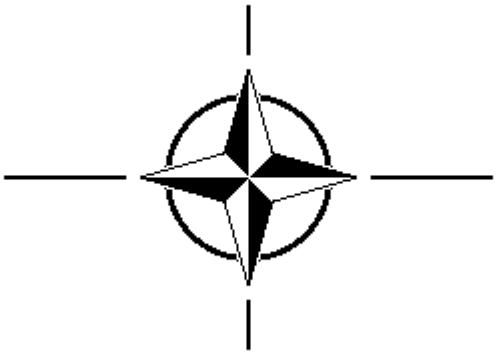
STANAG No. 4626
Part I
Draft 1

NORTH ATLANTIC TREATY ORGANIZATION

(NATO)



MILITARY AGENCY FOR STANDARDIZATION

(MAS)

STANDARDIZATION AGREEMENT

(STANAG)

SUBJECT: MODULAR AND OPEN AVIONICS ARCHITECTURES
PART I - ARCHITECTURE

Promulgated on

NORTH ATLANTIC TREATY ORGANIZATION
ORGANISATION DU TRAITE DE L'ATLANTIQUE NORD

MILITARY AGENCY FOR STANDARDIZATION (MAS)
BUREAU MILITAIRE DE STANDARDISATION (BMS)
1110 BRUSSELS
Tel: 707.43.09

….. MAS …..

STANAG 4626 (DRAFT 1) – MODULAR AND OPEN AVIONICS ARCHITECTURES
PART I: ARCHITECTURE

1. The enclosed NATO Standardization Agreement is herewith promulgated for ratification.

ACTION BY NATIONAL STAFFS

2 National staffs are requested to examine page iii of the STANAG and, if they have not already done so, advise the …… Division of their intention regarding its ratification and implementation.

Enclosure:
STANAG 4626 Part I (Draft 1)

RECORD OF AMENDMENTS

No.	Reference/date of amendment	Date entered	Signature

EXPLANATORY NOTES

AGREEMENT

1. This NATO Standardization Agreement (STANAG) is promulgated by the Chairman MAS under the authority vested in him by the NATO Military Committee.

2. No departure may be made from the agreement without consultation with the tasking authority. Nations may propose changes at any time to the tasking authority where they will be processed in the same manner as the original agreement.

3. Ratifying nations have agreed the national orders, manuals and instructions implementing this STANAG will include a reference to the STANAG number for purposes of identification.

DEFINITIONS

4. Ratification is “In NATO Standardization, the fulfillment by which a member nation formally accepts, with or without reservation, the content of a Standardization Agreement” (AAP-6).

5. Implementation is “In NATO Standardization, the fulfillment by a member nation of its obligations as specified in a Standardization Agreement (AAP-6).

6. Reservation is “In NATO Standardization, the stated qualification by a member nation that describes the part of a Standardization Agreement that it will not implement or will implement only with limitations (AAP-6).

RATIFICATION, IMPLEMENTATION AND RESERVATIONS

7. Page iii gives the details of ratification and implementation of this agreement. If no details are shown it signifies that the nation has not yet notified the tasking authority of it intentions. Page iv (and subsequent) gives details of reservations and proprietary rights that have been stated.

FEEDBACK

8. Any comments concerning this publication should be directed to NATO/MAS – Bvd Leopold III – 1110 Brussels - BE

RATIFICATION AND IMPLEMENTATION DETAILS
STADE DE RATIFICATION ET DE MISE EN APPLICATION

NATION	NATIONAL RATIFICATION REFERENCE DE LA RATIFICATION NATIONALE	NATIONAL IMPLEMENTING DOCUMENT / DOCUMENT NATIONAL DE MISE EN APPLICIATION	IMPLEMENTATION / MISE EN APPLICATION					
			INTENDED DATE OF IMPLEMENTATION / DATE PREVUE POUR MISE EN APPLICATION			DATE IMPLEMENTATION WAS ACHIEVED / DATE REELLE DE MISE EN APPLICATION		
			NAVY MER	ARMY TERRE	AIR	NAVY MER	ARMY TERRE	AIR
BE								
CA								
CZ								
DA								
FR								
GE								
HU								
IT								
LU								
NL								
NO								
PO								
SP								
TU								
UK								
US								

RESERVATIONS

RESERVES

STANAG 4626 (Part I)
(Draft 1)

NAVY / ARMY / AIR

NATO STANDARDIZATION AGREEMENT
(STANAG)

MODULAR AND OPEN AVIONICS ARCHITECTURE
PART I: ARCHITECTURE

Related Documents:

(a) STANAG 4626 Part II – Software

(b) STANAG 4626 Part III – Common Functional Modules

(c) STANAG 4626 Part IV – Packaging

(d) STANAG 4626 Part V – Networks and Communication

(e) STANAG 4626 Part VI – Guidelines for System Issues

- Vol. 1: System Management
- Vol. 2: Fault Management
- Vol. 3: System Initialisation and Shutdown
- Vol. 4: System Configuration/Reconfiguration
- Vol. 5: Time Management
- Vol. 6: Security Aspects
- Vol. 7: Safety

AIM

1. The aim of this agreement is to define and standardize essential technical characteristics which shall be incorporated in the design of avionics architectures.

AGREEMENT

2. Participating nations agree to adopt the avionic architectures of future aircraft developments and upgrades to the Standards and Guidelines of open avionics architectures as described in this STANAG.

DEFINITIONS

3. The definition of terms and abbreviations used in this Agreement are given in each Part of the Standard.

GENERAL

4. t.b.d.

DETAILS OF AGREEMENT

5. The details of the agreement are given as follows:

- in Part I: Architecture Standard and Annex “Rationale Report for Architecture Standards”
- in Part II: Software and Annex “Rationale Report for Architecture Software Standards”
- in Part III: Common Functional Modules and Annex “Rationale Report for Common Functional Modules Standards”
- in Part IV: Packaging and Annex “Rationale Report for Packaging Standards”
- in Part V: Networks and Communication and Annex “Rationale Report for Communications / Network Standards”
- in Part VI: Guidelines for System Issues consisting of:
  - Vol. 1: System Management
  - Vol. 2: Fault Management
  - Vol. 3: System Initialisation and Shutdown
  - Vol. 4: System Configuration/Reconfiguration
  - Vol. 5: Time Management
  - Vol. 6: Security Aspects
  - Vol. 7: Safety

each Part being published separately as STANAG 4626 (Part I), STANAG 4626 (Part II), STANAG 4626 (Part III), STANAG 4626 (Part IV), STANAG 4626 (Part V) and STANAG 4626 (Part VI).

STANAG 4626 (Part I)
(Draft 1)

Study n°:	Draft n°: I02	Date: 07/04/04

ENGLISH VERSION

ASAAC

Final Draft of Proposed Standards for Architecture

Proposition Finale des Standards pour l'Architecture

Entgültiger Entwurf des Standards für Architektur

Table of Contents

Table of Figures

Table of Tables

1 Introduction

1.1 Purpose

This document is produced under contract ASAAC Phase II Contract n°97/86.028.
The purpose of the ASAAC Programme is to define and validate a set of open architecture standards, concepts and guidelines for Advanced Avionics Architectures (A3) in order to meet the three main ASAAC drivers. The standards, concepts and guidelines produced by the Programme are to be applicable to both new aircraft and update programmes from 2005.
The three main drivers for the ASAAC Programme are:

- Reduced life cycle costs,
- Improved mission performance,
- Improved operational performance.

