



The European Organisation for Civil Aviation Equipment
L'Organisation Européenne pour l'Équipement de l'Aviation Civile

INTEGRATED MODULAR AVIONICS (IMA) DEVELOPMENT GUIDANCE AND CERTIFICATION CONSIDERATIONS

This document is the exclusive intellectual and commercial property of EUROCAE.

It is presently commercialised by EUROCAE.

This electronic copy is delivered to your company/organisation for internal use exclusively.

In no case it may be re-sold, or hired, lent or exchanged outside your company.

ED-124

June 2007

INTEGRATED MODULAR AVIONICS (IMA) DEVELOPMENT GUIDANCE AND CERTIFICATION CONSIDERATIONS

This document is the exclusive intellectual and commercial property of EUROCAE.

It is presently commercialised by EUROCAE.

This electronic copy is delivered to your company/organisation for internal use exclusively.

In no case it may be re-sold, or hired, lent or exchanged outside your company.

ED-124

June 2007

TABLE OF CONTENTS

FOREWORD	VI
EXECUTIVE SUMMARY	VII
CHAPTER 1 INTRODUCTION	1
1.1 PURPOSE	1
1.2 SCOPE	1
1.3 BACKGROUND	2
1.4 RELATIONSHIP TO OTHER DOCUMENTS	2
1.5 REFERENCES	3
1.6 HOW TO USE THIS DOCUMENT	3
CHAPTER 2 INTEGRATED MODULAR AVIONICS OVERVIEW	5
2.1 IMA DESIGN AND CERTIFICATION TERMINOLOGY	5
2.1.1 IMA Design Terminology	5
2.1.2 Certification Terminology	7
2.2 ARCHITECTURAL CONSIDERATIONS	8
2.3 KEY CHARACTERISTICS	8
2.3.1 Platforms and Hosted Applications	9
2.3.2 Shared Resources	10
2.3.3 Robust Partitioning	11
2.3.4 Application Programming Interface (API)	11
2.3.5 Health Monitoring and Fault Management	11
2.4 STAKEHOLDERS	12
2.4.1 Certification Authority	12
2.4.2 Certification Applicant	12
2.4.3 IMA System Integrator	13
2.4.4 Platform and Module Suppliers	13
2.4.5 Application Supplier	13
2.4.6 Maintenance Organization	13
CHAPTER 3 GENERAL DEVELOPMENT CONSIDERATIONS	14
3.1 IMA SYSTEM DEVELOPMENT PROCESS	15
3.1.1 IMA Platform Development Process	15
3.1.2 Hosted Application Development Process	17
3.1.3 IMA System Development Process	17
3.2 IMA SYSTEM RESOURCE ALLOCATION ACTIVITIES	19
3.3 AIRCRAFT SAFETY AND SECURITY	20
3.4 DEVELOPMENT ASSURANCE AND TOOL ASSURANCE	20
3.5 PARTITIONING AND RESOURCE MANAGEMENT ACTIVITIES	20
3.5.1 Design for Robust Partitioning	22
3.5.2 Partitioning Analysis	23
3.6 HEALTH MONITORING AND FAULT MANAGEMENT	25
3.6.1 Components and Aspects to be Monitored	25
3.6.2 Health Determination of Each Application	26

3.6.3	Health Determination of the IMA System as a Whole	26
3.6.4	Response to Each Type of Failure	26
3.6.5	Flight Crew Annunciation and Messaging	26
3.6.6	Control of Maintenance Actions and Reporting	27
3.6.7	Redundancy Management	27
3.6.8	Single Event Upset (SEU) Faults	28
3.7	IMA SYSTEM CONFIGURATION MANAGEMENT	28
3.7.1	Configuration Data	29
3.8	GUIDANCE ON USE OF SHARED DATABASES	30
3.9	MASTER MINIMUM EQUIPMENT LIST (MMEL).....	30
3.9.1	Design Considerations for MMEL	30
3.9.2	Approval Considerations for an MMEL	30
3.10	HUMAN FACTORS CONSIDERATIONS.....	31
CHAPTER 4	CERTIFICATION TASKS	33
4.1	OVERVIEW OF THE CERTIFICATION PROCESS.....	33
4.2	TASK 1 – MODULE ACCEPTANCE	35
4.2.1	Module Acceptance Objectives.....	35
4.2.2	Module Acceptance Data	36
4.2.3	Module Acceptance Plan (MAP)	36
4.2.4	Module Requirements Specification (MRS)	38
4.2.5	Module Validation and Verification (V&V) Data	39
4.2.6	Module Quality Assurance (QA) Records	40
4.2.7	Module Configuration Index (MCI)	40
4.2.8	Module Acceptance Configuration Management (CM) Records.....	40
4.2.9	Module Acceptance Accomplishment Summary (MAAS)	40
4.2.10	Module Acceptance Data Sheet (MADS).....	41
4.2.11	Module Problem Reports.....	42
4.2.12	Additional Module Acceptance Life Cycle Data	42
4.3	TASK 2 – APPLICATION ACCEPTANCE.....	42
4.3.1	Application Acceptance Objectives	43
4.3.2	Application Acceptance Data	43
4.4	TASK 3 – IMA SYSTEM ACCEPTANCE	44
4.4.1	IMA System Acceptance Objectives	44
4.4.2	IMA System Acceptance Data	44
4.4.3	IMA System Certification Plan (IMASCP)	45
4.4.4	IMA System Validation and Verification Plan (IMASVVP)	46
4.4.5	IMA System Configuration Index (IMASCI).....	46
4.4.6	IMA System Accomplishment Summary (IMASAS).....	47
4.4.7	Other IMA System Life Cycle Data	47
4.5	TASK 4 – AIRCRAFT INTEGRATION OF IMA SYSTEM (INCLUDING V&V).	47
4.5.1	Aircraft Integration Objectives	48

