system, and even more difficult to prove that it has been achieved.

b) Selection and weighting of evaluation parameters – evaluation criteria are selected, weighted in terms of priority and are tailored to the system in a similar fashion to the production of TPMs.

c) Identification of data needs – data is required in order to evaluate operational requirements, the maintenance concept, major design features, production and/or construction plans, and anticipated system utilisation and product support requirements. In the early stages of system development, the nature and availability of this data may be unknown and hard to come by. It is therefore down to the analyst to base projections on past experience, intuition and comparisons with similar systems. The process of conducting LSA when determining the optimum support concept for a particular system is a tool available to the analyst that includes, for example, the identification of Software Support Significant Items (SSSIs), Urgent Operational Requirement (UOR) support items and Candidate Items (CI) for detailed analysis; the identification of Support Tasks and the specification and gathering of a data set for SSSIs; and a Comparative Analysis of a similar system in order to identify existent problems or, conversely, good practices [Gill, 2004].

d) Identification of evaluation techniques – assessing the problem itself and identifying the available tools that can possibly be used in attacking the problem are necessary prerequisites to the selection of a design model.

e) Selection and/or the development of a model – a model, as a tool used in problem solving aids in the development of a simplified representation of the real world as it applies to the problem being solved. An important objective is to select and/or develop a tool that will help to evaluate the overall system configuration, as well as the interrelations of its various components. The future supportability of this tool is an important factor that again merits consideration from the customer's perspective, as this may have an impact on long-term support issues.

f) Generation of data and model application – this stage involves testing the model in order to assess whether or not it is responsive to the analysis requirement.

g) Evaluation of design alternatives – each of the alternatives being considered is then evaluated using the techniques (identified in [d] above) and the selected model.

h) Accomplishment of a sensitivity analysis – key system parameters that are "unknowns", or for which data is poor or simply unavailable may severely impact upon overall system performance. These parameters will need to be subjected to a sensitivity analysis in order to determine whether or not small variations in them have a subsequent adverse effect on the system functionality. For example, small increases in the number of user requests for change may adversely impact upon the efficiency of the support system.

i) Identification of risk and uncertainty – the process of design evaluation leads to decisions having a significant impact on the future. The aspects of risk and uncertainty must be identified and integrated into the program risk management plan.

j) Recommendation of preferred approach – this is the final step in the evaluation process which recommends the preferred design alternative. In order to ensure traceability the results of the analysis process should be fully documented and made available to all project design personnel.

Design Integration

This is concerned with the creation of a design team formed to perform system-level design functions. As system development progresses, appropriate design specialists are added to the team. The selection of design specialists is highly dependant upon the requirements developed through the functional analysis and allocation process, the size of the system being developed and the competency and skill sets of the personnel involved.

System Test and Evaluation (or Validation)

As the system design progresses, there is a requirement for ongoing measurement and validation activities. In Figure 3.7 Blanchard identifies five categories of testing that are recommended throughout the system life-cycle:

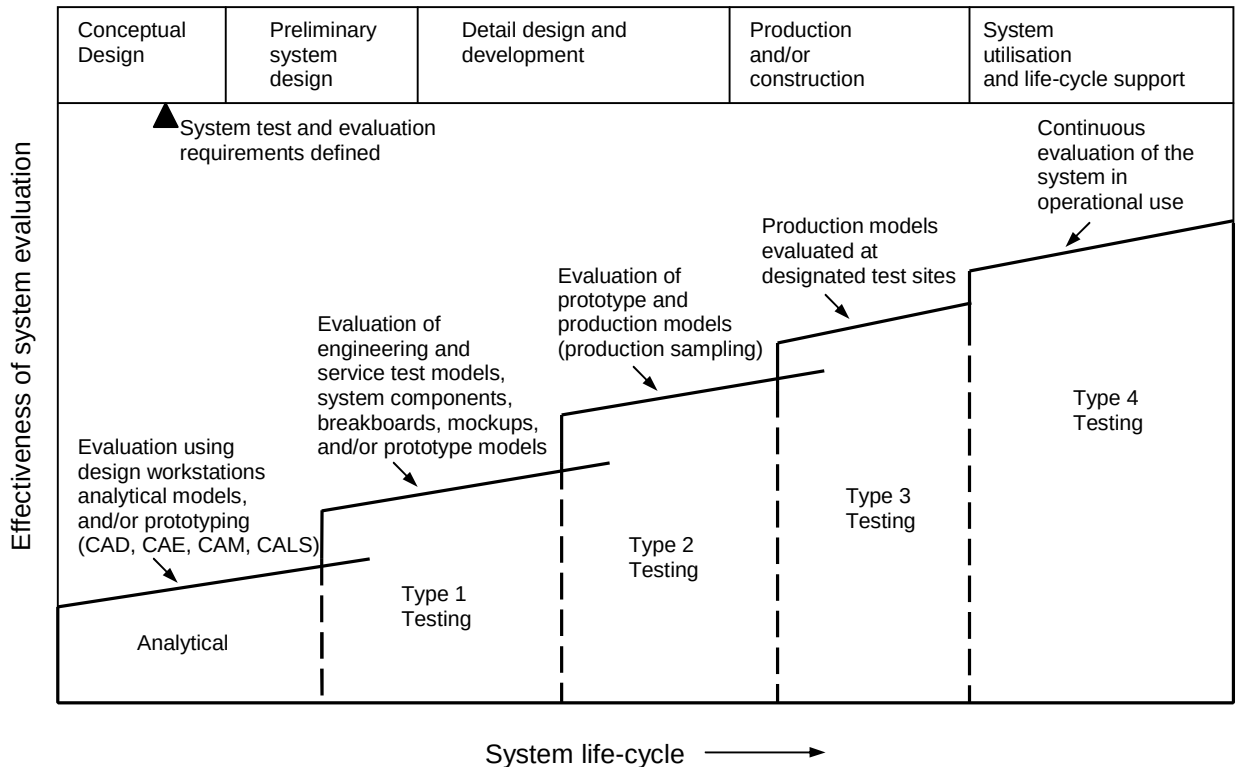


Figure 3.7 – System Evaluation Stages During the Life-Cycle [Blanchard, 1998]

a) Analytical testing refers to early design evaluation conducted using computerised techniques such as Computer Aided Design (CAD), Computer Aided Engineering (CAE), Computer Aided Manufacturing (CAM) and Computer Aided Logistic Support (CALS).

b) Type 1 testing refers to the evaluation of system components in the laboratory using engineering breakboards, bench test models, service test models, rapid prototyping etc.

c) Type 2 testing includes formal tests and demonstrations when preproduction prototypes are available. Checks are carried out in order to ensure that the same level of quality has been maintained throughout the development process.

d) Type 3 testing centres around the completion of formal tests at designated field test sites by user personnel over an extended period of time. Essentially all the elements of the complete system (prime equipment, software, and the elements of support) are operated and evaluated on an integrated basis in order to provide a representative assessment of the fully operational system.

e) Type 4 testing is accomplished at one or more user operational sites by operator and maintenance personnel during the operational use and life-cycle support phase of the system, the purpose of which is to gain further insight of the system in the user environment.

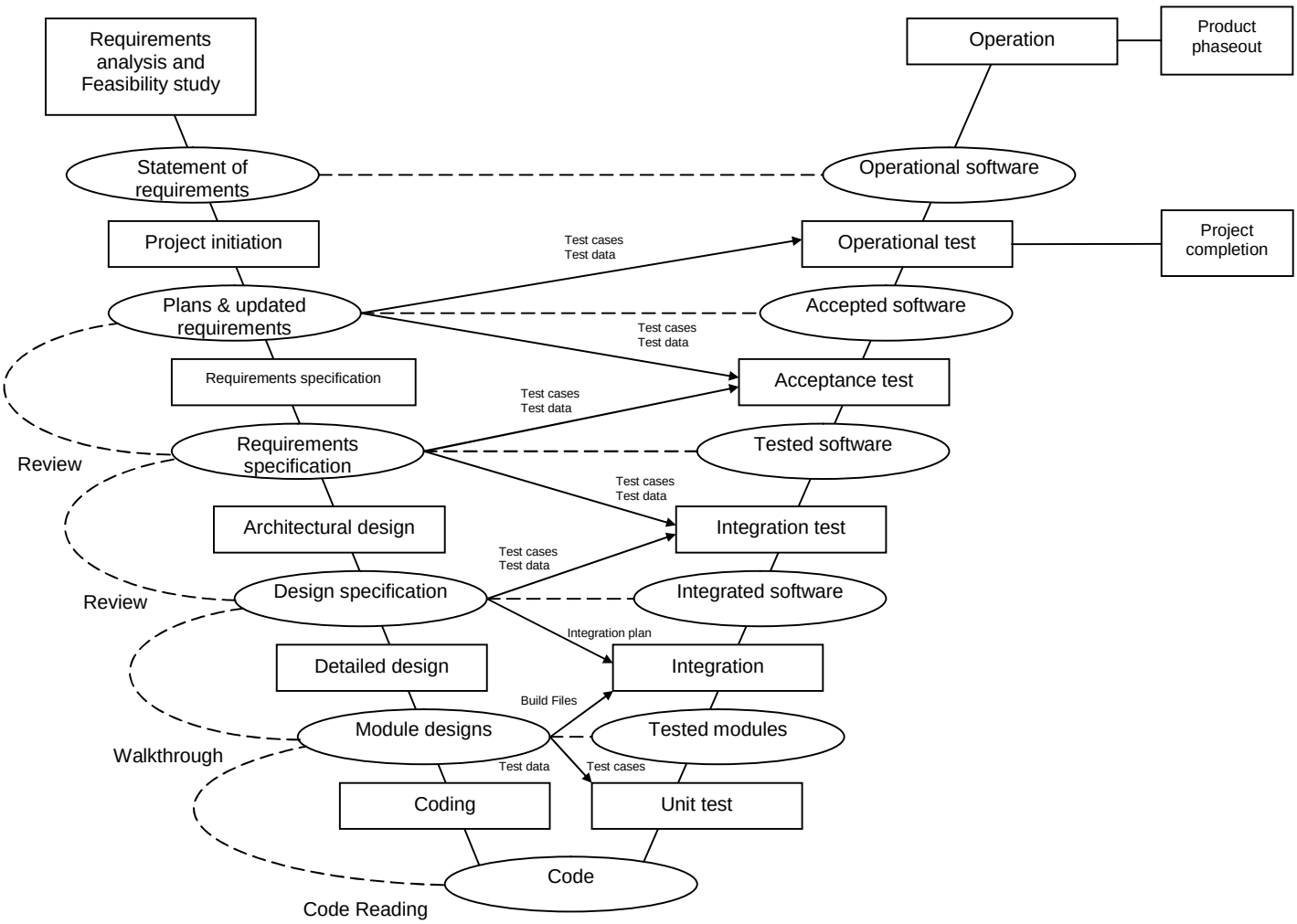


Figure 3.8 – 'V' Model Software Development Life-Cycle [McDermid, 1994]

Taking Blanchard's description of each testing category, we can draw comparisons with the stages within the 'V' software life-cycle model. Figure 3.8 is McDermid's example of the V-diagram software development life-cycle and Table 3.3 is my comparison of the two.

Blanchard's system life-cycle stages	Blanchard's life-cycle evaluation stages	McDermid's V-model software development phases
Conceptual and preliminary system design	Analytical	Reviews, walkthroughs and code reading of requirements specification, architectural design, detailed design, and code
Preliminary system and detailed design and development	Type 1	Unit test
Detailed design and development and production/construction	Type 2	Integration test
Production and/or construction	Type 3	Acceptance test
System utilisation and life-cycle support	Type 4	Operational test

Table 3.3 – Life-Cycle Evaluation Stage Comparison of System and Software Development

From this comparison table, it is possible to map system test and evaluation activities into both the development of new systems and the maintenance of developed ones. This analogy chimes well with Blanchard's identification of the requirement for integrated test planning early on in the system's development, effective system test and evaluation preparation, test and performance evaluation and the incorporation of the necessary change control procedures should a system modification become necessary.

Construction and/or Production

During this phase the system and/or multiple copies of it are produced. What is important to note here is that the high quality characteristics identified and incorporated during the design and testing phases are carried through into the system and not lost through production errors or omissions. This is perhaps the first mention by Blanchard of a link between SQA and SysE, but an important connection nonetheless.

System Operational Use and Life-Cycle Support

Although the system may be designed and produced with the required effectiveness characteristics incorporated, these characteristics need to be maintained within the system throughout its operational life. Blanchard states that this is to be accomplished through the use of good maintenance and support practices, but fails to detail what these practices should be. It is my opinion that personal integrity plays as much a part here as *"good maintenance and support practices"*.

System Retirement and Material Disposal

Blanchard's emphasis here is on the potential environmental impact of system disposal, essentially highlighting the need to design for disposability and the environment. However, from the RAFs perspective, the disposal of systems, especially software-intensive systems, poses more than just environmental concerns. Licensing issues [Gartner, 2003], equipment declassification and effective software removal/destruction are just three areas that may warrant consideration.

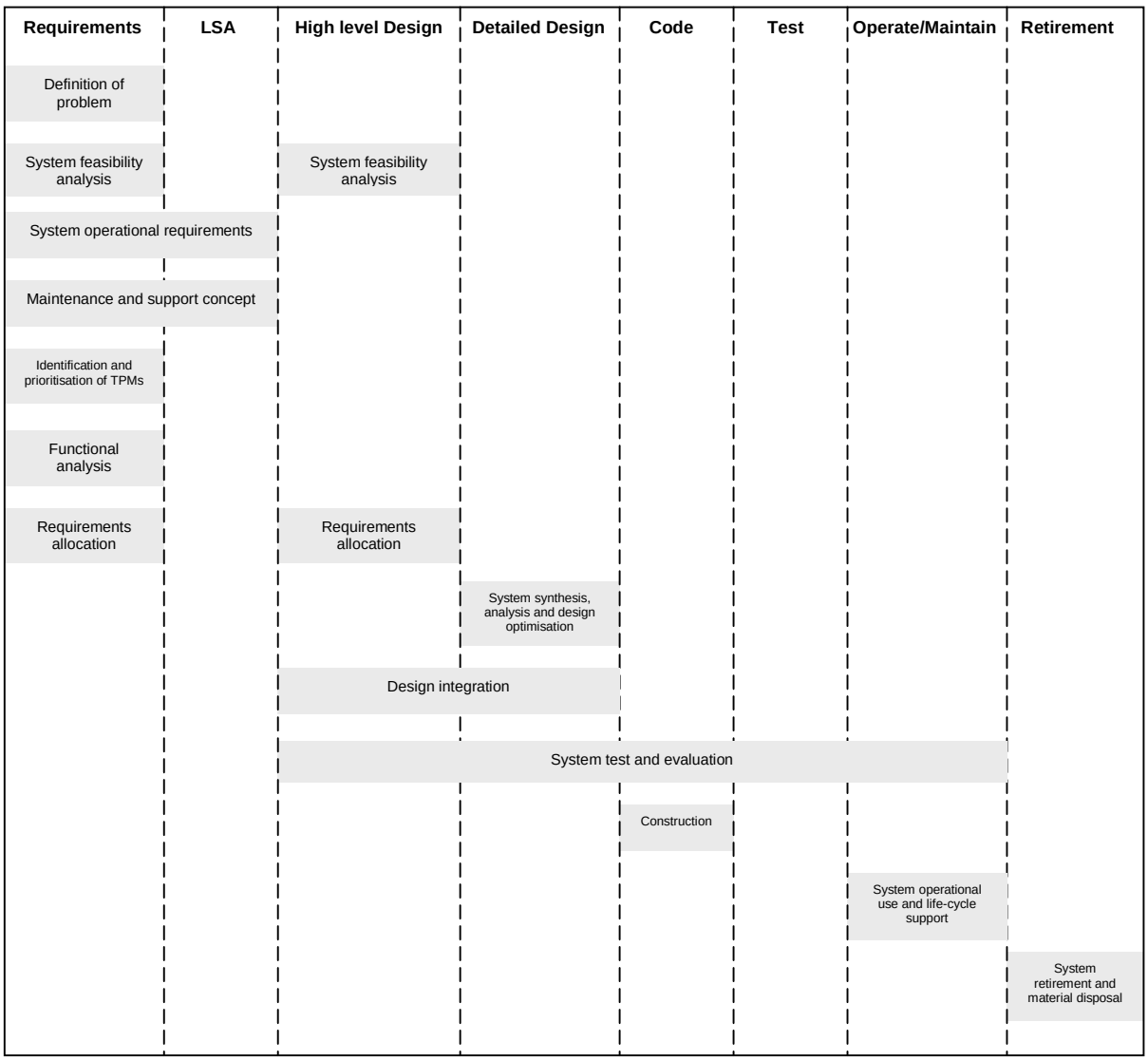


Figure 3.9 – System Activities Related to SM, after [Blanchard, 1998]

System Conclusions

What then can we conclude from this examination of SysE? It is certainly accurate to describe this as an over-arching life-cycle of system-related activities. Indeed, to a large extent, it is much like a very detailed classic waterfall life-cycle with feedback, and Blanchard does appear to have covered classic waterfall life-cycle categories that I have subsequently reproduced in Figure 3.9. His SysE activities span these categories in a way that can be related to, and ultimately drive, the SM function and I have illustrated this in Figure 3.9 by grouping and ordering them.

By way of signifying its importance, I have added the category of LSA to the classic waterfall model categories and shown two of Blanchard's activities (system operational requirements and maintenance and support concept) spanning both it and the requirements category. My reasoning for this is taken from one of the objectives of LSA, that being to *"exert readiness and economic influence on requirements and design"* [Biedenbender, 1993].

The linear representation of the classic waterfall categories is not meant to imply that they occur once-only; it merely serves to illustrate that they exist. Where Blanchard's activities are meant to span more than one classic waterfall category it is clearly represented by the grey bars; however, one should bear in mind that there may well be a requirement to revisit each category more than once during system evolution or maintenance. For this reason I will represent the SysE function in the SE Activity Model as an iterative, repeatable process.

So far I have postulated that SM, SA and SysE are functions resident within a SE Activity Model. However, there is one final, critical function that is often overlooked when identifying the major components of a system - its Supporting Infrastructure.

3.4 The Supporting Infrastructure Function

If we first refer back to the Software Support Model (Figure 2.2) we can see where a partial attempt has been made to identify the following support infrastructure elements (I have chosen to include Blanchard's equivalent descriptors in brackets):

a) Software Platform (or Prime Operating Equipment) - in most cases, where the RAF is concerned, this will be an air vehicle or equipment hardware residing within an air vehicle. Examples of air vehicles are manned and unmanned aircraft and missiles.

b) Released Software (Operating Software) – this will normally be a real time application pertaining to the display of flight, navigation and weapon ballistic data or control of flight surfaces.

c) Users (Operating Personnel) - this refers to any person requiring interaction with the operating software.

There is also the activity of Software Operations which is described as:

> *"The operations function includes the following activities that will normally be carried out by the user of the software:*
>
> *1. Software Operations Support – Actions necessary to load, re-load, replicate, copy, store, distribute, and carry out any handling activity on software, data or firmware.*
>
> *2. Mission preparation – Transfer of data to aircraft and/or equipment for mission and maintenance purposes.*
>
> *3. Post-mission recovery – mission and maintenance data extraction for analysis or sanitization.*
>
> *Users dealing with complex software-driven systems inevitably have questions about their system, discover problems and generate ideas for adaptations, improvements and new functions (enhancements). Once created, a query report, containing all relevant information, will be forwarded to the query evaluation function, along with any post-flight data (if available) that may assist in evaluating a request."* [JAP, 2004]

This description manages to overlap the Supporting Infrastructure and System Engineering functions by linking Software Operations to Query Evaluation in the *"questions"* and *"ideas"* element. It is an overlap that I shall keep as it demonstrates the inevitable interdependency of any SE Activity Model.

The Software Support Model (Figure 2.2) is, however, rather inadequate when compared to Blanchard's comprehensive analysis of logistics, serviceability and supportability engineering. This is not surprising, though, considering the current dearth of integrated logistics support guidance available to any military project manager[11]. We therefore need to revisit Figure 3.1 (RAF Software Support Team Functional Areas) with this in mind in order to identify Supporting Infrastructure elements and activities that are potentially missing.

[11] Logistic Support Analysis (LSA) within the RAF is based upon the US Military Standard, MIL-STD-1388-1A (11 April 1983). It was from this that the UK's Integrated Logistic Support Standard, Def Stan 00-60 was based, of which Part 3 gives guidance on the application of LSA to software. In addition there are a number of other publications, namely AP 100C-71, RAF Integrated Logistic Support Managers' Handbook and AP 100C-70 Integrated Logistic Support in the Royal Air Force (now obsolete), both of which, however, give little or no useful software support information.

However, let us be careful not to completely write-off Defence Standard 00-60 Part 3 [00-60-3, 1998], as this does provide us with useful reference material when compiling support infrastructure requirements. For example, the following statement does provide a useable (if not very well defined) list of supporting infrastructure elements:

> *"The System Support Package component list should include all software operation and support resources, including all elements of the PSE* [Project Support Environment]*, that will be evaluated/tested during software development, software production, operational tests and system logistic demonstrations. The component lists will include:*
>
> *(a) The software development platform.*
> *(b) Test equipment, rigs and data.*
> *(c) Facilities, hardware and software tools.*
> *(d) Documentation and source code.*
> *(e) Licences, design rights, etc.*
> *(f) Replication and transfer equipment.*
> *(g) Processes, procedures and controls (quality assurance, configuration management).*
> *(h) Consumables and parts.*
> *(i) Training and skills.*
> *(j) Personnel requirements."* [00-60-3, 1998]

Comparing the [00-60-3, 1998] list above with Blanchard's list:

> *"Prime Operating Equipment*
> *Operating Software*
> *Operating Personnel*
> *Technical Training*
> *Test and Support Equipment*
> *Maintenance Software*
> *Maintenance Data*
> *Supply Support*
> *Maintenance Facilities*
> *Technical Data*
> *Maintenance Personnel*
> *Transportation and Handling Equipment*
> *Consumable Resources"*

we can immediately see similarities. If we also factor in the Software Support Model, Software Support Team Functional Areas and Basic RAF SST Manning Model (Figures 2.2, 3.1 and 3.2), we can compile the following comprehensive list of Supporting Infrastructure elements that require further consideration:

a) Software Platform or Prime Operating Equipment

b) Released Software or Operating Software

c) Users
- operating personnel
- maintenance personnel
- personnel requirements

d) Software Operations

e) Technical Training
- skills

f) Test and Support Equipment
- rigs
- software development platform
- hardware and software tools
- licences and design rights
- maintenance software
- maintenance data
- documentation and source code

g) Supply Support
- consumables and parts

h) Maintenance Facilities

i) Technical Data
- enabling services
- data communications

j) Transportation and Handling Equipment
- replication and transfer equipment

Processes, procedures and controls (quality assurance and configuration management), however, are excluded from the supporting infrastructure as they have already been identified as SAs.

In the first instance, all of the above support infrastructure elements (activities) will require identifying and managing in order for the SE activity model to work. Although touched upon by Blanchard in the SysE section, it is the management of the support infrastructure that concerns us here. As I have previously outlined the LSA identification process within the SysE activities of system operational requirements and maintenance and support concept, I will use these activities to link

to the function I shall subsequently refer to as Support Infrastructure Management (SIM).

Taking each SIM activity in turn, therefore, I shall examine what its functional relationship is to the SE Activity Model and how it addresses the RAFs software support requirements.

Software Platform (SP) or Prime Operating Equipment (POE)

Management, as far as the SP/POE is concerned, will primarily be configuration-based. This assertion is based on the fact that it may not function correctly, as intended or indeed at all, if the wrong or superseded software is uploaded. It will therefore fall upon the operating and/or maintenance personnel to ensure that not only is the correct software uploaded, but accurate records of this are kept. This is essentially a localised extension of the change management Supporting Activity.

The same can be said of downloaded software and data required for performance monitoring or diagnostic purposes. Records of these activities will need to be kept for inspection with the SP/POE in order to ensure traceability. In order for this to be achieved work instructions, operating procedures and processes will be required to be developed, promulgated, maintained, assessed for quality and utilised by the appropriate personnel. These in themselves will require appropriate configuration management to ensure their usability. Effectively, we are necessitating the requirement for appropriate quality assurance and configuration (or change) management of the SP/POE, global activities already identified as elements of the SA function, but locally applicable here.

Released Software or Operating Software

This will inevitably be the prime output of any SST, as defined by the Software Support Model (Figure 2.2). Actions required here will be the planning, design and product creation in order to be delivered to the customer. Again, as with the SP/POE, this will require assessing for quality and effective configuration management (both globally and locally) as will any supporting documentation or source code issued along with it, in order to ensure that all personnel are using the correct information and products required for their particular working activities.

Users

From a military perspective, there can be a multitude of different types of software user. For example, if we assume that the SP/POE is an aircraft, the various users can be illustrated as per Figure 3.10. As can be seen, there are both operating personnel required to either use the software (aircrew) or upload it for use (1st and 2nd line technicians), and maintenance personnel required to carry out modification activities on the software itself (SSTs at 3rd or 4th line). All of these users will be

subject to management in some way shape or form, examples being local (enabling service e.g. line management), general (organisational e.g. administrative management) and third party (service provider e.g. support management).

In order to illustrate third party management in a SE context, end users (those external to a SST) would require instruction in the use of software releases or the resolving of queries. In order for this to be achieved, a SST would at least need to undertake some form of proactive, interactive self-promotion in order to advertise their existence and ensure that they were effectively utilised.

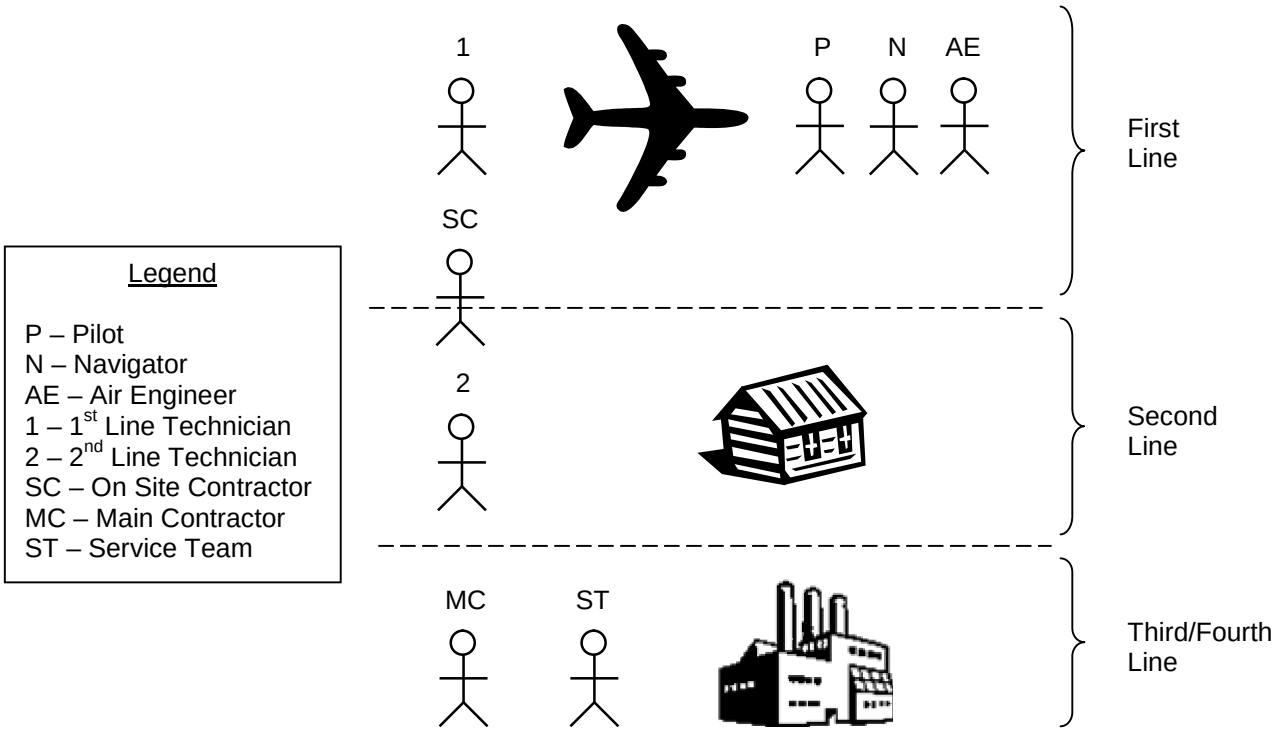


Figure 3.10 – Illustration of Software Users From a Military Perspective

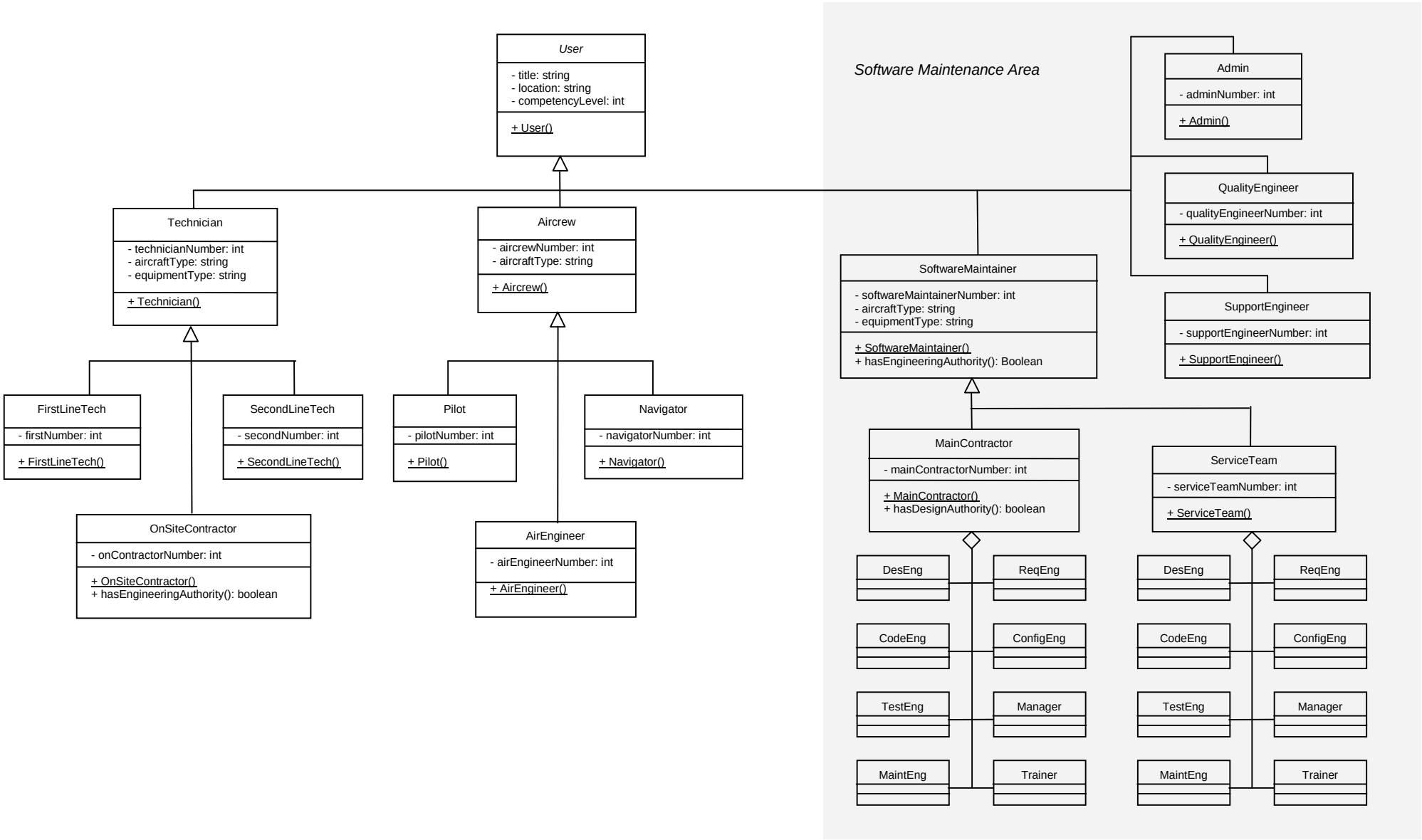


Figure 3.11 – Illustrative User Class Diagram

I have taken the general management of users further by representing Software Users in a class diagram (Figure 3.11). In it, I have identified illustrative attributes required for specific posts acknowledging that, in the maintenance area in particular, certain posts do not necessarily require operating software (application) domain knowledge in order for them to carry out their duties. Examples of this are Quality Engineering, Admin and Support Engineering, which in this case could fall into the realm of the Communications Information Systems trade group. Note that I have stated that no operating software (application) domain knowledge is necessary for the Quality Engineering user. This is a different assertion to the one that I made when discussing Supporting Activities, where I stated that a software development background is essential for this type of User, meaning that an understanding of the software development process is required, as opposed to an understanding of the end application's use.