The Standards are organised as a set of documents including:

- A set of agreed standards that describe, using a top down approach, the Architecture overview to all interfaces required to implement the core within avionics systems,
- The guidelines for system implementation through application of the standards.

The document hierarchy is given hereafter: *(in this figure, the current document is highlighted)*

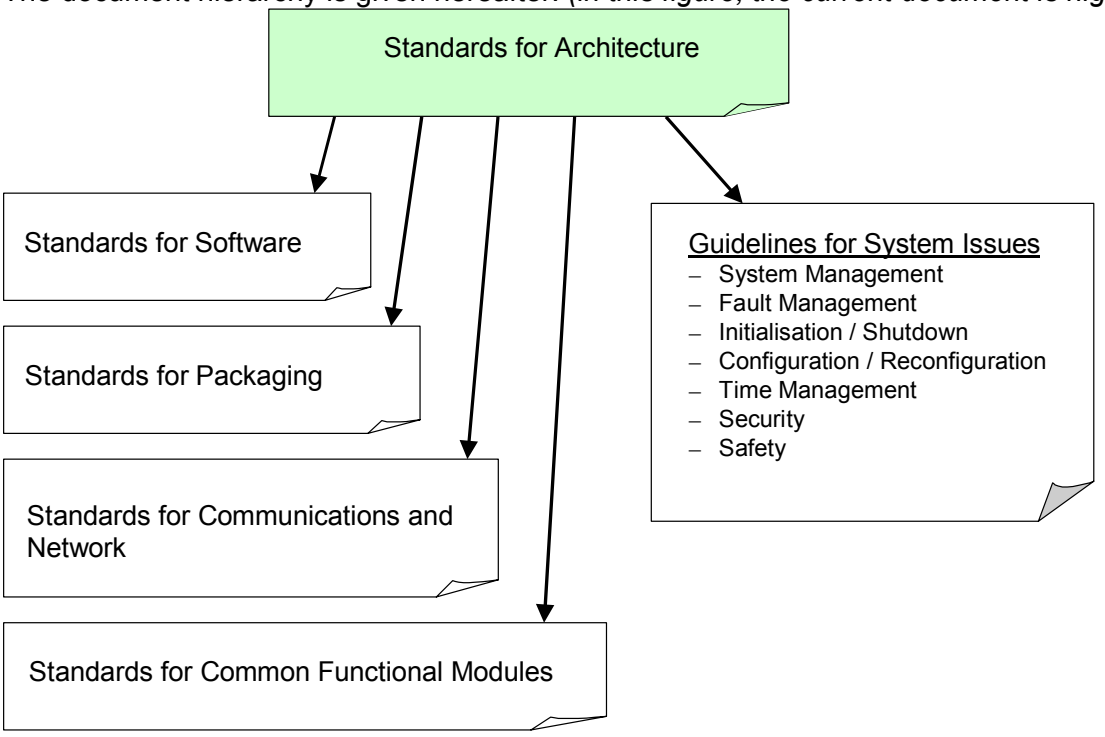


Figure 1 - ASAAC Standard Documentation Hierarchy

1.2 Document Structure

The document contains the following sections:

Section 2, gives the scope of the document,

Section 3, identifies normative references,

Section 4, gives the terms, definitions and abbreviations,

Section 5, presents the set of architecture drivers and characteristics as well as an introduction to IMA,

Section 6, defines the architecture standard, and introduces the other standards,

Section 7, introduces the guidelines for implementing an IMA architecture,

Annex A, presents the power supply architecture.

2 Scope

The purpose of this standard is to establish uniform requirements for the architecture for Integrated Modular Avionic (IMA) systems as defined by the ASAAC Programme.
The IMA architecture can be built by using common components. These components are specified in separate standards. Ways of using these components are described in a set of guidelines. This document gives references to these Standards and Guidelines as well as a short introduction to IMA.

3 Normative References

This European Standard incorporates by dated or undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this European Standard only when incorporated in it by amendment or revision. For updated references the latest edition of the publication referred to applies (including amendments).

[1] Final Draft of Proposed Standards for Software
STANAG 4626 Part II – Software

[2] Final Draft of Proposed Standards for Common Functional Modules
STANAG 4626 Part III – Common Functional Modules

[3] Final Draft of Proposed Standards for Communication /Network
STANAG 4626 Part V – Networks and Communication

[4] Final Draft of Proposed Standards for Packaging
STANAG 4626 Part IV – Packaging

[5] Final Draft of Proposed Guidelines for System Issues
STANAG 4626 Part VI – Guidelines for System Issues
Volume 1 – System Management
Volume 2 – Fault Management
Volume 3 – System Initialisation and Shutdown
Volume 4 – System Configuration / Reconfiguration
Volume 5 – Time Management
Volume 6 – Security Aspects
Volume 7 – Safety

[6]

[7]

4 Terms, Definitions and Abbreviations

4.1 Terms and Definitions

Use of “shall”, “should” and “may” within the standards observe the following rules:

- The word SHALL in the text expresses a mandatory requirement of the standard.
- The word SHOULD in the text expresses a recommendation or advice on implementing such a requirement of the standard. It is expected that such recommendations or advice will be followed unless good reasons are stated for not doing so.
- The word MAY in the text expresses a permissible practice or action. It does not express a requirement of the standard.

4.2 Abbreviations

A3	:	Advanced Avionics Architectures
AM	:	Application Management
AL	:	Application Layer
APOS	:	Application Layer / Operating System Layer Interface
ASAAC	:	Allied Standard Avionics Architecture Council
BIT	:	Built-In Test
BW	:	Band-Width
CFM	:	Common Functional Modules
CNI	:	Communication / Navigation / Identification
COMSEC	:	Communication Security
COTS	:	Commercial Off The Shelf
CPU	:	Computer Processing Unit
DC	:	Direct Current
DPM	:	Data Processing Module
EO	:	Electro-Optic
EMI	:	Electro-Magnetic Interference
EW	:	Electronic Warfare
GPM	:	Graphic Processing Module
GSM	:	Generic System Management
HDD	:	Head-Down Display
HUD	:	Head-Up Display
HW	:	Hardware
IED	:	Insertion / Extraction Device
IF	:	Interface
IFF	:	Identification Friend or Foe
IMA	:	Integrated Modular Avionics
LRC	:	Line Replaceable Chamber
LRM	:	Line Replaceable Module
MMM	:	Mass Memory Module
MOS	:	Module Support Layer / Operating System Layer Interface
MPI	:	Module Physical Interface
NSM	:	Network Support Module
OS	:	Operating System
PCM	:	Power Conversion Module
PCU	:	Power Conversion Unit
PSE	:	Power Supply Element
SPM	:	Signal Processing Module

TD&T	:	Target Detection and Tracking
TRANSEC	:	Transmission Security
UAV	:	Unmanned Aerial Vehicle

4.3 Definitions

IMA System is a full system that is built from an IMA Core System and non-Core equipment.

IMA Core System is an avionics system comprising one or a series of avionic racks containing sets of standardised CFMs linked together by a unified communication network and executing reusable functional applications that are hardware independent, operating systems and system management software.

Common Functional Modules (CFM) are line replaceable items and provide an IMA Core System with a computational capability, network support capability and power conversion capability.

Software Layered Architecture is a common software model based on the concept of a layered software architecture. Within this model, the layers are separated by standardised interfaces in order to provide independence of these layers.

System Management is the management of the resources and services of an IMA Core System during initialisation, all operational phases in flight and on ground, and system shutdown.