	4.5.2	Aircraft-level IMA System Compliance Data	48
	4.5.3	Aircraft-level IMA System Certification Plan (IMASCP)	49
	4.5.4	Aircraft-level IMA Validation & Verification Plan.....	49
	4.5.5	Aircraft-level IMA System Configuration Index (IMASCI).....	49
	4.5.6	Aircraft-level IMA System Accomplishment Summary (IMASAS).....	49
	4.5.7	Other Aircraft-level Data.....	49
4.6		TASK 5 – CHANGE	50
	4.6.1	Changes to IMA System Modules, Resources and Applications	50
	4.6.2	Change Objectives	50
	4.6.3	Change Management Process.....	50
	4.6.4	Change Impact Analysis (CIA)	52
	4.6.5	Change Data	53
4.7		TASK 6 – REUSE OF MODULES OR APPLICATIONS	53
	4.7.1	Objectives of the Reuse Process	53
	4.7.2	Reuse of a Software Module or Application	54
	4.7.3	Reuse of a Complex Electronic Hardware Module or Application	54
	4.7.4	Reuse of Environmental Qualification Test Data	55
	4.7.5	Reuse of a Module that Contains Software and Hardware.....	55
	4.7.6	Reuse Compliance Data	55
CHAPTER 5		INTEGRAL PROCESSES	57
5.1		SAFETY ASSESSMENT	57
	5.1.1	Responsibilities of the Certification Applicant	57
	5.1.2	Responsibilities of the IMA System Integrator	58
	5.1.3	Responsibilities of the IMA Platform Supplier	58
	5.1.4	Responsibilities of the Application Supplier	58
	5.1.5	Safety Assessment Activities	59
5.2		SYSTEM DEVELOPMENT ASSURANCE	63
	5.2.1	Software Guidance	63
	5.2.2	Electronic Hardware Guidance	63
	5.2.3	Integration Tool Qualification	63
	5.2.4	Shared Design Assurance.....	64
	5.2.5	IMA System Configuration Management	64
	5.2.6	Environmental Qualification Testing (EQT).....	64
5.3		VALIDATION	65
5.4		VERIFICATION.....	66
5.5		CONFIGURATION MANAGEMENT (CM).....	68
	5.5.1	IMA System Configuration Management Plan	68
	5.5.2	Configuration Control	69
5.6		QUALITY ASSURANCE (QA)	69
5.7		CERTIFICATION LIAISON	70
	5.7.1	Certification Liaison Process.....	70