Looking at the MainContractor and SoftwareMaintainer/OnSiteContractor classes, I have identified the Boolean methods of 'hasDesignAuthority' and 'hasEngineeringAuthority' respectively. I have identified these particular methods as they warrant further consideration. Any User's level of interaction with a particular software product will depend upon the following criteria:

a) The tasks upon which they are employed to perform,

b) Their level of experience and training, and

c) The degree of authority allocated to them.

All of these criteria are relatively easy to define. For example, the job description for a software implementer could typically comprise:

a) A brief description of the application (e.g. real-time avionics, electronic warfare database) and the tasks required to be performed upon it (e.g. corrective, perfective, adaptive, enhancement),

b) Any necessary background required (e.g. years experience in a similar field, knowledge and use of particular tools and techniques), and

c) The degree of responsibility associated with the post.

It may initially appear that I have taken authority and responsibility to mean one and the same. Although not my intention, it does serve to illustrate my next point.

Unfortunately, an endemic culture of responsibility without accountability (such as that perpetuated by the now removed Crown Immunity[12]) can cause conflict when considering the allocation of authority within a SST. RAF SSTs almost exclusively employ non-commissioned ranks (referred to as Other Ranks [OR]) to conduct life-cycle maintenance activities, whilst Commissioned Officers (CO) are employed to manage these activities. This decision is typically cost and usage-driven as the training and salaries of ORs are relatively fundamental compared with that of COs.

However, whilst the position of CO comes with an accepted level of managerial responsibility (typified by the 'officer and a gentleman' ethos), the same cannot be said of ORs. ORs may well have the talent, education, skills and creativity necessary to perform the most complex SE tasks, but in the eyes of the RAF they will always be regarded as producers or supervisors with limited responsibility, and therefore, accountability.

Whilst the RAF is content for ORs to maintain complex and critical systems, accountability for their actions, both technically and legally, ultimately rests in the hands of COs who may not necessarily possess the technical skills to assess their work. And there lies the problem when considering the allocation of Design and Engineering Authority: how can it be done in a way that is fair, effective and robust? What criteria must be satisfied in order for this to be achieved? Is it enough to rely on educational qualifications and skills alone when conducting this allocation process, or is a fundamental change required in the way the RAF conducts its SE activities?

Users of software are not necessarily confined to the SP/POE, and the activities they undertake upon and with that software will depend on where and what they are employed to do. Equally, their responsibility and effectiveness will depend on a number of factors – background, training, motivation, experience, seniority and responsibility. In order to effectively manage Users, therefore, I propose that there is one common attribute that should be monitored amongst them: that of competency level. Quite how this competency level is defined, assessed, recorded, tracked and managed is to be explored in the next section, but it will undoubtedly be a key SIM activity.

Software Operations

As discussed earlier, the [JAP, 2004] lists Software Operations as including actions necessary to load, re-load, replicate, copy, store, distribute, and carry out any handling activity on software, data or firmware; the transfer of data to aircraft and/or equipment for mission and maintenance purposes; and the extraction of mission and maintenance data for analysis or sanitization. At first glance this activity appears to

[12] Up until 1987, the Crown could not be sued in tort if members of the armed forces died or were injured in the course of their duties [CSJ, 2002].

be primarily concerned with 1st and 2nd line activities, although maintenance and analysis activities may well be performed at 3rd/4th line. In any case, for Users to be able to carry out these tasks they will require training and the provision tools and resources, all of which will require to be managed either wholly or in part by a SST.

Technical Training

Technical training is an essential part of any support system, and in the case of SIM it is primarily (though not exclusively) concerned with the management of software maintenance training (the shaded area of Figure 3.11) and the subsequent continuance of personnel skill levels. As has been identified, however, there are different "types" of personnel within this area each fulfilling distinct duties within the SST. Examples with sample training requirements are:

a) Software development personnel. Requirements, Design, Code, Test and Maintenance training that includes Pre-Employment Training (PET), On the Job Training (OJT) and on-going training for appropriate software language(s), the development environment and the SP/POE.

b) Support engineer (proposed Communications Information Systems trade group). Network, test rig software/hardware/emulator, development environment support, administrative tool and quality assurance PET and OJT.

c) Training personnel. Course design and instructor training, in addition to that required for software development personnel. My reasoning here is that for any trainer to be effective, they not only have to be able to instruct but must also have a thorough understanding of what it is they are required to instruct on.

Test and Support Equipment

This is an area that will require the procurement, maintenance and obsolescence management of the equipment and resources required to support operation, transportation, and scheduled and unscheduled maintenance actions associated with the system and/or operating software:

a) Development environment. Maintenance software that includes computer resources, networks and Computer Aided Software Engineering (CASE) tools, and maintenance data such as test results.

b) Support environment. Test rigs that include hardware, software, SP/POE Line Replaceable Items (LRIs), interconnecting cabling and other test equipment, such as Digital Volt Meters (DVMs), signal generators, calibration equipment and oscilloscopes.

c) Training environment. Personal Computers (PCs), projectors, stationery, data, and the production, storage and maintenance of documentation.

d) Administrative environment. PCs (hardware and software), printers, scanners, facsimile, internet, telecommunications and stationery.

e) Consumable resources. Maintenance contracts, licences and design rights.

Supply Support

The management and control of all inventory items, such as spares, parts, consumables, special supplies, warehousing, software, test and support equipment, transportation and handling equipment, training equipment, facilities and material distribution and personnel needed to support prime mission-oriented equipment.

Maintenance Facilities

The management and control of physical plant, personnel accommodation, real estate, furniture, ablutions, rest rooms, lighting, heating, power, air conditioning, fire suppression systems and building security.

Technical Data

The management of system installation and checkout procedures, operating and maintenance instructions, inspection and calibration procedures, modification instructions, drawings and specifications, data communications and enabling services and associated databases that are necessary for the performance of system operation and maintenance functions.

Transportation and Handling Equipment

The management of all special provisions, secure handling, materials, containers, labelling and supplies necessary to support the replication, packaging, preservation, storage, handling and/or transportation of prime mission-oriented equipment and software.

Supporting Infrastructure Conclusions

The Supporting Infrastructure is an often overlooked element of any system. Its elements are chiefly identified through the application of LSA via the SysE system operational requirements and maintenance and support concept activities. Once identified, however, its elements require management leading to the overall function

of SIM. Each activity within SIM is therefore a distinct managerial activity associated with a respective element, and there is therefore the potential for this managerial role to be assigned to the SA of SE Management.

This may or may not be possible, however, dependant on the level and extent of managerial authority associated with a particular post. For example, I would not expect a SST SE managerial post to be responsible for the direct management of all maintenance facilities. Equally, the same post would not be expected to be responsible for the general management of all Users. I would, however, expect that post to at least *liaise* with the managers' responsible for the elements identified, in order to achieve SST synergy within the context of the RAF. In particular the User competency level should be defined, identified and tracked as this is especially relevant when considering the allocation of Design and/or Engineering Authority status.

3.5 Software Engineering Activity Model

It is clear that, just as a relationship exists between the [SWEBOK, 2001] related disciplines (see Figure 3.12), there is also a distinct relationship between the four SE functional areas (see Figure 3.13). The relationships documented between each SE function will provide the basis from which to create a SST C&S assessment framework. Where activities are linked between functions (e.g. SysE Functional Analysis and SM Software Requirements), the C&S assessment criteria will subsume both activities. What is first required, however, is a strategy for identifying and assessing these competencies, and this will form the basis of the next section.

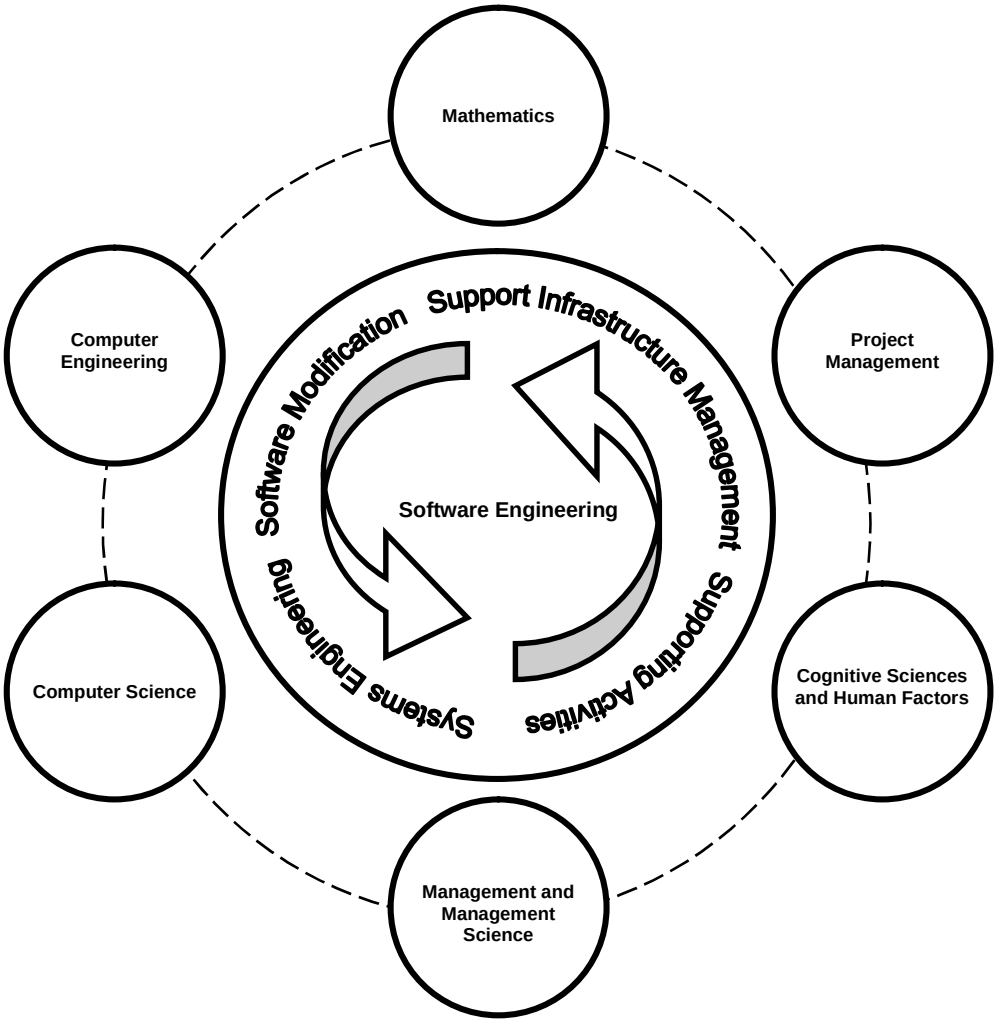


Figure 3.12 – Related Disciplines, after [SWEBOK, 2001]

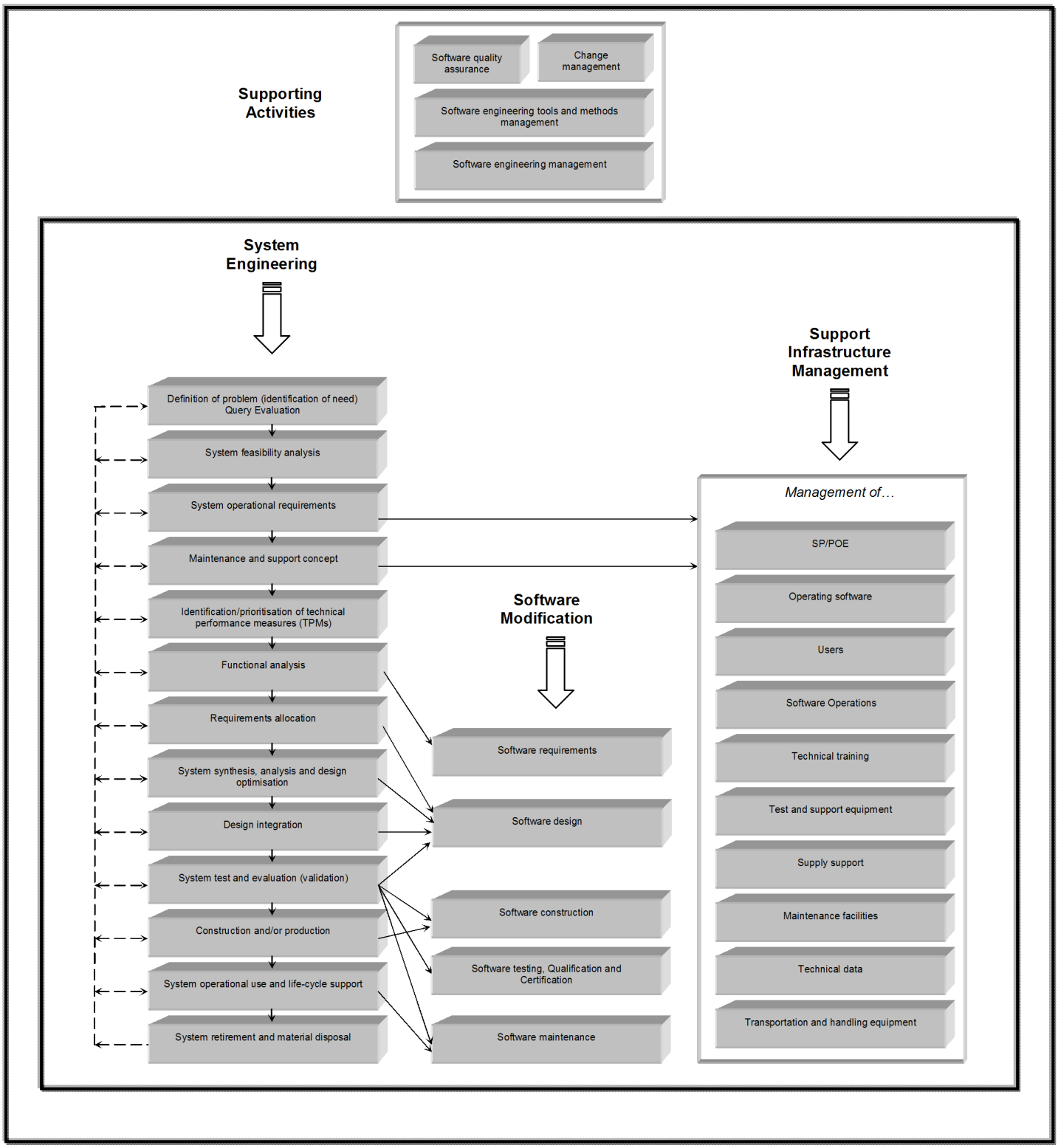


Figure 3.13 – Software Engineering Activity Model

Software Engineering Assessment Framework

Throughout sections 1 to 3, I have embarked upon the creation of a SE Activity Model. However, identifying these activities has only resulted in half of what is actually required. In order to eventually conduct a SE competency evaluation of RAF SSTs, there must not only exist a well defined model of SE activities, but also some way of measuring an individual's competency when undertaking those activities. To this end I now intend to formulate a strategy for the creation of a C&S assessment framework, utilising, extracting and adapting the most suitable elements from several related assessment tools.

4.1 Software Engineering Competencies and Skills

If we begin first with competencies, we must first ask ourselves what we actually mean when we state that an individual is indeed competent to perform a particular task or activity. Competence can be defined as:

> *"the ability to do something successfully or efficiently"* [Oxford, 2003]

and we could possibly use this definition as a basis for the creation of an assessment framework. However, it does introduce the problem of having to define what is actually meant by *"successfully or efficiently"* (s/e). What constitutes an acceptable definition of s/e can vary dependant upon the circumstances of its use. For example, if we compare the production of nuclear power plant control software to that of the latest PlayStation$_{SM}$ game release, it is possible to appreciate the obvious differences between the two that would perhaps justify a more stringent (and hence successful) approach to one development process over another.

One way of determining what we may mean by s/e would be by a thorough examination of the consequences of software failure whilst in use[13]. Again, however, different organisations may well put a different and altogether subjective value on failure (human life versus corporate insolvency for instance) although given the scope of this dissertation determining what is actually meant by s/e may not necessarily be an insurmountable problem. It is reasonable to pursue a thread of analysis along the lines of real-time avionics applications availability given that the RAF is predominantly concerned with the support of these. As a first step towards determining mission availability criteria, we can utilise the failure condition categorisation as defined in [RTCA, 1992] which lists five failure categories thus:

> *"Catastrophic*
>
> *Hazardous/Severe-Major*

[13] For example, by applying Failure Mode and Effects Analysis (FMEA).

Major

Minor

No Effect"

As the "catastrophic" category is only concerned with safe flight and landing and not functionality, it will not be used here. However, the other four categories refer either to safety and/or function and operation thus:

> *"Hazardous/Severe-Major: Failure conditions which would reduce the capability of the aircraft or the ability of the crew to cope with adverse operating conditions to the extent that there would be:*
>
> > *(1) a large reduction in **safety** margins or **functional** capabilities,*
> > …
> > …
>
> *Major: Failure conditions which would reduce the capability of the aircraft or the ability of the crew to cope with adverse operating conditions to the extent that there would be, for example, a significant reduction in **safety** margins or **functional** capabilities…*
>
> *Minor: …Minor failure conditions may include, for example, a slight reduction in **safety** margins or **functional** capabilities…*
>
> *No Effect: Failure conditions which do not affect the **operational** capability of the aircraft…"* [RTCA, 1992]

It appears that the [RTCA, 1992] places the same level of importance on safety and functional capability when determining failure condition categories. One can reinforce this premise with the notion that operational (or mission) failure may indeed be attributable to a loss of functionality that could indirectly lead to casualties. An example of this could be the non-completion of an operational sortie during either hostilities (air strike) or peacetime (search and rescue) that results in the loss of life not directly associated with the platform.

We can therefore make the assumption that safety criticality and mission availability can be treated as somewhat complimentary aspects of the same software product. Extending this notion to the determination of a s/e definition for use in a SE Activity Model, we can first apply existing safety-related competency assessment techniques to SE activities in order to produce a competency assessment framework that will ultimately meet this requirement.

The Institute of Electrical Engineers (IEE) - Competencies

> *"In any well-run organisation staff are required to be competent to perform the tasks assigned to them"* [IEE, 1999].

The IEEs competency guidelines for safety-related system practitioners is based on twelve safety functions each of which is defined by a set of tasks that the safety professional is required to perform in order to address a particular function. Each task within a particular function has an associated set of competencies directly related to the performance of that task, and typically consists of technical skills and knowledge. In addition, each function has an associated set of competencies that apply across all tasks (see Figure 4.1) and are typically behavioural skills and understanding. As a result, the IEE have developed a set of *"competency statements"* and guidance on an *"assessment procedure"* for safety-related system practitioners. The identified functions can be selected on a pick and mix basis in order to allow competencies to be tailored to particular job requirements. I have chosen not to list all the IEE competencies as many are not relevant to this dissertation and would only serve to consume space.

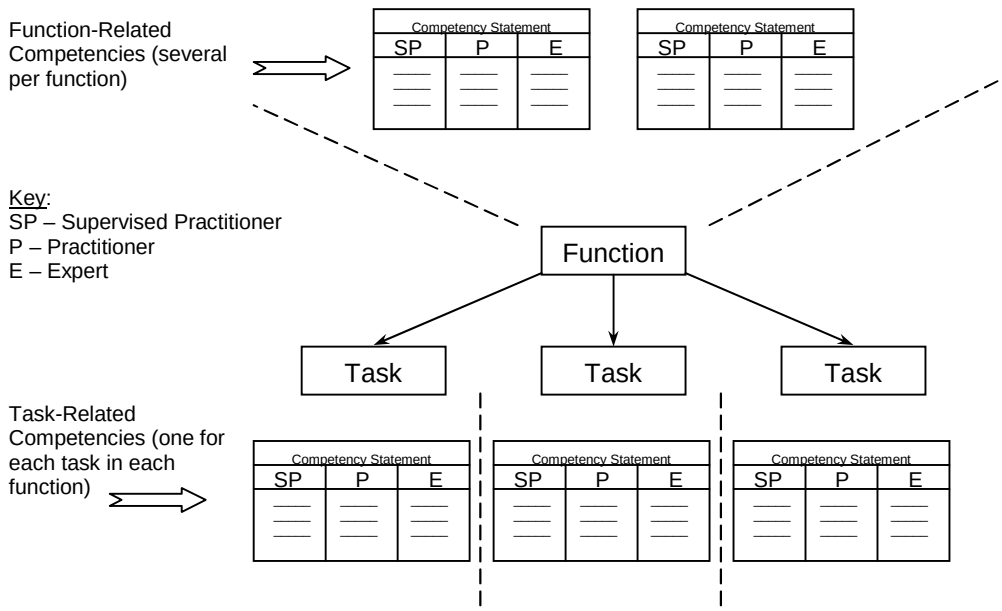


Figure 4.1 – Assessment Guidance [IEE, 1999]

We can draw a direct comparison between the IEE safety functions and tasks and our SE activity model (see Figure 4.2). Essentially, each safety function can be directly related to our four SE functional areas (System Engineering, Software Modification, Support Infrastructure Management and Supporting Activities). Likewise, we can utilise selected IEE competencies to map to our identified SE activities. The mechanism I have adopted for this initial competency selection is based on one of immediate relevance:- if the competency can be linked to the SE

activity with little or no modification, it has been retained; if not, it has not been included. For example "assessing safety analysis" has been omitted because in its present form it is only relevant to the safety field and would require substantial alteration to make it relevant to a SE activity. Similarly, "designing hardware" has been omitted but for the different reason that we are only concerned with software. However, "allocation of responsibilities" has been retained because of its broad relevance to all fields and the relative ease with which it can be adapted to accommodate our needs. The outcome of this competency selection process is documented at Appendix B in Table B.1.

There is, of course, the risk that, as my selection process is entirely subjective, this could lead to the creation of an incomplete framework. However, I believe I have mitigated this risk by intending the framework to be adaptive, to effectively mature from input with respective iterations of use.

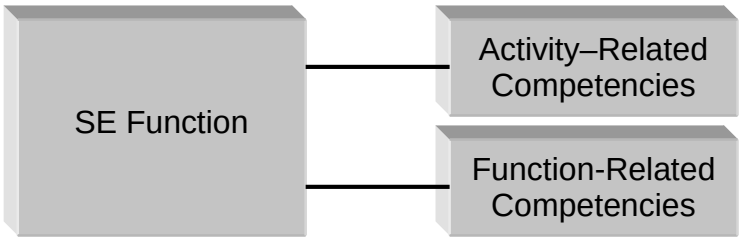


Figure 4.2 – SE Competency Assessment, after [IEE, 1999]

Basing a SE competency assessment framework on one similar product, however, is not good science. There are a number of other tools and reference material available that can be used to provide an alternative viewpoint on skills assessment, one of which is the Skills Framework for the Information Age [SFIA, 2004].

Skills Framework for the Information Age (SFIA) - Skills

> *"…a model for describing what Information and Communications Technologies (ICT) practitioners do"* [SFIA, 2004]

The SFIA model is constructed as a two-dimensional matrix (see Figure 4.3). One axis divides the whole of ICT into 'skills'. These skills are grouped for convenience into subcategories or business roles. Subcategories are grouped into five categories or work areas – strategy & planning, management & administration, development and implementation, service delivery and sales & marketing. These skills, subcategories and categories are detailed at Appendix C.

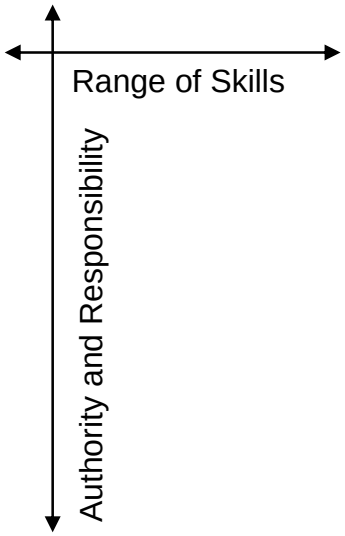


Figure 4.3 – SFIA Framework [SFIA, 2004]

The other axis defines the level of responsibility and accountability required of ICT practitioners and users. Each of seven levels – from new entrant to strategist level – is defined in terms of autonomy, influence, complexity and business skills. These responsibility levels are detailed at Appendix D and illustrated here in Figure 4.4.

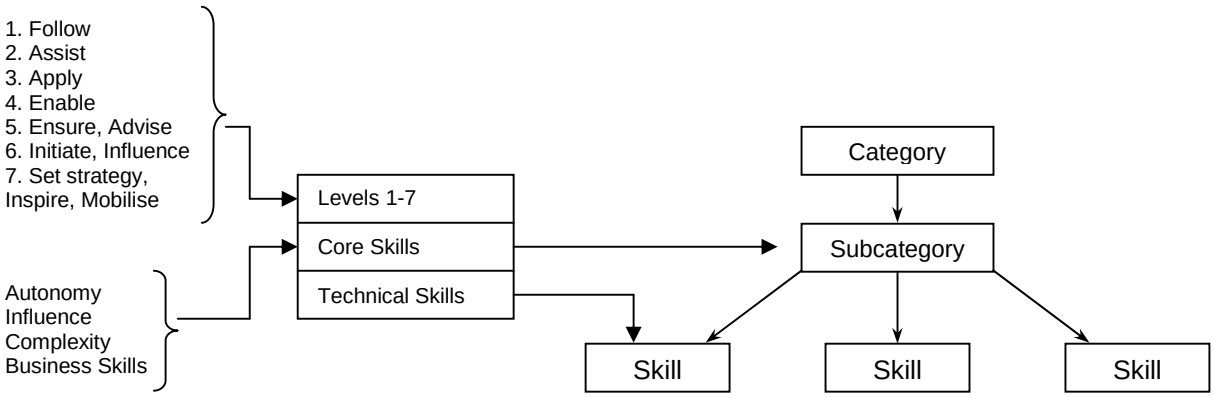


Figure 4.4 – Assessment Guidance [SFIA, 2004]

For each skill at each level, descriptive text provides examples of typical tasks undertaken. A typical task for systems design at level 5, for example, is *"reviews others' system design to ensure selection of appropriate technology, efficient use of resources, and integration of multiple systems and technology."*

Again, as with the IEEs competency guidelines, I have initially selected SFIA skills for use based on their immediate relevance to SE activities, with the same subjective risk involved. The outcome of this skill selection process is again documented at Appendix B in Table B.1.