5 IMA Drivers and Characteristics

5.1 Drivers

The three principle drivers for the architecture are:

- Reduced Life Cycle Cost:
  - A major objective is to reduce the accumulated costs over the life cycle of a system i.e. the development, acquisition and support costs.
- Improved Mission Performance:
  - The system must be capable of fulfilling the missions and satisfy all possible airborne platforms in terms of functionality, capability, reliability, accuracy, configurability and interoperability under the full scope of operating conditions.
- Improved Operational Performance:
  - The goal adopted is that the system (aircraft) should achieve a combat capability of 150 flying hours or 30 days without maintenance, with an availability of at least 95%.
  - This goal far exceeds that achievable today and an IMA System will be required to exhibit fault tolerance so that it can survive the occurrence of faults with the required level of functionality.

5.2 Introduction to IMA Concepts

5.2.1 Non-IMA Systems

Non-IMA systems (e.g. federated systems) often comprise avionics units supplied by different equipment suppliers. These units invariably contain custom embedded computer systems in which the functional software is habitually bound to the hardware. It is not uncommon practice for these units to communicate via a number of different data busses, with perhaps two or three communication standards being the norm. Figure 2 depicts a simplified federated system architecture.

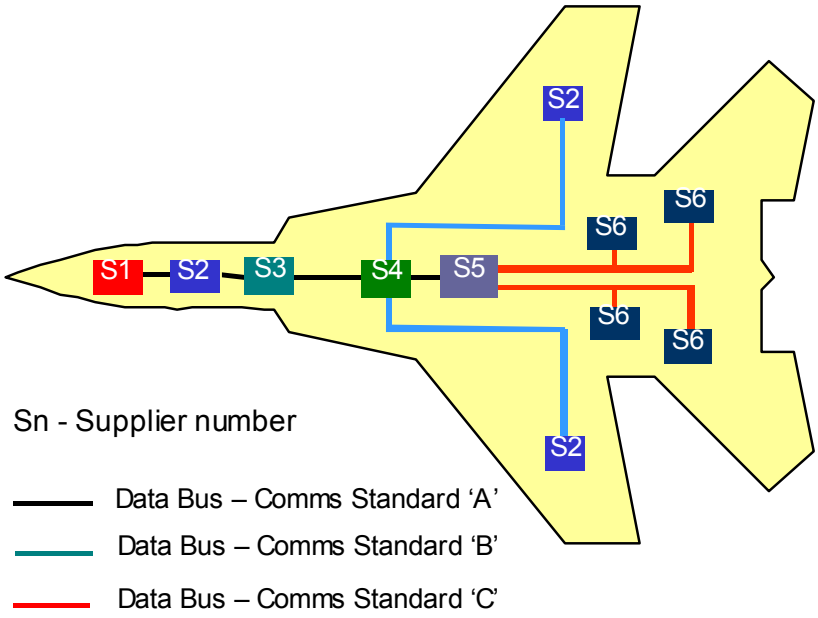


Figure 2 - A Typical Federated Aircraft System

It is widely accepted within the aerospace community that the consequences of continuing to develop aircraft along these lines are: frequent maintenance, low aircraft availability, low hardware and software re-use and large spares inventories - all of which contribute to higher costs for the initial production and the subsequent maintenance of avionics systems. Aircraft systems are becoming increasingly larger and more complex, driven as they are by current mission and operational requirements, while market availability of components is getting so short that systems are often becoming obsolete during their development.

5.2.2 Characteristics for an IMA System

The first step in defining a solution to meet the drivers defined in section 5.1 is to establish a suite of derived requirements or architecture characteristics that would collectively lend themselves to the main drivers being met.
The key architectural characteristics (ultimately there are many) derived from the three main drivers are identified in Table 1.

Table 1 - Architectural Characteristics

Architectural Characteristics	Mission Performance	Operational performance	Life Cycle Costs
Define a small module set with wide applicability	-	✔	✔
Design modules to be replaceable at 1st line	-	✔	✔
Maximise interoperability and interchangeability of modules	-	✔	✔
Adopt the use of an open system architecture	-	-	✔
Maximise the use of commercial off-the-shelf technology	-	✔	✔
Maximise technology transparency for both hardware and software components	-	-	✔
Minimise impact of Hardware & OS upgrades	-	-	✔
Maximise software reuse & Portability	-	✔	✔
Define comprehensive BIT and fault tolerance techniques to allow deferred maintenance	✔	✔	✔
Provide support for a high degree of both functional and physical integration	✔	-	✔
Ensure growth capability with reduced re-certification	✔	-	✔

5.2.3 IMA System Design

Once the three high level drivers are translated into architectural characteristics, the next step is to define the scope of what these new standards, concepts and guidelines should be applicable to. The boundaries are drawn at the IMA Core System.
The IMA Core System can be defined as a set of one or more racks comprising a set of standardised modules from a limited set of module types communicating across a unified digital network. The IMA Core System processes inputs received from the platform's low and high bandwidth sensors and transmits its outputs to the platform's low and high bandwidth effectors. Figure 3 shows an IMA Core System within a representative aircraft system.

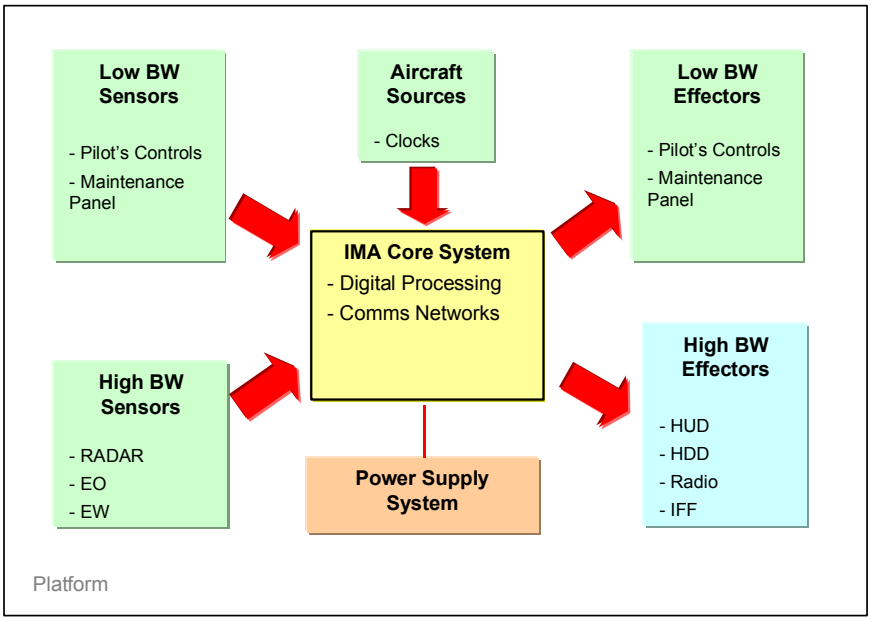


Figure 3 - IMA Core System

The IMA Core System can be viewed as a single entity comprising many integrated processing resources which can be used to construct any avionics system regardless of size and complexity. The concept of the IMA Core System is therefore equally applicable to smart missiles, UAVs, fast jets, large military aircraft...
The digital processing that occurs within the IMA Core System includes all the typical functional applications normally associated with avionics platforms: Vehicle Management, Mission Management, Stores Management, CNI, Target Detection & Tracking, HUD & HDD Displays, etc, as shown in Figure 4. The unified network used as the communication medium within the IMA Core System is also used to enable the functional applications to communicate with the platform's sensors and effectors. This communication is made possible by the use of interfaces to the network.