5.7.2	Means of Compliance and Planning Data.....	70
5.7.3	Development Life Cycle Data.....	71
5.7.4	Compliance Substantiation.....	72
5.7.5	Life Cycle Data Submittals	73
5.7.6	Certification Liaison Process When Changes Are Made	74
5.7.7	Certification Liaison Process For Reuse of Modules	74
CHAPTER 6	CONSIDERATIONS FOR CONTINUED AIRWORTHINESS OF IMA SYSTEMS	75
6.1	TRAINING.....	75
6.2	MAINTENANCE.....	75
6.3	POST-CERTIFICATION MODIFICATIONS	76
ANNEX A	OBJECTIVE TABLES.....	77
ANNEX B	GLOSSARY OF TERMS	86
ANNEX C	LIST OF ABBREVIATIONS AND ACRONYMS	96
ANNEX D	IMA SYSTEM DESIGN EXAMPLES	99
D.1	EXAMPLE 1: SINGLE LRU PLATFORM.....	99
D.1.1	Purpose of this example.....	99
D.1.2	Definition of platform and modules.....	99
D.1.3	Key characteristics of IMA found in this system.....	100
D.2	EXAMPLE 2: DISTRIBUTED IMA PLATFORM.....	101
D.2.1	Purpose of this example.....	101
D.2.2	Definition of platform and modules.....	101
D.2.3	Key characteristics of IMA found in this system.....	102
D.3	EXAMPLE 3: IDENTIFYING BOUNDS OF A DISTRIBUTED COMPLEX IMA SYSTEM	104
D.3.1	Purpose of this example.....	104
D.3.2	Definition of platform and modules.....	104
D.3.3	Key characteristics of IMA found in this system.....	104
D.4	EXAMPLE 4: SOFTWARE DESIGNED RADIO EXAMPLE.....	106
D.4.1	Purpose of this example.....	106
D.4.2	Definition of platform and modules.....	106
D.4.3	Key characteristics of IMA found in this system.....	107
LIST OF WORKING GROUP 60 MEMBERS.....		108

LIST OF FIGURES

Figure 1: Chapters and their relationships.....	4
Figure 2: Relationship of IMA design terms.....	6
Figure 3: Example of a typical design highlighting potential shared resources.....	14
Figure 4: IMA system certification tasks illustration	33
Figure 5: Planning data for IMA system.....	71
Figure 6: Life cycle data for IMA system.....	72
Figure 7: Compliance summaries for IMA	73
Figure D-1: Configured single LRU platform.....	100
Figure D-2: Distributed modular platform.....	101
Figure D-3: Common module structure	102
Figure D-4: Distributed complex IMA system.....	104
Figure D-5: Fault reporting hierarchy.....	105
Figure D-6: SDR IMA platform architecture	106

LIST OF TABLES

Table 1: Key IMA platform characteristics	9
Table 2: Key application characteristics	10
Table 3: Relationship among integration activities and acceptance tasks	19
Table 4: Overview of IMA certification tasks.....	34
Table 5: Overview of typical validation activities.....	66
Table 6: Overview of typical verification activities.....	67
Table 7: CC1/CC2 definition	69
Table 8: Life cycle data to be submitted to certification authority.....	73
Table A-1: IMA module/platform development process (Task 1) objectives	77
Table A-2: Hosted application development and acceptance (Task 2) objectives.....	79
Table A-3: IMA system-level development and acceptance (Task 3) objectives	80
Table A-4: Aircraft-level integration (Task 4) objectives	82
Table A-5: Change (Task 5) objectives	84
Table A-6: Module or application reuse (Task 6) objectives	85

FOREWORD

1. This document, jointly prepared by EUROCAE Working Group 60 and RTCA Special Committee 200, was accepted by the Council of EUROCAE on 30 June 2007.
2. EUROCAE is an international non-profit making organization. Membership is open to manufacturers and users in Europe of equipment for aeronautics, national civil aviation administrations, trade association and, under certain conditions, non-European members. Its work program is principally directed to the preparation of performance specifications and guidance documents for civil aviation equipment, for adoption and use at European and world-wide levels.
3. The findings of EUROCAE are resolved after discussion among its members and in cooperation with the RTCA Inc., Washington DC, and/or the Society of Automotive Engineers (SAE), Warrendale PA, USA, through their appropriate committees.
4. EUROCAE Performance Specifications and Guidance Documents are recommendations only; EUROCAE is not an official body of the European Governments. Its recommendations therefore are not valid as statements of official policy unless adopted by a particular government or conference of governments.
5. Copies of this document may be obtained from:

EUROCAE

102 rue Etienne Dolet

92240 MALAKOFF

France

Tel: 33 1 40 92 79 30

Fax: 33 1 46 55 62 65

Email: eurocae@eurocae.net

Web Site: www.eurocae.net

EXECUTIVE SUMMARY

The use of Integrated Modular Avionics (IMA) is rapidly expanding and is found in all classes of aircraft. In recognition of this rapid growth EUROCAE established Working Group 60 (WG-60) and RTCA established Special Committee 200 (SC-200) to jointly develop a document that could be used as guidance in the design, development, and application of IMA. Participants in the development of the document included government, industry, and academic personnel.

IMA is a shared set of flexible, reusable, and interoperable hardware and software resources that, when integrated, form a platform that provides services, designed and verified to a defined set of safety and performance requirements, to host applications performing aircraft functions.

This document provides guidance for IMA developers, integrators, applicants, and those involved in the approval and continued airworthiness of IMA systems. It provides specific guidance for the assurance of IMA systems as differentiated from traditional federated avionics.