4.2 Competency and Skill Assessment Methodology

The IEE defines three types of safety professional for the purpose of assessing competencies: Supervised Practitioner, Practitioner and Expert thus:

> *"Supervised Practitioner A Supervised Practitioner has sufficient knowledge and understanding of best practice, within the organization or within the relevant industry sector, to be able to work on the Tasks associated with the overall Function without placing an excessive burden on the Practitioner or Expert…It will be the responsibility of a Practitioner or an Expert to check the work of the Supervised Practitioner.*
>
> *Practitioner A Practitioner has sufficient knowledge and understanding of best practice, and sufficient demonstrated experience, to be able to work on the Tasks associated with the overall Function without the need for detailed supervision. A Practitioner will maintain their knowledge and be aware of the current developments in the context in which they work. The Practitioner may be required to perform detailed checks on the work carried out by a Supervised Practitioner.*
>
> *Expert An Expert will have sufficient understanding of why things are done in certain ways, and sufficient demonstrated managerial skills, to be able to undertake overall responsibility for the performance of a Function…*
>
> *An expert* (sic) *will keep abreast of technologies, architectures, application solutions, standards, and regulatory requirements…An expert* (sic) *will have sufficient breadth of experience. knowledge and deep understanding to be able to work in novel situations.*
>
> *An expert* (sic) *is able to deal with a multiplicity of problems under pressure…"* [IEE, 1999]

In order to align this dissertation with the requirements of RAF SSTs, I have utilized the Royal Air Force Engineering Orders and Procedures [100B-01, 1997] classification of engineering tradesmen: Producer, Supervisor and Manager. Each classification has the following functional definitions:

> *"**Management**. It is the managerial function to calculate, allocate and direct resources in terms of manpower (supervisors and producers), materials, tools and time to a particular maintenance operation, and to be responsible for controlling the quality and quantity of work produced…*
>
> ***Supervision**. It is the supervisory function to direct the tradesman to do a job in accordance with the manager's directive. The supervisor is to ensure*

that the tradesman is competent to do the job, that he is adequately briefed and that he has the necessary tools, equipment, publications and instructions. He is also to…certify, through his countersignature…that he has taken all reasonable care to ensure the quality and completeness of the job [and] in deciding the extent to which supervision must be exercised…is to take due account of the difficulty and nature of the job, the skill and experience of the tradesman involved, and the prevailing circumstances.

***Production**. It is the function of the producer to perform a job in accordance with the supervisor's directive [and to be] responsible for the quality and completeness of his work…"* [100B-01, 1997]

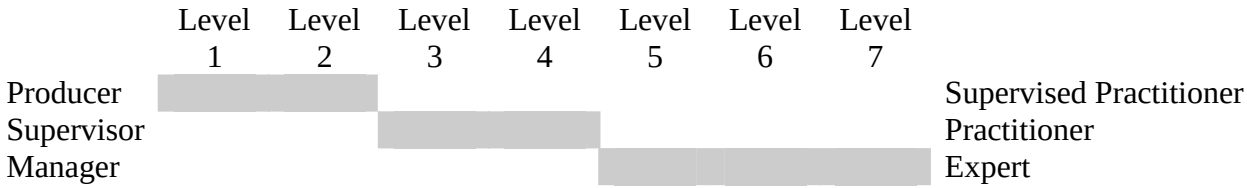
Applying a simple comparison, one can identify similarities between the [IEE, 1999] safety professional definitions and the [100B-01, 1997] classification to the extent that they can be mapped thus:

Supervised Practitioner → Producer
Practitioner → Supervisor
Expert → Manager

The [SFIA, 2004] framework reference, on the other hand, divides skills into seven levels:

"1. Follow
2. Assist
3. Apply
4. Enable
5. Ensure and advise
6. Initiate and influence
7. Set strategy, inspire and mobilize"

reflecting the autonomy, influence, complexity and business skills which are typical of someone at that level. These levels are detailed at Appendix D, and, utilising these level definitions, we can cross-reference further the [100B-01, 1997] classifications and [IEE, 1999] safety professional definitions to the SFIA levels thus:



This cross-referencing helps tie the raw source material thus far utilised together, and map an outline competency and skills assessment framework to individual levels of responsibility. The progress made so far is illustrated in Figure 4.5.

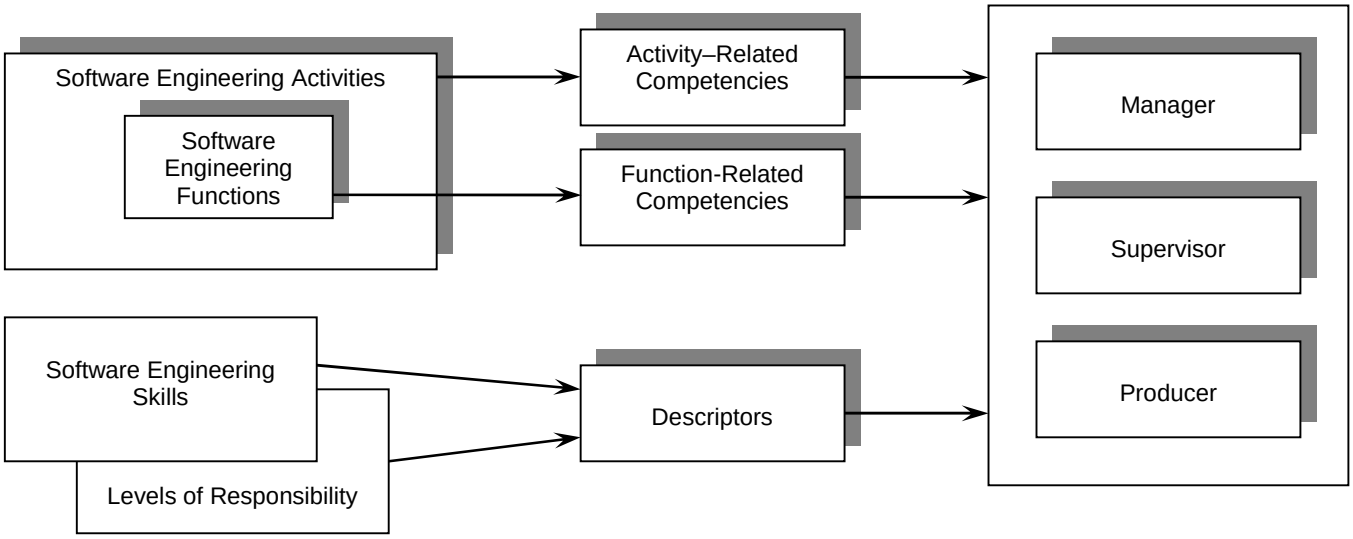


Figure 4.5 – Outline SE C&S Assessment Framework

This outline strategy has identified, at an initial level, C&S relevant to SE, and I have worked under the postulation that the definition and application of this framework will be relatively straightforward. However, there are two specific problems that I have not yet addressed:

a) The C&S that I have identified are still basically source-specific. What I mean by this is that the competencies are still essentially safety-related and the skills are almost all ICT-related. I therefore need some way of objectively modifying them for use with the SE functions and activities that I have identified.

b) When undertaking an assessment, I will need to base it on a workplace observation and oral (managerial evidence and response to questions) basis in order to maintain objectivity. I would expect the outcome of this assessment to be:

i) Assessment against each C&S statement for the function or activity,

ii) Recommendations, for example training, and

iii) Information to help in team building.

However, whilst I expect this process to be straightforward, I do not expect the outcome to be a simple comply/does not comply. There will certainly be

instances where evidence will suggest that partial compliance is being achieved, and actually grading this objectively may prove difficult.

In order to address these problems, I have decided to seek guidance from the Capability Maturity Model for Software (SW-CMM), applying some principals as and where required.

Capability Maturity Model for Software

The SW-CMM has been a model for judging the maturity of the software processes of an organization for many years now. It provides software organizations with guidance on how to gain control of their processes for developing and maintaining software and how to evolve toward a culture of software engineering and management excellence [CMM, 1993].

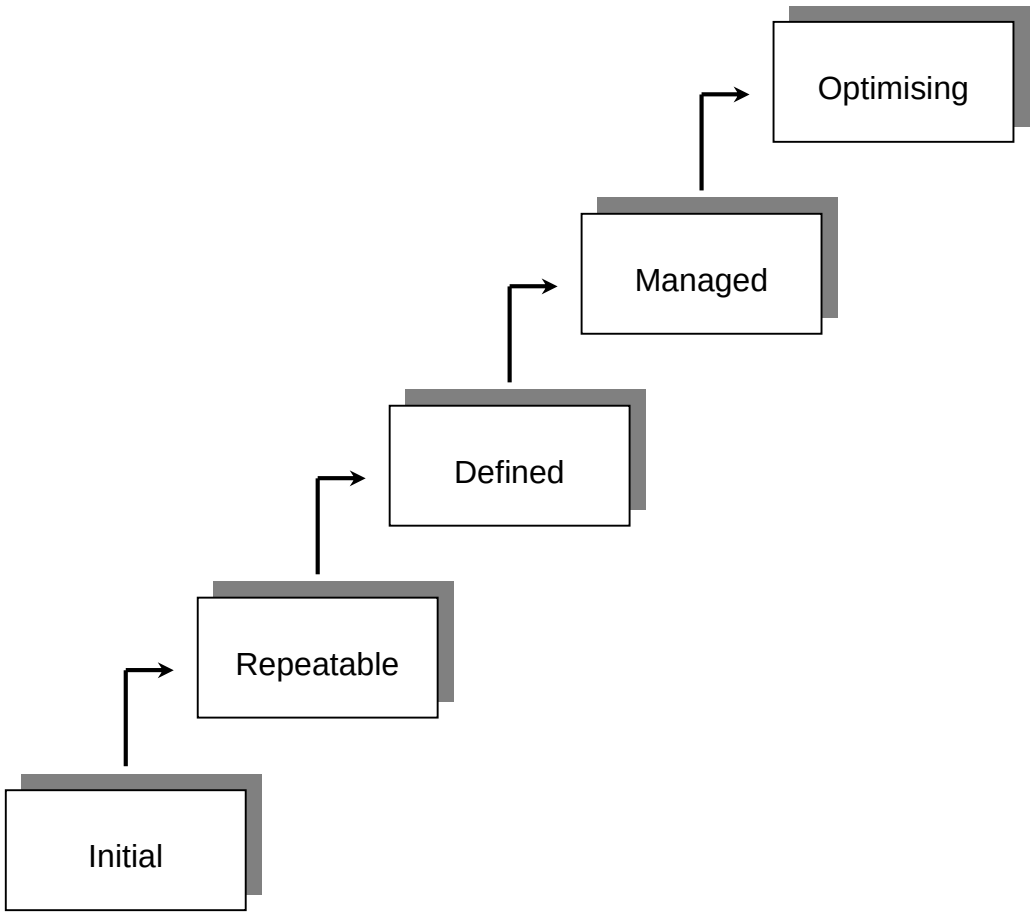


Figure 4.6 – The 5 Levels of Software Process Maturity [SW-CMM, 1997]

The SW-CMM is based on five levels of software process maturity that lay successive foundations for continuous process improvement (see Figure 4.6). Each

of these levels comprises a set of process goals that must be achieved in order for an organisation to progress.

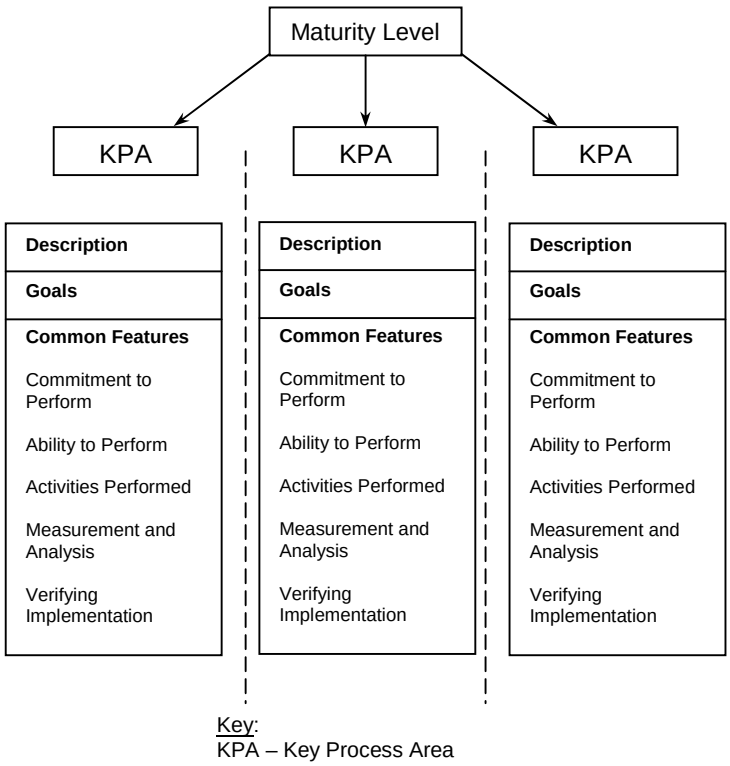


Figure 4.7 – Assessment Guidance [SW-CMM, 1997]

With the exception of Initial, each maturity level comprises several well-defined Key Process Areas (KPAs) against which an organisation can be measured. These KPAs are in turn organised by common features, which are attributes that indicate whether the implementation and institutionalisation of a KPA is effective, repeatable and lasting (see Figure 4.7). The SW-CMM KPAs and common features are listed in Table 4.1, and I intend to utilise these in the SE competency framework to help define and measure individual C&S.

Maturity Level	Key Process Area (KPA)	Common Features
Initial	-	-
Repeatable	Software Configuration Management	Commitment to Perform, Ability to Perform, Activities Performed, Measurement and Analysis and Verifying Implementation
	Software Quality Assurance	
	Software Subcontract Management	
	Software Project Tracking and Oversight	
	Software Project Planning	
	Requirements Management	
Defined	Peer Reviews	
	Intergroup Coordination	
	Software Product Engineering	
	Integrated Software Management	
	Training Program	
	Organisation Process Definition	
	Organisation Process Focus	
Managed	Software Quality Management	
	Quantitative Process Management	
Optimising	Process Change Management	
	Technology Change Management	
	Defect Prevention	

Table 4.1 – SW-CMM KPAs and Common Features

Combining the SW-CMM together with the outline SE C&S assessment framework gives a SE C&S assessment framework creation strategy (as depicted in Figure 4.8). It is from this and Table B.1 in Appendix B that I shall create assessment guidance in the next section.

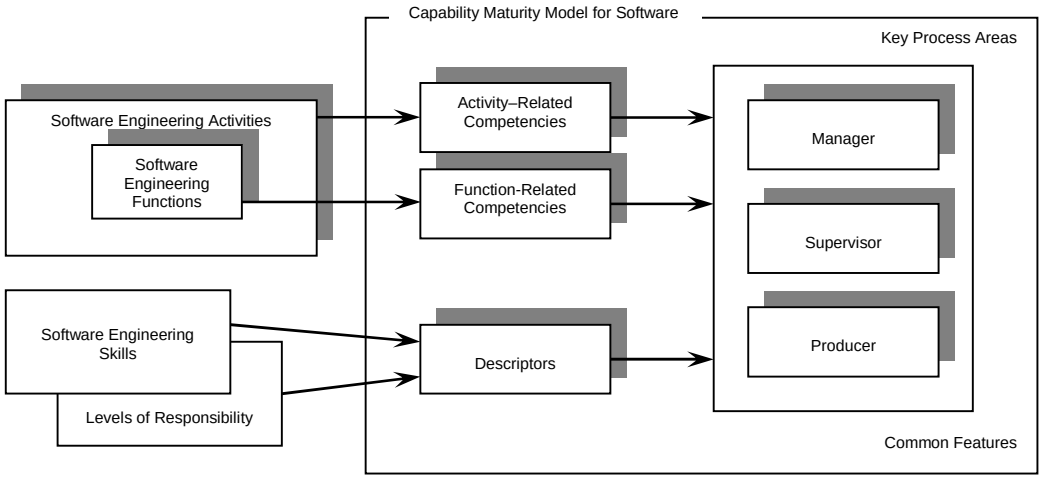


Figure 4.8 – SE C&S Assessment Framework Creation Strategy

Theoretical Framework Creation

This section is concerned with the synthesis of the individual [IEE, 1999], [SFIA, 2004], [SW-CMM, 1997] and [100B-01, 1997] elements in order to produce SE C&S assessment guidance. Figure 5.1 is representative of the combining process I have developed and documented in Appendix E. For reasons of clarity, I have decided to reference each assessment by competency alone, adding cross-referenced skills as necessary. Where there is obvious symmetry between competencies [IEE, 1999] and skills [SFIA, 2004], they have been merged.

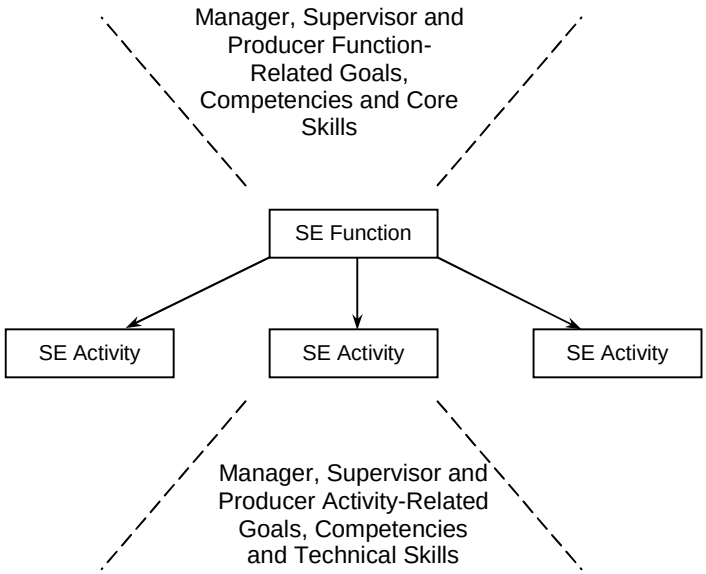


Figure 5.1 – SE C&S Assessment Guidance Creation

Table 5.1 represents how each competency in the guidance framework is individually documented at Appendix E. It is based primarily on the [IEE, 1999] assessment guidance utilising [100B-01, 1997] levels of competency, and combines skills [SFIA, 2004] and common feature [SW-CMM, 1997] amplifying material as appropriate. As the [IEE, 1999] guidance is cross-referenced to the life-cycle phases in IEC 61508, it has provided a useful check when mapping the competencies across to the SE Functions and Activities.

Competency *SE Function or SE Activity related*		
Description		
Producer	**Supervisor**	**Manager**
Required demonstrable proficiency level utilising [IEE, 1999], [SFIA, 2004] and [SW-CMM, 1997], documenting core skills for Function Related Competencies and technical skills for Activity Related Competencies →		

Table 5.1 – SE C&S Assessment Table

Framework Creation Conclusions

The [IEE, 1999] competencies and [SFIA, 2004] skills have been allocated to SE functions and activities by matching them to the discussion descriptions that I outlined in section 3. [IEE, 1999] competencies have been utilised "as published" as far as possible; however, there are certain competencies that do reference safety-related topics in their descriptive text. In these cases, they have been adjusted by myself in order to make them SE-specific, necessitating the creation of the new terms and definitions of 'Erroneous Situation' and 'Software Engineering Management System' which I have listed in the Definition of Terms section.

[SFIA, 2004] skills have proven to be less amenable to the matching process I employed for the [IEE, 1999] competencies. This can be attributed to the fact that:

a) SFIA "…provides a common reference model for the identification of the skills needed to develop effective information systems (IS) making use of information and communications technologies (ICT)", and

b) The secondary aim of this dissertation being the enabling of the identification and definition of RAF SST manning requirements, of which the entire SIM functional area was provisionally expected by myself to fall within the IS and ICT field of responsibility.

As such, it was unacceptable to simply discard IS- and ICT-titled skills or modify the descriptive texts of those that appeared at first hand to be generic. In this case, I evaluated each skill on a simple merit-basis: if it appeared generic and could be directly related to a previously defined competency with minimal modification, then it was; if not, it was assigned to a relevant SIM activity (or activities).

Again, as before, there is the risk that the subjective nature of my adjustment and assignment processes has lead to the creation of an incomplete or flawed framework. However, I believe that its practical application and subsequent suggestions for

modification will drive out any errors that I may have unintentionally introduced, ultimately leading to a mature, reliable product.

Porting the IEE competencies and SFIA skills across to the C&S framework has inevitably resulted in a less than 100% coverage of my identified SE functions and activities, especially in the SIM area. This has necessitated the requirement for the creation of new competencies (prefixed NC) with which to conduct an assessment. In a similar vein, it was not possible to completely match all the SFIA skills with all the IEE competencies, again effectively necessitating the requirement for the creation of NCs. Where possible I have created NC tables basing them on the associated SFIA skills; however, it has not been entirely possible to fully populate these new tables. In this case, I have identified absent material "to be defined", a proportion of which I expect to populate during the empirical assessment.

It was initially planned that Appendix B, the C&S table, would lend itself easily to cross-referencing; alas, this has not been the case. Whilst functions/activities and competencies (on the left hand side) are co-ordinated in a relatively straightforward *xy* configuration, functions/activities and skills (on the right hand side) are *xy* co-ordinated only to a certain extent. This is due to the fact that some skills present themselves in more than one competency, making logical cross-referencing difficult to achieve. Ideally, the C&S table would be better presented in a relational database thus allowing for multiple data configurations to be presented.

To summarise, in conjunction with Appendices B, E and F:

a) Where SFIA technical skills could not be related to activity-related competencies, I have created new competencies (1-29). The only exception to this is for technical skills 21-24 where they are referenced in the SIM activities of Test and Support Equipment and Technical Data. In these specific cases, I felt that they were unique enough to warrant not being related to any specific competencies, even though they are competency-related elsewhere in the framework. What this effectively means is that for the SIM activities of Test and Support Equipment and Technical Data there is the requirement to create specific competencies based around technical skills 21-24.

b) I felt that some of the SFIA technical skills were more appropriate as core skills, as they appeared to be applicable to multiple functions, so I adapted them accordingly (1-8).

c) The core levels of responsibility skills of Autonomy, Influence, Complexity and Business Skills (9-12) have been assigned to each of the 23 Function Related Competencies. I have taken the highest SFIA level definitions for each of these core skills when assigning them to the producer, supervisor and manager categories (i.e. level 2 for producer, level 4 for

supervisor and level 7 for manager), as the differences between each grouping are only minor.

d) Some new competencies are original, created by myself on the basis of tacit domain knowledge and available reference material, in order to fill gaps where no existing competencies were appropriate for particular SE activities. These are NC30 (for functional analysis), NC31 (for software requirements) and NC32 (for system operational requirements and maintenance and support concept).

e) Some activity-related competencies don't have any associated technical skills. These are ARC1, ARC5, ARC7, ARC9, ARC10, ARC13, ARC15, ARC16, ARC17, ARC18, ARC19, ARC22, ARC24, ARC25, ARC26, ARC30 and ARC33. I expect these to be populated in the long term as the C&S Assessment Framework matures.

f) I have adapted the [IEE, 1999] assessment pro-forma for the purpose of recording competency assessments, and recoded this at Appendix F. I do, however, expect this pro-forma to require modification once I conduct the empirical assessment.

Empirical Assessment

This section is concerned with the application of the SE C&S Assessment Framework. A critical review of the results is undertaken in the next section.

6.1 SE C&S Assessment of HSMU

In order to evaluate the framework and the assumptions made thus far, it was submitted to a real-world test at the Harrier Software Maintenance Unit (HSMU). However, as the evaluation process only had myself as a resource, and given that it was not possible to interview all HSMU personnel, the evaluation strategy was as follows:

a) An identification of the activities currently undertaken by HSMU was conducted.

b) This activity identification was then compared and mapped to the SE Activity Model (Figure 3.13).

c) A percentage selection of activities was made (based upon personnel and time availability), and then an assessment of those activities undertaken using the pro-forma at Appendix F. In particular:

i) The "Context Summary" was completed by summarising the required context for which the candidate was being assessed. This included available information such as application and technology details, applicable standards etc.

ii) For each competency, a "Competency Statement" was formulated that summarised the evidence presented. Associated FRCs were applied at the assessor's discretion to each ARC where applicable. If possible, I utilised this area to formulate statements for those competencies that I had not yet populated.

iii) If possible, I would formulate an "Assessment Summary" and "Action Plan", effectively deriving an individual competence profile histogram for each SE Function.

The completed pro-forma is documented at Appendix G.

d) An assessment of whether or not individuals perform activities ranging across multiple SE Functions was also made. In addition, there may have been "extra" activities that HSMU undertake that may not be present in the SE Activity Model.

e) The assessment results are presented in the next section consisting of a critical review of the SE C&S Assessment Framework.

Current HSMU Maintenance Activities

The HSMU support an AYK14 assembler OFP and associated ground-based software for the RAFs Harrier GR7/T10 fleet. In achieving this, they undertake several broad functions which are presented in Table 6.1. These functions have subsequently been broken down further utilising HSMUs software life-cycle (Figure 6.1), and an overall analogous mapping to the SE Activity Model has been carried out.

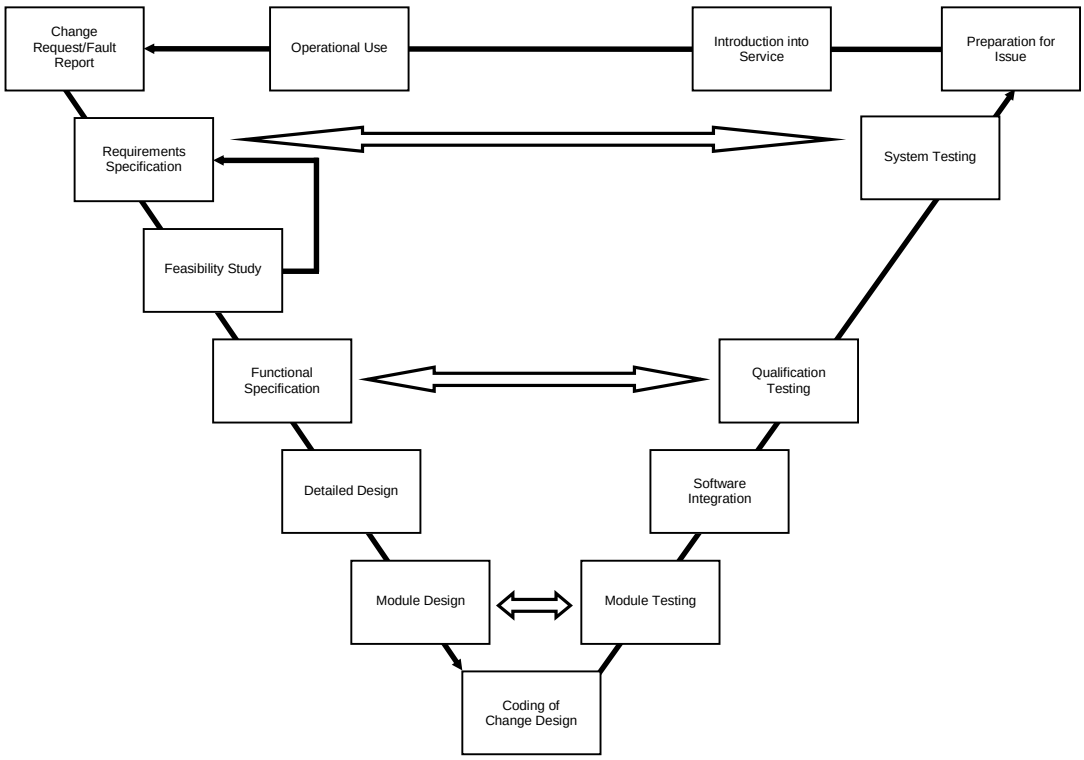


Figure 6.1 – HSMU Software Life-Cycle

I should stress that my mapping of HSMUs functions to the SE Activity Model is chiefly subjective, based upon limited available data. It is not my intention to "bend" HSMUs functions to fit my model, nor is it my intention to justify the model by utilising all of its activities. As a check, however, I have also mapped HSMUs manning structure (see Figure 6.2) to the model to ensure as complete a coverage as possible, and provide an initial list of potential interviewees.