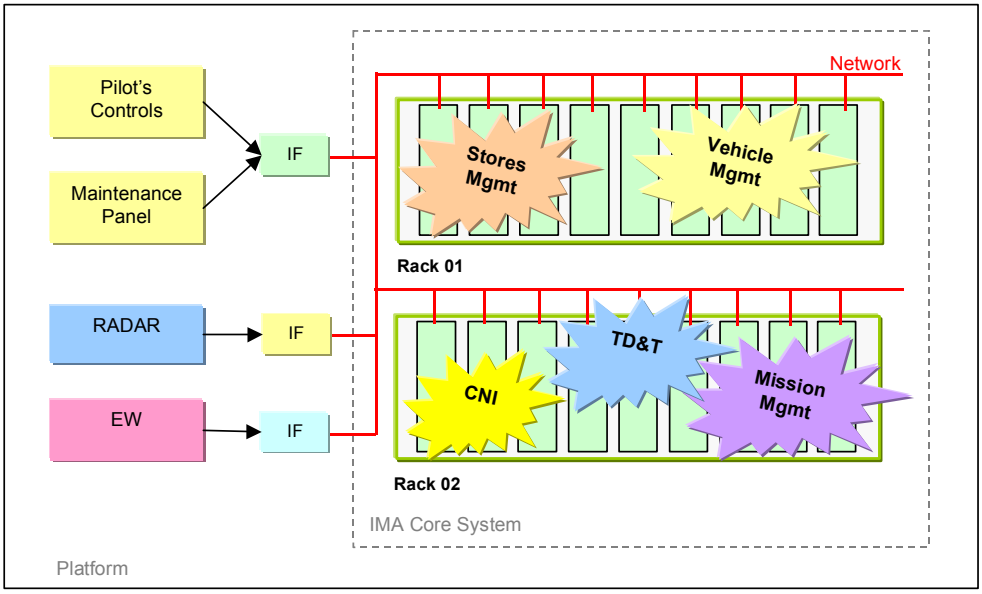


Figure 4 - IMA System

The main conceptual difference between the IMA Core System and current federated systems' Line Replaceable Units is that the functional application software does not remain resident on the modules on which it is ultimately to be processed. In the IMA Core System, all software is held on mass memory storage devices and downloaded to the modules upon which they are to execute as part of the system initialisation and configuration processes. This concept is instrumental in deferring maintenance and ensuring that modules can be replaced during first line maintenance.
Figure 5 shows how such a system could look in a platform:

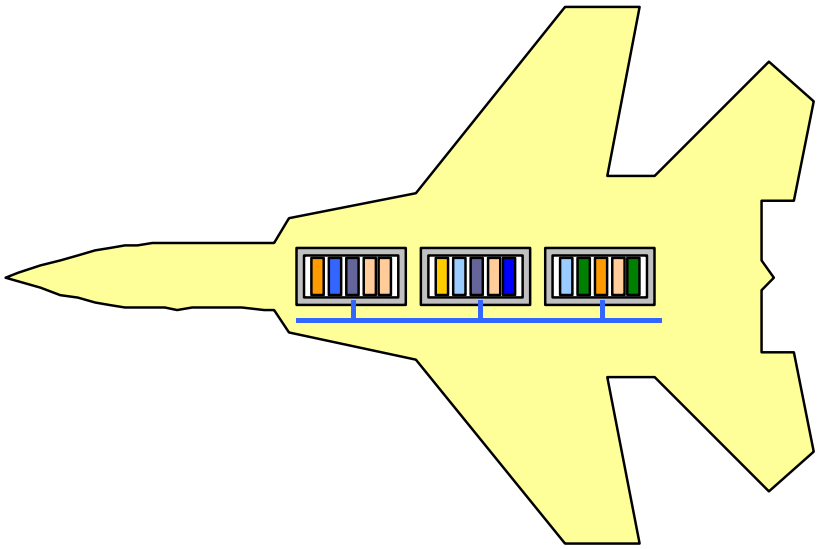


Figure 5 - An IMA System

The essence of IMA is to use a minimum set of common parts: Common Functional Modules, software and interfaces.

6 Requirements and the Architecture Standard

The architecture of the IMA Core System as described in section 5 above shall use:

- The distributed software layered architecture,
- The Common Functional Modules in form, fit and function,
- The unified communication network and associated protocols,
- The system management hierarchy.

The above requirements are embodied in a set of standards, which are introduced below. These standards are mandatory.

6.1 Software Architecture

The purpose of the Software Standard is to establish uniform requirements for design and development of a software architecture for IMA Core Systems.
The software architecture is based on the separation into horizontal layers. Figure 6 below shows a simplified view of the three-layer software architecture.

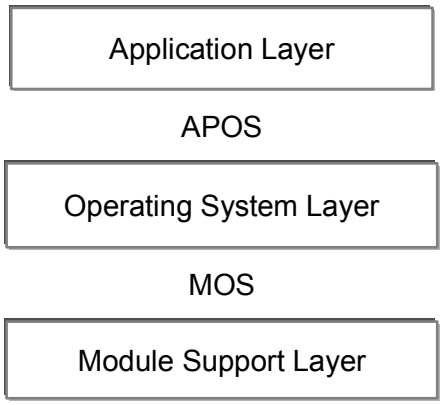


Figure 6 - Three Layer Software Architecture

Each layer can be described in terms of dependence / independence on both the aircraft system and the underlying hardware, see Table 2.

Table 2 - Software Layer Independence

Software Layer	Aircraft Dependency	Hardware Dependency
Application Layer	Dependent	Independent
Operating System Layer	Independent	Independent
Module Support Layer	Independent	Dependent

The two major interfaces, APOS and MOS, ensure the independence for each of the three layers as described in Figure 6.
The software concept is based on three main aspects:

- System management, that is carried out in two layers:
  - The Application Management function (AM), located in the Application Layer,

  - The Generic System Management function (GSM), located in the Operating System Layer.

- Blueprints,
  The Blueprints configuration files contain the information required by the GSM function, operating in a hierarchical manner, with the information they need to manage the resources under their jurisdiction.

- Virtual Channel,
  Virtual Channels are the message-based means of communication between processes which are network implementation independent.

The complete software architecture standard is available within the document referenced [1].

6.2 Common Functional Module

The Common Functional Module Standard defines the functionality and principle interfaces for the CFM to ensure their interoperability and provides design recommendations to assist in implementation of such a CFM. The following set of modules have been defined for use within an IMA Core System:

- Signal Processing Module (SPM),

- Data Processing Module (DPM),

- Graphics Processing Module (GPM),

- Mass Memory Module (MMM),

- Network Support Module (NSM),

- Power Conversion Module (PCM), for power distribution architecture see Annex A.

The definition of interfaces and functionality allows a CFM design that is interoperable with all other CFMs to this standard, that is technology transparent, that is open to a multi-vendor market and that can make the best use of COTS technologies.
Although the physical organisation and implementation of a CFM should remain the manufacturer's choice, in accordance with the best use of the current technology, it is necessary to define a structure for each CFM in order to achieve a logical definition of the CFM with a defined functionality. The structure is for building blocks and is independent of the implementation. The definition includes:

- The Generic CFM, which defines the generic functionality applicable to the complete set of CFMs.

- The processing capability, which defines the unique functionality associated with each CFM type within the set.

- The logical and physical interfaces that enable CFMs to be interoperable and interchangeable.

