AVIONIC SYSTEMS STANDARDISATION COMMITTEE

PROCESSING WORKING GROUP

Doc No : ASSC/330/2/159 *

Issue 1 - January 1999

EMERGING COMMERCIAL SOFTWARE DEVELOPMENT STANDARDS *

0 EXECUTIVE SUMMARY

The introduction of the standards ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 will have an impact on software development strategy for ASSC member companies. ERA has been tasked by the ASSC to review these three standards, compare them to other software development standards and assess their impact on ASSC member companies.

This report is the result of ERA's investigations. ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 are described and contrasted. Then they are briefly compared to other software development standards used by ASSC members. They are found to be fundamentally different to those other software development standards, apart perhaps from MIL-STD-498. The responses to a questionnaire circulated amongst ASSC member companies are presented.

It is found that ISO/IEC12207 is not currently being called up in ASSC member company contracts. However, there is little doubt that it will have a considerable impact on member companies and their software trade. It is concluded that the adoption of ISO/IEC12207 for military avionics software trade should be advanced, building upon the progress already made in IEEE/EIA12207 but also specifically considering the UK and military avionics contexts.

Table of Contents **Page no.**

Abbreviations List

COTS		Commercial Off The Shelf
DID		Data Item Description
DOD		Department of Defense
EIA		Electronic Industries Association
IEC		International Electrotechnical Commission
IEEE		Institute of Electrical and Electronic Engineers
ISO		International Organization for Standardization
OOD		Object Oriented Development
RAD		Rapid Application Development
SPICE		Software Process Improvement Capability dEtermination

1 Introduction

The introduction of the standards ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 will have an impact on software development strategy for ASSC member companies. ERA has been tasked by the ASSC to review these three standards, compare them to other software development standards and assess their impact on ASSC member companies.

This report is the result of ERA's investigations. ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 are described and contrasted. Then they are briefly compared to other software development standards used by ASSC members. They are found to be fundamentally different to those other software development standards, apart perhaps from MIL-STD-498. The responses to a questionnaire circulated amongst ASSC member companies are presented.

1.1 Objective of Task

The objectives of the task are as follows.

- To review the scope of ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 and to describe their relationships and key differences.
- To provide some comparison of these standards with existing MoD and other commonly used software development standards.
- To ascertain the extent of use of ISO/IEC12207 amongst ASSC member companies.
- To assess the impact on UK companies of the DoD's adoption of commercial standards for software development.

1.2 Background

There are a number of strands to the background of the development of the related standards ISO/IEC-12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016. Firstly, in recent years, the American Department of Defense (DoD) has shifted towards the use of commercial, rather than military, standards. IEEE/EIA-12207 and J-STD-016 were developed with this policy shift in mind.

Secondly, there was a general appreciation of the over-proliferation of frameworks for managing and engineering software. ISO/IEC12207 was an international response to this and attempts to provide a single, universally applicable framework that can be instantiated for individual organizations or projects as necessary, by tailoring and the integration of other standards, guidance, etc.

Thirdly, many problems had been identified with the predecessors of ISO/IEC12207. For example, previous standards tended to:

- focus solely on software development, with only implicit provisions for the processes of operation and maintenance,

- focus solely on the software developer, neglecting the non-technical roles of acquisition and supply,
- focus on a single contract or project in isolation, rather than viewing software in the context of its enclosing system,
- describe a single, monolithic process, rather than describing a flexible set of processes that could be configured to specific needs and
- entail additional costs, estimated at 20-50%, in documentation and formal reviews, by addressing those activities at an excessively prescriptive level rather than dealing with principles and objectives (ref.7).

A particular example is DOD-STD-2167A which; was not sympathetic to RAD and OOD or any development method significantly different to the waterfall model, entailed cumbersome review processes, made an over-rigid distinction between requirements and design and placed an over-emphasis on paper documents. The response to these particular problems was the development of MIL-STD-498[1]. Many of the key people involved in the development of MIL-STD-498, notably manager Raghu Singh (Project Editor of the ISO initiative), were also prominently involved in the development of ISO/IEC-12207 and hence this latter standard was also able to address many of the problems of DOD-STD-2167A.