HSMU Functions	SE Activity Model Functions	HSMU Life-Cycle Activities	SE Activity Model Activities	HSMU Manning Areas
OFP Support and the undertaking of software investigations	SysE	Change Request/Fault Report	Definition of Problem/ Query Evaluation	Queries and Prototypes
		Requirements Specification	Functional Analysis	SNCO Requirements SNCO SST3
		Feasibility Study	System Feasibility Analysis	
		Software Integration	System Test and Evaluation	SNCO Testing
				SNCO SST1
				OFP Testers
		Qualification Testing	System Test and Evaluation	
		System Testing	System Test and Evaluation	SNCO Testing
				SNCO SST1
				OFP Testers
	SM	Functional Specification	Software Design	SNCO OFP Production SNCO SST2 Programmers
		Detailed Design	Software Design	
		Module Design	Software Design	
		Coding of Change Design	Software Construction	
		Module Testing	Software Testing	SNCO Testing
				SNCO SST1
				OFP Testers
	SIM	Preparation for Issue	Operating Software Management	
		Introduction into Service	Operating Software Management	
Query Answer Service	SysE	Change Request/Fault Report	Definition of Problem/ Query Evaluation	Queries and Prototypes
Control of OFP issue	SIM		Operating Software Management	
Provision of training for HSMU personnel	SIM		Technical Training	SNCO Training
Quality Assurance	SA		Software Quality Assurance	FS QAM
Configuration Control	SA		Change Management	Configuration Manager
Advice and Assistance	SysE		Definition of Problem/ Query Evaluation	
Project Planning and Management	SA		Software Engineering Management	OC HSMU
				OC Requirements & Testing
				OC Engineering Flt
	SIM		Test and Support Equipment Technical Data	Systems Support
				Systems Operator
				Support Programmer
	SIM		Supply Support	Administration
			Maintenance Facilities	
			Transportation and Handling Equipment	

Table 6.1 – HSMU Coupling to SE Activity Model

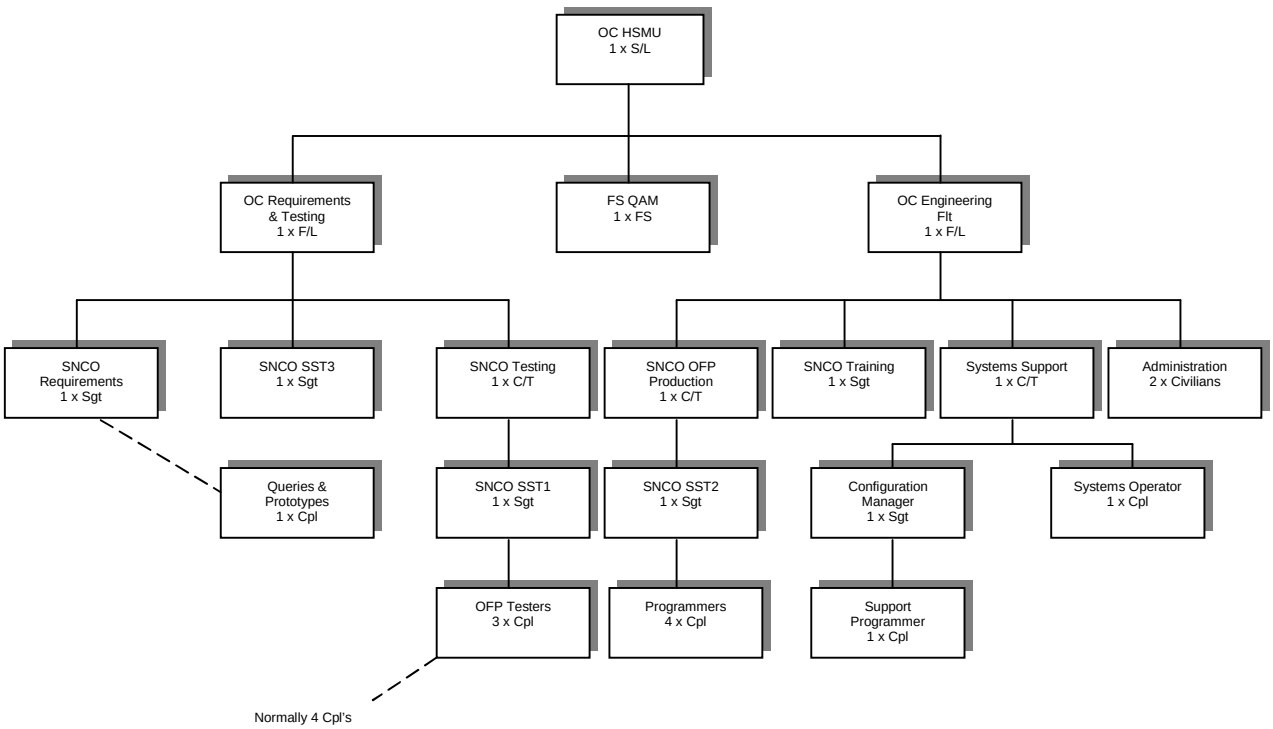


Figure 6.2 – HSMU Manning Structure Chart

Competency Assessment of HSMU

Completed Assessment Pro-forma, with additional candidate notes, have been recorded at Appendix G. For the "Context Summary" I have chosen to reference the product and its relationship to the tasking process, as I felt that this would enable each candidate to put the competency statements into perspective.

During the assessment process I was able to develop statements in conjunction with the candidates for competencies that I had not yet populated. These are listed at Appendix H.

Conclusions

7.1 Framework Evaluation Results

Although not entirely unexpected, the assessment phase did begin quite slowly. Once the first candidate was finished, however, I was able to reflect on how that assessment had gone and found that the framework had been relatively easy to apply. Due (predominately) to time constraints, I was only able to trial the Assessment Framework on four HSMU individuals and 28 competencies. This equated to 14.8% of HSMU personnel, 41.8% of ARCs and 31.1% of all competencies. With this in mind, however, I do not believe that this constraint had too detrimental an effect, as I was still able to evaluate the assessment process, even though the entire framework content was not covered.

There was a lot of reliance on assessor/candidate discussion when arriving at a qualitative assessment; this was as much for my benefit in order to contextually relate the quantitative information presented, as I felt it was for the candidate's self esteem. I conducted the assessment by allowing each candidate to read the relevant competency and conduct an initial self-assessment, thus facilitating the identification of evidence to support their decision.

I found that adding the Context Summary to the assessment supported the assessment process; however, I believe that one drawback from this method was that the assessments ultimately become team-specific, and this may present problems when comparing like-assessments between teams. I did, however, find that the Framework and its application had the additional effect of identifying what a team actually does by allowing the individuals to relate their tasking to the competencies.

Although I concentrated on Activity Related Competencies during the assessment process, I did not ignore Function Related Competencies. As these apply across all activities and relate to behavioural skills and understanding, I felt that they were best applied qualitatively by the assessor.

In addition, I found that the Manager, Supervisor and Producer classifications created confusion when viewed from a military perspective, as they are normally associated with levels of responsibility and not necessarily competency. To illustrate, a Manager under assessment attempted to associate with the "Manager" competencies rather than with his "true" competency level simply because he felt that he had to. At the risk of creating unintentional confusion, I shall utilise the [IEE, 1999] categories in this and the remaining sections as I wish to make recommendations that now rely upon their usage.

The Framework does need to be made easier to cross-reference between Functions, Activities and Competencies, as I found Table B.1 and Appendix E difficult to use in an assessment situation. Likewise, for the assessment to be robust traceability is

all important, and I believe that the addition of notes covering all decisions should be recorded and justified where applicable in future evaluations.

In order to assist the pre-preparation process, it would be wise to request Terms Of Reference (TORs) to assist in the mapping of the team and individuals to the SE Activity Model. The strategy for the initial mapping of RAF SSTs to the model would then be:

a) Request copy of life-cycle, manning structure and TORs.

b) Conduct initial mapping of functions and activities.

c) Forward this mapping to the SST and get them to compare it with what they actually do, potentially identifying where they undertake cross-function tasks.

I found that the histogram as used by the IEE and myself on the assessment pro-forma at Appendix F was unsuitable. As it can be used to identify areas for improvement as well as identifying a team's overall competency rating for cross-team comparison purposes, its importance should not be underestimated. My suggested improvements to it, therefore, are as follows:

a) It should show all applicable categories that are relevant during the assessment e.g. Supervised Practitioner and Practitioner, if the subject is assessed as Expert.

b) For the individual's overall Function/Activity competency rating, the following reckoner could be used:

i) Assign the following values: Supervised Practitioner = 1, Practitioner = 2, Expert =3.

ii) Add up the highest values in each competency column.

iii) Divide this total by the number of competencies and round up.

iv) Assign an overall competency rating based on that total, e.g.

Supporting Activities																		
	Change Management			Software Engineering Management							Software Engineering Tools and Methods Management		Software Quality Assurance					
	ARC13	NC7	NC8	ARC1	ARC5	ARC19	ARC28	ARC29	NC9	NC10	ARC8	NC11	ARC18	ARC20	ARC21	ARC26	ARC27	ARC30
Supervised Practitioner	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Practitioner	X		X	X	X		X	X		X		X		X	X	X	X	
Expert	X			X	X					X							X	
Overall	Practitioner			Practitioner							Practitioner		Practitioner					

c) For the team's overall competency rating, the following reckoner could be used:

i) Create a table based on Function with respective Activities.

ii) Add up the number of Supervised Practitioners, Practitioners and Experts for each Activity.

e.g. taking the above individual example:

Supporting Activities				
	Change Management	Software Engineering Management	Software Engineering Tools and Methods Management	Software Quality Assurance
Supervised Practitioner				
Practitioner	1	1	1	1
Expert				

All of the above said, my overall analysis is that the framework and its trial application is a success. With this in mind, I propose the following thesis.

7.2 Thesis

The view one takes of Software Engineering (SE) can no longer be confined to the production and/or modification of software alone. Converting a set of requirements into the software component of a system may seem a relatively straightforward goal; however, if we are ever to produce software that is both useable and cost effective, we must first approach the SE process from a wider perspective. What this means is considering it not only from the system within which it will ultimately reside, but the system that will ultimately support it. And in order to achieve this we must make use of a SE Activity Model, that forces us to consider the product both from within and without its project boundaries.

But this is not all. A SE Activity Model alone will not guarantee the production of useable software; who we employ to produce and maintain it will. This distinction can impact considerably on the supportability of a product over its life time, and we must bear this in mind when going through our deliberations. We can, however, alleviate at least some of these potential problems by assessing and managing the personnel employed to undertake these tasks for us. In the end, it will be our need for useable and cost effective software that will drive forward the use of a SE Activity Model that is underpinned by the competencies and skills of the personnel who work within it.

7.3 Proposal

To this end, I propose the use of a SE Activity Model both during and post-development that forces the consideration of four areas: System Engineering, Software Modification, Support Infrastructure Management and Supporting Activities. In addition to this, I propose the Competency and Skill assessment and subsequent career management of personnel employed to undertake SE duties for RAF SSTs.

7.4 Recommendations

It is recommended that:

a) The SE Activity Model and C&S Assessment tables be adopted as the RAF policy standard.

b) Liaison be undertaken with Training Development Wing, RAF Halton to develop the SE Activity Model and C&S Assessment tables further.

c) Statements be formulated in conjunction with SSTs for competencies and skills where currently unpopulated.

d) The [IEE, 1999] classifications of Supervised Practitioner, Practitioner and Expert be used in the C&S Framework.

e) A full C&S review of RAF SSTs be undertaken by an assessment team with the aim of categorising both personnel and teams. The makeup of this assessment team will be critical given the personal nature of the assessment process, and only personnel with excellent inter-personal skills should be utilised.

f) A planned programme of assessments designed to track and monitor personnel be put in place.

Discussion

8.1 Taking the Conclusions Further

This C&S assessment framework should be viewed as a work in progress. It does not make any claim to be complete nor does it make any suggestion that an external evaluation of it is not required. It is intended as a first draft from which successive users of it are expected to offer constructive criticism and suggestions for change.

All the competencies are based around 3 levels of expertise: supervised practitioner, practitioner and expert. However, this appraisal strategy is not meant to imply that managers within a SST should possess all the expert competency levels. Likewise, it is not meant to imply that all the competencies should be covered in a SST either. How it is meant to be utilised is threefold:

a) as a strategy for identifying competency requirements in a SST.

b) as a complimentary method for the objective appraisal and career management of personnel.

c) as a process for the identification of training should it be deemed necessary.

The intention to use the framework for post profiling will effectively detail competency levels required for each post and rank. For example, a SST manager's post profile may consist of the requirement for expert grading in 50% of all competencies, practitioner grading in 30% and supervised practitioner grading in 20%. Conversely, a SST producer's post profile may only require expert grading in 5% of all competencies. Post profiling in all SSTs will ultimately enable the cross transfer of personnel for career development purposes, leading to true experienced-based promotion.

Throughout this dissertation, I have deliberately avoided any detailed discussion of safety and safety-related issues. My reasoning for this is twofold:

a) Safety engineering and the maintenance of safety-related systems are topics of which much complex dialogue already exists.

b) The issue of safety-related software maintenance by RAF SSTs is currently being investigated under a separate task.

SE Management System and QMS appear at first glance to be one and the same. There are, however, slight nuances in their respective definitions that effectively denote a QMS residing *within* a SE Management System. Specifically, a SE Management System is an overarching system concerned primarily with the wider

engineering context, implementing functions and activities as required (amongst these a QMS) in order to produce engineered software.

It must be borne in mind that, although the C&S framework was created to all intents and purposes, objectively, the subsequent assessment exercise has been entirely subjective. Analogies in this instance can be drawn to the job application and interview process – the prospective employee may have the necessary qualifications, but it is the interviewer's job to determine whether or not they are competent to fulfil a particular post. As such, the assessment process would be better served if it were undertaken fully in line with the SW-CMM appraisal guidelines:

> *"- Appraisal team selection, based on SE experience and knowledge of the assessment process.*
>
> *- Scope assessment of appraisal area, mapping SE Functions and Activities to work undertaken.*
>
> *- Identification of key areas for analysis.*
>
> *- Site visit, conducting interviews and applying professional judgement to determine competency levels.*
>
> *- Report production, with recommendations for improvement opportunities.*
>
> *- Histogram preparation, giving overall appraisal area profile."*
>
> after [SW-CMM, 1997]

An extension of the evaluation process could involve the creation of a C&S database in order to aid the assessment, recording and tracking of SE personnel. This would enable post profiling and continual professional development as personnel's experience gaps are identified and jobs and training are allocated. In addition, its use could be extended to the membership of competency schemes potentially enabling the awarding of EA and DA status to RAF SSTs.

In addition, a competency grading alphanumeric based on Function, Activity and Competency Level can be used to mark post requirements. For example, the post C&S requirement for Maintenance and Support Concept could read:

> SysE, M&SC, ARC9(S) ARC22(P) ARC33(P)

which would translate to:

- Systems Engineering
    - Maintenance and Support Concept

- Development of Maintenance and Modification Procedures to Supervised Practitioner level
- Operational Analysis to Practitioner level
- Task Analysis to Practitioner level

A further extension to this dissertation could involve a study into the psychology and management of the RAF SST environment, with the aim of establishing whether or not certain psyches are more suited to specific roles, i.e.

> *"Action oriented roles*
> *Shaper, Implementer (company worker), Completer-finisher*
>
> *People oriented roles*
> *Chairman (co-ordinator), Team worker, Resource Investigator*
>
> *Cerebral roles*
> *Plant, Monitor-evaluator, Specialist"* [Belbin, 2003]

This could lead to the psyche profiling of RAF software maintenance personnel, recording the results alongside their C&S assessment in order to aid team formation and/or sustainment. My examples of role-psyche matching are:

> Requirements – Passive, receptive.
>
> Design – Creative, innovative.
>
> Test – Obstinate, persistent.

This could also be used to compliment the career management process of personnel with the aim of establishing job-employment suitability through the use of trained neutral observers.

8.2 How This Work Should be Viewed

The pedigree of the C&S Framework can be directly traced back to its main constituent elements – the IEE which was created in consultation with UK industry, and the SFIA which has been validated and extended by public and private sector organisations.

The 'Definition of Terms' section was compiled using material from [00-55, 1997], [SW-CMM, 1997] and [IEE, 1999], with additional original input from myself. In particular, I have introduced the expressions of 'Software Engineering Management System' and 'Erroneous Situation' which I believe to be original.

The concept of SE C&S assessment together with the notion of a SE Activity Model is currently under consideration by the RAF, and is ultimately expected to be adopted as policy across all three Services.

In this dissertation I have attempted to adapt current views on SE to the practical environment of military software support solutions. In doing this, I have adapted also some of the accepted definitions and boundaries. It is my belief that this was necessary in order to produce a useable model and framework. That it was at least sufficient has been borne out by its successful trialling at a RAF SST.

8.3 Intelligent Customers

Although this work has been directed at the needs of the Royal Air Force as an intelligent customer, many of the results should be equally applicable to other organisations. For instance, the ability of an organisation to assess the C&S of its personnel will no doubt provide valuable capability evidence for presentation to certification authorities and customers alike. Likewise, considering SE from a SysE perspective will undoubtedly lead to the production of software that is more useable, adaptable and hence, supportable.

The analysis of intelligent customer participation in the SE process also has implications for education and training. Curricula may need to be extended or modified to include, for example, the teaching of support analysis and its application to software.

There are other issues that will require attention – not least is the question of how to assess the C&S assessors – but the simple introduction of the intelligent customer perspective should have considerable benefits in terms of the awareness, acceptance, and management of change.

References

[00-55, 1997] Annex A, Definitions and Abbreviations, 1997. Defence Standard 00-55 (Part 1)/Issue 2, Requirements for Safety Related Software in Defence Equipment, Part 1: Requirements.

[00-60-3, 1998] Annex D, Guidance on the Application of LSA Tasks to Software, 1998. Defence Standard 00-60 (Part 3)/Issue 2, Integrated Logistic Support, Part 3: Guidance for Application, Software Support.

[100B-01, 1997] Order 0121, Responsibilities of Engineering Tradesmen for Management, Supervision and Production, 1997. AP 100B-01, Royal Air Force Engineering Orders and Procedures.

[100D-10, 1998] Leaflet 101, Terms, Definitions and Abbreviations, 1998. AP 100D-10, Support for Mission Software in RAF Systems.

[2237/01, 2003] Gill, D., HSMU Post Visit Report, 17 Oct 2003. ES(Air)(WYT)/SS/2237/01/AD SYS.

[Belbin, 2003] Belbin, R.M., Management Teams: Why they succeed or fail, 2003. Butterworth Heinemann.

[Biedenbender, 1993] Biedenbender, R.E. et al, The ILS Manager's Toolkit: Availability Engineering, 1993. McGraw-Hill Inc.

[Blanchard, 1998] Blanchard, B.S., System Engineering Management, 1998. John Wiley & Sons, Inc.

[Brooks, 2004] Brooks, J., Joint Strike Fighter Concept Phase Logistic Support Analysis For Software, 2004. ES(Air)(WYT)/SS/2333/01/0/06/AD SYS.

[CMM, 1993]. Paulk, M.C. et al, Capability Maturity Model for Software Version 1.1 Technical Report CMU/SEI-93-TR-024 ESC-TR-93-177, 1993. Software Engineering Institute, Carnegie Mellon University.

[CSJ, 2002] Court Service Judgements, Case No: S/01/0219 [2002] EWHC 13 (QB). Web article, http://www.courtservice.gov.uk/View.do?id=0272, The Court Service.

[Gartner, 2003] O'Brien, F. and Park, A.R., Software disposal: Old software never dies, 2003. Web article, http://techrepublic.com.com/5100-6270-5055940.html, TechRepublic.

[Gilb, 1990] Gilb, T., Principles of Software Engineering Management, 1990. Addison-Wesley Publishing Company.

[Gill, 2004] Gill, D., Logistics Support Analysis for Harrier GR9/GR9A/TMK12 Software, 2004. ES(Air)(WYT)/SS/2237/01/AD SYS.

[IEE, 1999] Safety, Competency and Commitment, 1999. Competency Guidelines for Safety-Related System Practitioners. IEE.

[IEEE, 1999] IEEE Standard 610.12-1990, IEEE Standards Software Engineering: Customer and Terminology Standards, 1999. IEEE.

[ISO 9001, 2000] BS EN ISO 9001:2000, Quality management systems – Requirements, 2000. BSI.

[JAP, 2004] Airborne Software and Data Management, 2004. Military Aviation Engineering Policy, Regulation and Documentation, JAP 100A-01 Issue 6.

[McDermid, 1994] McDermid, J.A., Software Engineer's Reference Book, 1994. Butterworth-Heinemann Ltd.

[Naur, 1969] Naur, P. and Randall, B., Software Engineering: A Report on a Conference Sponsored by the NATO Science Committee, 1969. NATO.

[Open, 1997] M880 Software Engineering Module,1997. The Open University.

[Oxford, 2003] Oxford Dictionary of English, 2003. Oxford University Press.

[RTCA, 1992] RTCA/DO-178B, Software Considerations in Airborne Systems and Equipment Certification, 1992.

[SFIA, 2004] The Skills Framework for the Information Age, 2004. The SFIA Foundation.

[Sommerville, 1998] Sommerville, I., Software Engineering, 1998. Addison Wesley Longman Ltd.

[Stebbing. 1986] Stebbing, L., Quality Assurance the Route to Efficiency and Competitiveness, 1986. Ellis Horwood Limited.

[SW-CMM, 1997] Paulk, M.C. et al, The Capability Maturity Model: Guidelines for Improving the Software Process, 1997. Addison Wesley Longman, Inc.

[SWEBOK, 2001] Software Engineering Body of Knowledge, Trial Version 1.00, 2001. IEEE.

[TickIT, 2001] The TickIT Guide, Issue 5.0, 2001. British Standards Institution.

Definition of Terms

Activity	An Activity is a task or related set of tasks that an individual or groups of individuals carry out in order to discharge a responsibility placed on them by an organisation.
Assessment	The judgement of evidence of performance, as specified by a Competency Statement, as to whether or not competency has been achieved.
Competency	The ability to undertake responsibilities and to perform activities with regard to specified standards.
Erroneous Situation	An incorrect or wrong condition that a system or component thereof may be in, or subjected to, that could lead to a fault situation occurring.
Error	A system state, resulting from a fault, human mistake or erroneous situation, that is liable to lead to a failure.
Failure	The inability of a system or component to fulfil its operational requirements. Failures may be systematic or due to physical change.
Fault	*See* Failure.
Function	A set of responsibilities that could be attached to an individual, or group of individuals. In practice a Function is more easily defined by a set of Activities that are required to be carried out in order to discharge the responsibilities attached to a Function.
Goals	A summary of the scope, boundaries and intent that can be used to determine whether an organisation or project has effectively implemented a competency.

Process	The means of achieving something, as distinct from the outcome itself.
Quality Management System	The organisational structure, responsibilities, procedures, activities, capabilities and resources that together aim to ensure that products, processes or services will satisfy stated or implied needs.
Skill	An ability gained by practice, knowledge and understanding. Skill contributes to competence, but on its own lacks the characteristic of “outcome”.
Software Engineering Management System	A designated system that implements the functions and activities necessary to create or maintain engineered software that meets its requirements.
System	A collection of elements organised to accomplish a specific function or set of functions.

Abbreviations

AP	Air Publication
ARC	Activity Related Competency
AWC	Air Warfare Centre
C&S	Competencies and Skills
C/I	Configuration Item
C/T	Chief Technician
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CALS	Computer Aided Logistic Support
CAM	Computer Aided Manufacturing
CASE	Computer Aided Software Engineering
CI	Candidate Item
CO	Commissioned Officer
COTS	Commercial Off The Shelf
Cpl	Corporal
CS	Core Skill
CSP	Communicating Sequential Process
DA	Design Authority
DBMS	Database Management System
Def Stan	Defence Standard
DITMC	Defence IT Management Centre
DLO	Defence Logistics Organisation
DVM	Digital Volt Meter
EA	Engineering Authority
Eng Tech Av	Engineering Technician Avionics
ESDB	Equipment Specific Database
EW	Electronic Warfare
F/L	Flight Lieutenant
Flt	Flight
FRC	Function Related Competency
FS	Flight Sergeant
HOL	High Order Language
HQSTC	Headquarters Strike Command
HSMU	Harrier Software Maintenance Unit
ICT	Information and Communications Technologies
ILS	Integrated Logistic Support
IPT	Integrated Project Team
IS	Information Systems
ISADS	In Service Aviation Design Support
KA	Knowledge Areas
KPA	Key Process Area
LRI	Line Replaceable Item
LSA	Logistics Support Analysis

OC	Officer Commanding
OEU	Operational Evaluation Unit
OFP	Operational Flight Program
OJT	On the Job Training
Ops	Operations
OR	Other Ranks
PC	Personal Computer
PET	Pre Employment Training
PFM	Pre Flight Message
POE	Prime Operating Equipment
PSE	Project Support Environment
QA	Quality Assurance
QAM	Quality Assurance Manager
QMS	Quality Management System
RAF	Royal Air Force
RFC	Request for Change
s/e	successfully and efficiently
S/L	Squadron Leader
SA	Supporting Activities
SCMB	Software Configuration Management Board
SCR	Software Change Request
SD	Software Development
SE	Software Engineering
SEM	Service Engineered Modification
SEP	Software Engineering Programme
Sgt	Sergeant
SIM	Support Infrastructure Management
SM	Software Modification
SNCO	Senior Non-Commissioned Officer
SOS	Software Operations Support
SP	Software Platform
SQA	Software Quality Assurance
SSSI	Software Support Significant Item
SST	Software Support Team
STC	Strike Command
Stim	Stimulator
SWEBOK	Software Engineering Body Of Knowledge
SysE	System Engineering
TISMT	Tornado In-service Software Maintenance Team
TOR	Terms Of Reference
TPM	Technical Performance Measure
TS	Technical Skill
UML	Unified Modelling Language
UOR	Urgent Operational Requirement

Appendix A – SE Topics

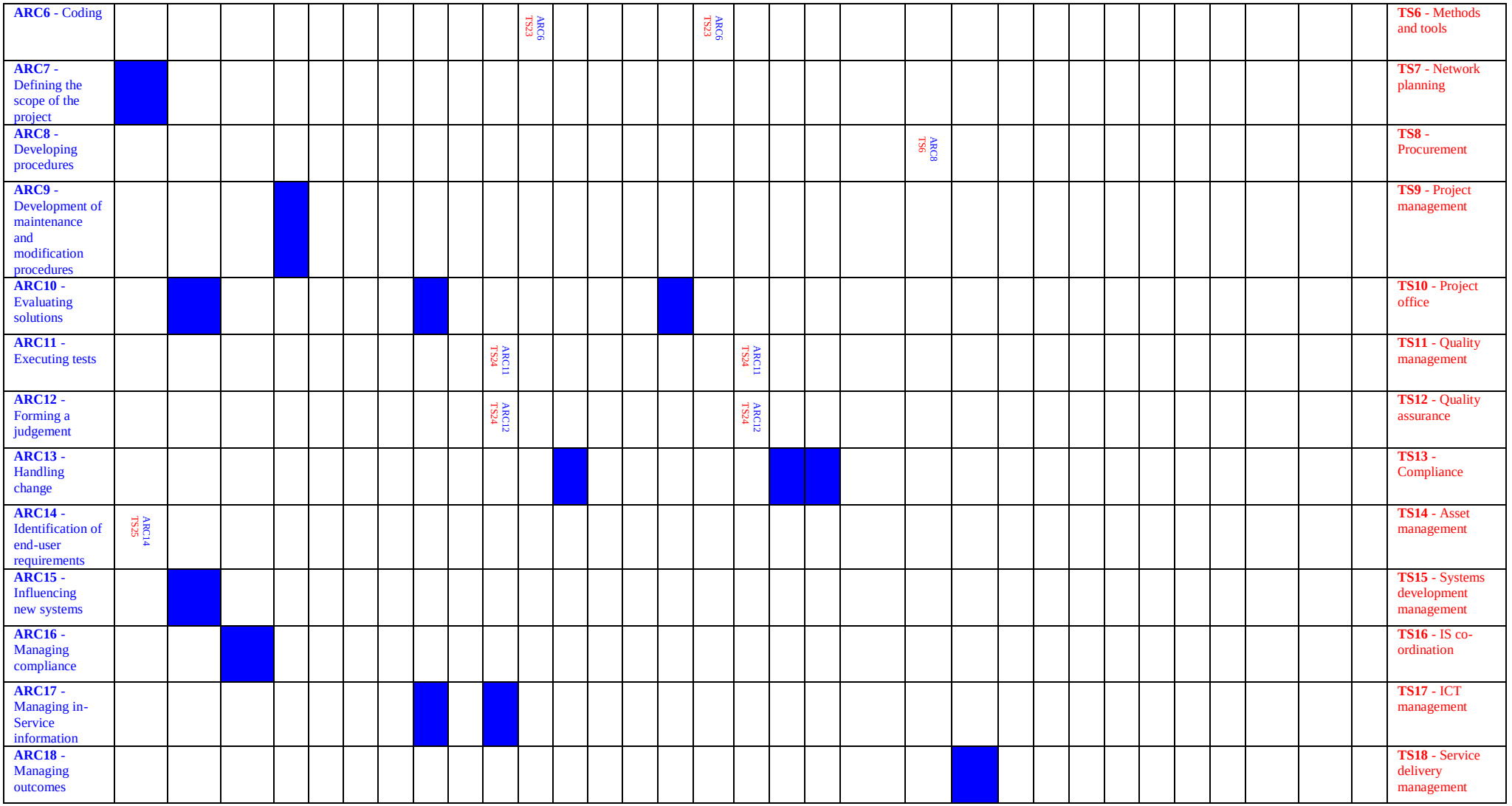
Software – The Process and its Management	The Nature of Software	The evolution of the role of software
		Problems in developing software
		Types of software system
	Software Development Life-Cycles	An introduction to life-cycles and life-cycle models
		Practical life-cycle models
		Object-oriented and package-based life-cycle models
	Planning and Management	The need for project management
		Project initiation
		Planning - estimation
		Planning – work breakdown and scheduling
		Tracking and control
System and Software Requirements Analysis	Requirements Specification	Introduction
		Requirements specification
		The structured approach
		The object-oriented approach
		The formal methods approach
		Choosing an approach to requirements analysis
The Design and Implementation of Software	Software Design	The nature of software design
		Four aspects of software design
		The structured approach
		The object-oriented approach
		Towards coding
	Reuse	Introduction
		Incorporating reuse into software development
		Reuse through objects and classes
		Making reuse happen
Ensuring, Verifying and Maintaining Software Integrity	Software Quality Management	Introduction
		What is software quality?
		Quality management systems
		The ISO 9000 family of standards
		Processes and process improvement
	Testing	Introduction
		The management of testing
		Structural testing
		Functional testing
		Testing tools
		Bringing it all together
	Maintenance	Introduction
		Maintainability and the control of maintenance costs
		The maintenance process and the organisation
		Preventive maintenance and legacy systems
	Human Factors	Introduction
		Classifying human factors
		Person-related factors
		Task-related factors
		Environment-related factors
		Automation
	Professional Issues	Being professional
		Ethical decision-making

Overall Process		Legal issues
		Professional bodies
	Software Process Review	Software engineering: meeting the users' needs
		Reconsidering the software development life-cycle
		Establishing an initial set of requirements
		Making the requirements and delivered system converge
		Acceptance testing and installation
		Summary
		Further reading

Table A.1 – Software Engineering Topics [Open, 1997]

Appendix B – C&S Mapping to SE

SE Functions	System Engineering													Software Modification					Supporting Activities				Support Infrastructure Management										
SE Activities / Competencies	Definition of problem (identification of need)/Query Evaluation	System feasibility analysis	System operational requirements	Maintenance and support concept	Identification/prioritisation of technical performance meas ures (TPMs)	Functional analysis	Requirements allocation	System synthesis, analysis and design optimisation	Design integration	System test and evaluation (validation)	Construction and/or production	System operational use and life -cycle support	System retirement and material disposal	Software Requirements	Software Design	Software Construction	Software Testing/Qualification and Certification	Software Maintenance	Change Management	Software Engineering Management	Software Engineering Tools and Methods Management	Software Quality Assurance	Software Platform/Prime Operating Equipment	Operating Software	Users	Software Operations	Technical Training	Test and Support Equipment	Supply Support	Maintenance Facilities	Technical Data	Transportation and Handling Equipment	Skills
ARC1 - Allocation of responsibilities																																**TS1** - Information resource management
ARC2 - Analysing test results										ARC2 TS24							ARC2 TS24																**TS2** - Consultancy
ARC3 - Analysing the code										ARC3 TS24							ARC3 TS24																**TS3** - IS strategy and planning
ARC4 - Analysing the design										ARC4 TS24							ARC4 TS24																**TS4** - Business risk management
ARC5 - Assuring staff competency																																	**TS5** - Change control