The complete CFM Standard is available within the document referenced [2].

6.3 Communication / Network

The Communication / Network Standard details the functionality and principle interfaces for the network to ensure the interoperability of Common Functional Modules and design recommendations to assist in implementation of such a network.
The purpose of this Standard is to establish by means of well defined interfaces and functionality, a network design that is technology transparent, that is open to a multi-vendor market and that can make the best use of COTS technologies. Therefore the associated data communication network topology, protocols and technologies are not identified in the Standard. Instead it identifies the issues that should be considered when defining a specific network implementation.
Although the physical organisation and implementation of the network shall remain the system designers choice, in accordance with the best use of the current technology, it is necessary to define interfaces and parameter sets in order to achieve a logical definition of the network with a defined functionality. This definition includes:

- The generic functionality applicable to all networks,
- The logical interfaces to the OS and MSL,
- Physical interfaces to the CFMs.

The complete Communication / Network Standard is available within the document referenced [3].

6.4 Packaging

The purpose of the Packaging Standard is to establish uniform requirements for packaging for IMA Core System components. It defines the module physical properties and the Module Physical Interface (MPI) definitions together with recommendations for IMA rack and the operational environment.
The characteristics addressed by the Packaging Standard are:
Interchangeability:

- For a given cooling method all modules conforming to the packaging standard will function correctly when inserted into any rack slot conforming to the standard for the cooling method.
- All modules conforming to the MPI definitions for connector, IED and cooling interface will function correctly when inserted into any rack slot conforming to the same MPI definition.

Maintainability:

- All modules are easily removable at first line,
- No special tools required at first line,
- No manual adjustment is necessary when installing modules. No tool is required for installation or removal of the modules.

The equipment design shall take into account:

- Environmental conditions,
- Mechanical conditions,
- Cooling conditions,
- Power supply architecture,

- Electromagnetic compatibility.

The complete Packaging Standard is available within the document referenced [4].

7 Guidelines

The standards, in section 6, mandate the functionality, and in some cases the implementation of that functionality that a system must adopt in order to be considered 'standard compliant'. In addition to these standards, a series of guidelines are offered in order to support the IMA system integrator in defining and building a system. These guidelines are not mandatory, they represent the guidance which arose during the validation phase of the standard.
The System Issues (7 volumes) provides guidelines supplementary to the Architecture Standard. They are defined in the “Final Draft of Proposed Guidelines for System Issues”, reference [5].

7.1 System Management

System management supports functional applications with safety requirements ranging from non-essential to flight critical and functional applications managing/processing data marked from unclassified to top secret. In meeting these requirements, the individual implementations of IMA architectures may require segregation of differing levels of safety criticality or security. Consequently, the system management framework shall be sufficiently flexible to allow the local management of such functions.
System management is responsible for:

- Controlling mission mode selection according to the requests made by the pilot and/or functional applications.
- Identifying, masking, confining and localising any faults or errors that occur.
- Providing integrated test and maintenance facilities to the ground crew thus enabling them to ascertain the state of the system and correct it if necessary.
- Controlling the system initialisation and shutdown processes.
- Offering functional applications security-related services.

The configuration of the system management is defined by Run Time Blueprints.

System management is defined in Volume 1 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5]).

7.2 Fault Management

Fault Management is a combination of relevant fault management techniques that apply to individual components of the complete system. Each fault management technique should be assessed for coverage, accuracy, speed, resources used (network bandwidth, memory, CPU time, etc.). The sum of the selected techniques should meet the system requirements for fault tolerance and integrated test and maintenance support.

Fault management is defined in Volume 2 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5]).

7.3 System Initialisation and Shutdown

The System Initialisation consists of:

- Initialisation of the initial configuration,

- Subsequent CFM initialisation,
- Initialisation of a system management hierarchy (including time).

The aim of these mechanisms is to reach a basic configuration which can then be used to support the loading and subsequent running of an entire system configuration. These further steps use configuration / reconfiguration mechanisms.
The system shutdown consists of:

- System management shutdown,
- CFM shutdown,
- Final platform shutdown.

System Initialisation and shutdown are defined in Volume 3 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5]).

7.4 System Configuration / Reconfiguration

The flexible nature of IMA systems allows the possibility of different configurations being used, depending upon resources available or required functionality. The process of transition between such configurations is known as system reconfiguration.
System configurations / reconfigurations can occur as a result of:

- A system mode change,
- Fault management,
- Ground crew maintenance and test actions,
- Phases of system initialisation and shutdown.

System configuration / reconfiguration is defined in Volume 4 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5]).

7.5 Time Management

IMA Systems require the concept of a distributed time reference in order to facilitate the following:

- The co-ordination of multiple aircraft taking part in the same mission,
- The recording of the time when system events occurred. (e.g. for fault management),
- The recording the time when data arrives in and/or leaves a system. (e.g. for time stamping),
- The scheduling of system management and functional application processes (to cater for a range of scheduling algorithms),
- The synchronisation of the components of an IMA Core System.

Time management is defined in Volume 5 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5]).

7.6 Security Aspects

IMA Systems are embedded computing systems with communication channels to the external environment. It is assumed that all of the processing resources required will be contained within the confines of an aircraft fuselage. In this case, security refers to the separation of data between processors within the IMA Core System. It is not concerned with encryption off the aircraft “COMSEC” and “TRANSEC”.
A number of security mechanisms may be adopted within the core and these are defined in Volume 6 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5]).

7.7 Safety

Steps to include safety aspects are as follows:

- Ensure that, at all levels, safety is designed into the system from the beginning, and not added on afterwards,
- Tailor a system safety activity to meet specific program needs,
- Manage residual hazards.

Safety is defined in Volume 7 of the Final Draft of Proposed Guidelines for System Issues (see Ref. [5])

.

Annex A Power Distribution Architecture

A.1 General Description

The Power Supply Distribution Architecture applies to the IMA Core system. and describes the functionality of each element in the architecture in terms of the electrical input and output characteristics. The internal design, manufacture and implementation of the power supply distribution elements is left to the vendor/systems designers.
The Electrical power quality characteristics are described with a reference to ISO/CD 1540 " Aerospace - Characteristics of aircraft electrical systems - ISO/TC20/SC 1/WG 13 - Date : 20/04/1998" and apply at the electrical input terminals of each of the Power supply elements defined below. These characteristics include both normal and abnormal operating conditions of the aircraft electrical power system during all phases of flight and ground operation.

A.2 The Double Conversion Architecture

The Power Supply Distribution is based on a Double Conversion Architecture, presented in Figure A. 1 where there are two stages of DC/DC conversion between the rack input voltage and the electronic components located on the boards installed within the LRM
It should be noted that the 'Rack input Voltages' are derived from the primary 'Platform Voltages' via a Line Replaceable Chamber (LRC). The LRC forms the interface between the Platform's power distribution and that of the IMA Core System specified here. For continuity, the LRC characteristics are also specified.