1.3 Report Structure

Section2 reviews the standards ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 in turn. Section2.4 considers common features of the three standards. Section3 gives a short comparison of the three standards with other software development standards. Section4 presents the results of the questionnaire circulated to obtain information on the use of various software development standards. Section5 presents conclusions and recommendations. AnnexA describes the processes of ISO/IEC12207. AnnexB summarizes the differences between MIL-STD-498 and EIA/IEEEJ-STD-016. Finally, the questionnaire is contained in AnnexC.

2 ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016

In this section, the three standards that are the focus of this report, ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016, are described in sequence. At the end of the section, features common to all three standards are discussed.

[1] In response to the development of the alternative standards described in this report, MIL-STD-498 has been cancelled.

2.1 ISO/IEC12207

ISO/IEC12207 (ref.2) is an attempt to provide a very widely applicable, best-practice framework for software life cycle processes within which other standards, procedures, etc. can be integrated. The standard was published in1995, having grown out of a wide range of other standards, frameworks, etc., including the ISO9000 series, IEEE 1074 ("IEEE standard for developing software life cycle processes") and MIL-STD-498. It is, however, very different to those predecessors.

ISO/IEC12207 is claimed (ref.1) to be the first international standard that provides a complete set of processes for acquiring and supplying software products and services. It encompasses both technical and commercial concerns and applies to acquirers, suppliers, developers, operators and maintainers (these are the five primary processes in the standard — see AnnexA). The standard is intended for use because of internal policy directive or as specified by contractual agreement. It can be applied to individual projects. It is also claimed (ref.1) that it can be applied to organizations (such that a whole organization is compliant and all suppliers to that organization will be compliant)[2].

One potentially important constraint on the standard is that it is not intended to be applicable to the purchase of COTS software products (ref.2, Paragraph1.2), though it is not immediately clear why this should be the case. The standard is primarily intended to be applied in the context of a two party contract. Perhaps, the use of such a contract for the purchase of COTS is considered excessive and hence the standard is considered inappropriate. In any case, there does not appear to be any reason why ISO/IEC-12207 should not be applied, voluntarily, to the development, operation and maintenance of COTS software, if not its acquisition and supply.

The standard encapsulates a high level life cycle architecture for software from conception to retirement (not just development). The architecture is built from defined processes and interrelationships between those processes[3]. Each process describes responsibilities that must be achieved and maintained during the life of that process (in contrast to the step-by-step requirements characteristic of procedures). The processes are listed and described in AnnexA of this document. Processes consist of activities which are cohesive sets of tasks. A task is an elementary responsibility or action.

A key feature of ISO/IEC12207 is that it should be tailored for application to individual projects and organisations. For any particular project, or organization, it is intended that a subset of processes, activities and tasks be chosen. Compliance with the standard is achieved by complying to a properly tailored subset of the standard. Normative rules and guidance are given for this tailoring process.

ISO/IEC12207 tends to address the functions to be performed rather than the organizations that execute them. For example, the standard describes a quality assurance process but this does not imply that a conforming organization must establish a quality assurance department.

---

[2] This claim is contradicted by proponents of IEEE/EIA12207 who assert that a major motivation for developing IEEE/EIA-12207 was to provide a version of12207 applicable to organizations (ref.7).

[3] Whereas the basic units of the CMM (ref.5), and its derivatives, are *requirements* that good processes meet.

There are very few specific requirements on documentation (ref.2, Paragraph1.5), for example there are no DIDs as in MIL-STD-498. Existing company documentation standards can be used in the context of ISO/IEC12207.

ISO/IEC12207 is very high level[4] and abstract, and consequently can be difficult to apply (especially in a general organizational sense). JTC1SC7WG7, the author of ISO/IEC12207, is developing a guidebook on the standard, detailing basic concepts, application of the standard and guidance for tailoring. Various other guides and related and supporting standards (for example the ISO/IEC 12220 series, SPICE) are also being developed which will help in the implementation of ISO/IEC 12207.

Future revision of ISO/IEC12207 is likely to position the software life cycle processes in the context of an overall system life cycle (ref.6).

2.2 IEEE/EIA12207

IEEE/EIA12207 (ref.4) is a compliant American adaptation of ISO/IEC12207 developed by the EIA and the IEEE in collaboration with the DoD, to suit current American practice. Essentially, it is ISO/IEC 12207 plus extra material heavily influenced by the engineering and data requirements in MIL-STD-498 and J-STD-016. Most of the extra material is concerned with the development process, rather than the other primary processes. The objectives of IEEE/EIA12207 are:

- to represent best commercial practice,
- to be suitable for application in defence acquisition and
- to be compatible with other standards in the global marketplace for software.