ARC6 - Coding
ARC7 - Defining the scope of the project
ARC8 - Developing procedures
ARC9 - Development of maintenance and modification procedures
ARC10 - Evaluating solutions
ARC11 - Executing tests
ARC12 - Forming a judgement
ARC13 - Handling change
ARC14 - Identification of end-user requirements
ARC15 - Influencing new systems
ARC16 - Managing compliance
ARC17 - Managing in-Service information
ARC18 - Managing outcomes
ARC6 TS23
ARC6 TS23
ARC8 TS6
ARC11 TS24
ARC11 TS24
ARC12 TS24
ARC12 TS24
ARC14 TS25
TS6 - Methods and tools
TS7 - Network planning
TS8 - Procurement
TS9 - Project management
TS10 - Project office
TS11 - Quality management
TS12 - Quality assurance
TS13 - Compliance
TS14 - Asset management
TS15 - Systems development management
TS16 - IS co-ordination
TS17 - ICT management
TS18 - Service delivery management

ARC19 - Managing resource allocation																																	**TS19** - Marketing
ARC20 - Monitoring compliance																						ARC20 TS13											**TS20** - Technical authority
ARC21 - Monitoring the engineering development																						ARC21 TS12						TS21			TS21		**TS21** - Systems design
ARC22 - Operational analysis																												TS22			TS22		**TS22** - Database design
ARC23 - Performing analysis										ARC23 TS24							ARC23 TS24											TS23			TS23		**TS23** - Programming/ software development
ARC24 - Planning																												TS24			TS24		**TS24** - Systems testing
ARC25 - Producing assessment reports																																	**TS25** - Systems ergonomics
ARC26 - Promoting awareness																																	**TS26** - Media creation
ARC27 - Regulatory and legal compliance																						ARC27 TS11											**TS27** - Systems integration
ARC28 - Resource allocation																				ARC28 TS1 TS15 TS18													**TS28** - Systems installation/ decommissioning
ARC29 - Risk assessment																				ARC29 TS4													**TS29** - Education and training management
ARC30 - Scope and context appreciation																																	**TS30** - Development and training
ARC31 - Specifying software tests										ARC31 TS24							ARC31 TS24																**TS31** - Training materials creation

ARC32 - Specifying tests										ARC32 TS24							ARC32 TS24																			**TS32** - Education and training delivery
ARC33 - Task analysis																																				**TS33** - Configuration management
ARC34 - Transposing from requirements into design							ARC34 TS21 TS22	ARC34 TS21 TS22	ARC34 TS21 TS22						ARC34 TS21 TS22																					**TS34** - Network control
ARC35 - Witnessing and executing tests										ARC35 TS24							ARC35 TS24																			**TS35** - Capacity management
NC1 – Query Evaluation	NC1 TS37																																			**TS36** - Security administration
NC2 – Helpdesk Administration	NC2 TS42																							NC2 TS42	NC2 TS42		NC2 TS42			NC2 TS42					**TS37** - Application and system support
NC3 – Product Dissemination											NC3 TS26												NC3 TS26												**TS38** - ICT operations
NC4 - Construction											NC4 TS27				NC4 TS27												NC4 TS27			NC4 TS27				**TS39** - Database administration
NC5 – Installation and De-commissioning												NC5 TS28	NC5 TS28														NC5 TS28			NC5 TS28				**TS40** - Service level control
NC6 – Database Support												NC6 TS39															NC6 TS39			NC6 TS39				**TS41** - Network administration and support
NC7 – Change Management																		NC7 TS5				NC7 TS5												**TS42** - User support
NC8 – Configuration Control																		NC8 TS33				NC8 TS33												
NC9 – SE Project Management																			NC9 TS9																

NC10 – Service Level Management																				NC10 TS40												
NC11 – Project Management Support																					NC11 TS10											
NC12 - Salesmanship																								NC12 TS19								
NC13 – SE Training Management																										NC13 TS29						
NC14 – Training Needs Analysis																										NC14 TS30						
NC15 – Course Design																										NC15 TS31						
NC16 - Instructing																										NC16 TS32						
NC17 – Professional Advice																											NC17 TS2			NC17 TS2		
NC18 – IS Strategy Management																											NC18 TS3			NC18 TS3		
NC19 – Network Management																											NC19 TS7			NC19 TS7		
NC20 – IS/ICT Procurement																											NC20 TS8			NC20 TS8		
NC21 – SE Asset Management																											NC21 TS14	NC21 TS14		NC21 TS14	NC21 TS14	
NC22 – IS Harmonisation																											NC22 TS16				NC22 TS16	
NC23 – SE Support System Management																											NC23 TS17			NC23 TS17		

NC24 – Technical Direction																												NC24 TS20			NC24 TS20		
NC25 – SE Support System Control																												NC25 TS34		NC25 TS34	NC25 TS34		
NC26 – Capability Provision																												NC26 TS35		NC26 TS35	NC26 TS35		
NC27 – IS Security Management																												NC27 TS36		NC27 TS36	NC27 TS36		
NC28 – ICT Administration																												NC28 TS38		NC28 TS38	NC28 TS38		
NC29 – Network Administration																												NC29 TS41		NC29 TS41	NC29 TS41		
NC30 – System Structural Analysis						NC30																											
NC31 – Software Requirements Specification														NC31																			
NC32 – Logistics Support Analysis				NC32	NC32																												
FRC1 - Application domain knowledge	CS11 CS12	FRC1 CS9 CS10																															**CS1** - Business process improvement
FRC2 - Attention to accuracy and detail	CS11 CS12	FRC2 CS9 CS10																															**CS2** - Technical specialism
FRC3 - Business objectives	CS10 CS11 CS12	CS6 CS7 CS9	FRC3 CS1 CS5																														**CS3** - Systems architecture
FRC4 - Clarity	CS11 CS12	FRC4 CS9 CS10																															**CS4** - Emerging technology monitoring

FRC5 - Conceptual thinking and open-mindedness	CS11 CS12	FRC5 CS9 CS10																														**CS5** - Contract management
FRC6 - Decision making	CS11 CS12	FRC6 CS9 CS10																														**CS6** - Programme management
FRC7 - Effective communication	CS11 CS12	FRC7 CS9 CS10																														**CS7** - Business analysis
FRC8 - Effective working relationships	CS11 CS12	FRC8 CS9 CS10																														**CS8** - Data analysis
FRC9 - Eliciting information	CS11 CS12	FRC9 CS9 CS10																														**CS9** – Autonomy
FRC10 - Influencing and Negotiating	CS11 CS12	FRC10 CS9 CS10																														**CS10** – Influence
FRC11 - Methodical approach	CS11 CS12	FRC11 CS9 CS10																														**CS11** – Complexity
FRC12 - Multi-discipline systems viewpoint	CS11 CS12	FRC12 CS9 CS10																														**CS12** – Business Skills
FRC13 - Openness	CS11 CS12	FRC13 CS9 CS10																														
FRC14 - Organisation systems	CS11 CS12	FRC14 CS9 CS10																														
FRC15 - Professional standing and personal integrity	CS11 CS12	FRC15 CS9 CS10																														
FRC16 - Regulatory and legal compliance	CS11 CS12	FRC16 CS9 CS10																														

FRC17 - Report writing	CS11 CS12	FRC17 CS9 CS10																														
FRC18 - Systematic approach	CS11 CS12	FRC18 CS9 CS10																														
FRC19 - Systems viewpoint	CS10 CS11 CS12	FRC19 CS3 CS9																														
FRC20 - Team management	CS11 CS12	FRC20 CS9 CS10																														
FRC21 - Team-working	CS11 CS12	FRC21 CS9 CS10																														
FRC22 - Technology	CS9 CS10 CS11 CS12	FRC22 CS2 CS4																														
FRC23 - Test and analysis methods and techniques	CS10 CS11 CS12	FRC23 CS8 CS9																														

Table B.1 – C&S Mapping to SE Functions and Activities

Key:
FRC - Function Related Competency
ARC - Activity Related Competency
TS – Technical Skill
CS – Core Skill
NC – New Competency

Appendix C – SFIA Skills and Levels

SFIA summary: skills and levels

Category	Subcategory	Skill	1	2	3	4	5	6	7
Strategy & planning	Information management	Information resource management							
	Advice & guidance	Consultancy							
		Technical specialism							
	Business/IS strategy and planning	Business process improvement							
		IS strategy & planning							
		Business risk management							
	Technical strategy and planning	Systems architecture							
		Change control							
		Business continuity planning							
		Emerging technology monitoring							
		Methods and tools							
		Network planning							
Management & administration	Supply management	Contract management							
		Procurement							
	Project management	Programme management							
		Project management							
		Project office							
	Quality management	Quality management							
		Quality assurance							
		Compliance							
	Resource management	Asset management							
		Systems development management							
		IS co-ordination							
		ICT management							
		Service delivery management							
Sales & marketing	Sales and marketing	Account management							
		Marketing							
		Selling							
		Sales support							
Development & implementation	Systems development	Business analysis							
		Data analysis							
		Technical authority							
		Systems design							
		Database design							
		Programming/software development							
		Systems testing							
	Human factors	Systems ergonomics							
		Media creation							
	Installation & integration	Systems integration							
		Systems installation/decommissioning							
Service delivery	Education and training	Education & training management							
		Development & training							
		Training materials creation							
		Education & training delivery							
	Infrastructure	Configuration management							
		Network control							
		Capacity management							
		Security administration							
	Operation	Application & system support							
		ICT operations							
		Database administration							
		Service level control							
	User support	Network administration & support							
		User support							

Appendix D – SFIA Level Definitions

The Skills Framework for the Information Age (SFIA) is a matrix of levels of responsibility and accountability on one axis, and areas of work on the other. The standard levels of responsibility and accountability are described here.

Level 1: *follow*

Autonomy
Works under close supervision. Uses little discretion. Expected to seek guidance in unexpected situations.

Influence
Interacts with department.

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

Business skills
Uses basic IS functions, applications, and processes. Demonstrates an organized approach to work. Capable of learning new skills and applying newly acquired knowledge. Basic oral and written communication skills. Contributes to identifying own development opportunities.

Level 2: assist

Autonomy
Works under routine supervision. Uses minor discretion in resolving problems or enquiries. Works without frequent reference to others.

Influence
Interacts with and may influence department. May have some external contact with customers and suppliers. May have more influence in own domain.

Complexity
Performs range of varied work activities in variety of structured environments.

Business skills
Understands and uses appropriate methods tools and applications. Demonstrates a rational and organised approach to work. Awareness of health and safety issues. Identifies and negotiates own development opportunities. Sufficient communication skills for effective dialogue with colleagues. Able to work in a team. Able to plan, schedule and monitor own work within short time horizons. Can absorb technical information when it is presented systematically and apply it effectively.

Level 3: apply

Autonomy
Works under general supervision. Uses discretion in identifying and resolving complex problems and assignments. Specific instruction is usually given and work is reviewed at frequent milestones. Determines when problems should be escalated to a higher level.

Influence

Interacts with and influences department/project team members. Frequent external contact with customers and suppliers. In predictable and structured areas may supervise others. Decisions may impact work assigned to individual/phases of project.

Complexity
Broad range of work, sometimes complex and non routine, in variety of environments.

Business skills
Understands and uses appropriate methods tools and applications. Demonstrates analytical and systematic approach to problem solving. Takes initiative in identifying and negotiating appropriate development opportunities. Demonstrates effective communication skills. Contributes fully to the work of teams. Can plan, schedule and monitor own work (and that of others where applicable) competently within limited time horizons and according to health and safety procedures. Is able to absorb and apply new technical information. Is able to work to required standards and to understand and use the appropriate methods, tools and applications. Appreciates wider field of IS, how own role relates to other IS roles and to the business of the employer or client.

Level 4: enable

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

Influence
Influences team, and specialist peers internally. Influences customers at account level and suppliers. Some responsibility for work of others and allocation of resources. Participates in external activities related to specialisation. Decisions influence success of projects and team objectives

Complexity
Broad range of complex technical or professional work activities, in a variety of contexts.

Business skills
Selects appropriately from applicable standards, methods, tools and applications and use. Demonstrates analytical and systematic approach to problem solving. Communicates fluently orally and in writing and can present complex technical information to both technical and non-technical audiences. Is able to plan, schedule and monitor work activities in order to meet time and quality targets and in accordance with health and safety procedures. Is able to absorb rapidly new technical information and apply it effectively. Good appreciation of wider field of IS, how IS is used in relevant employment areas and how IS relates to the business activities of the employer or client. Maintains awareness of developing technologies and their application and takes some responsibility for personal development.

Level 5: ensure, advise

Autonomy
Works under broad direction. Full accountability for own technical work or project/supervisory responsibilities. Receives assignments in the form of objectives. Establishes own milestones, team objectives and delegates assignments. Work is often self-initiated.

Influence
Influences organisation, customers, suppliers and peers within industry on contribution of specialisation. Significant responsibility for the work of others and for the allocation of resources. Decisions impact on success of assigned projects i.e. results, deadlines and budget. Develops business relationships with customers.

Complexity
Challenging range – variety of complex technical or professional work activities. Work requires application of fundamental principles in a wide and often unpredictable range of contexts. Understands relationship between specialism and wider customer/organisational requirements.

Business skills
Advises on the available standards, methods, tools and applications in own area of specialisation and can make correct choices from alternatives. Can analyse, diagnose, design, plan, execute and evaluate work to time, cost and quality targets. Communicates effectively, formally and informally, with colleagues, subordinates and customers. Demonstrates leadership. Clear understanding of the relationship between own area of responsibility/specialisation to the employing organisation and takes customer requirements into account when making proposals. Takes initiative to keep skills up to date. Maintains awareness of developments in the industry. Can analyse user requirements and advise users on scope and options for operational improvement. Demonstrates creativity and innovation in applying IS solutions for the benefit of the user.

Level 6: initiate, influence

Autonomy
Has defined authority and responsibility for a significant area of IS work, including technical, financial and quality aspects. Establishes organisational objectives and delegates assignments. Accountable for actions and decisions taken by self and subordinates.

Influence
Influences policy formation on contribution of specialisation to business objectives. Influences at level of division internally and influences customer/suppliers and industry at senior management level. Decisions impact IS work of employing organisations, achievement of organisational objectives and financial performance. Develops high-level relationships with customers suppliers and industry leaders.

Complexity
Highly complex work activities covering technical, financial and quality aspects and contributing to formulation of IS strategy. Work involves creative application of wide range of technical and/or management principles.

Business skills
Can absorb complex technical information and communicate effectively at all levels to both technical and non-technical audiences. Is able to assess and evaluate risk and to understand the implications of new technologies. Demonstrates clear leadership skills and the ability to influence and persuade. Has a broad understanding of all aspects of IS and deep understanding of area(s) of specialisation. Understands and communicates the role and impact of IS in the employing organisation. Takes initiative to keep both own and subordinates skills up to date and to maintain awareness of developments in the IS industry.

Level 7: set strategy, inspire, mobilise

Autonomy
Has authority and responsibility for all aspects of a significant area of IS work, including policy formation and application. Is held fully accountable for actions taken and decisions made, both by self and subordinates.

Influence

Decisions critical to organizational success. Influences developments within IS industry at highest levels. Advances exploitation of IS within one or more organisations and/or the advancement of IS knowledge. Develops long-term strategic relationships with customers and industry leaders.

Complexity

Leads on formulation and application of IS strategy. Work involves application of highest level management and leadership skills. Has deep understanding of IS industry and emerging technologies and implications for the wider
business environment.

Business skills

Full range of strategic management and leadership skills. Understands, explains and presents complex technical ideas to both technical and non-technical audiences at all levels up to the highest in a persuasive and convincing manner. Has a broad and deep IS knowledge coupled with equivalent knowledge of the activities of those businesses and other organisations who use and exploit IS. Is able to understand and communicate the potential impact of emerging technologies on organisations and individuals and can analyse the risks of using or not using such technologies. Takes initiative to keep both own and subordinates skills up to date and to maintain awareness of and, in own area(s) of expertise.

Appendix E – C&S Assessment Tables

E.1 Systems Engineering Function

Summary

The Systems Engineering function is an iterative, repeatable pr ocess that comprises 13 separate activities which essentially form the core framework of the SE process. This function is linked to the support infrastructure management and software modification function via related enabling activities. Key activities w ithin this function are:

- Definition of Problem/Query Evaluation
- System Feasibility Analysis
- System Operational Requirements
- Maintenance and Support Concept
- Identification and Prioritisation of Technical Performance Measures
- Functional Analysis
- Requirements Allocation
- System Synthesis, Analysis and Design Optimisation
- Design Integration
- System Test and Evaluation
- Construction
- System Operation Use and Life-Cycle Support
- System Retirement and Material Disposal

Systems Engineering Function Related Competencies

<table>
<tr><th colspan="3">FRC1 Application Domain Knowledge</th></tr>
<tr><td colspan="3">a) Considers the process, external equipment, the operating environment, maintenance activities and other human interactions, etc., associated with the system throughout the performance of analysis activities.
b) Applies knowledge of the application domain of the system including processes, modes of operation, and human interaction, to requirements specification activities.
c) Addresses the overall process, equipment, operating environment, human interactions, and environmental aspects etc., associated with a system.
d) Addresses the process, the equipment under control, the operation and maintenance environment, human interactions etc., associated with a system.
e) Considers the process, equipment, operating environment, human interactions, etc., associated with a system throughout the performance of system software construction activities.</td></tr>
<tr><th>Producer</th><th>Supervisor</th><th>Manager</th></tr>
<tr><td>a) & c) Has had practical work experience within the relevant industry sector and with the relevant system applications. Can explain the reasons why analyses are performed and their place in a SE management system.

b) Has done practical work on similar applications within the relevant industry sector. Can describe the key requirements for the system.

d) & e) Has worked within the relevant industry sector and with the relevant system applications. Knows the key issues relating to the environment in which systems are required to operate, their key modes of operation and typical architectural design solutions.</td><td>a) & c) Can illustrate, through memos and meeting notes, how domain specific requirements have been addressed during analysis and validation exercises for a system. Can explain the reasons why analyses and validation exercises are performed and their place in the SE management system.

d) & e) Has consistently reflected domain specific requirements in system architectural and software design solutions.</td><td>a) Can explain, how consideration of application domain specific issues is key to successful assessment performance. Can illustrate, through analysis procedures, training course notes, how actions have been taken to ensure that application domain specific requirements are adequately considered during analysis activities.

c) Can explain, through examples (real or hypothetical), how potentially erroneous situations have arisen as a result of insufficient consideration of application domain specific issues.

c) Can illustrate, through validation procedures and training course notes, how actions have been taken to ensure that application domain specific requirements are adequately considered during validation activities.

d) Can explain the importance of domain specific requirements in designing system architectures. Can identify, through regular design reviews and monitoring, potentially erroneous situations which could arise as a result of insufficient understanding of the application domain.

e) Assists others, through training courses and mentoring, to appreciate the importance of domain specific requirements in designing software. Can identify, through regular design reviews and monitoring, potentially erroneous situations which could arise as a result of insufficient understanding of the application domain.</td></tr>
<tr><td></td><td colspan="2">b) Has written requirements specifications, and can illustrate the key requirements for a system within the domain. Can explain how the system impacts upon the wider environment, including operation and maintenance.</td></tr>
<tr><td></td><td colspan="2">a), c), d), & e) Is familiar with the history of the development of the SE philosophy and standards for the domain and the way in which previous projects have influenced that development.</td></tr>
<tr><td colspan="3">c) Can identify and can describe the main functional components that make up a system developed, operated or maintained by the organisation. Can identify the main modes of operation of the system and its key functions.</td></tr>
</table>

Core Skills **CS9 Autonomy** Works under routine supervision. Uses minor discretion in resolving problems or enquiries. Works without frequent reference to others. **CS10 Influence** Interacts with and may influence department. May have some external contact with customers and suppliers. May have more influence in own domain. **CS11 Complexity** Performs range of varied work activities in variety of structured environments. **CS12 Business Skills** Understands and uses appropriate methods tools and applications. Demonstrates a rational and organised approach to work. Awareness of health and safety issues. Identifies and negotiates own development opportunities. Sufficient communication skills for effective dialogue with colleagues. Able to work in a team. Able to plan, schedule and monitor own work within short time horizons. Can absorb technical information when it is presented systematically and apply it effectively.	**Core Skills** **CS9 Autonomy** Works under general direction within a clear framework of accountability. Substantial personal responsibility and autonomy. Plans own work, to meet given objectives and processes **CS10 Influence** Influences team, and specialist peers internally. Influences customers at account level and suppliers. Some responsibility for work of others and allocation of resources. Participates in external activities related to specialisation. Decisions influence success of projects and team objectives. **CS11 Complexity** Broad range of complex technical or professional work activities, in a variety of contexts. **CS12 Business Skills** Selects appropriately from applicable standards, methods, tools and applications and use. Demonstrates analytical and systematic approach to problem solving. Communicates fluently orally and in writing and can present complex technical information to both technical and non-technical audiences. Is able to plan, schedule and monitor work activities in order to meet time and quality targets and in accordance with health and safety procedures. Is able to absorb rapidly new technical information and apply it effectively. Good appreciation of wider field of SE, how SE is used in relevant employment areas and how SE relates to the business activities of the employer or client. Maintains awareness of developing technologies and their application and takes some responsibility for personal development.	**Core Skills** **CS9 Autonomy** Has authority and responsibility for all aspects of a significant area of SE work, including policy formation and application. Is held fully accountable for actions taken and decisions made, both by self and subordinates. **CS10 Influence** Decisions critical to organizational success. Influences developments within SE industry at highest levels. Advances exploitation of SE within one or more organisations and/or the advancement of SE knowledge. Develops long-term strategic relationships with customers and industry leaders. **CS11 Complexity** Leads on formulation and application of SE strategy. Work involves application of highest level management and leadership skills. Has deep understanding of SE industry and emerging technologies and implications for the wider business environment. **CS12 Business Skills** Full range of strategic management and leadership skills. Understands, explains and presents complex technical ideas to both technical and non-technical audiences at all levels up to the highest in a persuasive and convincing manner. Has a broad and deep SE knowledge coupled with equivalent knowledge of the activities of those businesses and other organisations who use and exploit SE. Is able to understand and communicate the potential impact of emerging technologies on organisations and individuals and can analyse the risks of using or not using such technologies. Takes initiative to keep both own and subordinates skills up to date and to maintain awareness of and, in own area(s) of expertise.

FRC2 Attention to Accuracy and Detail		
Recognises incomplete, inaccurate and misleading specifications and reports and can devise tests which exploit such deficiencies.		
Producer	**Supervisor**	**Manager**
Has successfully performed work requiring a high degree of accuracy and attention to detail to complete.	*Has reviewed, with consistent accuracy, system documentation as part of validation activities.*	*Can describe examples where insufficient accuracy or attention to detail in system documentation has led to uncertainty with regard to the validation of a system .* *Can illustrate, through validation plans, how actions are taken to resolve such uncertainties.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC3 Business Objectives		
Reflects an organisation's business objectives in the system or services procurement process.		
Producer	**Supervisor**	**Manager**
Can describe the key business objectives of the organisation and how functional assurance impacts on these business objectives.	*Can cite examples where issues of functional assurance have had an impact upon the organisation's business objectives. Can explain the actions that have been taken to ensure that functional assurance is seen within projects as a factor which serves the organisation's business objectives.* **Core Skills** **CS1 Business Process Improvement** Analyses business processes; identifies alternative solutions, assesses feasibility, and recommends new approaches, where the changes include major SE-related components. Evaluates the financial, cultural, technological, organizational and environmental factors which must be addressed in the change programme. Establishes requirements for the implementation of significant changes in organizational mission, business functions and process, organizational roles and responsibilities, and scope or nature of service delivery.	
Core Skills See CS9, CS10, CS11, CS12.	**CS5 Contract Management** Monitors contracts to ensure service level agreements are met, collects information about contract performance and reports to management on the performance of supplier. Investigates and acts to correct problems in contract delivery. Liases between supplier and users and arbitrates as necessary.	**CS5 Contract Management** Initiates development and ensures proper management of contracts with suppliers to meet key performance indicators and agreed targets. Develops mutual understanding between the organization and suppliers and a commitment to improving contract performance to bring added value to the organization through high level relationships with suppliers. Ensures contracts are properly monitored and uses the information to review and correct the performance of suppliers. Advises on organization procedures for monitoring, establishing service level agreements and evaluating the performance of suppliers. Influences policy and procedures covering the approval of suppliers, conduct of tendering process and purchasing procedures. Leads major service review meetings, enforces or renegotiates contracts as required. Is the senior arbiter within the demand-side organization through which users can ensure problems with suppliers are resolved.

	CS7 Business Analysis Creates requirements specification and business case for development of SE solutions by investigating business processes and business needs. Finds out whether and how an existing system supports business processes and recommends incremental improvements in both system and process. See CS9, CS10, CS11, CS12.	**CS6 Programme Management** Sets the business objectives for SE activities and authorizes the selection and planning of all projects and activities. Plans, directs, and co-ordinates activities to manage and implement complex interrelated projects from contract/proposal initiation to final operational stage; plans, schedules, monitors, and reports on activities related to the programme; leads the programme team(s) in determining customer requirements and translating requirements into operational plans; determines, monitors, and reviews all programme economics to include programme costs, operational budgets, staffing requirements, programme resources, and programme risk. Ensures that programme is managed to maximize business benefits and monitors and evaluates changes to programme management practices and initiates improvement to organization practices. **CS7 Business Analysis** Initiates and influences enterprisewide business process analysis. Proposes and champions SE solutions to emerging opportunities. Acts as consultant to team and customers on integrating business needs and SE solutions. See CS9, CS10, CS11, CS12.