The development of this document is based on earlier EUROCAE/RTCA documents, for example EUROCAE ED-12/RTCA DO-178, Software Considerations in Airborne Systems and Equipment Certification and EUROCAE ED-80/RTCA DO-254, Design Assurance Guidance for Airborne Electronic Hardware. Concepts from other EUROCAE and RTCA documents, as well as SAE and ARINC documents, also guided the document preparation.

CHAPTER 1

INTRODUCTION

1.1 PURPOSE

This document contains guidance for Integrated Modular Avionics (IMA) developers, application developers, integrators, certification applicants, and those involved in the approval and continued airworthiness of IMA systems in civil certification projects. The guidance describes the objectives, processes, and activities for those involved in the development and integration of IMA modules, applications, and systems to incrementally accumulate design assurance toward the installation and approval of an IMA system on an approved aviation product as differentiated from traditional federated aviation system architectures.

IMA system concepts are presented, including the platform and modules, and their relationships to the hosted applications and avionics functions used in an aircraft installation. This includes the description of how the developers and integrators can accumulate incremental acceptance of the modules, platform, and application integration which will provide a means for applicants to achieve design assurance of an IMA system on an approved aviation product.

During the IMA system development the certification applicant for a Type Certificate (TC) or Supplemental Type Certificate (STC) program should develop an effective system of communication among the module and platform developers and system integrators. This is especially important when these suppliers are from different companies. Otherwise, there may be a misunderstanding of the implementation during final integration and approval of the IMA system installation.

Six tasks define the incremental acceptance of IMA systems in the certification process:

- Task 1: Module acceptance.
- Task 2: Application software or hardware acceptance.
- Task 3: IMA system acceptance.
- Task 4: Aircraft integration of IMA system - including Validation and Verification (V&V).
- Task 5: Change of modules or applications.
- Task 6: Reuse of modules or applications.

Approval of an IMA system installation may be based on the accumulation of incremental acceptance, culminating in the complete design assurance needed to demonstrate that the installed system and functions comply with the applicable regulations and guidance.

The incremental acceptance, if appropriate, may be granted in the form of an acceptance letter, stamped type design data, or other means for the specific project.

1.2 SCOPE

This document is applicable to all parties involved in the development, integration, verification, and validation of IMA systems. The guidance in this document is focused on IMA-specific aspects of design assurance. As such it relies on other recognized documents that provide additional guidance on overall aircraft certification, system development, software assurance, and hardware assurance.

As defined in this document, IMA is described as a shared set of flexible, reusable, and interoperable hardware and software resources that, when integrated, form a platform that provides services, designed and verified to a defined set of requirements, to host applications performing aircraft functions. Key IMA and certification terms are described in Section [2.2](#) and in the glossary.

The primary industry-accepted guidance for satisfying airworthiness requirements for IMA components is included in Section [1.5](#). The ability to obtain incremental acceptance of individual items of the IMA platform (including the core software) and hosted applications enables the reduction of follow-on certification efforts without compromising system safety.

This document describes application properties as they relate to their integration with a platform; however, it does not address the specific functionality of applications nor specific Technical Standard Order (TSO)/European Technical Standard Order (ETSO) requirements, Minimum Operational Performance Specifications (MOPS), or Minimum Aviation System Performance Specifications (MASPS).

1.3 BACKGROUND

The evolution of software and microelectronics technology enables the introduction of new aircraft functions, new capabilities, and increased levels of complexity. The need to perform these complex functions necessitates the use of high-performance computing platforms that can host multiple applications on a single processor or a distributed network of processors.

Aircraft functions are increasingly being implemented by electronic systems. IMA is a means to accommodate this increase and reduce the number of units per aircraft. Additional benefits are reduced weight and volume of aircraft systems and equipment.

The IMA platform should be capable of providing robust partitioning and other protection means that allow multiple applications to share a platform and its resources, or to support functions distributed across a fault-tolerant network. An IMA platform may be provided by the system integrator or by a third party supplier. Likewise, IMA applications may be provided by the platform supplier or third party suppliers.

Economic factors, including the needs to provide cost-effective upgrade paths, introduce new operational capabilities (e.g., CNS/ATM functions), fast and efficient maintenance, and to avoid premature obsolescence are the primary incentives for the IMA concept. In light of this background, this document describes a set of processes and guidance to be used for the development and certification of IMA systems.

1.4 RELATIONSHIP TO OTHER DOCUMENTS

In addition to the airworthiness regulations and requirements, various national and international standards for software, avionics, complex electronics, and safety are available. In some communities, compliance with these standards may be required. However, it is outside the scope of this document to invoke specific national or international standards, or to propose a means by which these standards might be used as an alternative or supplement to this document.