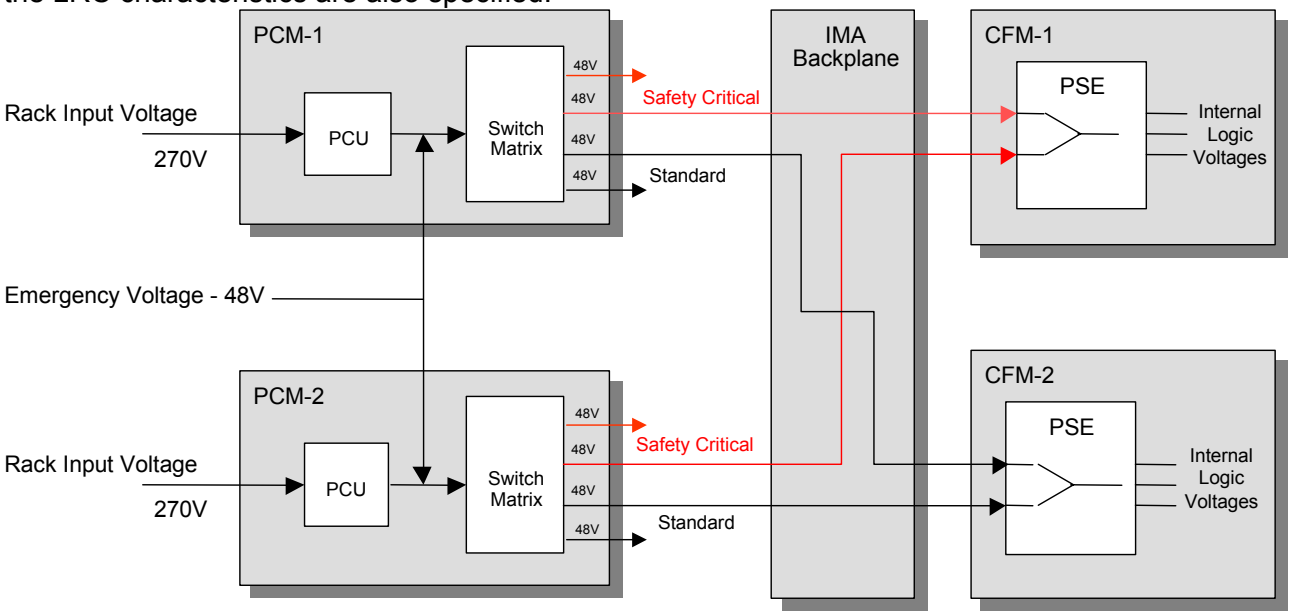


Figure A. 1 - Double Conversion Architecture

The two stages of conversion are carried out in:

- the Power Conversion Module (PCM), which converts the rack input voltage down to a medium voltage which is then distributed via the backplane to the LRMs,

- the Power Supply Element (PSE), which converts the LRM input voltage, equal to the backplane medium voltage, down to the electronic components voltage(s).

These two items are fully defined in the Final Draft of Proposed Standards for CFM [2].

A.3 The Line Replaceable Chamber

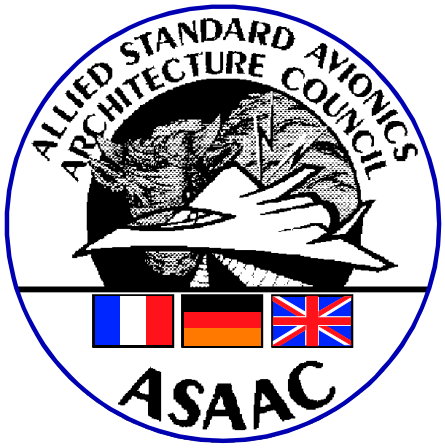
The LRC is platform specific and provides the interface from the platform to the IMA Core System power supply architecture and more specifically to the racks. The purpose of having a LRC is:

- To separate the platform voltages from the IMA System rack input voltages: the rack input voltages being fixed, the LRC allows the installation of the ASAAC rack on different platforms which may have ac fixed frequency, ac variable frequency or dc supplies,

- To adapt the platform voltages to the necessary rack input voltages: as a consequence of the previous point, the LRC includes a rectifying function when ac supplies are used at the platform level,

- To filter the low frequency perturbations: the latter require large passive components which cannot fit within CFM envelopes and if they are filtered outside the rack, internal interference (EMI) is avoided.

The LRC may supply more than one rack and there may be more than one LRC on the platform. The LRC characteristics are to be specified by the system design specification.



ASAAC Phase II Stage 2

Rationale Report for Architecture standards

Issue 02

IPR Category 1

Issue Date:	14/12/04
Pages:	46

Change Log

Version	Date	Reason for Change
Issue 01	05/02/04	Initial version
Issue 02	17/04/04	Table 3 is updated due to editing mistakes (missing crosses) and 48VD**C** in TLR_10

Table of Contents

List of Tables

1 Introduction

1.1 Scope of this Document

This document is produced under the ASAAC Phase II Contract. It is the second part of the deliverable attached to the Work Package 32460 " Final Draft of Proposed Standards For Architecture" and constitutes the Architecture Standards Rationale Document, which is included in the Poste 3D of the contract.

1.2 Work package Objectives

The objective of work package WP32460 was to produce the final draft of the Standards that define an IMA system, its architecture, software and the Common Functional Modules (CFMs) to operate within it.
The objective of this document is to give the justification for the final draft of proposed standards for architecture.

1.3 Abbreviations

APOS Application Layer / Operating System Layer Interface
ASAAC Allied Standard Avionics Architecture Council
BIT Built-In Test
CFM Common Functional Modules
CNI Communication / Navigation / Identification
GLI Generic System Management Logical Interface
GSM generic System Management
IMA Integrated Modular Avionics
LRC Line Replaceable Chamber
MLI Module Logical Interface
MOS Module Support Layer / Operating System Layer Interface
OLI Operating System Logical Interface
OS Operating System
SMBP System Management to Blueprint Interface
SMLI System Management Logical Interface
SMOS System Management to Operating System Interface
TLR Top Level Requirements

2 Related Documents

[1] Final Draft of Proposed Standards for Architecture
STANAG 4626 Part I – Standards for Architecture

[2] Final Draft of Proposed Standards for Software
STANAG 4626 Part II – Software

[3] Final Draft of Proposed Standards for Common Functional Modules
STANAG 4626 Part III – Common Functional Modules

[4] Final Draft of Proposed Standards for Communication /Network
STANAG 4626 Part V – Networks and Communication

[5] Final Draft of Proposed Standards for Packaging
STANAG 4626 Part IV – Packaging

[6] Final Draft of Proposed Guidelines for System Issues
STANAG 4626 Part VI – Guidelines for System Issues
Volume 1 – System Management
Volume 2 – Fault Management
Volume 3 – System Initialisation and Shutdown
Volume 4 – System Configuration / Reconfiguration
Volume 5 – Time Management
Volume 6 – Security Aspects
Volume 7 – Safety

[7] ASAAC Phase II Stage 1 - Requirements Review Report - REF-WP : 11170
Reasons for selecting the proposed Standards for ASAAC architecture

Key Architectural Characteristics

The key architectural characteristics are defined in Table 1.

Architectural Characteristics	Mission Performance	Operational performance	Life Cycle Costs
Define a small module set with wide applicability	-	✔	✔
Design modules to be replaceable at 1st line	-	✔	✔
Maximise interoperability (1) and interchangeability of modules	-	✔	✔
Adopt the use of an open system architecture	-	-	✔
Maximise the use of commercial off-the-shelf technology	-	✔	✔
Maximise technology transparency for both hardware and software components	-	-	✔
Minimise impact of Hardware & OS upgrades	-	-	✔
Maximise software reuse & Portability	-	✔	✔
Define comprehensive BIT and fault tolerance techniques to allow deferred maintenance	✔	✔	✔
Provide support for a high degree of both functional and physical integration	✔	-	✔
Ensure growth capability with reduced re-certification	✔	-	✔

Table 3- Architectural Characteristics

It is the aim of the architecture standard to address these characteristics.