FRC4 Clarity		
Produces clear and precise specifications and can present requirements clearly during discussions.		
Producer	**Supervisor**	**Manager**
Has written a clear specification.	*Has written software requirements specifications and can explain why the requirements are sufficiently clear and not open to misinterpretation.*	
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC5 Conceptual Thinking and Open-Mindedness		
Presents requirements without bias towards particular design solutions and is open to radical technologies and architectural design solutions, additional and modified requirements, conceptualising about their effect on the integrity of the system in its environment.		
Producer	**Supervisor**	**Manager**
Is aware of current and new technological developments in the field of architectural design and potentially different ways of designing systems. Presents requirements which allow different design solutions.	*Has produced requirements specifications which allow radical designs. Has incorporated new technology in the architectural design for a system. Is open to different design solutions and understands the importance of innovation for performance.*	*Can show how the organisation ensures that requirements specifications are free from implementation bias, and how changes are incorporated. Can show how new design solutions have been encouraged where appropriate. Evaluates the impact of technological advances in the field of architectural design, incorporating the findings within organisation awareness programmes and procedures.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC6 Decision Making		
Uncovers the key facts associated with a situation and communicates a firm, rational decision based on an analysis of those key facts.		
Producer	**Supervisor**	**Manager**
Given a set of information regarding a hypothetical si tuation, can identify the key facts and proposes a decision that relates to the identified key facts.		*Can cite examples from his/her own experience where it has been necessary to make difficult decisions relating to the functional assurance of a system, an d can illustrate, via memos, letters, reports, witness testimonies, how the key facts were uncovered and how decisions were taken and communicated.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC7 Effective Communication		
Communicates effectively, both orally in writing and electronically at all levels in and outside an organisation, with people of varying skill and groups of varying size, such that the objectives for the communicat ion are achieved.		
Producer	**Supervisor**	**Manager**
Understands the principles of good presentation. Communicates well with peers.	*Has made successful formal presentations. Communicates well in a team and in one-to-one situations at most levels within an organisation.*	*Is acknowledged as proficient in communicating information orally in all situations. Is able to communicate key requirements to potential suppliers and can liase effectively with both maintenance and modification staff such that their issues ar e represented fairly to project team members and management. Has established effective liaison with the organisation's management such that any issues are raised at the highest level. Has effective relationships with relevant external organisations, such as regulatory bodies.*
Understands the principles of clear report writing. Has written at least one report which can demonstrate basic literacy skills and the ability to present written information in an organised, logical and unambiguous manner.	*Has consistently produced written work of a quality which is well organised, accurate (both technically and grammatically), complete, logical, concise, unambiguous and to the point.*	*Has consistently produced written work of a quality which is well organised, accu rate (both technically and grammatically), complete, logical, concise, unambiguous and to the point. Is aware of the wider implications and purpose of communications.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC8 Effective Working Relationships		
Develops and maintains effective working relationships with other members of the project including: ▪ software engineers and managers within the supplier's organisation ▪ personnel within suppliers to the organisation of systems or services ▪ personnel within the purchaser's organisation and other organisations, e.g. operators and maintainers, independent qualifiers and certifiers and regulatory authorities.		
Producer	**Supervisor**	**Manager**
Has worked as an effective member of a project team coordinating own activities with those of peers and reporting to a supervisor.	*Has worked as an effective member of a project team coordinating the activities of a group of individuals and reporting to a project manager within his/her own organisation.*	*Has worked as an effective leader of a project team coordinating the activities of more than one organisation and reporting directly to the project stakeholders.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC9 Eliciting Information		
Proactively elicits all necessary information from relevant personnel at whatever level (e.g. stakeholders, peers, etc.) such that the tasks associated with the role can be properly scoped and undertaken.		
Producer	**Supervisor**	**Manager**
Collects the relevant facts about tasks from peers.	*Collects and understands the relevant information from personnel at all levels. Can identify more important issues from a wider range of points.*	*Has established a mechanism for the collection of information across the whole organisation.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC10 Influencing and Negotiating		
Convincingly argues a point of view or position and obtains buy -in from personnel at all levels of the organisation and is able to compromise on detail, if necessary, whilst still achieving the key objectives of the quality assurance plan.		
Producer	**Supervisor**	**Manager**
Understands the principles of negotiation and has taken part in practical training exercises in influencing/negotiating.	*Can cite examples from his/her own experience where it has been necessary to exert influence to satisfactorily resolve a situation relating to the quality assurance of a system.*	*Can cite examples from his/her own experience where it has been necessary to exert influence to satisfactorily resolve a situation relating to the quality assurance of a system and c an illustrate, via memos, letters, witness testimonies, how negotiations were brought to a satisfactory conclusion.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC11 Methodical Approach		
Works in a methodical, clearly structured manner, applying a methodical approach to assignments that incorporates analytical and systematic techniques as appropriate.		
Producer	**Supervisor**	**Manager**
For specific tasks undertaken, can explain the method followed in performing the tasks and indicates the resulting structure of the work backing up the explanation with documentary evidence.	*Can explain how and why particular methods were chosen to perform the different SE tasks required for the project or maintenance and modification tasks.*	*Can explain how the work performed on different projects or maintenance and modification tasks undertaken within the organisation is monitored and controlled to ensure a methodical approach, backing up the explanation with documentary evidence from previous projects.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC12 Multi-Discipline Systems Viewpoint		
Recognises, distinguishes and specifies the inter-relationships between equipment, procedures and people, for a multi -discipline environment.		
Producer	**Supervisor**	**Manager**
Understands the main system elements, including non-equipment elements that make up a typical system.	*Has analysed the inter-relationships between the elements of a system and documented the results with system block diagrams and requirements analysis reports.*	
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC13 Openness		
Openly admits to, and highlights, mistakes or potential weaknesses arising during the performance of system software realisation activities.		
Producer	**Supervisor**	**Manager**
Is prepared willingly to report mistakes made in the performance of software design activities, and can give illustration through for example memos, software fault reports and design change documents.	*Is prepared to willingly describe situations in which mistakes which have been made resulting from insufficient supervision/mon itoring of a software development team, the underlying reasons and the lessons learned. Encourages openness in software development teams, for example through mentoring and the provision of appropriate design review procedures.*	
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC14 Organisation Systems		
▪ Has a knowledge and understanding of existing systems in the organisation (e.g. Quality Management Systems) and functional SE practi ces, including application and technology appropriate to the organisation and industry sector, sufficient to ensure that the development and maintenance of the SE Management System is cost effective and appropriate to the organisation. ▪ Reflects the organisation's SE management system and associated methods and procedures in the project quality assurance plan. ▪ Addresses the organisation's SE management system and associated methods and procedures in the performance of system or service procurement activities .		
Producer	**Supervisor**	**Manager**
As well as the SE Management System, is aware of the organisation's Quality Management System, Financial and Project Management Systems and can explain how they operate.	*Can explain how the SE Management System fits in, and relies on the Quality Management System and the Financial/Project Management Systems.*	*Can explain how the SE Management System fits in with other systems in the organisation to produce an efficient solution. Can describe ways in which the SE Management Sy stem could be realigned and the impacts of the change.*
Can identify the relevant documentation relating to the organisation's SE management system and can describe the key features of the system. Can describe the key methods and procedures associated wit h the organisations SE management system.	*Can illustrate, through project plans, audit reports, design documents, operation and maintenance and modification manuals, fault reports and impact analysis reports, how the requirements of the organisation's SE m anagement system and the associated methods and procedures are referred to in the system project and maintenance and modification activities carried out by the organisation and to which he/she has been a main contributor.*	*Can identify current or past inade quacies in the SE management system and can describe the importance of these with regard to typical projects and maintenance and modification activities carried out within the organisation. Can illustrate, via letters, memos etc. how an attempt has been made to improve the organisation's SE management system and associated methods and procedures.*
	Can illustrate, via requirements specifications and audit reports, how the requirements of the organisation's SE management system and the associated methods an d procedures are employed in the system or services procurement activities carried out by the organisation.	*Can illustrate, via letters, memos, etc., how an attempt has been made to improve the organisation's SE management system and associated methods and procedures with regard to the procurement of systems. Can identify inadequacies in the SE management system or associated methods and procedures and can describe the importance of these with regard to typical system or services procurement activities carried out within the organisation.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC15 Professional Standing and Personal Integrity		
Has the professional standing to provide credible judgements that are generally acknowledged as authoritative, coupled with sufficient strength of character not to compromise sincerely held beliefs when under pressure.		
Producer	**Supervisor**	**Manager**
Typically a degree or equivalent in a relevant discipline.	*Typically a Chartered Engineer with a degree in a relevant discipline. Has had practical SE experience within the relevant industry sector.*	*Typically a Chartered Engineer who is acknowledged as an authority in the field of SE. Likely to have presented papers on SE issues.*
Aware of the importance of personal integrity when pressed to compromise a judgement.	*Has defended a judgement when under external pressure to compromise position.*	*Has a reputation for integrity that indicates candidate will never all ow a judgement on SE issues to be compromised by outside interference.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC16 Regulatory and Legal Compliance		
Has a knowledge and understanding of all relevant regulatory and legal requirements, together with organisation-specific procedures.		
Producer	**Supervisor**	**Manager**
Has read the relevant functional standards appropriate to the industry sector. *Can explain the key principles underlying the relevant regulatory regime and associated legal issues.*	*Can illustrate through plans and modification manuals, how regulatory requirements and associated legal issues are addressed in the performance of system maintenance and modification activities.*	*Can illustrate through memos, reports, maintenance and modification manuals, how the operational and maintenance and modification requirements of the relevant regulatory authorities are continually reviewed and, where appropriate, incorporated within the organisation's SE management system, especially with regard to maintenance and modification activities.* *Supports others, through regular design reviews and the provision of organisation procedures and mentoring, in addressing regulatory and legal requirements. Can identify cases where there may be uncertainty regarding compliance with regulatory and legal requirements and can describe a practicable approach.*
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC17 Report Writing		
Produces technical reports, procedures etc., incorporating a logical document structure with the content grammatically correct using a non-verbose style.		
Producer	**Supervisor**	**Manager**
Can show an example of a technical report of which he/she is the principle author.	*Contrasts reports which are clear and to the point with reports where key evidence is hidden by poor writing or superfluous technical detail. Can show a range of technical reports which he/she has written on maintenance and modification issues.*	
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC18 Systematic Approach		
Employs systematic methods of identification, analysis and assessment to ensure that all aspects of the behaviour of the system in its environment are addressed.		
Producer	**Supervisor**	**Manager**
Has successfully performed activities requiring the use of relevant systematic techniques and can illustrate contribution through e.g. design documents, design analysis reports.	*Can illustrate, through analysis and assessment reports, how systematic techniques have been employed in analyses associated with a system application.*	*Can illustrate, through examples, where inappropriate techniques have been employed, or techniques have been incorrectly employed, in analysis and assessment. Can illustrate, through review records and analysis procedures, training course programmes how actions have been taken to ensure the appropriateness and correct implementation of techniques.*
Typically, a degree in a numerate discipline would be expected. Someone without A-level mathematics or equivalent would be unlikely to have the logical and numeracy skills to undertake or understand the analyses required.		
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC19 Systems Viewpoint		
Considers requirements in the context of the overall system within the operation, the process and th e wider environment and is able to abstract away from unimportant detail.		
Producer	**Supervisor**	**Manager**
Knows the main system elements and functions that make up a typical system.	*Can analyse and understand the inter -relationships between system and sub -system elements and the environment using for example block diagrams, and has systematically documented assumptions relating to these inter -relationships in requirements analysis reports. Can identify key system elements which relate to the function of the system in its environment.* **Core Skills** **CS3 Systems Architecture** The specification of systems architecture, identifying all components and their interrelationships, needed to meet the present and future needs of the business as a whole.	
Core Skills See CS9, CS10, CS11, CS12.	See CS9, CS10, CS11, CS12.	See CS9, CS10, CS11, CS12.

FRC20 Team Management		
Organises, supervises and checks the activities carried out by other SE staff such that the overall SE management role responsibilities are adequately disch arged and the collective ability and resources of a team of individuals are effectively combined.		
Producer	**Supervisor**	**Manager**
Has not had the opportunity to show competence in leading a SE team.	*Can illustrate, through the presentation of supporting docu mentation, how the work carried out by others is supported and checked to ensure that the key objectives of Project SE Management are met.*	
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC21 Team Working		
Works well within a software development team environment and contributes effectively during analysis and assessment meetings/discussions.		
Producer	**Supervisor**	**Manager**
Has worked as an effective member of a software development team , co-ordinating his or her own analysis and assessment activities with those of peers and reporting to a supervisor.	*Has worked as an effective member of a software development team, co -ordinating the activities of the team and reporting to a project manager within his or her own organisation. Can identify key attributes of a software development team and can describe situations where team dynamics has led to incomplete or insufficient analysis.*	
Core Skills See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.	**Core Skills** See CS9, CS10, CS11, CS12.

FRC22 Technology		
Applies knowledge of different engineering technologies, their strengths and weaknesses, and how they can be used to produce an efficient architecture (including the use of object technology, structured systems analysis, Jackson System Development, the Unified Process etc.)		
Producer	**Supervisor**	**Manager**
Understands current engineering technologies and efficient architectural design techniques. Has practical experience of the use of relevant technologies. **Core Skills** See CS9, CS10, CS11, CS12.	*Has reviewed and evaluated different engineering technologies in relation to the selection of optimum architectural design solutions and has documented the results in system analysis reports.* **Core Skills** See CS9, CS10, CS11, CS12.	*Ensures that system architectures are adequately reviewed with regard to the best use of available technology.* **Core Skills** **CS2 Technical specialism** Maintains an in-depth knowledge of specific technical specialisms in SE, provides expert advice regarding their application. Can supervise specialist technical consultancy. The specialism can be any SE, information or communication technology, technique, method, product or application area. Provides organizational leadership and guidelines for provision of high level technical knowledge. **CS4 Emerging Technology Monitoring** Monitors market to gain knowledge and understanding of currently emerging technologies and co-ordinates the identification and assessment of new and emerging hardware, software and communication technologies, products, methods and techniques. Evaluates likely relevance of these for the business. Provides regular briefings to staff and business management. See CS9, CS10, CS11, CS12.

FRC23 Test and Analysis Methods and Techniques		
Has knowledge of a range of suitable test and analysis methods, techniques and tools for incorporation into a validation activity and is aware of their practical implementation.		
Producer	**Supervisor**	**Manager**
Can identify and can describe a range of test and analysis methods and techniques normally employed within the organisation or industry sector for carrying out system validation activities. Given a typical validation scenario, can select an appropriate set of test and analysis methods and techniques.	*Has selected appropriate test and analysis methods and techniques for validating a system within the organisation or the relevant industry sector, and can illustrate by reference to validation plans.*	*Can explain the strengths and weaknesses of alternative testing and analysis approaches with regard to their practical implementation. Can illustrate, through memos, reports and validation procedures, how best industry practice in validating systems is continually reviewed and incorporated within the organisation's validation process.*
Core Skills **CS8 Data Analysis** Assists in establishing the data requirements for systems which meet defined system requirements. See CS9, CS10, CS11, CS12.	**Core Skills** **CS8 Data Analysis** Works with customer and/or system design team to define the data requirements for systems which meet defined system requirements. See CS9, CS10, CS11, CS12.	**Core Skills** **CS8 Data Analysis** Sets standards for data analysis tools and techniques, advises on their application, and ensures compliance. See CS9, CS10, CS11, CS12.

Definition of Problem/Query Evaluation Related Competencies

Goal 1: Software estimates are documented for use in planning and tracking the software project.

Goal 2: Software project activities and commitments are planned and documented.

Goal 3: Affected groups and individuals agree to their commitments related to the software project.

Goal 4: Primary and secondary functions of the system are identified and specified.

Goal 5: The customer's requirements are agreed to by all affected groups.

Goal 6: The commitments between the engineering groups are agreed to by the affected groups.

Goal 7: The engineering groups identify, track, and resolve inter -group issues.

ARC7 Defining the Scope of the Project		
Seeks out and evaluates information in order to define the scope, objectives, context and significance of a SE project.		
Producer	**Supervisor**	**Manager**
Can identify the main categories of information required to define the scope, context and significance of a SE project and describe how this information is obtained and evaluated. **Technical Skills** *To be defined.*	*Can illustrate, through design documents, working notes, minutes of meetings etc., how information has been collected to define the scope, context and significance of SE projects carried out within the organisation or relevant industry sector.* **Technical Skills** *To be defined.*	

ARC14 Identification of End-User Requirements		
Facilitates, and manages if required, end-user participation in the realisation of systems, often through prototyping such that any system design addresses the requirements placed on humans for operation and maintenance.		
Producer	**Supervisor**	**Manager**
Understands the standard methods and processes used for consulting, engaging and managing input from users (e.g. formalised questionnaires, interviews, and brainstorms). Has obtained input from users during the development of a system and has documented the results.	*Understands how failure to adequately engage the users of a system might lead to an erroneous situation. Regularly obtains and documents input from users, using standard strategies, during the development of systems.*	*Champions the involvement of users in determining operator and maintainer requirements. Can identify key user issues with a direct impact on performance.*
	Technical Skills **TS25 Systems Ergonomics** Observes and analyses user behaviour. Evaluates software. Enables user requirements to be implemented by recommending improvements in human interface.	**Technical Skills** **TS25 Systems Ergonomics** Is responsible for organizational commitment to high standards in human factors. Specifies ergonomics standards and methods to meet organisational objectives. Advises development team on human factors.

ARC24 Planning		
Originates and maintains a plan which encapsulates an agreed set of activities, including their interrelationship, scheduling and responsibilities which, if conformed with, results in the objectives for the plan being satisfied in a cost-effective manner.		
Producer	**Supervisor**	**Manager**
Can explain and can illustrate through examples of his/her own work, how plans have been developed and then continually updated to reflect the current status of a project. **Technical Skills** *To be defined.*	*Presents an assessment plan to which the candidate has contributed. Can illustrate through checklists, how technical criteria have been identified and specified in performing an independent assessment.* **Technical Skills** *To be defined.*	*Has been the major contributor in the preparation of assessment plans for projects carried out within the organisation or industry sector and can show how assessment plans have been maintained during the course of a project. Can illustrate the limits to the extent of applicability of the technical criteria typically used for independent assessments and can explain additional technical criteria that could be used.* **Technical Skills** *To be defined.*

NC1 Query Evaluation		
The provision of applications & systems maintenanc e and support services. Support may be provided both to users of the systems and to service delivery functions. Support typically takes the form of investigating and resolving problems and providing information about the systems. It may also include mon itoring their performance. Problems may be resolved by providing advice or training to users about the application systems' functionality, correct operation or constraints, by devising workarounds, by correcting faults, by making general or site-specific modifications, by updating system documentation, by manipulating data, or by defining enhancements - often in close collaboration with the systems' developers.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** **TS37 Application and System Support** Assists in investigation and helps resolve problems relating to applications and systems. Assists with specified maintenance procedures.	*To be defined.* **Technical Skills** **TS37 Application and System Support** Enhances applications and systems to improve b usiness performance. Identifies and resolves problems with applications and systems to maintain underlying business processes and/or continuity of service. Maintains support process and checks all requests for support are dealt with according to set standards and procedures.	*To be defined.* **Technical Skills** **TS37 Application and System Support** Manages application and system enhancements to improve business performance. Ensures all requests for support and/or system changes are dealt with according to set standards and procedures.

NC2 Helpdesk Administration		
Receiving and resolving system problems. The provision of support to enable users to make effective use of systems. Keeping records of systems supported, together with logs of users, problems and resolutions.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** **TS42 User Support** Receive and record problems from users, gather and verify prescribed customer information and resolve or escalate according to given procedures. Investigates and helps resolve problems relating to routine use of information systems and related equipment.	*To be defined.* **Technical Skills** **TS42 User Support** Receives escalated problems, gathers further information and ensures all problems are resolved or channelled to appropriate support function. Resolves complex problems affecting use of systems to maintain underlying business process. Manages help desk or call centre teams. Analyses records to identify potential improvements in the use of systems.	*To be defined.* **Technical Skills** **TS42 User Support** Specifies and owns problem resolution system to ensure all problems are closed in accordance with established procedures. Managing the support team, with particular emphasis on using resources efficiently and assisting t he team in resolving complex/difficult problems. Advises business process owners and suppliers on improvements in the use of systems.

System Feasibility Analysis Related Competencies

Goal 1: The most rigorous functions are selected as a basis for defining system-level design requirements.

ARC10 Evaluating Solutions		
Selects and justifies an architectural design solution through: ▪ an evaluation of competing solutions against a pre-defined set of criteria (e.g. degree of functionality, cost effective ness) ▪ consideration of the effect of the use of diversity in design and technology in both achieving and demonstrating that the functional requirements have been met.		
Producer	**Supervisor**	**Manager**
Understands how a comparison between competing architectures would normally be carried out and the key criteria that would influence such a comparison. Understands the relative benefits of at least two typical architectures.	*Has compared competing architectures using a standard approach and a pre-defined set of criteria and has documented the results in system analysis reports.*	*Has defined, through organisation procedures, the standard approach, and the set of criteria to be used, in comparing competing architectures. Can explain the use of criteria for comparing non-typical projects.*
Understands the advantages and disadvantages of diversity of design and technology in relation to demonstrating that functional requirements have been met. **Technical Skills** *To be defined.*	*In justifying a particular choice of system architecture, has taken into account the advantages and disadvantages of alternative diversities in design and technology, and has documented the rationale in system analysis reports.* **Technical Skills** *To be defined.*	*Supports others in understanding the advantages to be gained by a choice of design and technology over another. Can identify and resolves issues arising from the use of new or non-typical technologies through regular design reviews.* **Technical Skills** *To be defined.*

ARC15 Influencing New Systems		
Influences the realisation of new systems such that the requirements to maintain a system satisfactorily are addressed.		
Producer	**Supervisor**	**Manager**
Can describe the main operation and maintenance and modification procedures associated with typical systems developed or operated by the organisation.	*Can describe the key functional issues associated with the operation and maintenance and modification of typical systems developed or operated by the organisation.*	*Can illustrate by examples (real or hypothetical), how failure to address maintenance and modification requirements in the design of a system has led to a potentially erroneous situation. Can illustrate with review records how potential system designs are reviewed for their impact on maintenance and modification.*
Understands the principles of negotiation and has taken part in practical training exercises in influencing/negotiating. **Technical Skills** *To be defined.*	*Can cite examples from his/her own experience where it has been necessary to exert influence to resolve a situation relating to the maintenance and modification of a system. Can illustrate through memos, letters and witness testimonies, how the necessary influence was brought to bear and how each situation was resolved.* **Technical Skills** *To be defined.*	

System Operational Requirements Related Competencies

Goal 1: The operational distribution, mission profile and performance parameters are identified and documented.

Goal 2: Anticipated utilisation, efficiency parameters, operational life-cycle and system operating envelope are identified and documented.

Goal 3: Supportability factors are established and defined through the application of LSA.

ARC16 Managing Compliance		
Manages compliance with the requirements incorporated in a system or services procurement contract by establishing appropriate monitoring procedures and negotiating with and exerting influence on suppliers and other personnel within the project organisation.		
Producer	**Supervisor**	**Manager**
Has worked with systems suppliers or subcontractors and is aware of the ways in which their different perspectives can lead to problems. **Technical Skills** *To be defined.*	*Has had day-to-day responsibility for management of systems suppliers or subcontractors.* **Technical Skills** *To be defined.*	*Has had responsibility for supply of procured systems or subcontracted services.* **Technical Skills** *To be defined.*
	Can illustrate, through examples, problems that can arise with suppliers and subcontractors.	
Can explain the mechanisms (e.g. audits) that have been put in place for specific projects to monitor compliance with requirement specifications.		*Can explain the advantages and disadvantages of different mechanisms for monitoring compliance with specified requirements.*
Understands the principles of negotiation and has taken part in practical training exercises in influencing/negotiating.	*Can cite examples from his/her own experience where it has been necessary to negotiate and to exert influence to satisfactorily resolve a situation relating to the procurement of a system or service and can illustrate via memos, letters and witness testimonies, how influence was brought to bear in resolving each situation.*	

ARC22 Operational Analysis		
Uses a hazard and risk approach to the analysis of a system's use such that: ▪ human factors can be correctly incorporated into a system design ▪ associated operating and maintenance activities can be successfully specified ▪ wider environmental problems affecting human performance are identified and resolved so that functionality is not compromised.		
Producer	**Supervisor**	**Manager**
Understands the main operation and maintenance procedures associated with typical systems. Has had practical experience of the operation and maintenance of such systems through, for example, participation in installation and commissioning.	*Has assisted in the hazard analysis and risk assessment of systems through the provision of written hazard analysis and risk assessment reports dealing with the operational and maintenance aspects of systems.*	*Has performed design reviews and monitored the work of others to ensure that operational issues are adequately addressed. Can identify situations in which failure to adequately address operation and maintenance requirements in the design of systems could lead to potentially erroneous situations.*
Knows the key environmental issues that have been shown to affect the performance of humans in developing or operating systems. **Technical Skills** *To be defined.*	*Regularly assesses the environmental conditions under which systems are developed or operated and has proposed solutions to identified environmental problems in written environmental assessment reports.* **Technical Skills** *To be defined.*	

ARC33 Task Analysis		
Analyses in detail specific tasks performed by the operators and maintainers such that human hazards that affect the functionality of a system are identified and the risk -reduction measures are specified.		
Producer	**Supervisor**	**Manager**
Understands the key tasks carried out by humans in relation to typical systems. **Technical Skills** *To be defined.*	*Has analysed the tasks carried out by humans and documented the findings in system analysis reports and requirements specifications. Understands how humans interact with systems and is aware how detailed changes in system designs can impact day-to-day operation and maintenance tasks and through life costs.* **Technical Skills** *To be defined.*	*Has performed systematic analyses of human tasks to identify key operator and maintainer activities which need to be carried out to ensure functionality.* **Technical Skills** *To be defined.*

NC32 Logistics Support Analysis		
Identifies software supportability issues through the application of LSA, exerting readiness and economic influence on req uirements and design, optimally integrating ILS elements and identifying detailed ILS element resource requirements.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** *To be defined.*	*To be defined.* **Technical Skills** *To be defined.*	*To be defined.* **Technical Skills** *To be defined.*

Maintenance and Support Concept Related Competencies

Goal 1: Levels of maintenance and organisational responsibilities are defined through the application of LSA.

Goal 2: Maintenance support elements pertaining to s upply support, test and support equipment, personnel and training, transportation and handling equipment, facilities, data and computer resources are defined through the application of LSA.

Goal 3: Factors associated with the support capability such as t urn round time, skill levels, error rates, training rates, errors per mission or line of code, change traffic and expansion capability are identified and defined.

Goal 4: The environment e.g. location, dispersion etc. as it pertains to maintenance and su pport is defined.

ARC9 Development of Maintenance and Modification Procedures		
Identifies appropriate test and monitoring strategies and techniques and encapsulates these in procedures which, if complied with, result in satisfactory operation for both nor mal and degraded (maintenance, modification, failure, sabotage, etc.) modes of operation.		
Producer	**Supervisor**	**Manager**
Has written maintenance and modification procedures. **Technical Skills** *To be defined.*	*Has written maintenance and modification procedures. Can explain how maintenance and modification procedures ensure satisfactory operation.* **Technical Skills** *To be defined.*	*Has written a suite of maintenance and modification procedures for several systems. Can explain how maintenance and modification procedures ensure satisfactory operation.* **Technical Skills** *To be defined.*

See ARC22, ARC33, NC32.

Identification and Prioritisation of Technical Performance Measures (TPM) Related Competencies

Goal 1: Requirements are weighted according to import ance and measurable goals (or TPMs) are assigned to each.

ARC25 Producing Assessment Reports		
Produces technical reports, etc., incorporating a logical document structure with the content grammatically correct using a non -verbose style.		
Producer	**Supervisor**	**Manager**
Has written a technical report which is well structured and to the point. **Technical Skills** *To be defined.*	*Has written a range of assessment reports. Can show how the assessment activities and results are reported and how these are summarised to present a relevant and clear set of conclusions on the evidence for a system.* **Technical Skills** *To be defined.*	

Functional Analysis Related Competencies

Goal 1: The basic system requirements structural and functional interfaces are organised through, for example, the application of functional flow block diagrams.

Goal 2: System requirements allocated to software are controlled to establish a baseline for software engineering and management use.

Goal 3: Software plans, products and activities are kept consistent with the system requirements allocated to software.

NC30 System Structural Analysis		
Identifies functional interfaces, input design criteria and constraints through the decomposition of requirements down to sub system level, typically using functional flow block diagrams.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** *To be defined.*	*To be defined.* **Technical Skills** *To be defined.*	*To be defined.* **Technical Skills** *To be defined.*

Requirements Allocation Related Competencies

Goal 1: System decomposition is conducted in order to achieve component partitioning. Component design is as independent as possible.

ARC34 Transposing from Requirements into Design		
Transposes requirements into an easily understood, testable, software design specification through the correct interpretation and use of appropriate notations and appropriate consideration of relevant constraints (e.g. process, hardware design, hardware reliability, etc.).		
Producer	**Supervisor**	**Manager**
Has contributed to the production of a software design specification. Knows the relevant notations, standards or guidelines applicable to software design. Has made practical use of the notations in producing software design specifications.	*Has written a software design specification for a complex system. Produces software design specifications for systems using the relevant software design notations.*	*Has written software design specification for different types of system. Supports others, for example through training courses, the provision of organisation software design standards, work instructions and regular software design reviews, in making best use of structured software design methods for software design tasks.*
Knows the typical constraints that would be imposed on the software designs for a typical system.	*Has addressed software design constraints for systems in the preparation of software design specifications.*	*Through regular software design reviews, and by monitoring the work of others, can identify constraints that could lead to potentially erroneous situations.*
Has produced software design specifications which explicitly document, using a standard approach, the relationship between each requirement and the corresponding software design features so as to facilitate understanding of the design.		*Has developed organisation procedures and run regular design reviews, so that all requirements can be easily traced to the resulting software design.*
Knows the key requirements of relevant standards and guidelines relating to testability (e.g. IEE Guidelines for assuring testability).	*Has produced source code and associated module design specifications for systems which consistently address the requirement for testability.*	*Ensures, for example through regular software design reviews, that sufficient attention is paid to testability in designing software for systems. Can identify cases where insufficient attention to testability in designing software for a system could result in inadequate functional assurance.*

Technical Skills **TS21 Systems Design** Undertake complete design of simple applications using simple templates and tools. Assist as part of a team on design of components of larger systems. **TS22 Database Design** Assists in detailed database design. Carries out data design required by report generation for small ad hoc jobs.	**Technical Skills** **TS21 Systems Design** Recommends and designs systems or subsystems to meet defined objectives following established process. Designs structures and tools for systems which meet business needs. Delivers 'technical visualisation' of proposed system for approval by customer and execution by system developers. Maps work to user specification and removes errors and deviations from specification to achieve user - friendly systems. **TS22 Database Design** Transforms data models to appropriate physical database design. May understand and use more than one DBMS.	**Technical Skills** **TS21 Systems Design** Controls system design within an enterprise or industry architecture. Influences industry-based models for the development of new systems. Develops effective implementation strategies and procurement. Overall responsibility for managing and coordinating the architecture of systems. Reviews others' system design to ensure selection of appropriate technology, efficient use of resources, and integration of multiple systems and technology. Establishes policy for selection of architecture components. Ensures that the system architecture balances functional, service quality and systems management requireme nts. **TS22 Database Design** Initiates multi-system or multienterprise data architectures, creating and enhancing industry-based data models. Reviews designs to ensure quality of database design.

System Synthesis, Analysis and Design Optimisation Related Competencies

Goal 1: Design candidates and evaluation parameters are identified and tailored to the system in order to eliminate unfeasible solutions.

Goal 2: Data required in order to evaluate operational requirements, the maintenance concept, major design features, production plans and anticipated system utilisation and support requirements is identified and managed effectively through the application of LSA.

Goal 3: Design evaluation tools, techniques and development model are identified and selec ted.

Goal 4: Design alternatives are evaluated and a sensitivity analysis conducted in order to identify risk and uncertainty and recommend a preferred approach.

See ARC10, ARC17, ARC34.

Design Integration Related Competencies

Goal 1: The appropriate allocation of design specialists according to system development, functional requirements and size is carried out.

See ARC34.

System Test and Evaluation Related Competencies

Goal 1: Integrated test planning and defect prevention activities are planned for early on in the system's development.

Goal 2: Effective system test and evaluation preparation and test and performance evaluation seeking out and identifying common causes of defects is undertaken.

Goal 3: The prioritisation and systematic elimination of the common causes of defects whilst incorporating the necessary change control procedures is undertaken.

Goal 4: Defects in the software work products are identified and removed.

ARC2 Analysing Test Results		
Analyses and categorises test and other observations such that failures with an impact on mission criticality are clearly highlighted and an objective decision can be taken as to whether a system is fit for service.		
Producer	**Supervisor**	**Manager**
Can identify and can explain potential test failure categories (e.g. test rig or test equipment fault, fault in the test procedure, actual system fault) and can illustrate with test reports in which test failures are clearly categorised.	*Can illustrate, through test reports, how test failures have been analysed and categorised in terms of their potential impact on functional operability and their underlying causes, and can show how important side-effects of system validation activities have been highlighted and recorded for subsequent action.*	
Technical Skills **TS24 Systems Testing** Assists in the execution of test plans, recording and reporting outcomes.	**Technical Skills** **TS24 Systems Testing** Scopes and creates test plans and test data, mapping back to requirements. Maintains the integrity of the test environment. Executes test plans, recording and reporting outcomes.	**Technical Skills** **TS24 Systems Testing** Sets objectives, standards and techniques for systems testing function in the enterprise, manages application, ensures compliance and ensures function focuses on delivering business advantage.