The intention is that, in the fullness of time, from the process viewpoint, IEEE/EIA12207 will serve as a single entry point to all the standards of the IEEE software engineering collection (ref. 7).

Proponents of the IEEE/EIA standard consider that its most important improvement over ISO/IEC12207 is a set of alternatives for conformance with the standard, besides just project conformance (ref. 7). A total of four situations are addressed.

- An organization adopts the standard and implements policies, procedures and infrastructure to implement the provisions of the standard.
- A project directly conforms to the standard.
- A multi-supplier program adopts the standard and, viewed as a whole, conforms to the provisions of the standard regardless of whether individual suppliers qualify for conformance.

---

[4] For example, MIL-STD-498 and its Data Item Descriptions (DIDs) contain 117pages of engineering and data requirements that correspond to the seven page development process in ISO/IEC 12207 (ref.8).

- A regulatory program conforms to a tailored version of the standard provided by the regulator.

The preferred usage is that an organization develops its own set of processes and procedures that comply with all the requirements of IEEE/EIA 12207 and are applied across the organization. For any individual project conducted by the organization, a selection of appropriate processes and procedures is chosen and parameterized for application on that project (ref.6). Hence, as compared to ISO/IEC 12207 (and MIL-STD-498), the focus of the standard is more on compliance at an organizational level, rather than for individual projects. Partly as a result of this, IEEE/EIA12207 is more discouraging of tailoring than ISO/IEC 12207. This in turn will make organizational claims of compliance to IEEE/EIA12207 more meaningful than similar claims of compliance to ISO/IEC12207.

It may also be that the restriction of ISO/IEC12207 concerning COTS (Section2.1 refers) does not apply to IEEE/EIA12207. This is because the emphasis is more on voluntary single party adoption rather than two party contractual agreements.

IEEE/EIA12207 is packaged into three parts, designated12207.0, 12207.1 and12207.2. Part0 consists of the full text of ISO/IEC 12207 together with extra American material, including:

- a set of process and data objectives that assist in determining the intent of the process requirements specified in ISO/IEC12207 and in guiding adaptation of the requirements in unusual situations and

- a replacement compliance clause that shifts emphasis toward compliance at the organizational level and which requires documentation of the means of compliance.

Part 1 is a guidance document providing recommendations expanding upon the data objectives of Part 0 (described above). Recommendations for the content of various possible documents are given along with references to helpful standards and guidance. Part 2 is a guidance document providing recommendations for the implementation of the 12207 processes in the context of American best practices.

It is noted that some of this extra American content may be included in future revisions of ISO/IEC12207 (ref.6).

2.3 EIA/IEEEJ-STD-016

EIA/IEEE J-STD-016 (ref.3) is an interim or trial-use American standard developed by the same group that produced IEEE/EIA 12207, and was published in January 1996. It is essentially a demilitarized version of MIL-STD-498 that removes government language (such as references to the Federal Aviation Requirements) and introduces concepts and terminology from international standards (detailed differences between MIL-STD-498 and J-STD-016 are given in AnnexB). With its roots in MIL-STD-498, J-STD-016 focuses very much on development rather than the other primary processes of ISO/IEC 12207 (despite the presence of “acquirer” & “supplier” in the title of J-STD-016). However, it has been described as the first step in the implementation of ISO/IEC12207 and future versions are intended to implement additional software life cycle processes and supplier roles, such as the operator and maintainer.

In a sense, J-STD-016 is a bridge between existing practice encapsulated in previous defence standards (MIL-STD-498) and IEEE/EIA 12207 practice. Since it is recognized that the implementation of organization level processes (in the style of IEEE/EIA 12207) can be very expensive and time-consuming, the choice is available, in the short term, to adopt J-STD-016 which may be more closely related to currently implemented processes. J-STD-016 is suitable for use in continuing projects that began under MIL-STD-498 or one of its predecessor military standards, or by companies that have implemented organizational processes based on MIL-STD-498 or one of its predecessors.