Where this document uses the term “standards” it should be interpreted to mean the use of project-specific standards as applied to the aircraft systems, equipment, and engines. Such standards may be derived from general standards produced or adopted by the applicant for their certification activities.

1.5

REFERENCES

The latest version of the following documents apply:

- [1] EUROCAE ED-14 / RTCA DO-160, Environmental Conditions and Test Procedures for Airborne Equipment.
- [2] EUROCAE ED-12 / RTCA DO-178, Software Considerations in Airborne Systems and Equipment Certification.
- [3] EUROCAE ED-76 / RTCA DO-200, Standards for Processing Aeronautical Data.
- [4] EUROCAE ED-77 / RTCA DO-201, Industry Requirements for Aeronautical Information.
- [5] EUROCAE ED-94 / RTCA DO-248, Final Report for Clarification of DO-178B "Software Considerations in Airborne Systems and Equipment Certification".
- [6] EUROCAE ED-80 / RTCA DO-254, Design Assurance Guidance for Airborne Electronic Hardware.
- [7] EUROCAE ED-79 / SAE ARP4754, Certification Considerations for Highly Integrated or Complex Aircraft Systems.
- [8] SAE ARP4761, Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment.
- [9] Federal Aviation Administration (FAA) AC 20-148, Reusable Software Components.
- [10] FAA TSO-C153, Integrated Modular Avionics Hardware Elements.
- [11] FAA Order 8110.49, Software Approval Guidelines.
- [12] ARINC 615A, Software Data Loading.
- [13] ARINC 653, Avionics Application Software Standard Interface.
- [14] ARINC 664, Aircraft Data Network.

NOTE: *When FAA Advisory Circulars are referenced, they are intended as material that may supply topics and areas for the applicant to consider. All requirements should be coordinated with the applicant's local certification authority.*

1.6

HOW TO USE THIS DOCUMENT

This document is intended to be used by the international aviation community. To aid in such use, references to specific national regulations and procedures are minimized. Instead, generic terms are used. For example, the term "certification authority" is used to mean the organization or person granting approval on behalf of the country responsible for aircraft and/or engine certification. Where a second country or a group of countries validates or participates in this certification, this document may be used with due recognition given to bilateral agreements or memoranda of understanding among the countries involved.

This document recognizes that the guidance herein is not mandated by law, but represents a consensus of the aviation community. Therefore, use of the word "shall" is avoided. The word "should" is used to describe the method required to adhere to this document, but it is recognized that alternative methods to those described herein may be acceptable.

The document is structured to introduce the reader to IMA and highlight the attributes of IMA that affect the development, acceptance, and installation approval of IMA systems on aircraft.

The document consists of the following chapters:

- Chapter 1 gives the reader an overview of this document.
- Chapter 2 introduces the concept of IMA and highlights the system, hardware, and software characteristics.
- Chapter 3 provides a description of IMA-specific development and integration guidelines.
- Chapter 4 provides guidelines for the acceptance of IMA and describes the relationship to the approval of the installed IMA system.
- Chapter 5 describes integral processes for IMA development.
- Chapter 6 includes guidelines for continued airworthiness of IMA systems.

Figure 1 illustrates the relationships of these chapters.

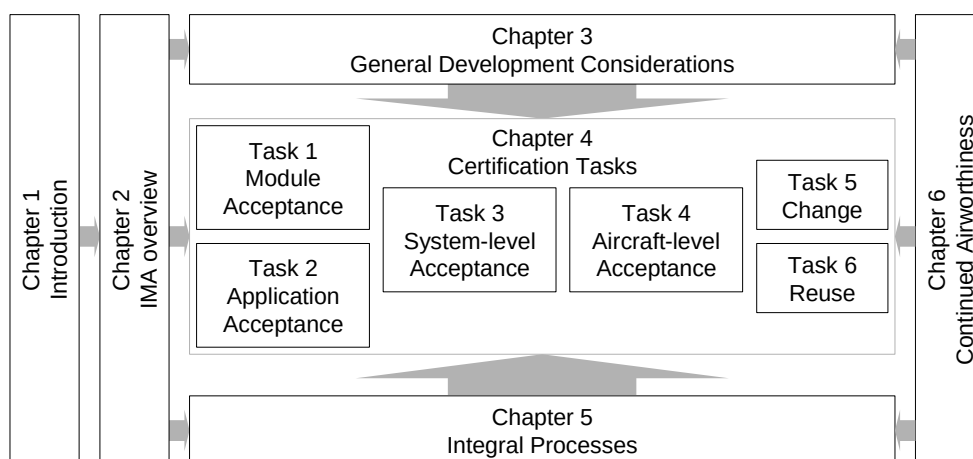


FIGURE 1 : CHAPTERS AND THEIR RELATIONSHIPS

At the end of the document the following annexes are included:

- Annex A summarizes the objectives to be satisfied to demonstrate compliance with this document. Objectives are listed in tables, using a summary of the tasks described in Chapters 3 and 4.
- Annex B is the Glossary of Terms. Note that some terms have a specific meaning in this document.
- Annex C is the list of acronyms and abbreviations used in this document.
- Annex D contains IMA examples.