ARC3 Analysing the Code		
Demonstrates, through the appropriate use of static and dynamic software analysis techniques, that constraint limitations will not adversely affect the operation of the system and that the software realisation is consistent with the software requirements.		
Producer	**Supervisor**	**Manager**
Has made practical use of relevant software analysis techniques (e.g. complexity analysis, data flow analysis, control flow analysis, object code analysis, timing analysis, stack analysis, build analysis). Has taken part in code walkthroughs.	*Has analysed the functionality of the software elements of a system using the relevant techniques, and has documented the results in software analysis reports that could be used to support a justification that the system is fault -free. Has taken part in and led code walkthroughs.*	*Has analysed the functional aspects of software code using appropriate software analysis procedures and regular code walkthroughs. Can identif y situations in which inappropriate techniques or the incorrect use of analysis techniques could lead to an error situation. Can explain the types of failures that are identified via code analysis and walkthroughs and can discuss how these relate to softw are functionality.*
Typically, a degree in a numerate discipline would be expected. Someone without A -level mathematics or equivalent would be unlikely to have the logical skills to undertake or understand the analyses required.		
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC4 Analysing the Design		
Can demonstrate, through the use of appropriate software analysis techniques, that a design meets the given functional requirements.		
Producer	**Supervisor**	**Manager**
Has an understanding of software analysis techniques (e.g. UML, CSP and Z) and an appreciation of their differences.	*Has analysed software designs for systems employing software analysis techniques and has documented the results in software analysis reports. Understands the contribution of software analysis in the overall assessment process and how the results are used in further verification and validation activities (e.g. determining the required rigour of testing of different areas of the software design).*	*For non-typical systems, can identify areas in which additional analysis is required to provide adequate functional assurance evidence for software designs. Can describe typical problems with software designs and can show how software analysis uncovers these problems.*
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC11 Executing Tests		
Executes test procedures precisely, accurately and reliably such that items of importance are not overlooked during the execution of test c ases.		
Producer	**Supervisor**	**Manager**
Has applied a standard organisational method to execute and record typical test cases.	*Has produced test reports for systems that clearly identify and highlight, for subsequent action, the side-effects of system software testing activities. Has reviewed and approved software test specifications to ensure that software designs are tested fully. Can identify key functions in a software design.*	*Typically not involved in executing tests.*
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC12 Forming a Judgement		
Makes an unambiguous judgement, through a reasoned and documented argument, on whether a system has satisfied its functional objectives, including the systematic aggregation of evidenc e obtained through a combination of audits, reviews and analyses.		
Producer	**Supervisor**	**Manager**
Has constructed and presented a clear and reasoned argument from unstructured information.	*Has constructed and presented an argument to justify a set of conclus ions and recommendations arising from the conduct of an independent assessment carried out within the organisation or the relevant industry sector and can illustrate this with independent assessment reports.*	
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC17 Managing in-Service Information		
Proactively collects, analyses and effectively uses data obtained during in -Service operation such that increased assurance is obtained on existing systems and is available for new designs.		
Producer	**Supervisor**	**Manager**
Can illustrate, through reports and presentations, how data analysis techniques have been used in a practical work situation.	*Can illustrate, through incident reports, change documents and reliability growth modelling, h ow data analysis techniques are used in the provision of evidence of the operational performance of a system and used to improve its performance.*	
Understands how performance information relating to the performance of systems is collected within the organisation. Knows the basic techniques of data collection and the use of analysis equipment (e.g. data analysers, oscilloscopes). **Technical Skills** *To be defined.*	*Can illustrate, through working notes, data recorder printouts, oscilloscope traces etc., how operational performance information has been collected from a variety of sources and analysed to arrive at a conclusion regarding operational functionality.* **Technical Skills** *To be defined.*	*Can cite examples where insufficient information has been obtained with regard to the operational performance of a system. Can illustrate, using for example checklists, how such examples can be avoided.* **Technical Skills** *To be defined.*

ARC23 Performing Analysis		
Ensures that adequate analysis is carried out, in acc ordance with the software validation plan, to complement the evidence obtained through functional testing.		
Producer	**Supervisor**	**Manager**
Is familiar with the typical analysis techniques used to validate systems within the organisation or industry sector.	*Can illustrate, through extracts from validation reports, how analysis has been carried out to validate the implementation of a system.*	*For a chosen system, can explain why particular analysis techniques were selected and how the analysis complements the fu nctional testing.*
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC31 Specifying Software Tests		
Produces software test specifications, employing a complementary set of approaches to software testing, that are consistent with the planned validation strategy, that contain rigorous test cases, which take account of the environment, that have a high probability of detecting faults in the realisation of a system through the practical and creative use of proven test methods and techniques.		
Producer	**Supervisor**	**Manager**
Understands the range of software test methods and techniques normally employed for projects and the different types of fault that are found by the different types of test.	*Has prepared software design test specifications using the relevant software test methods and techniques to demonstrate compliance with requirements.*	*Has monitored and reported on best industry practice in testing software and, when appropriate, incorporated the findings within organisational te st procedures.*
Has developed test specifications, and contributed to the development of test rig designs, using a practical and creative approach.		*Typically not involved in detailed specification of tests.*
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC32 Specifying Tests		
Produces practical test specifications and procedures which are consistent with the validation plan and which have a high probability of detecting faults in the system.		
Producer	**Supervisor**	**Manager**
Can describe the content of typical test specifications and procedures (e.g. initial conditions, hazards, space to record results, acceptance criteria) appropriate to the organisation or industry sector and has had recent project experience illustrated by t est specifications.		*Can explain, with examples, how the accepted format of test specifications and procedures has evolved with regard to the particular requirements of the organisation or the relevant industry sector.*
Can demonstrate a practical approach in devising means of validating a system, illustrated by e.g. test rig designs, test specifications.		*Typically not involved in detailed specification of tests, although has reviewed some examples.*
Has shown creativity in devising means of validating a sys tem, illustrated by e.g. test rig designs, test specifications.		*Typically not involved in detailed specification of tests, although has reviewed some examples and is able to discuss novel validation strategies.*
Not yet fully aware of a wide range of weakn esses in test specifications and how they are identified.		*Can illustrate, using test specifications and review records, how weaknesses in test specifications have been identified and can explain the weaknesses and suggests alternative approaches.*
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

ARC35 Witnessing and Executing Tests		
Ensures that validation tests are executed accurately and reliably such that anomalies are identified and that results are reported in a form that aids subsequent analysis.		
Producer	**Supervisor**	**Manager**
Can describe and can illustrate through test records and test reports, the process involved in executing system validation tests (e.g. calibration of test equipment, recording the system and test equipment configuration, recording test data).	*Can illustrate through site acceptance and factory acceptance test reports, how the significance of side-effects observed during the performance of testing has been assessed.*	*Can illustrate, through the presentation of independent test reports, how validation testing has been observed (e.g. as an independent witness) to ensure that the testing has been carried out in accordance with the defined strategy and procedures. Can explain the significance of test failures or testing that has not been performed in accordance with the defined strategy and procedures and can explain alternative strategies which may have been used.*
Not fully considered the possibility that performance of validation tests may pose a hazard.	*Can illustrate, through examples (real or hypothetical), instances where proposed validation tests have, in themselves, been potentially erroneous and can illustrate, through test procedure review records and training course programmes, how actions are taken to ensure that functionality is adequately considered during the performance of validation activities.*	
Technical Skills See TS24	**Technical Skills** See TS24	**Technical Skills** See TS24

Construction Related Competencies

Goal 1: The software engineering tasks are defined, integrated, and consistently performed whilst producing the software in order to maintain the high quality characteristics identified and incorporated in the design and testing phases.

Goal 2: Software work products are kept consistent with each other.

ARC6 Coding		
Translates the specified software functional and design requirements into easily understood, analysable source code through the correct use of an appropriate programming language. Pays due heed to the requirements of a relevant coding standard (with particular regard to the functional implications of different constructs and the environment in which the code is to operate).		
Producer	**Supervisor**	**Manager**
Has coded individual models using the relevant programming language(s) in accordance with the organisation's programming style and commenting strategy. **Technical Skills** **TS23 Programming/Software Development** Assists in coding, testing, debugging, and documenting simple programs, and assists in the implementation of software which forms part of a properly engineered system.	*Has coded complete software sub-systems for typical systems, using a programming language in accordance with a defined coding standard.* **Technical Skills** **TS23 Programming/Software Development** Designs, codes, tests, debugs, and documents systems to a clear design specification. Takes technical responsibility for several stages in the software development process. Contributes to project and quality plans.	*Is abreast of the latest developments in SE research, particularly with regard to language limitations and the circumstances in which they should be avoided, and maintains latest understanding in an organisational coding standard.* **Technical Skills** **TS23 Programming/Software Development** Sets standards for programming tools and techniques, advises on their application and ensures compliance. Takes technical responsibility for all stages in the software development process. Prepares project and quality plans and advises systems development teams.

NC3 Product Dissemination		
The planning, design and creation of information to be delivered electronically or otherwise. This includes managing the quality assurance and publication process.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** **TS26 Media Creation** Assists in the creation of design and content for information to be delivered by following style guidelines.	*To be defined.* **Technical Skills** **TS26 Media Creation** Controls the creation of all system/user information to meet customer requirements and organisational standards in multiple media. Assists in setting design standards, creates and revises design and content for information to be delivered.	*To be defined.* **Technical Skills** **TS26 Media Creation** Initiates the research and development of new media. Delivers effective information to the organisation or customer. Assists management and exploitation of the organisation's information assets. Works with systems designers to create and maintain standards for the technology and ergonomics of media and information. Manages the design and authoring process for information to be delivered. Sets design and publishing guidelines for the organisation.

NC4 Construction		
The incremental and logical integration and testing of components and/or sub-systems and their interfaces in order to form complete systems.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** **TS27 Systems Integration** Assists in integrating software and hardware sub-systems in new systems, upgrades, enhancements and conversions.	*To be defined.* **Technical Skills** **TS27 Systems Integration** Controls the integration of software and hardware sub-systems in new systems, upgrades, enhancements and conversions.	*To be defined.* **Technical Skills** **TS27 Systems Integration** Designs and builds integration components and interfaces. Leads practical implementation work under the technical direction of the system designer/architect. May contribute to the overall design of the solution. May define the technical criteria for product/com ponent selection. Contributes to decisions about tools, methods and approaches to be used in the project.

System Operation Use and Life-Cycle Support Related Competencies

Goal 1: Maintenance and support practices are defined and evaluated for effectiv eness.

ARC13 Handling Change		
Analyses the impact on functionality of any change to a system; ensures that the implementation of any change does not result in an erroneous situation, and provides a reversion strategy.		
Producer	**Supervisor**	**Manager**
Understands how to analyse the potential impact of changes to systems. Can identify ways in which changes to a system would impact upon maintenance procedures. **Technical Skills** *To be defined.*	*Can illustrate, through analysis reports, how proposed changes to systems are assessed for their impact upon functionality and the maintenance and modification procedures.* **Technical Skills** *To be defined.*	*Can illustrate through examples (real or hypothetical) how the incorrect assessment of the implications of proposed chan ges have led to potentially erroneous situations especially regarding maintenance and modification activities. Can illustrate, through procedures, work instructions, training course notes etc., the actions that have been put in place to ensure that risks are correctly assessed.* **Technical Skills** *To be defined.*

NC5 Installation and Decommissioning		
The installation or decommissioning of hardware and/or software, following plans and instructions and in accordance with agreed standards. The testing of hardware and software components affected, resolving malfunctions found and recording the results. The reporting of details of hardware and software installed so that the organisation's configuration management records can be updated.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** **TS28 Systems Installation/Decommissioning** Assists in installation/removal of hardware and software components of a system.	*To be defined.* **Technical Skills** **TS28 Systems Installation/Decommissioning** Plans and controls the installation/decommissioning of hardware and software components of a system.	*To be defined.*

NC6 Database Support		
The installation, configuration, upgrade, administration, monitoring and maintenance of databases in support of systems.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** **TS39 Database Administration** Assists in database management system support activities, investigates database problems.	*To be defined.* **Technical Skills** **TS39 Database Administration** Configures and administers databases.	*To be defined.* **Technical Skills** **TS39 Database Administration** Evaluates and advises on database administration techniques and tools. Responsible for ensuring data integrity.

System Retirement and Material Disposal Related Competencie s

Goal 1: System disposal plans are formulated that take into consideration the specific exigencies of the military (e.g. equipment declassification) as well as environmental concerns.

See NC5.

E.2 Software Modification Function

Summary

The Software Modification function is concerned with the development and maintenance of software for use within, or by, a system. Its activities are directly related to activities within the SysE function, and as such have related competencies. Key activities within this function are:

- Software Requirements
- Software Design
- Software Construction
- Software Testing/Qualification and Certification
- Software Maintenance

Software Modification Function Related Competencies

See FRC1, FRC2, FRC3, FRC4, FRC5, FRC6, FRC7, FRC8. FRC9, FRC10, FRC11, FRC12, FRC13, FRC14, FRC15, FRC16, FRC17, FRC18, FRC19, FRC20, FRC21, FRC22, FRC23.

Software Requirements Related Competencies

NC31 Software Requirements Specification		
Working from system requirements, identifies and specifies software requirements.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical Skills** *To be defined.*	*To be defined.* **Technical Skills** *To be defined.*	*To be defined.* **Technical Skills** *To be defined.*

Software Design Related Competencies

See ARC10, ARC34.

Software Construction Related Competencies

See ARC6, NC4.

Software Testing/Qualification and Certification Related Competencies

See ARC2, ARC3, ARC4, ARC11, ARC12, ARC 23, ARC31, ARC32, ARC35.

Software Maintenance Related Competencies

See ARC13.

E.3 Supporting Activities Function

Summary

The Supporting Activities function comprises 4 separate global SE activities that interact with and influence each of the other 3 SE functions. Key activities within this function are:

- Change Management
- Software Engineering Management
- Software Engineering Tools and Methods Management
- Software Quality Assurance

Supporting Activities Function Related Competencies

See FRC3, FRC6, FRC7, FRC8, FRC10, FRC11, FRC14, FRC15, FRC17, FRC20, FRC21.

Change Management Related Competencies

Goal 1: Software configuration management activities are planned.

Goal 2: Selected software work products are identified, controlled and available.

Goal 3: Changes to identified software work products are controlled.

Goal 4: Affected groups and individuals are informed of the status and content of software baselines.

See ARC13.

NC7 Change Management		
The management of all changes to the components of a live infrastructure, from request for change (RFC) through to impl ementation and review, to support the continued availability, effectiveness and safety of the infrastructure.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.* **Technical Skills** **TS5 Change Control** Assesses, analyses, develops, documents and implements changes based on Requests For Change. Applies change control procedures.	*To be defined.* **Technical Skills** **TS5 Change Control** Develops implementation plans for dealing with more complex requests for change, evaluates risks to integrity of infrastruct ure inherent in proposed implementations, seeks authority for those activities, undertakes review of effectiveness of change implementation, suggests improvement to organizational procedures governing change management. Sets the organization's assessment, analysis, development, documentation and implementation of changes based on Requests For Change.

NC8 Configuration Control		
The systematic management of documentation, software, hardware and firmware SE assets in terms of their identification as configu ration items (C/Is), with the definition of their structures and relationships. Storage, access, problem reporting and change control of C/Is and the application of status accounting and auditing, often in line with acknowledged external criteria such as ISO 9000, throughout all stages of the system life -cycle.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.* **Technical Skills** **TS33 Configuration Management** Ensures appropriate versions of a product's or system's or organisation's assets or compon ents are available and managed while protecting those assets and components from unauthorised change, diversion, and inappropriate use. Uses configuration management tools to identify, track, and log system components and their changes to maintain a recor d of the status of hardware and changes to the system.	*To be defined.* **Technical Skills** **TS33 Configuration Management** Investigates, recommends, maintains and is responsible for tools, techniques and processes for ensuring appropriate versions of a product' s or system's or organisation's assets or components are available and managed as well as managing those assets.

Software Engineering Management Related Competencies

Goal 1: Technical and managerial activities within the SE process that are performed d uring software acquisition, development, maintenance and retirement are planned and implemented.

Goal 2: Actual results and performance are tracked against the software plans.

Goal 3: Corrective actions are taken and managed to closure when actual resu lts and performance deviate significantly from the software plans.

Goal 4: Changes to software commitments are agreed to by the affected groups and individuals.

Goal 5: The project's defined software process is a tailored version of the organisation's standard software process.

Goal 6: The project is planned and managed according to the project's defined software process.

ARC1 Allocation of Responsibilities		
Identifies roles and organisational relationships required to implement the corporate SE mana gement system, and allocates or arranges staff responsibilities for the performance of these roles.		
Producer	**Supervisor**	**Manager**
Can identify relevant documentation relating to the allocation of responsibilities, and understands the way in which appropriate allocation and organisation contributes to the effective and robust operation of the SE management system. **Technical Skills** *To be defined.*	*Has participated in the definition of specific roles and their relationships with other roles so as to ensure the effective and robust operation of a SE management system.* **Technical Skills** *To be defined.*	*Has allocated responsibilities for SE management system roles, monitored the efficacy of the allocation with respect to effective and robust operation of the syste m, and improved allocation accordingly.* **Technical Skills** *To be defined.*

ARC5 Assuring Staff Competency		
Ensures that all staff involved with SE work are competent to execute their assigned tasks. For example, instigating a formal training programme, w ork place supervised experience etc.		
Producer	**Supervisor**	**Manager**
Can describe the methods currently used within the organisation to assess and justify the competency of SE project team members. **Technical Skills** *To be defined.*	*Can illustrate, via the organisation's procedures and project plans how a competency justification system is implemented within the organisation for SE projects.* **Technical Skills** *To be defined.*	*Can illustrate through examples (real or hypothetical) how insufficient attention to th e competency of individuals employed on SE projects could lead to project failure. Can illustrate, via the organisation's procedures, how actions have been taken to ensure competent individuals are assigned to projects.* **Technical Skills** *To be defined.*

ARC19 Managing Resource Allocation		
Advises and facilitates (and manages if appropriate) the deployment of the allocation of sufficient resource of relevant competence, such that the needs of the SE management system can be met.		
Producer	**Supervisor**	**Manager**
Is familiar with an accepted estimating method and associated techniques and is able to present supporting documentation to show how the method has been applied in practice. **Technical Skills** *To be defined.*	*Can illustrate, through e.g. estimating sheets, how advice has been provided to projects with regard to the necessary resource requirements for carrying out the project.* **Technical Skills** *To be defined.*	*Can cite examples (real or hypothetical) where resource issues have or could lead to an erroneous situation on a project. Can explain how the organisation's procedures have been developed to ensure adequate resources.* **Technical Skills** *To be defined.*
Can provide 'rule of thumb' estimates for typical projects carried out by the organisation.	*Can provide 'rule of thumb' estimates for complex or innovative projects carried out by the organisation.*	

ARC28 Resource Allocation		
Advises and facilitates (and manages if directed) the deployment of resources (competent staff, spares, tools, etc.), sufficient to ensure operation, maintenance and modification.		
Producer	**Supervisor**	**Manager**
Can identify the key resource requirements that are necessary for the successful maintenance and modification of a system. Given a typical project scenario, correctly estimates the necessary resource requirements.	*Can illustrate, through project plans and memos, how resource requirements have been derived, reviewed, updated in line with operational experience gained during the use of a system and updated to meet revised needs after modification.*	*Can illustrate through examples (real or hypothetical) how inadequate resources have led to compromises on functionality. Can illustrate, through for example, review procedures and checklists, how actions are taken to ensure adequate, trained resources and the actions required to accumulate changed resource requirements after systems modification.* **Technical Skills** **TS1 Information Resource Management** Establishes and manages information resource management strategy and policies for the business. Plans and implements processes to take forward the strategy and policies. Coordinates strategy for information resource management with strategies for data storage and management. **TS15 Systems Development Management** Sets strategy for resource management within systems development, identifies and manages resources necessary for all stages (planning, estimation, execution) of individual systems development projects to ensure technical financial and quality targets are met, authorises allocation of resources for programmes of system development projects, maintains overview of contribution of programme to organisational success **TS18 Service Delivery Management** Set strategy which inspires a commitment to excellence in service delivery, authorise allocation of resources for system delivery monitoring arrangements, provide leadership within the industry on the identification of future trends (e.g. technical, market, industrial, socio-economic, legislative, etc.) develop relationships with customers at the highest level to identify potential areas of mutual commercial interest for future development, maintain overview of contribution of service delivery arrangements to organisational success. Agrees and manages service levels in the enterprise. Plans and delivers operational resources to meet service levels over time. Manages user expectations throughout the business.

ARC29 Risk Assessment		
Determines the consequences and frequencies of project failure associated with the occurrence of unmanaged e vents by selecting and applying an appropriate risk assessment method (e.g. the risk graph).		
Producer	**Supervisor**	**Manager**
Understands the principles of risk assessment and can explain the process of risk assessment.	*Has carried out risk assessments as part of a wider team. Can illustrate, through risk assessment reports, how an accepted method for assessing the risks associated with a system application has been used.*	*Has been involved in a number of risk assessments, and led some of these activities. Ca n illustrate, through risk assessment procedures, work instructions, training course notes the actions that have been taken to ensure that appropriate methods are used to assess the risks associated with systems.* **Technical Skills** **TS4 Business Risk Managem ent** Plans and manages the implementation of organization -wide processes and procedures, tools and techniques for the management of risk in all SE activities and projects.
Can describe the key factors that affect the tolerability of risks within the relevant industry sector and general formulae for arriving at accepted figures.	*Can illustrate, through risk assessment reports, how the tolerability of risks has been addressed for projects carried out within the organisation or within the relevant industry sec tor.*	
Can explain the key principles underlying the relevant regulatory regime and associated legal issues and how these relate to the assessment of risks.	*Can illustrate, through risk assessment reports, how regulatory requirements and associated legal issues have been addressed during risk assessment activities.*	*Can illustrate how risk assessment activities have failed, or could fail to adequately address the relevant regulatory requirements or associated legal issues and can explain how the risk assessment process used within the organisation counters such examples.*

NC9 Software Engineering Project Management		
The management of projects, involving the development and implementation of systems to meet identified business needs, acquiring and utilising th e necessary resources and skills, within agreed parameters of cost, timescales, and quality.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS9 Project Management** Sets organizational strategy governing the dire ction and conduct of project management. Authorizes the management of large scale projects. Leads project planning, scheduling, controlling and reporting activities for corporate, strategic, high impact, high risk projects. Develops, recommends and prom otes the application of sound project management technologies. Influences the preparation and maintenance of realistic project, quality and risk plans. Controls manpower and all costs against project budget and ensures system is delivered within planned cost, timescale and resource budgets, project deliverables are signed off. Manages risk and sees that solutions to problems are implemented in line with change control processes.

NC10 Service Level Management		
The negotiation and execution of service level agreements, and the planning, scheduling, measurement and control of operational facilities to provide the agreed levels of service. The monitoring and reporting of actual service levels compared with the targets set in service level agreements. The use of tools, as required, to capture, analyse, store and report accurate service quality details.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical skills** **TS40 Service Level Control** Monitors the performance of systems.	*To be defined.* **Technical skills** **TS40 Service Level Control** Monitors, gathers and analyses service level information to ensure compliance with service level agreements and to help plan future requirements. Identify operational resources to meet service levels over time.	*To be defined.* **Technical skills** **TS40 Service Level Control** Ensures that operational resources meet service levels over time. Initiates, owns, develops and manages monitoring systems which meet real customer and organisational needs to monitor, measure and report on service delivery arrangements, identify any changes to business and user service requirements through regular contact with customers, share information on emerging trends (e.g. new technologies, changes in the structure of the market etc.) with customers and identify areas for improvement.

Software Engineering Tools and Methods Management Related Competencies

Goal 1: Incorporation of technology changes is planned.

Goal 2: New technologies are evaluated to determine their effect on quality and productivity.

Goal 3: Appropriate new technologies are transferred into normal practice across the organisation.

Goal 4: Software development environments that assist in the software development process are managed.

Goal 5: Development methods that impose structure on the software development process are defined.

Goal 6: Continuous software process development and improvement activities are planned and coordinated across the organisation.

Goal 7: Participation in the organisation's software process improvement activities is organisation-wide.

Goal 8: The organisation's standard software process and the project's defined software processes are developed, maintained and improved continuously. Information relating to their use is collected, reviewed and mad e available.

Goal 9: The strengths and weaknesses of the software processes used are identified relative to a process standard.

Goal 10: Organisation-level process development and improvement activities are planned.

Goal 11: The quantitative process management activities are planned.

Goal 12: The process performance of the project's defined software process is controlled quantitatively.

Goal 13: The process capability of the organisation's standard software process is known in quantitative terms.

<table>
<tr><td colspan="3">ARC8 Developing Procedures</td></tr>
<tr><td colspan="3">Facilitates the origination of operational and maintenance procedures, typically through style guides, such that clear, unambiguous instructions are available to humans in performing SE tasks.</td></tr>
<tr><td>Producer</td><td>Supervisor</td><td>Manager</td></tr>
<tr><td>Has developed operation and/or maintenance procedures in similar environments to the required context. Knows the key standards (internal and external) used by the organisation in developing or operating software intensive systems.</td><td colspan="2">Has developed operation and maintenance procedures for software intensive systems. Has written clear specifications for user tasks. Has developed, or tailored, organisation standards, procedures and style guides for use during the development or operation of software intensive syst ems.

<u>Technical Skills</u>
TS6 Methods and Tools
Sets direction, provides expertise and support, and leads in the introduction and use of techniques, methodologies and tools.</td></tr>
</table>

NC11 Project Management Support		
The provision of support and guidance on project management processes, procedures, tools and techniques to programme and project managers and their teams. The use of project management software. The development, production and maintenance of time, resource, cost and exception plans. The tracking and reporting of progress and performance of SE projects (including those performed by third parties under contract). The maintenance of programme and/or project files. The servicing of project control boards, project assurance teams and quality review meetings. The analysis of performance and the maintenance of metric data and estimating models. The administration of project change control, including use of configuration management systems.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical skills** **TS10 Project Office** Assists with the compilation of project management reports. maintains programme and project files from supplied actual and forecast data.	*To be defined.* **Technical skills** **TS10 Project Office** Uses and recommends project control solutions for planning, scheduling, and tracking projects. Sets up project files, compiles and distributes reports and provides detailed guidance on project management software, procedures, processes, tools and techniques. Attends project control boards, project assurance teams and quality review meetings. Provides guidance on project management software, procedures, processes, tools and techniques. Provides administrative services to project control boards, project assurance teams and quality review meetings.	*To be defined.* **Technical skills** **TS10 Project Office** Advises on the available standards, procedures, methods, tools and techniques. Evaluates project performance and recommends changes where necessary. Contributes to reviews and audits of project and programme management to ensure conformance to standards.

Software Quality Assurance Related Competencies

Goal 1: Software quality assurance activities are planned.

Goal 2: Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.

Goal 3: Affected groups and individuals are informed of software quality assurance activities and results.

Goal 4: Non-compliance issues that cannot be resolved within the software project are addressed by senior management.

Goal 5: Improvement of the QMS is achieved by regular analysis and management activities.

Goal 6: The project's software quality management activities are planned.

Goal 7: Measurable goals for software product quality and their priorities are defined.

Goal 8: Actual progress toward achieving the quality goals for the software products is quantified and managed.

ARC18 Managing Outcomes		
Contributes as required to the management of assessment results, such that any necessary act ions are addressed and appropriately resolved.		
Producer	**Supervisor**	**Manager**
Has presented positive and constructive criticism to peers, which can be illustrated via review records and presentations. **Technical Skills** *To be defined.*	*Can illustrate, through independent assessment reports, how the findings resulting from independent assessments have been presented in a positive and constructive manner.* **Technical Skills** *To be defined.*	
Can describe the key commercial, legal and political issues associated with typical systems assessed by the organisation.	*Can illustrate, through independent assessment reports and associated letters and presentations, how commercial, legal or political issues have been taken into account in presenting the findings of independe nt assessments.*	*Has handled contentious issues arising from independent assessments in a way which was appreciated by customers.*

ARC20 Monitoring Compliance		
Achieves adherence with the SE management system, by performing audits against a schedule and instigating improvements to the SE management system when identified as necessary.		
Producer	**Supervisor**	**Manager**
Can explain the mechanisms (e.g. audits) that have been put in place across the organisation to monitor compliance of these projects with the SE management system.	*Can explain the advantages and disadvantages of different mechanisms for monitoring compliance with a SE management system, backing up the explanation with documentary evidence of the performance of such monitoring. Can show how monitoring has been achieved within an organisation, and how the results of the monitoring process are fed back into the SE management system.* **Technical Skills** **TS13 Compliance** Plans a programme to audit and collect and collate evidence of a SE activity, process, product or service. Analyses evidence collated and examines records as part of specified testing strategies for evidence of compliance with management directives, or the identification of abnormal occurrences. Drafts part, or all, of formal reports commenting on the conformity found to exist in the audited part of a SE environment.	*Can cite examples (real or hypothetical) where a lack of adequate monitoring has or could lead to a potentially erroneous situation and can explain how monitoring within the organisation has been improved to counter such examples. Can explain the advantages and disadvantages of different mechanisms for monitoring compliance with a SE management system. Has been responsible for monitoring within an organisation, and can describe the way in which the results of the monitoring process are fed back into the SE management system.* **Technical Skills** **TS13 Compliance** Specifies organizational procedures for the internal or third party assessment of a SE activity, process, product or service, against recognized criteria, such as BS EN ISO 9000/14000. Evaluates and independently appraises the internal control of SE mechanisms or processes based on investigation evidence and assessments undertaken by self or team. Ensures a udit activities meet their objectives. Recommends changes in processes and SE control procedures based on audit findings in order to improve the effectiveness and efficiency of SE control mechanisms. Involves the establishment, maintenance and management of the assessment framework and practices that support wider business objectives, and the identification and evaluation of associated risks and how they can be reduced.