J-STD-016 defines a set of software development activities and resulting software products. It provides a framework for software development planning and engineering. J-STD-016 incorporates twenty two software product descriptions derived from the twenty two MIL-STD-498 Data Item Descriptions (DIDs). Like ISO/IEC12207, J-STD-016 is meant to be tailored to ensure that only necessary and cost-effective requirements are applied. Unlike ISO/IEC12207, J-STD-016 fully covers software reuse (COTS).

J-STD-016 was due to be balloted for updating to a new version in 1998 but it is believed that development of the standard is behind schedule.

2.4 Common Features of ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016

Since IEEE/EIA12207 is an adaptation of ISO/IEC12207, and IEEE/EIA12207 and EIA/IEEEJ-STD-016 were written by the same working group, it is not surprising that all three standards share many similarities. These similarities are described in this section. It is worth noting that most of the crucial differences that set apart ISO/IEC12207, IEEE/EIA12207, and J-STD-016 from related software life cycle standards, frameworks, etc. are shared among the three standards and hence are listed below.

ISO/IEC12207, and IEEE/EIA12207 and J-STD-016 are expressed in contractual terms to facilitate their application. All three standards attempt to avoid temporal constraints, that is explicit or implicit time-ordering of events. In part, this is achieved by moving away from large formal reviews which tend to cause bottlenecks in projects.

The standards are applicable to any size or complexity of project, to stand-alone or embedded/integrated software, in any market sector (commercial, industry, defence, etc.). All three standards can be used within an organisation or between two parties.

The standards are compatible with ISO9001.

All the standards, to varying degrees, concentrate on what to do, not how to do it (ISO/IEC12207 most exhibits this characteristic, followed by IEEE/EIA12207, thenJ-STD-016). In particular, the standards do

not mandate or prefer any particular life cycle model[5] (unlike, for example, DOD-STD-2167A which leant towards the waterfall model), design method, programming language, documentation standard[6], etc.

Specification of management indicators (such as cost expenditure) and software attributes (such as reliability, maintainability, etc.) is required or encouraged, but no detail is prescribed. In particular, no metrics or even use of metrics is prescribed.

3 Comparisons

In this section, we compare ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 with MoD and other commonly used software development standards; specifically JSP188, Def Stan00-55 and RTCA/DO-178B[7]. The differences are so fundamental and pervasive, as will be apparent to anyone familiar with JSP188, Def Stan00-55 or RTCA/DO-178B, that detailed comparisons are nugatory. Hence this section is brief.

The scopes of JSP188, Def Stan00-55 and RTCA/DO-178B are all considerably smaller than those of ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016, and Def Stan00-55 and RTCA/DO-178B do not admit tailoring. Furthermore, JSP188, Def Stan00-55 and RTCA/DO-178B are all considerably more prescriptive than ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016. In the case of JSP188, "Specification for Technical Publications for the Services - Documentation of Software in Military Operational Systems", the content and format of required documentation is more completely specified. Def Stan00-55 and RTCA/DO-178B are more prescriptive in their mandating of techniques.

4 Analysis of Questionnaire Responses

As part of the task documented in this report, a questionnaire was prepared and circulated to ASSC member companies. Four responses were received and two respondents discussed their responses with the author. ERA gratefully acknowledges the assistance provided by the respondents.

The questionnaire is quoted in full in AnnexC. This section summarizes and analyses the responses to questions5–12 from the questionnaire (questions1–4 were concerned with identifying respondents and contact details) and the ensuing conversations. In view of the small number of responses, it is not possible to draw definitive conclusions and the rest of this section should be read with this in mind.

---

[5] It is intended that one or more life cycle models be determined for any given project. IEEE1074 can help in this activity.

[6] There is no requirement for paper documentation. CASE tools can be used.

[7] The comparison of ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 with DOD-STD-2167A and MIL-STD-498 (standards that directly influenced the development of ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016) is covered in Section and AnnexA.

5. Nature of company business:

As expected, the respondents are active in the defence, avionics and aerospace fields

6. How familiar are you with ISO 12207, "Information technology – Software life cycle processes":

Responses varied from “not at all” to “very familiar with contents”. Additional comments included:

- “used [ISO/IEC12207] to help define our processes”
- “To date [deleted] has not been required to work to ISO 12207. We believe that our underlying processes are consisting with this standard.”
- “I have studied the standard because I believe it will be increasingly important for us. However, as yet there is no customer push for us to address it.”