NOTE: Annexes A through C are considered normative; Annex D is informational only.

CHAPTER 2

INTEGRATED MODULAR AVIONICS OVERVIEW

This chapter provides a description of an IMA system. It introduces terminology that applies to IMA and describes the relationship of terms. Key characteristics are introduced to acquaint the reader with the IMA concept.

2.1 IMA DESIGN AND CERTIFICATION TERMINOLOGY

The design, certification, and IMA terms needed to understand this document are described below. In some cases, the description of these terms expands upon the definitions included in the glossary and may be used in a slightly different fashion than in traditional aviation design and guidance material.

2.1.1 IMA Design Terminology

The following terms are used to describe IMA systems. These terms are identical to those defined in [Annex B](#), expanded to provide more understanding of the terms as they apply to IMA design.

Aircraft Function – The capability of the aircraft that may be provided by the hardware and software of the systems on the aircraft. Functions include flight control, autopilot, braking, fuel management, flight instruments, etc. IMA has the potential to broaden the definition of avionics to include any aircraft function.

Application - Software and/or application-specific hardware with a defined set of interfaces that, when integrated with a platform, performs a function.

Component - A self-contained hardware part, software part, database, or combination thereof that is configuration controlled. A component does not provide an aircraft function by itself.

Core Software - The operating system and support software that manage resources to provide an environment in which applications can execute. Core software is a necessary component of a platform and is typically comprised of one or more modules.

IMA System – Consists of an IMA platform(s) and a defined set of hosted applications.

Interoperable – The capability of several integrated modules to operate together to accomplish a specific goal or function. This requires defined interface boundaries between the modules and allows the use of other interoperable components. To describe this concept in physical terms, an IMA platform may include interoperable modules and components such as physical devices (processor, memory, electrical power, Input/Output (I/O) devices), and logical elements, such as an operating system, and communication software.

Module – A component or collection of components that may be accepted by themselves or in the context of IMA. A module may also comprise other modules. A module may be software, hardware, or a combination of hardware and software, which provides resources to the IMA-hosted applications. Modules may be distributed across the aircraft or may be co-located.

Partitioning - An architectural technique to provide the necessary separation and independence of functions or applications to ensure that only intended coupling occurs. The mechanisms for providing the protection in an IMA platform are specified to a required level of integrity.

Platform - Module or group of modules, including core software, that manages resources in a manner sufficient to support at least one application. IMA hardware resources and core software are designed and managed in a way to provide computational, communication, and interface capabilities for hosting at least one application. Platforms, by themselves, do not provide any aircraft functionality. The platform establishes a computing environment, support services, and platform-related capabilities, such as health monitoring and fault management. The IMA platform may be accepted independently of hosted applications.

Resource - Any object (processor, memory, software, data, etc.) or component used by an IMA platform or application. A resource may be shared by multiple applications or dedicated to a specific application. A resource may be physical (a hardware device) or logical (a piece of information).

Reusable - The design assurance data of previously accepted modules and applications may be used in a subsequent aircraft system design with reduced need for redesign or additional acceptance.

Figure 2 shows the relationships between many of these terms.