ARC21 Monitoring the Engineering Development		
Monitors the engineering development to ensure consistency with a design philosophy which contributes to quality assurance.		
Producer	**Supervisor**	**Manager**
Has not had the opportunity to monitor engineering developments.	*Can show how monitoring of engineering development is achieved within an organisation. Can describe ways in which divergence from design philosophy can occur and how quality can therefore be compromised.* **Technical Skills** **TS12 Quality Assurance** Investigates and documents the internal control of specified aspects of a SE mechanism or process and assesses how this system compares to the relevant standard. Takes responsibility for the control, distribution and filing of quality standards. Creates and distributes procedures and standards, according to documented procedures. Makes technical changes to quality and environmental standards, according to documented procedures. Uses appropriate methods tools and applications in the development, maintenance, control and distribution of SE quality and environmental standards. Identifies changes required to quality and/or environmental standards as a result of an audit or changes to current practice and takes responsibility for ensuring that they are made, either directly or indirectly.	*Has been responsible for monitoring engineering development within an organisation. Can illustrate, by examples, project situations in which quality has been compromised by divergence or potential divergence from a design philosophy.* **Technical Skills** **TS12 Quality Assurance** Develops organizational commitment to ongoing quality and environmental improvement by ensuring that the quality assurance process is robust and based on the best information from colleagues, customers, and industry as a whole. Undertakes communications and training activit ies to update colleagues on implication of revisions to quality standards, uses standards to review past performance and plan future activities, identifies opportunities for maintaining and updating quality standards in the light of emerging best practice, monitors and reports on the outputs from the quality assurance and audit processes. Considers implications emerging technological developments, economic and social trends etc. Advises on the development, maintenance, control and distribution of SE quali ty and environmental standards and ensures that this process supports organizational objectives. Reviews audit process to ensure it continues to meet needs of the standards.

ARC26 Promoting Awareness		
Ensures that all staff who can affect the achievemen t of functionality are aware of their obligations by: identifying target audience, implementing an appropriate dissemination programme, measuring achievement and applying corrective action as necessary.		
Producer	**Supervisor**	**Manager**
Can illustrate, through training programme course notes, follow-up questionnaires, audit reports etc., how awareness of safety information has been promoted within an organisation. **Technical Skills** *To be defined.*	*Can illustrate, through training programme course notes, follow-up questionnaires, audit reports etc., how awareness of a SE management system has been promoted within an organisation, how the extent of that awareness has been checked and how corrective actions have been taken to increase awareness.* **Technical Skills** *To be defined.*	*Can identify key areas of a SE management system where difficulty has been encountered in promoting awareness of the underlying issues and can illustrate specific actions that have been taken to overcome them. Can explain how different promotional strategies achieve awareness of SE issues within an organisation.* **Technical Skills** *To be defined.*

ARC27 Regulatory and Legal Compliance		
Ensures that all relevant regulations and legal requirements and organisation -specific standards are satisfied by the organisation by determining the requirements, encapsulating the requirements into the SE management system and managing the interface with the Regulator including successful conflict resolution.		
Producer	**Supervisor**	**Manager**
Is aware of the requirements of the relevant functional standards appropriate to the industry sector. Can describe and explain the key principles underlying the relevant regulatory regime and associated legal issues.	*Can illustrate, through corporate management procedures, how regulatory requirements and associated legal issues have been reflected in the organisation's SE management system.*	*Can illustrate, through memos, reports and SE management procedures, how the requirements of the regulatory authorities are continually reviewed, and where appropriate incorporated, within the organisation's SE management system.* **Technical Skills** **TS11 Quality Management** Sets quality strategy for SE, for approval and adoption by organization management. Prioritises areas for quality and/or environmental improvement in light of strategy, wider business objectives and advice from team. Measures achievement of the quality policy in terms of meeting the organization's needs and objectives and reviews it as necessary. Initiates the application of appropriate quality management techniques in these areas. Initiates improvements to processes by changing approaches and working practices, typically using recognized models. Achieves and maintains compliance against national and international standards if 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. Plans, resources (either directly or indirectly) and monitors the internal quality audit schedule. Defines and reviews SE quality and environmental systems. Ensures that adequate technology, procedures and resources are in place to support the quality system. Facilitates improvements to processes by changing approaches and working practices, typically using recognized models. Advises on the application of appropriate quality and/or environmental management techniques to any aspect of SE.

ARC30 Scope and Context Appreciation		
Acquires an appreciation of the context of a system and establishes the scope and objectives of an assessment, such that all necessary requirements of an assessment are capable of being satisfied.		
Producer	**Supervisor**	**Manager**
Can identify the main categories of information (e.g. system boundaries, technical expertise, organisational boundaries, applicable standards) that would be required to adequately define the scope of a typical assessment carried out within the industry sector and describe how this information would typically be obtained and evaluated. **Technical Skills** *To be defined.*	*Can illustrate through design documents, interview notes and minutes, how the necessary information has been collected to correctly define the scope of an assessment. Can explain how issues with scope and context of a system are important in the performance of an effective independent assessment.* **Technical Skills** *To be defined.*	
Can describe the main system elements that make up a typical system that could be the subject of an independent assessment.	*Can illustrate through system block diagrams, how the interrelationships between system elements of a system have been analysed in order to define the scope of the elements of a system to be assessed.*	

E.4 Support Infrastructure Management Function

Summary

The Support Infrastructure Management function introduces a precedent within the SE model, as it is concerned with activities traditionally considered as being external to the SE process. The supporting infrastructure is defined via 2 SysE activities and its subsequent realisation establishes by default the requirement thereafter for it to be managed. Key elements requiring management within this function are:

- Software Platform/Prime Operating Equipment
- Operating Software
- Users
- Software Operations
- Technical Training
- Test and Support Equipment
- Supply Support
- Maintenance Facilities
- Technical Data
- Transportation and Handling Equipment

Support Infrastructure Management Function Related Competencies

See FRC3, FRC6, FRC7, FRC8, FRC10, FRC11, FRC14, FRC17, FRC20.

Software Platform/Prime Operating Equipment Related Competencies

Goal 1: Localised quality assurance and configuration management of uploaded software, work instructions, operating procedures and processes is undertaken.

See ARC26, NC7, NC8.

Operating Software Related Competencies

Goal 1: Quality assurance and configuration management of the planning, design and product creation process is undertaken.

See NC3.

Users Related Competencies

Goal 1: Promotion of SST awareness is undertaken.

Goal 2: Individual's competencies and skills are managed.

NC12 Salesmanship		
The research, analysis and stimulation of potential or existing markets for SE products and services, both to provide a sound basis for their development and to generate a satisfactory flow of enquiries.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.* **Technical skills** **TS19 Marketing** Maintains successful internal and external business relationships. Plans and conducts market research. Investigates and analyses customer dynamics and uses research to inform marketing plans. Organises marketing events and drafts marketing support materials such as brochures and mailshots. Works with technical and non-technical customer representatives to identify needs and sales opportunities. Selects from and uses marketing tools appropriate to a project. Maintains database of marketing information. Conducts market research. Contributes to marketing plans.	*To be defined.* **Technical skills** **TS19 Marketing** Responsible for making strategic decisions regarding marketing plans and the planning process. Determines and oversees the overall marketing strategy for the organization to meet its business objectives. Manages marketing campaigns within specified budgets to meet specified objectives. Develops and maintains successful internal and external business relationships. Manages and monitors market research, analysis and the marketing planning process. Overall responsibility for production of marketing materials and staging of events. Finds innovative solutions to marketing problems. Uses experience to make informed recommendations to senior management.

See NC2.

Software Operations Related Competencies

Goal 1: User training, tools and resources to carry out software operations is provide d.

See NC2.

Technical Training Related Competencies

Goal 1: Individuals in the SE group and software -related groups receive the training necessary (e.g. software development, support systems, instructing) to perform their roles. This training is plann ed.

Goal 2: Training for developing the skills and knowledge needed to perform software management and technical roles is provided.

NC13 Software Engineering Training Management		
The overall management responsibility for the development and provision of a teaching/training service, covering all levels and any areas of expertise within the scope of SE.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS29 Education and Training Management** Manage the development and provision of a training service for the organization.

NC14 Training Needs Analysis		
The management of, or provision of expertise in, the development and training of SE practitioners, including determination of training and development needs in line with organisation or business requirements, production of training plans, design and delivery of training, liaison with external training providers, and evaluation of the benefits of training and development activities.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS30 Development and Training** Determine organizational development needs in line with cultural and strategic direction and business requirements. Generates a training strategy to achieve requi red change and outcomes for training materials development.

NC15 Course Design		
The creation of materials for use by SE teachers or SE students as training aids. Training materials covering any subject within the scope of SE and its applications are rel evant.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.* **Technical skills** **TS31 Training Materials Creation** Specifying content and structure of training to deliver agreed outcomes. Author and customise training materials to existing standards.	*To be defined.*

NC16 Instructing		
The teaching of SE knowledge, and techniques and training in SE skills to help students and staff fulfil SE roles.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.* **Technical skills** **TS32 Education and Training D elivery** Customise and deliver training to a variety of audiences using a range of instructional techniques.	*To be defined.* **Technical skills** **TS32 Education and Training Delivery** Customise and deliver specialist training to specialist audiences using a rang e of instructional techniques.

Test and Support Equipment Related Competencies

Goal 1: The development, support, training and administrative environments, and consumable resources are managed and controlled.

NC17 Professional Advice		
The provision of advice, assistance, and leadership in any area associated with the planning, procurement, provision, delivery, management, maintenance or effective use of information systems and their environments. The consultancy can deal with one specific aspect of IS and the business, or it can be wide ranging and address strategic business issues.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS2 Consultancy** Provides organizational leadership and guidelines for provision of specialist knowledge over a range of IS/ICT topics and the role of IS in the business; provides knowledge and advice in own areas of expertise.

NC18 IS Strategy Management		
The management of, and provision of expert advice on a specific technical specialism in ICT. Examples of specialism can be any ICT technology, technique, method, product or application area.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS3 IS Strategy and Planning** Leads the creation or review of an IS strategy which meets the requirements of the business. The development of processes which ensure that the strategic management of IS is embedded in the management of the organization.

NC19 Network Management		
The creation and maintenance of overall network plans, encompassing data, voice, text and image, to underpin IS and ICT strategies in the support of an organization's business strategy. This includes participation in the creation of service level agreements and the planning of all aspects of infrastructure necessary to ensure provision of network services to meet such agreements.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS7 Network Planning** Creates and maintains overall network plans to support the organization's business strategy, agrees service level agreements with customers and plans all aspects of the infrastructure necessary to ensure provision of network services to meet such agreements.

NC20 IS/ICT Procurement		
The management, or provision of advice on, the procurement of IS and/or ICT goods and services.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS8 Procurement** Leads the procurement process from clarifying a specificat ion to placing contracts, including identifying opportunities for business improvement. Investigates the technical and commercial options for fulfilling the requirements, including possible sources of supply, and agrees the preferred options and potential suppliers with the business. Ensures implementation of procurement strategies and evaluation criteria in line with procurement legislation. Where no corporate policy exists, establishes procurement strategy, policy, standards, methods and processes. Manages tender, evaluation & acquisition process. Negotiates with preferred suppliers, drafts contracts and technical schedules, develops acceptance procedures and criteria. Places contracts. Implements, maintains and disseminates procurement strategy, pol icy, standards, methods and processes. Agrees and meets budgets for the procurement of products and services.

NC21 SE Asset Management		
The management of the inventory of SE assets (hardware, software and user intellectual) held within an organisation, aiming to optimise the total cost of ownership of all SE assets, by minimising operating costs, improving investment decisions, and capitalising on potential opportunities.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS14 Asset Management** Plan audit of SE resources, set criteria for evaluating effectiveness, agree priority areas for investment, set maintenance procedures. Manages hardware, software and design assets held by the organisation to maximise the availability and reuse whilst minimising costs.

NC22 IS Harmonisation		
The co-ordination of IS matters where the adoption of a common approach would benefit the organisation. This could be within a large organisation in which the IS function is devolved to autonomous units, or within a collaborative enterprise of otherwise independent organisations.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS16 IS Co-ordination** Establish organisational objectives for the coordination of IS/ICT matters as part of a wider corporate IS strategy, inspire development of partnerships and collaborative arrangements with other organisations in order to maximise exploitation of IS/ICT resources. Develop common approaches or organisational standards governing use, security, maintenance and acquisition of IS/ICT resources in line with IS organizational objectives.

NC23 SE Support System Management		
The overall management of the resources required to plan for, develop, deli ver and support properly-engineered information and telecommunication system services and products to a customer or customers, including the management of change and the maintenance of professional standards. The management of performance of ICT -related systems and services in relation to their contribution to business performance.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS17 ICT Management** Set strategy for management of IS/ICT resources, authorise alloc ation of resources for the planning, development and delivery of all information systems services and products, authorise organisational porticoes governing the conduct of management of change initiatives and standards of professional conduct, maintain ove rview of contribution of programme to organisational success. Inspires creativity and flexibility in the management and application of IS/ICT. Set strategy for monitoring and managing the performance of ICT -related systems and services, in respect of the ir contribution to business performance and benefits to the business. Identify and manage resources needed for the planning, development and delivery of specified information and communications systems services and products, influence senior level custome rs and project teams through change management initiatives, ensure that professional standards are maintained. Planning and managing implementation of processes and procedures, tools and techniques for monitoring and managing the performance of ICT -related systems and services, in respect of their contribution to business performance and benefits to the business.

NC24 Technical Direction		
The provision of direction and guidance on all technical aspects of the development of, and modifications to, in formation systems to ensure that they take account of relevant ICT technical strategies, policies, standards and practices and that they are compatible with existing and planned systems and ICT infrastructure.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS20 Technical Authority** Is responsible for specifying and implementing technical standards in tools, methods and processes in ICT systems projects.

NC25 SE Support System Control		
The day-to-day support, operation and control of all equipment within an IT or telecommunications network infrastructure. Includes data backup and restore, production of network performance statistics, provision of network diagnostic information, and site surveys.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.*	*To be defined.* **Technical skills** **TS34 Network Control** Plans upgrades and improvements to networking capability, supporting and maintaining hardware and software as necessary. In telecomms, this may include remot e testing and adjustment, and monitoring of network traffic for data capture to enable customer billing.

NC26 Capability Provision		
The matching of hardware, software, network and overall system capability and capacity to meet current and predicted needs for ICT services in a cost effective manner, and to cope with varying levels of traffic on telecommunications networks.		
Producer	**Supervisor**	**Manager**
To be defined.	*To be defined.* **Technical skills** **TS35 Capacity Management** Monitors, maintains capacity in storage and hardware availability to ensure adequate system performance. In telecomms, monitors levels of traffic on a network and ensures sufficient capacity.	*To be defined.* **Technical skills** **TS35 Capacity Management** Is responsible for providing adequate capacity to provide system or network performance. Examines issues pertaining to capacity and plans and recommends changes to hardware and software to ensure adequate system or network performance. In telecomms, ensures sufficient capacity to cope with varying levels of network traffic.

NC27 IS Security Management		
The authorisation and monitoring of access to any part of the IS facilities or infrastructure in accordance with established organizational policy. Includes investigation of unauthorized access, compliance with data protection and the performance of other administrative duties relating to security management.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical skills** **TS36 Security Administration** Maintains security records and access levels to organisation assets.	*To be defined.* **Technical skills** **TS36 Security Administration** Maintains security records and access levels to organisation assets. Monitors access to organisation IS facilities to ensure proper usage. Maintains inventory of personal data and deals with access requests according to data protection legislation. Drafts and maintains procedures, checklists and employee data protection legislation awareness materials.	*To be defined.* **Technical skills** **TS36 Security Administration** Examines and investigates security breaches and makes recommendations for policy improvements. Drafts and maintains policy standards for data protection, specifies and advises on appropriate systems architecture design, defines procedures for disclosure o f personal information, deals with complaints and advises management on compliance with data protection.

NC28 ICT Administration		
The administration and operation of hardware and software in support of the delivery of an agreed ICT service. Includes inc ident handling, availability and performance monitoring, routine start-up/close-down, and the maintenance of operating plans and schedules.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical skills** **TS38 ICT Operations** Carries out routine operation of hardware and software. Operates and maintains equipment under instruction.	*To be defined.* **Technical skills** **TS38 ICT Operations** Provides technical expertise to operations management and staff. Contributes to the planning of operational and maintenance schedule. Executes operational tasks to meet schedules and targets. Enables the deployment of operational resources in order to meet service levels. Evaluates results of implementation. Contributes to the planning of installation and upgrade work. Provides support to the team.	*To be defined.* **Technical skills** **TS38 ICT Operations** Ensures operational performance is maintained at agreed levels. Provides technical operations management. Plans operational and maintenance schedule to meet business demands. Provides operational input on planning of installation and upgrade work.

NC29 Network Administration		
The provision of day-to-day network administration and support, including resolution of network user problems.		
Producer	**Supervisor**	**Manager**
To be defined. **Technical skills** **TS41 Network Administration and Support** Assists with identification and implementation of remedial solutions for hardware, software components and subassemblies.	*To be defined.* **Technical skills** **TS41 Network Administration and Support** Day to day network administration, resolving network problems and enabling agreed levels of support to be met. This will require familiarity with telecommunications concepts and protocols.	*To be defined.*

See TS21, TS22, TS23, TS24.

See NC2, NC4, NC5, NC6.

Supply Support Related Competencies

Goal 1: All spares, parts, consumables, special supplies, warehousing, material distribution and personnel needed to support prime mission-oriented equipment are managed and controlled.

See NC21.

Maintenance Facilities Related Competencies

Goal 1: Physical plant, personnel accommodation, real estate, furniture, ablutions, rest rooms, lighting, heating, power, air conditioning, fire suppression systems and building security are managed and controlled.

See NC17, NC18, NC19, NC20, NC21, NC23, NC25, NC26, NC27, NC28, NC29.

Technical Data Related Competencies

Goal 1: All data necessary for the performance of system operation and maintenance functions is managed and controlled..

See TS21, TS22, TS23, TS24.

See NC2, NC4, NC5, NC6, NC17, NC18, NC19, NC20, NC21, NC22, NC23, NC24, NC25, NC26, NC27, NC28.

Transportation and Handling Equipment Related Competencies

Goal 1: All special provisions required to support the prime mission -oriented equipment and software are managed and controlled.

See NC21.

Appendix F – Assessment Pro-forma

FUNCTION			Reference

Context Summary

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

Assessor	Print name	Signature	Date
Candidate	Print name	Signature	Date

Competency Statement:		
Summary of evidence provided, including context.	Evidence Type:	

Manager	
Supervisor	
Producer	

Competency Statement:		
Summary of evidence provided, including context.	Evidence Type:	

Manager	
Supervisor	
Producer	

Competency Statement:		
Summary of evidence provided, including context.	Evidence Type:	

Manager	
Supervisor	
Producer	

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

Appendix G – Completed HSMU Assessment Pro-forma

G.1 Candidate A

FUNCTION	System Engineering	ACTIVITY	Definition of Problem/Query evaluation	Reference SNCO Requirements

Context Summary

Maintenance of an avionics real-time assembler based application.
Tasking is externally driven.
Task is internally evaluated.

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

Assessor	Print name	Signature	Date
Candidate	Print name Candidate A	Signature	Date

Competency Statement: ARC7 Defining the Scope of the Project

Summary of evidence provided, including context. | Evidence Type: OR/DC

Has produced operational requirements documents for defining the development of individual projects.
Has created, maintained and utilised this information – evidence supplied in form of files, meeting minutes and letters.

Manager	X
Supervisor	X
Producer	X

Competency Statement: ARC14 Identification of End-User Requirements

Summary of evidence provided, including context. | Evidence Type: OR/DC

Has produced operational requirements documents for defining the development of individual projects.
Has created, maintained and utilised this information – evidence supplied in form of files, meeting minutes and letters.
Meets Technical Skill TS25.

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC24 Planning		
Summary of evidence provided, including context.	Evidence Type:	
Not carried out by this post.		

Manager	
Supervisor	
Producer	

Competency Statement: NC1 Query Evaluation		
Summary of evidence provided, including context.	Evidence Type:	OR
Runs Query Answering service. Has identified problems in existing QA service and is currently developing new database and procedures (TS37).		

Manager	X
Supervisor	X
Producer	X

Competency Statement: NC2 Helpdesk Administration		
Summary of evidence provided, including context.	Evidence Type:	OR/DC
Receives and documents customer queries. Manages helpdesk. Advises senior management on improvements to the QA system.		

Manager	X
Supervisor	X
Producer	X

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

FUNCTION	System Engineering	ACTIVITY	System Feasibility Analysis	Reference SNCO Requirements

Context Summary

Maintenance of an avionics real-time assembler based application.
Tasking is externally driven.
Task is internally evaluated.

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

	Print name	Signature	Date
Assessor			
Candidate	Print name Candidate A	Signature	Date

Competency Statement: ARC10 Evaluating Solutions

Summary of evidence provided, including context. | Evidence Type: OR

Evaluates proposed changes against existing system in order to identify alternative courses of action, ultimately mandating the direction of the change.
Has done this on TIALD and Maverick changes.

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC15 Influencing New Systems

Summary of evidence provided, including context. | Evidence Type: OR

Has influenced the design of Maverick/OFP interface software in order to address supportability issues.
Understands the importance of interfaces as key functional support areas.
Has influenced the means by which the ADR data is decoded and analysed in order to address flight safety issues.

Manager	X
Supervisor	X
Producer	X

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

FUNCTION	System Engineering	ACTIVITY	Functional Analysis	Reference SNCO Requirements

Context Summary

Maintenance of an avionics real-time assembler based application.
Tasking is externally driven.
Task is internally evaluated.

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

	Print name	Signature	Date
Assessor			
Candidate	Print name Candidate A	Signature	Date

Competency Statement: NC30 System Structural Analysis

Summary of evidence provided, including context.	Evidence Type:	OR

Identifies system interfaces through the use of interface control documents.
Sets out system constraints in the operational requirement document.

Manager	
Supervisor	X
Producer	X

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

Candidate A notes

There was a difference in understanding of the terms product and project. A product in HSMU terms equates to an individual change, whereas a project equates to a collection of products.

ARC7 (Defining the Scope of the Project) is a SNCO Requirements (Functional Analysis/System Feasibility Analysis) competency and not a Queries and Prototypes one. Likewise, the Queries and Prototypes SNCO does not carry out the ARC24 (Planning) competency for the Definition of Problem activity.

Managing the Query Evaluation service is the responsibility of SNCO SST3.

G.2 Candidate B

FUNCTION	Supporting Activities	ACTIVITY	Change Management	Reference Deputy Configuration Manager

Context Summary

Maintenance of an avionics real-time assembler based application.
Tasking is externally driven.
Task is internally evaluated.

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

Assessor	Print name	Signature	Date
Candidate	Print name Candidate B	Signature	Date

Competency Statement: ARC13 Handling Change		
Summary of evidence provided, including context.	Evidence Type:	

Not assessed against this competency due to erroneous oversight on part of assessor.

Manager	
Supervisor	
Producer	

Competency Statement: NC7 Change Management		
Summary of evidence provided, including context.	Evidence Type:	OR

Has applied company configuration control procedures to OFP software.

Manager	
Supervisor	
Producer	X

Competency Statement: NC8 Configuration Control		
Summary of evidence provided, including context.	Evidence Type:	OR
Has utilised Dimensions configuration control tool to manage HSMUs C/Is. Has produced change document reports to facilitate status accounting.		

Manager	
Supervisor	X
Producer	X

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

Candidate B notes

Undertakes duties relating to NC8 (Configuration Control) more than NC7 (Change Management).

G.3 Candidate C

FUNCTION	Supporting Activities	ACTIVITY	Software Engineering Management	Reference * OC OSEF *(OC HSMU)*

Context Summary

Maintenance of an avionics real-time assembler based application.
Tasking is externally driven.
Task is internally evaluated.
* Normal role is OC OSEF and is deputising for OC HSMU.

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

Assessor	Print name	Signature	Date
Candidate	Print name Candidate C	Signature	Date

Competency Statement: ARC1 Allocation of Responsibilities		
Summary of evidence provided, including context.	Evidence Type:	OR

Has produced and reviewed TORs for HSMU personnel.
Has managed the HSMU organisational structure to optimise resources.

Manager	X
Supervisor	X
Producer	X

Competency Statement: ARC5 Assuring Staff Competency		
Summary of evidence provided, including context.	Evidence Type:	OR

Has instigated HSMUs competency and training matrix for personnel and directed and monitored the activities of training personnel.

Manager	X
Supervisor	X
Producer	X

Competency Statement: ARC19 Managing Resource Allocation		
Summary of evidence provided, including context.	Evidence Type:	OR
Has managed resource allocation for HSMUs Technical Expert (OC HSMU).		

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC28 Resource Allocation		
Summary of evidence provided, including context.	Evidence Type:	
Not carried out by this post.		

Manager	
Supervisor	
Producer	

Competency Statement: ARC29 Risk Assessment		
Summary of evidence provided, including context.	Evidence Type:	OR
Is Technical Expert on risk management for HSMU. Has produced a generic risk management plan for all HSMU activities. Has been involved in risk assessment activities and currently leads these activities. Has signed off risk assessment reports.		

Manager	X
Supervisor	X
Producer	X

Competency Statement: NC9 Software Engineering Project Management		
Summary of evidence provided, including context.	Evidence Type:	OR
Has conducted project management activities on a number of projects.		

Manager	
Supervisor	X
Producer	X

Competency Statement: NC10 Service Level Management		
Summary of evidence provided, including context.	Evidence Type:	OR
Has reported on quality in several projects and identified future preventative action. Manages test rig resources in order to meet service level requirements of HSMU.		

Manager	
Supervisor	X
Producer	X

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

Candidate C notes

Does not undertake ARC28 (Resource Allocation) as this is carried out by SNCO OFP Production.

G.4 Candidate D

FUNCTION	System Engineering	ACTIVITY	System Test and Evaluation	Reference OFP Tester

Context Summary

Maintenance of an avionics real-time assembler based application.
Tasking is externally driven.
Task is internally evaluated.
Currently not fulfilling a SNCO testing post.

Assessment Summary

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Manager																
Supervisor																
Producer																

Action Plan

	Date for next assessment	

	Print name	Signature	Date
Assessor			
Candidate	Print name Candidate D	Signature	Date

Competency Statement: ARC2 Analysing Test Results		
Summary of evidence provided, including context.	Evidence Type:	OR

Has participated in the production and analysis of test reports.
Has participated in the production and application of test specifications.

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC3 Analysing the Code		
Summary of evidence provided, including context.	Evidence Type:	OR

Has participated in the testing of requirements through static analysis techniques.
Has carried out dynamic code testing using software development rig.
Has applied black and white box testing techniques to OFP systems.
Is training in SE to degree standard.

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC4 Analysing the Design		
Summary of evidence provided, including context.	Evidence Type:	OR

Has analysed and reviewed high level design requirements for OFP changes.
Has applied logical testing technique check sheets to design requirements.

Manager	
Supervisor	
Producer	X

Competency Statement: ARC11 Executing Tests		
Summary of evidence provided, including context.	Evidence Type:	OR

Has applied tests to systems and produced test logs.

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC12 Forming a Judgement		
Summary of evidence provided, including context.	Evidence Type:	OR

Has conducted functional testing and is involved in the OFP coding process, analysing it from an implementer's perspective.
Has contributed to the production of test analysis logs that present an argument to justify a set of conclusions that categorise errors.
Has made recommendations that influence the design and code.

Manager	X
Supervisor	X
Producer	X

Competency Statement: ARC17 Managing in-Service Information		
Summary of evidence provided, including context.	Evidence Type:	

Not carried out by this post.

Manager	
Supervisor	
Producer	

Competency Statement: ARC23 Performing Analysis		
Summary of evidence provided, including context.	Evidence Type:	OR
Has produced test analysis logs that illustrate the application of test analysis techniques. Has applied BVA techniques to the OFP system to compliment a functional timing check.		

Manager	X
Supervisor	X
Producer	X

Competency Statement: ARC31 Specifying Software Tests		
Summary of evidence provided, including context.	Evidence Type:	OR
Has specified design strategy test specifications in order to demonstrate compliance with requirements.		

Manager	
Supervisor	X
Producer	X

Competency Statement: ARC32 Specifying Tests		
Summary of evidence provided, including context.	Evidence Type:	
Not carried out by this post.		

Manager	
Supervisor	
Producer	

Competency Statement: ARC35 Witnessing and Executing Tests		
Summary of evidence provided, including context.	Evidence Type:	OR
Has witnessed the execution of qualification tests undertaken by third parties. Has applied HSMUs test schedule.		

Manager	
Supervisor	
Producer	X

Evidence	Assignment Project	Competence Skills/Tests	Documentary Records	Oral	Workplace Observation	Witness Testimony
Code	AP	CT	DC	OR	WO	WT

Candidate D notes

Was an OFP tester before being recently promoted to Sgt.
Analysing the Design (ARC4) any further than Producer level is outside of responsibilities.

ARC17 (Managing in-Service Information) is not undertaken by this post.
ARC32 (Specifying Tests) is conducted by 3rd Party organisations (QinetiQ, OEU).

Appendix H – Proposed Statements for Unpopulated Competencies

NC30 – System Structural Analysis

Producer – Has produced system analysis documents.

Supervisor – Has produced system analysis documents. Understands the reasoning and principles behind production decisions concerning system analysis documents.

Manager – Manages the production of system analysis documents through adherence to the approved production process. Authorizes changes to that process.

NC7 – Change Management

Producer – Utilizes configuration control tools to manage C/Is. Applies documented change control procedures. Grants access to systems procedures.

Note: In TS5, assessment and analysis is more a requirements-oriented skill than a change management one.

NC9 – SE Project Management

Supervisor – Concerned with the practical application of project management – using tools, identifying and recognizing constraints, management of SE process as opposed to management of product. Implementing the SEMS.

Producer – Is aware of the process and its impact on the working environment.

Note: Manager role normally undertaken by OC HSMU, not OC OSEF.

NC10 – Service Level Management

Note: Manager role normally undertaken by OC HSMU, not OC OSEF.

ARC4 – Analysing the Design

Note: The “analysis technique” used by HSMU is a logical testing technique check sheet.