7. Which of the following standards have you developed software to:

All the quoted standards have been used apart from ISO/IEC12207. Additionally, other Def Stans, AQAPs, MIL-STDs and aircraft and customer specific standards were quoted. Those respondents who commented had not seen ISO/IEC12207 invoked by a contract. Respondents indicated that they are still being contracted to work to DOD-STD-2167A and MIL-STD-498.

8. How have you achieved compliance with these standards:

It appears that all the respondents use standard compliant company procedures, tailored and augmented as necessary.

9. What has determined the need for compliance with the standards

Three of the respondents indicated that the drivers were both specific contractual requirements and company quality objectives. The fourth indicated only specific contractual requirements.

10. In what contexts have you used the standards

All the respondents indicated military contexts. Additionally, industry and civil aviation were indicated.

11. How do the software development standards you have used compare in terms of prescriptiveness, ease of application, etc.

12. Any other comments ...

Responses to questions11 and12 blurred and hence are presented together here. None of the respondents had direct experience of working to ISO/IEC12207 but it is possible to extrapolate likely concerns from their comments about the standards with which they had worked. One respondent described the various standards as follows.

RTCA/DO-178B: “pragmatic and flexible”, albeit restricted to safety. Flexible on requirements for documentation.

JSP188: “totally outdated”, incompatible with modern development methods, having little value.

DOD-STD-2167A: “overly proscriptive”, too heavily focused on expensive to produce documentation of little value. Apparently, these problems can be mitigated by tailoring but this requires a cooperative customer. The standard is light in the area of verification, e.g. test coverage.

MIL-STD-498: better, more flexible than DOD-STD-2167A, notably in the area of documentation.

Def Stan00-55: “overly prescriptive”, “unlike DO-178B it describes what you should do rather than what you should aim to achieve”. “To demand compliance with a standard when it is acknowledged that it is beyond the state of the art seems unreasonable”.

The respondents were generally unhappy with the documentation requirements of existing software development standards. Much of the required documentation is considered to be of little use and the focus on paper documentation is also problematic. ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016 should suffer less than the other standards described above, in this respect.

There was also a generally expressed preference for requirements that state what must be done, not how to do it. This virtue is supported within the framework of ISO/IEC12207, IEEE/EIA12207 and EIA/IEEEJ-STD-016.

A generally highlighted problem was that of customers and suppliers interpreting standards differently, or even customers not understanding the standards at all. There is a tendency to call up standards “for forms sake”, with no thought as to what is required or expected. In particular, there is a tendency for customers to call up standards without being prepared or able to put in the necessary effort to tailor those standards, leading to significant problems later in the project lifecycle. Particular problem areas quoted were documentation, the review process, and the level and nature of testing. There is obviously a danger that such high-level abstract standards as ISO/IEC12207 and IEEE/EIA12207 will suffer similarly. However, it is to be hoped that the normative and pervasive status of the tailoring process in these standards and the encouragement they provide to use other more specific standards to “fill in the gaps” will force many of these issues to be confronted early in a project lifecycle.

5 Conclusions

5.1 Moving to Commercial Software Standards for Military Avionics

Following the DoD's decision to move to the use of commercial standards, including software standards, activities have been undertaken to build on the widely respected MIL-STD-498 and to provide a link to commercial and international standards. The result has been the US developed EIA/IEEEJ-STD-016 and IEEE/EIA12207, and the internationally developed ISO/IEC12207. For the US military avionics

software supplier, these standards provide a graded path from the customer-mandated military-specific world of MIL-STD-498 to the commercial world of ISO/IEC12207.

EIA/IEEEJ-STD-016, which is a demilitarised revision of MIL-STD-498, is the first step on the path; specific products of the software development process are described but the standard is meant to be tailored to ensure that only the cost-effective requirements are applied. Concepts and terminology from the international standards are introduced, but the scope is not much extended from MIL-STD-498.

As an international standard intended to cover all forms of software development, ISO/IEC12207 has a much wider scope; “development” is one of five primary processes, with a total of twelve further “supporting” and “organisational” processes. As a result, the standard is claimed to provide a complete set of processes for acquiring and supplying software products and services.