Note (1): “Interoperability” means operation with other aircraft in the fleet and with aircraft of other fleets (i.e. other nations). Typically: Radio frequencies, data exchange protocols, etc

2.1 Overview on the proposed ASAAC standards

The ASAAC architecture is a combination of Software, Processing, Networks, Packaging and System concepts brought together to define the key attributes of a Core Processor for an integrated modular avionics system.
These components are specified in separate standards. Ways of using these components are described in a set of guidelines. The Architecture Standard document gives references to these Standards and Guidelines as well as a short introduction to IMA

The architecture of the IMA Core System shall use:

- The distributed software layered architecture, as defined in the Final Draft of Proposed Standards for Software [2]
- The Common Functional Modules (CFM) in form, fit and function, as defined in the Final Draft of Proposed Standards for Common Functional Modules [3]
- The unified communication network and associated protocols, as defined in the Final Draft of Proposed Standards for Communication /Network [4]

- The uniform requirements for packaging for IMA Core System components, as defined in the Final Draft of Proposed Standards for Packaging [5]
- The system management hierarchy, As defined in the Final Draft of Proposed Guidelines for System Issues [6]

3 Justification for standard selection

3.1 Definition of the Top Level Requirements

The justification of the standards selection, of their definition and their allocation to the ASAAC architecture elements is based on a selection of 19 top level requirements as defined in the Requirements Review Report [7], the table given below lists these top level requirements (TLR) and the link with the Architectural Characteristics as defined in Table 1:

Ref.	Top Level Requirements	Architectural Characteristics
1	Small set of common modules	Define a small module set with wide applicability
2	Modules Applicable to Wide range of platforms	Define a small module set with wide applicability
3.1	Re-use of Software	Maximise software reuse & Portability
3.2	Module replaceable at first line	Design modules to be replaceable at 1st line
3.3	System level self-test	Design modules to be replaceable at 1st line
3.4	Deferred maintenance	Define comprehensive BIT and fault tolerance techniques to allow deferred maintenance
3.5	Comprehensive BIT and Testability	Define comprehensive BIT and fault tolerance techniques to allow deferred maintenance
8	Interoperability	Maximise interoperability and interchangeability of modules
9	Interchangeability	Maximise interoperability and interchangeability of modules
10	Technology Transparency	Maximise technology transparency for both hardware and software components
11	Use of Commercial components, technologies and processes	Maximise the use of commercial off-the-shelf technology
12	Maximise digital processing of functions	Provide support for a high degree of both functional and physical integration Maximise the use of commercial off-the-shelf technology
13.1	General system requirements and performance	Maximise technology transparency for both hardware and software components
13.2	Sensors and sub-system	Adopt the use of an open system architecture

Ref.	Top Level Requirements	Architectural Characteristics
13.3	Interface definitions	Maximise software reuse & Portability Adopt the use of an open system architecture
13.4	Criticality of functions	Provide support for a high degree of both functional and physical integration. • This Architectural Characteristic addresses particularly system management, especially Blueprints
13.5	System Processing performance and memory requirements	Provide support for a high degree of both functional and physical integration
14.1	Growth Capability	Ensure growth capability with reduced re-certification
14.2	Modularity and configurability	Ensure growth capability with reduced re-certification • Define a small module set with wide applicability
15	Certification and qualification	Ensure growth capability with reduced re-certification
16	Security	Provide support for a high degree of both functional and physical integration. • Provide support for multilevel secure data.
17	System management	Provide support for a high degree of both functional and physical integration
18	Environmental effects	Provide support for a high degree of both functional and physical integration
19	Buildability	All

Table 4- Top Level Requirements and Architectural Characteristics

The table given below summarises the justification of the standards selection.

Each of the columns is allocated to one of the ASAAC Standards.

In a box belonging to this column, an « X » is written if the standard contributes to meeting the Top Level Requirement represented in that row.

Reference	Top Level Requirements	Software Standard	CFM Standard	Comm./ Network Standard	Packaging Standard	System Management Guidelines
1	Small set of common modules		✔			
2	Modules Applicable to Wide range of platforms	✔			✔	
3.1	Re-use of Software	✔				
3.2	Module replaceable at first line				✔	
3.3	System level self-test	✔	✔			
3.4	Deferred maintenance	✔				✔
3.5	Comprehensive BIT and Testability	✔	✔			
8	Interoperability	✔		✔		
9	Interchangeability	✔	✔	✔	✔	✔
10	Technology Transparency	✔	✔	✔	✔	✔
11	Use of Commercial components, technologies and processes	✔	✔	✔	✔	
12	Maximise digital processing of functions		✔			
13.1	General system requirements and performance	✔		✔		✔
13.2	Sensors and sub-system	✔	✔			
13.3	Interface definitions	✔				✔
13.4	Criticality of functions	✔				✔
13.5	System Processing performance and memory requirements		✔			
14.1	Growth Capability	✔	✔	✔	✔	✔
14.2	Modularity and configurability			✔		✔
15	Certification and qualification	✔				✔
16	Security	✔				✔
17	System management	✔				✔
18	Environmental effects				✔	
19	Buildability	✔	✔	✔	✔	✔

Table 5- Overview on justification for Standards selection

3.1.1 Detailed reasons for standards selection

Detailed reasons are proposed in this section against each top level requirement.

TLR_1: Small set of common modules

The proposed CFM standard specifies a set of 6 ASAAC modules:

- Data Processing Module,
- Signal Processing Module,
- Graphics Processing Module,
- Mass Memory Module,

Free Use.

- Power Conversion Module,
- Network Support Module.

TLR_2: Modules Applicable to Wide range of platforms

To ease the integration of the ASAAC architecture in a wide range of platforms, the Power Supply Distribution, described in the Architecture Standard, specifies only the distribution inside the core processing system. The adaptation to a particular platform is made through the Line Replaceable Chamber (LRC), which receives the power supply from the aircraft generators and distributes it to the standardised input of the Power Conversion Modules. The LRC is not an ASAAC standard in order to provide appropriate adaptation to the platforms mainly for cooling technology and external connections.
The Packaging Standard takes into account the different cooling technologies that can be used in the platforms. The Packaging Standard specifies the mechanical interfaces for the following types of cooled modules:

- Conduction cooled modules,
- Air flow through cooled modules,
- Air flow around cooled modules,
- Direct air flow cooled modules.

The selection of the cooling technique to be used in a particular application is dependent upon the capabilities of the platform on which the system will be integrated.
The APOS interface, specified in the Software Standard, offers the services required by a wide range of avionics applications and thus usable in a wide range of platforms.

TLR_3.1: Re-use of Software

The definition of the APOS interface contributes to the reusability of Application Software. Application software compliant with this interface will be re-usable in any ASAAC System, provided it has sufficient system resources available. (It should be noted that the re-use of software is also dependent on the software development tools such as compiler).
The MOS interface**,** specified in the Software Standard, is also defined to allow the reusability/portability of the Operating System Layer software.
In addition, the SMLI interface**,** specified in the Software Standard, which defines a protocol of communication between application management and generic system management, contributes to the re-use of Application Management software.

TLR_3.2: Module replaceable at first line

The Packaging Standard includes the specifications to enable insertion and extraction of the CFMs at first line.
The main enabler for modules being replaced at first line maintenance is the fact that application and OSL software is not blown into prom on the CFMs, but is downloaded to them as part of the initialisation process. Consequently, a CFM can be replaced by another CFM of the same type. This concept is basically made possible due to the system management and software concepts.