To avoid the imposition of inappropriate requirements, ISO/IEC12207 is intended to be tailored to the meet the needs of a particular project or organisation. However, together with its wide scope, this makes the application of ISO/IEC12207 a difficult task. To assist US companies, the EIA and IEEE, in collaboration with the DoD, have developed IEEE/EIA12207, which is a compliant adaptation of ISO/IEC12207, intended to suit current US practice. Additional material has been added, mostly concerned with the development process, rather than the other primary processes of ISO/IEC12207, bridging the gap between earlier standards and the abstract framework of ISO/IEC12207. In future, it is intended that IEEE/EIA12207 will serve as the entry point for the IEEE software engineering standards, each of which covers a particular aspect of software development.

5.2 UK Adoption of ISO/IEC12207

As a result of the small number of replies received to the questionnaire, only tentative conclusions can be reached about the adoption of ISO/IEC12207 for military avionics by UK suppliers. However, it seems that the adoption of ISO/IEC12207 is at an early stage.

Respondents to the questionnaire showed an understanding of the need to move to a more flexible and comprehensive software engineering standard for military software development. However, little user requirement for compliance to ISO/IEC12207 has been seen yet. Indeed, respondents indicated that DOD-STD-2167A and MIL-STD-498 are still being called up in contracts, although both have been withdrawn by the DoD.

The respondents also commented that standards such as ISO/IEC 12207, which are high-level and require tailoring, place a significant onus on customers (acquirers) who mandate them in a contract. The supplier must both tailor the standard and implement it using detailed procedures. Compliance with ISO/IEC 12207 alone does not impose many restrictions on the tailoring and the detailed procedures. Hence, the tailoring and procedures need to be agreed by suppliers and customers (acquirers) early in the project lifecycle to avoid misunderstanding and confusion later as differing interpretations clash.

5.3 The Way Ahead

Since the nature of ISO/IEC12207 makes its adoption difficult, it seems likely that UK military avionics suppliers could benefit from the standards prepared in the US with DoD collaboration, which are

providing a path towards the use of the international standard. The long term benefits of moving towards ISO/IEC12207 include its flexibility, allowing the use of up to date software engineering techniques and tools. Using software engineering processes described within a common international framework also has potential benefits in the long term.

To move towards ISO/IEC12207, the UK should build on IEEE/EIA12207, rather than repeat the work that has already been carried out in the US. An overriding need is for this standard and its application to be more widely understood.

There are also specific questions that could be addressed by the ASSC in future. These include the following.

- Is IEEE/EIA12207 specific to US practices in any ways that make it inapplicable in the UK?
- Are there additional UK-only requirements that should be taken into account in the application of ISO/IEC12207?
- Are there avionics-specific requirements that should be incorporated into the application of ISO/IEC12207?

As an example of item 2, the safety-specific standards Def Stan00-56 and00-55 could be positioned within the wider framework of 12207.

5.4 Recommendations

The ASSC should prepare a concise handbook on the adoption of ISO/IEC12207 for military avionics software. Drawing on the material contained in this report, the handbook should:

- outline the scope and potential benefits of the adoption of ISO/IEC12207
- draw attention to the US standards which provide a graded path for moving towards ISO/IEC-12207
- address UK and avionics-specific issues.

In addition, the handbook should reference and summarise existing MoD policy on software engineering standards applicable to the development of military avionics.

6 References

1. Raghu Singh, Federal Aviation Administration
International Standard ISO/IEC12207 Software Life Cycle Processes

2. ISO/IEC
Information Technology — Software life cycle processes
ISO/IEC12207:1995(E)
First edition, 1995-08-01

3. EIA/IEEE
Standard for Information Technology, Software Life Cycle Processes, Software Development, Acquirer-Supplier Agreement
J-STD-016-1995
Issued for Trial Use, September1995

4. IEEE/EIA
Industry Implementation of International Standard ISO/IEC12207:1995, (ISO/IEC12207)
Standard for Information Technology — Software life cycle processes
IEEE/EIA12207.0-1996
March1998

5. M.C.Paulk, B.Curtis, M.B.Chrissis and C.V.Weber; Software Engineering Institute
Capability Maturity Model for Software
CMU/SEI-93-TR-24
Version1.1, February1993

6. JamesW. Moore, The MITRE Corporation; PerryR.DeWeese, Lockheed Martin Aeronautical Systems; Dennis Rilling, Joint Logistics Commanders, Joint Group on Systems
U. S. Software Life Cycle Process Standards
ASSC reportASSC/330/3/404