TLR_3.3: System level self-test
The CFM standard specifies that each CFM shall include BIT functionality to be performed either on the request of the system management and/or the ground crew. The BIT functionality in the CFM and the set of services dedicated to BIT in the MOS and SMOS interfaces**,** specified in the Software Standard, allow the location of failure to module level for first line.
Consequently no base or depot level maintenance is required.

TLR_3.4: Deferred maintenance
Fault tolerance mechanisms can be implemented by using the software interfaces completed by the System Management, which include dedicated services for management of faults and the establishment and removal of different system configurations.
It must be noted that the architecture includes provisions to implement a deferred maintenance policy. However, the satisfaction of this requirement has to be assessed for each platform because it is up to the system designer to implement this policy.

TLR_3.5: Comprehensive BIT and Testability
The proposed ASAAC standards include facilities to implement comprehensive BIT.
Examples of these facilities are:

- CFM standard specifies each CFM to have a Power-up BIT, a Continuous BIT and an Initiated BIT.
- Services in the MOS interface to request the execution of the CFM BIT or to get the result.
- Services in the SMOS interface to allow the GSM to request the execution of the CFM BIT and to get the result.
- Services in the MLI, specified in the Software Standard, to get result of PBIT and to execute a network BIT.

TLR_8: Interoperability
To obtain the interoperability of the ASAAC architecture elements, several logical interfaces were defined in the Software Standard:

- The MLI interface to specify the communication protocol between CFMs,
- The OLI interface to specify the communication protocol between Operating System instances.
- The GLI interface to specify the communication protocol between Generic System Management instances.

The ASAAC Standards provide independence with specific technologies, including network technologies. The Communication/Network Standard provides requirements on network

parameters to be defined by the designer. The interfaces relevant to the network are defined in the Software Standard.

TLR_9: Interchangeability
The specification of the interfaces of the ASAAC architecture elements ensures their interchangeability. The following elements are considered as interchangeable with regard to major interfaces

- An OS has to comply with Software interfaces.
- A GSM has to comply with Software interfaces.
- A CFM has to comply with Software Interface, Power Supply Distribution specification, CFM standard, Communication/Network Standard and System Management.

TLR_10: Technology Transparency
The software interfaces are defined to increase the independence of Application Layer software and of Generic System management software from the hardware:

- The MOS interface provides the OSL (OS & System management Software) with a level of abstraction from the underlying hardware such that if the hardware and its corresponding SW were to change, the software in the OSL could remain unchanged,
- The APOS interface provides the Application software with a level of abstraction from the underlying OS, so that should the OS need to change, the application software could remain unchanged.

The MOS interface is defined to minimise the dependence of the Operating System on the CFM implementation. In case of obsolescence of a CFM component, only the software below the MOS would need to be re-developed.
The CFM standard, Communication/Network Standard and power supply distribution also contribute also to the technology transparency. All CFMs shall include a Power Supply Element to adapt the standard 48VDC supplied by the PCMs to the voltage(s) required by the components used to implement the CFM.

TLR_11: Use of Commercial components, technologies and processes
The Standards associated with packaging are defined to ease the use of commercial components. As an example, the voltage chosen for the distribution of current between the Power Conversion Module and the other CFMs corresponds to a voltage commonly used in telecommunications technologies.
The set of CFMs are specified in the CFM standard so as to allow use of any specialised components that might be developed by the market, ensuring the most cost efficient module implementation.
The software interface contains provision to allow the implementation of the Operating System with existing COTS OS.

The Software Interface, mainly the NII (Network Independence Interface) and the Network Standard contribute to allow the implementation of COTS communication protocols.

TLR_12: Maximise digital processing of functions
To ensure that the ASAAC architecture is applicable to a wide range of digital processing functions, Data, Graphic and Signal Processing Modules are defined in the CFM standard to enable these functions to be implemented as efficiently as possible.

TLR_13.1: General system requirements and performance
The integration of an ASAAC system into a real weapon system is dependent upon the interconnection of the core processing with non-core modules such as front-end modules. The interconnection between the ASAAC system and the external sub-system is ensured by use of the ASAAC network, designed as specified in the Communication/Network Standard. The protocol of communications between non-core and core modules is contained in the Software Standard.

TLR_13.2: Sensors and sub-system
The APOS interface offers the services required by a wide range of avionics applications and thus usable by applications such as Radar, Electro-Optic, Electronic Warfare, Sensor Fusion, Vehicle Management System, Aircraft Control System, CNI, Mission Management and Man Machine Interface.
Data, Graphic and Signal Processing Modules are defined in the CFM standard to enable these functions to be implemented as efficiently as possible.

TLR_13.3: Interface definitions
The sharing of hardware resources will be defined at design time by the system designer. Associated information will be placed in the Run Time Blueprints. Using the SMBP services, as defined in the Software Standard, the System Management will obtain the required information to configure the shared resources and to control them.

TLR_13.4: Criticality of functions
The Standards provide facilities to implement functions having different criticality as security, mission critical, survival critical and safety critical.
The System Management Guidelines mainly describe policy and mechanisms for:

- Security
- Fault Management
- Safety

TLR_13.5: System processing performance and memory requirements
This requirement addresses the performance aspect of the architecture. The performance associated with each CFM is dependent upon the technology used for their implementation. However, the CFM standard specifies 6 different CFM, three specialised in technical areas such as Signal Processing, Graphics Processing and Data Processing, three specialised in support such as Mass Memory, Power Conversion and Network Switching. This provision is

expected to meet the performance requirements of the majority of applications that will be run in the core.
Scalability to improve performances can be reached by adding several CFMs from the same type.

TLR_14.1: Growth Capability
No limitation is identified in the ASAAC Standards, they are equally applicable to the smallest system through to the largest possible system.
The incorporation of new or enhanced capabilities can be made through:

- The incorporation of new functional application software (and associated blueprint).
- And/or the incorporation of new or additional CFMs.

New or additional CFMs can easily be integrated. When a CFM satisfies the MOS interface, the Network Interface and the Packaging Standard, it is interoperable with the other CFMs. New or additional CFMs, as well as new functional application software, are handled by the system management through the use of Run Time Blueprints, which can be read by GSM through the SMBP interface.

TLR_14.2: Modularity and configurability
The definition of the System Management has no limitation in terms of size of system or number of Integration Areas to be defined.

TLR_15: Certification and qualification
Services defined in the software interfaces allow the implementation of safety critical applications. However, qualification and certification are dependent on system implementation and are performed for each system, depending on the performance of the functions implemented in the System Management.
The architecture supports Virtual Machines and Trusted Function Calls mechanism.

TLR_16: Security
A set of services are defined in the software Interfaces to provide identification and authentication functionality. These services are an initial provision for implementing an IMA system security policy.
A set of technical measures for security implemented in the System Management was demonstrated during ASAAC Stage 2. Twelve Security Technical Measures have been validated which can be used to ensure the Integrity, Availability and Confidentiality of protectively marked data.

TLR_17: System management
To handle the system issues functions (initialisation, shutdown, configuration, reconfiguration, security, and fault management) services are defined in the Software Standard**.**
These services offer the system designer the functionality required to implement these system issues into a real IMA system. The System Management Guidelines provide recommendation for their implementation.

TLR_18: Environmental effects
Environmental effects are taken into account in the specification of the Packaging Standard and of the power supply *distribution shown in the Architecture Standard.

TLR_19: Buildability
The ability to implement the ASAAC architecture is demonstrated in Stage 2.