7. JamesW.Moore, The MITRE Corporation
IEEE/EIA 12207 as the Foundation for Enterprise Software Processes
Draft

8. Lewis Gray, Ada PROS, Inc.
ISO/IEC12207 Software Lifecycle Processes

9. Reed Sorensen, Software Technology Support Center
MIL-STD-498, J-STD-016, and the U.S. Commercial Standard
ASSC reportASSC/330/3/405

7 Annex A: ISO/IEC12207 Processes

The processes in ISO/IEC12207 are derived from the principles of system engineering. They are classed as primary, supporting or organizational. Each of these process classes and the seventeen constituent processes is described below. Additionally, the tailoring process is described below. It is worth noting that there are no activities in MIL-STD-498 that correspond specifically to the acquisition, supply, operation, maintenance or training processes in ISO/IEC 12207.

7.1 Primary Processes

The five primary processes correspond to the five key parties defined in the standard. Each of the five primary processes executes other primary processes or is itself executed by one or more of the primary processes. As they interact, the primary processes execute supporting processes defined by the standard.

7.1.1 Acquisition

This process defines the activities of the acquirer, the organization that acquires a system, software product or software service. The process begins with the definition of the need to acquire a product, continues with the preparation and issue of a request for proposals, selection of a supplier and management of the acquisition process through the acceptance of the product.

7.1.2 Supply

This process defines the activities of the supplier, the organization that provides the system, software product or software service to the acquirer. It can be initiated by a decision to prepare a proposal or by entering into a contract or agreement with an acquirer. The process continues with the identification of necessary procedures and resources, and development and execution of plans.

7.1.3 Development

This process defines the activities of the developer, the organization that defines and develops the product. For these purposes, development includes prototyping, and requirements and design analysis, as well as producing products.

7.1.4 Operation

This process defines the activities of the operator, the organization that provides the service of operating a computer system in its live environment for its users. The process covers operation of the software and operational support to users.

7.1.5 Maintenance

This process defines the activities of the maintainer, the organization that provides the service of maintaining the software product; that is, managing modifications to the product, migration and retirement. The process is started when a system undergoes modification to code and associated documentation. Whenever a software product requires modification, the development process is invoked.

7.2 Supporting Processes

The eight supporting processes support other processes, contributing to the success and quality of the software product

7.2.1 Documentation

This process defines the activities for recording the information produced by another process. Planning, design, development, editing, distribution and maintenance of documents are covered.

7.2.2 Configuration Management

This process defines the activities of configuration management. This includes identification, definition and baselining of software items, controlling modification and release of items, recording and reporting software item and modification request status, ensuring completeness and correctness of items and controlling storage, handling and delivery of items.

7.2.3 Quality Assurance

This process defines the activities for independently assuring that software products and processes are in conformance with their specified requirements and adhere to their established plans. Joint reviews, audits, verification and validation may be used. This process is compatible with ISO9001.

7.2.4 Verification

This process defines the activities, for the acquirer, supplier or independent party, for verifying software products. Verification can be applied to process, requirements, design, code, integration and documentation. Verification supplements any evaluation inherent in other processes.

7.2.5 Validation

This process defines the activities, for the acquirer, supplier or independent party, for validating software products. Validation supplements any evaluation inherent in other processes.

7.2.6 Joint Review

This process defines the activities for evaluating the status and products of an activity. The process may be employed by any two parties where one party reviews another in a joint forum.

7.2.7 Audit

This process defines the activities for determining compliance with requirements, plans and contract. The process may be employed by any two parties where one party audits the software products, services or activities of another.

7.2.8 Problem Resolution

This process defines the activities for analyzing, resolving and removing problems (taking corrective action), including non-conformances, that are discovered during the execution of other processes. The process requires identification and analysis of causes.

7.3 Organizational Processes

The four organizational processes are employed to establish, implement and continually improve an underlying corporate structure of processes and personnel. Typically, they are employed beyond or across projects though lessons from projects may contribute to improvement. The organizational processes help in establishing, controlling and improving other processes.

7.3.1 Management

This process defines the activities for management, including project management of other processes.

7.3.2 Infrastructure

This process defines the activities for establishing and maintaining the underlying structure of another process. Infrastructure can include hardware, software, standards, tools, techniques and facilities.

7.3.3 Improvement

This process defines the activities for assessing, measuring, controlling and improving other processes. The objective is to improve those other processes across an organization.

7.3.4 Training

This process defines the activities for providing adequately trained personnel for both management and technical work. The process requires timely development of a training plan, generation of training material and provision of training.

7.4 Tailoring Process

This process defines the activities for tailoring (deletion of non-applicable or in-effective processes, activities and tasks) of ISO/IEC12207. This is the only normative process in ISO/IEC12207. Guidance on application of the tailoring process is given

8 Annex B: Differences Between MIL-STD-498 and EIA/IEEEJ-STD-016

EIA/IEEEJ-STD-016 is essentially a demilitarized version of MIL-STD-498. Differences between the two standards are described in this annex.

A key high-level difference between the standards is that J-STD-016 is properly applied through voluntary adoption rather than contractual imposition.

There are a number of terminology differences between J-STD-016 and MIL-STD-498. See, for example, ref.9.

More substantial changes that J-STD-016 introduced, as compared to MIL-STD-498, include the following.

- There is a normative appendix requiring tailoring to be performed (just as in ISO/IEC12207).
- The developer is allowed to reference software practices, as well as standards, for software development in the software development plan.
- There is a requirement to ensure that each element of the software development environment performs its intended function before use.
- There is a traceability tasking clause clarifying that traceability is required both upwards and downwards.
- There is a requirement to update the system and software requirements descriptions to match the approved "as built" system and software.
- The default application to deliverable software has been removed. J-STD-016 applies to all project software whether deliverable or not.
- The default time limit (duration of the contract) for record keeping and retention of the software development library and software development file has been deleted. The acquirer should specify how long records should be kept, which may be shorter or longer than the duration of the contract.
- The requirement to obtain approval for any programming language not specified has been deleted. If the contract is silent on the issue, the developer is free to use any language unless other provisions or laws apply.
- The following reuse requirements have also been deleted.
- The requirement to use any reusable software identified for reuse. The developer has now only to identify and evaluate for potential reuse.
- The requirement to identify opportunities for reuse and evaluate benefits and costs.

- The requirement to interpret the standard when reusable software is incorporated according to AppendixB. The content of AppendixB is now informative rather than normative.

9 Annex C: Questionnaire

As part of the task documented in this report, a questionnaire was prepared and circulated to ASSC member companies. Four responses to the questionnaire were received. The analysis of those responses is contained in Section4. This annex quotes the questionnaire in full.

Avionics Software Development Strategy Questionnaire

ERA has been tasked by the ASSC Process Working Group to review software development strategies in the light of ISO 12207 ("Information technology - Software life cycle processes") and its adoption, in an American guise as EIA/IEEE 12207, by the United States Department of Defence (DoD).

To this end, the following questionnaire has been created to gauge the knowledge and use of ISO 12207 and related standards amongst ASSC member companies. The questionnaire should ideally be completed by a person with supervisory responsibilities for software development. Answers will be treated confidentially and used to assess the impact of ISO 12207, and its adoption by the DoD, on ASSC member companies. A report documenting the work will be made available to member companies.

Questionnaire responses should be returned to:

Dr Simon Hughes
ERA Technology Ltd
Cleeve Road
Leatherhead
KT22 7SA

Telephone: 01371 367000 Extn 2798
Fax: 01372 367072
Email: simon.hughes@era.co.uk

Thank you for your assistance.

1. Contact name:

2. Job title:

3. Name of organisation:

4. Contact address:

Tel: Fax:

Email:

5. Nature of company business:

6. How familiar are you with ISO 12207, "Information technology – Software life cycle processes" (please tick most appropriate):

not at all
aware of existence
some familiarity with contents
very familiar with contents
work with standard extensively

Comments:

7. Which of the following standards have you developed software to:

RTCA/DO-178B
JSP 188
DoD-STD-2167A
MIL-STD-498
DEF STAN 00-55
ISO 12207
others (please specify)

Comments:

8. How have you achieved compliance with these standards:

by developing project specific processes
by using compliant company standards
other (please specify)

9. What has determined the need for compliance with the standards

specific contractual requirements
company quality objectives
other (please specify)

10. In what contexts have you used the standards

military
government
industry
commerce
other (please specify)

11. How do the software development standards you have used compare in terms of prescriptiveness, ease of application, etc.

12. Any other comments ...