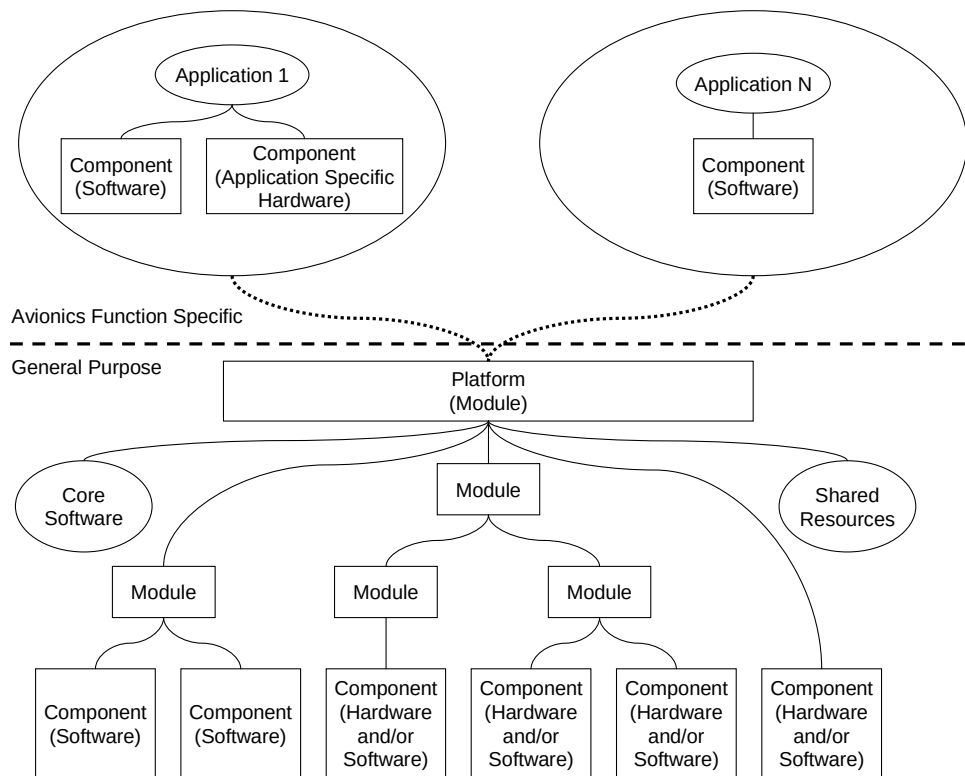


FIGURE 2 : RELATIONSHIP OF IMA DESIGN TERMS

2.1.2 Certification Terminology

A primary purpose of this document is to provide a means of compliance for IMA systems seeking acceptance, approval, and certification of their installation on an aircraft. To accomplish this it is important to understand the certification terminology. The primary certification terms relevant to IMA are defined below:

Certification - Legal recognition by a certification authority that a product, service, organization, or person complies with the requirements. Such certification includes the activities of technically checking the product, service, organization, or person, and a formal recognition of compliance with the applicable regulations and airworthiness requirements by issuance of a certificate, license, approval, or other documents as required by national laws and procedures. In particular, certification of a product includes:

- a. the certification applicant process of assessing the design of a product and demonstrating to the certification authority that it complies with the applicable regulations and a set of standards applicable to that type of product so as to demonstrate an acceptable level of safety;
- b. the process of assessing products to ensure they conform with the approved type design for that product;
- c. the certification authority process of finding compliance and the issuance of a certificate required by national laws to declare that compliance and/or conformity has been found with standards in accordance with items a or b above.

TSO Authorization - Legal recognition by the certification authority that a system, equipment, or part satisfies the TSO requirements and minimum specification applicable for that equipment. This term is equally applicable to European TSO (ETSO) authorizations.

Acceptance – Acknowledgement by the certification authority that the module, application, or system complies with its defined requirements. Acceptance is recognition by the certification authority (typically in the form of a letter or stamped data sheet) signifying that the submission of data, justification, or claim of equivalence satisfies applicable guidance or requirements. The goal of acceptance is to achieve credit for future use in a certification project.

Approval – The act or instance of giving formal or official acknowledgement of compliance with regulations. In the context of IMA there are typically two forms of approval:

- a. approval of submitted life cycle data by the certification authority (usually demonstrated by issuance of a stamped and signed data item or a letter),
- b. installation approval by the issuance of an aircraft or engine type certificate and/or airworthiness certificate.

Incremental acceptance - A process for obtaining credit toward approval and certification by accepting or finding that an IMA module, application, and/or off-aircraft IMA system complies with specific requirements. This incremental acceptance is divided into tasks. Credit granted for individual tasks contributes to the overall certification goal. Incremental acceptance provides the ability to integrate and accept new applications and/or modules, in an IMA system, and maintain existing applications and/or modules without the need for re-acceptance.

2.2 ARCHITECTURAL CONSIDERATIONS

An IMA system architecture is composed of one or more platforms and includes interfaces to other aircraft systems and users (for example, flight crew, maintenance personnel, etc.). The allocation of aircraft functions should be addressed in the IMA system architecture to ensure that applicable availability, integrity, and safety requirements are satisfied. Some of the primary considerations are listed and described below:

- a. Availability considerations
 - Functional performance – allocation of hosted aircraft functions (Section [3.2](#)) to the IMA system architecture.
 - Resource management – allocation of IMA platform resources, control of shared and dedicated resources, and protection of IMA platform resources used by multiple hosted aircraft functions or applications (Section [3.5](#)).
 - Reliability and maintainability – impact on Master Minimum Equipment List (MMEL), aircraft dispatchability, repair, replacement, and ease of performing maintenance activities to ensure continued airworthiness (Section [3.9](#)).
 - Health monitoring – monitoring the condition of the system and operational and maintenance concerns (Section [3.6](#)).
- b. Integrity considerations
 - Design assurance, IMA safety and protection features, fault detection and partitioning - ability to protect hosted functions and applications to ensure independence and to prevent unintended interactions while using shared resources (Section [3.5](#)).
- c. Safety considerations
 - Safety assessment - ensure appropriate architecture, design assurance, failure protection, address common mode and impact of combinational failures of aircraft functions, and airworthiness (Section [5.1](#)).
- d. Fault management, fault reporting, and recovery actions – detecting and identifying faults, failures and anomalous behavior, and providing appropriate responses (Section [3.6](#)).
- e. Composability considerations
 - An IMA platform is composable if integration of a new application will not invalidate any verified requirements of an already integrated application.
 - In a composable architecture, system requirements follow from requirements allocated to IMA applications. An IMA platform with well defined interface boundaries may be composable with respect to partial acceptance of integrated applications. Robust partitioning is one step toward this goal (Section [3.5.1](#)).

2.3 KEY CHARACTERISTICS

The key characteristics of IMA platforms and hosted applications influence the IMA system architecture, the detailed system design, and, ultimately, the IMA platform and system acceptance process. Sections [2.3.1](#) through [2.3.5](#) summarize the characteristics of IMA platforms that affect the certification process.

2.3.1 Platforms and Hosted Applications

Two primary building blocks of an IMA system are the platform and the hosted applications. The key characteristics for the IMA platform are provided in Table 1.

TABLE 1 : KEY IMA PLATFORM CHARACTERISTICS

Key IMA platform characteristics	Description
Platform resources are shared by multiple applications	Integration implies sharing of resources. An IMA platform is able to host multiple applications through partitioning and other protection capabilities provided by it (e.g., HIRF/IEL, power supply, Built-In-Test).
An IMA platform provides robust partitioning of shared resources	This characteristic ensures that shared platform resources are available to the hosted applications as needed, and that those resources are protected from any anomalous behavior of the applications using them. IMA platform resource management ensures that only specified, intentional usage, interactions, and interfaces are allowed by the platform and applications sharing the resources. Robust partitioning will ensure that any hosted application or function has no unintended effect on other hosted applications or functions.
An IMA platform only allows hosted applications to interact with the platform and other hosted applications through well defined interfaces	An IMA platform is a general purpose computing platform able to host one or more aircraft functions or applications. As such, platform behavior may be verified independent of specific applications (e.g., it may be shown to meet its module requirement specification). The IMA platform is viewed as a separately accepted component of an IMA system. This characteristic is necessary to isolate changes between the platform and hosted applications. The intent is to allow modification of the IMA platform with minimum impact on the hosted applications, and changes to applications with minimum impact on the platform. The platform provides documented (and verified) Application Programming Interfaces (API) to allow applications access to platform services and resources.
Shared IMA platform resources are configurable	IMA platform resources are configurable to support the resource requirements of hosted applications.

The key characteristics for IMA hosted applications are provided in Table 2.

TABLE 2 : KEY APPLICATION CHARACTERISTICS

Key Application Characteristics	Description
An application may be designed independent of other applications and obtain incremental acceptance on the IMA platform independently of other applications	Hosted applications may be individually verified on the platform without the full suite of intended applications. The incremental acceptance for each hosted application can be used to support the accumulation of credit toward approval of the IMA system installation on the aircraft.
Applications can be integrated onto a platform without unintended interactions with other hosted applications	As the different applications reach completion and are verified individually, they should be integrated on the platform as a complete suite of hosted applications. Interactions between applications should be verified.
Applications may be reusable	Application modularity and portability may enable and facilitate use on different projects and products.
Applications are independently modifiable	Each application is modifiable with little or no impact on other applications and platform resources and modules. Any impacts should be identified and coordinated with the affected components.

2.3.2

Shared Resources

IMA systems may host several applications that share resources. For example, resources may be shared by the method of access time. This applies to processing resources and hardware. Each shared resource has the potential to become a single point failure that can affect all applications using that resource. Accordingly, appropriate mitigation techniques should be applied as determined by the system safety assessment process.

Processing resource refers to a physical element that may contain Central Processing Unit(s) (CPU), memory and associated interfaces. Memory is capable of storing machine-readable computer programs and associated data. The IMA hosted applications will communicate using shared resources provided by the IMA platform (for example, I/O devices, data buses, shared memory, etc.). A resource or portion of a resource can be allocated per unit time (for example, processor cycles or communication bandwidth).

The IMA platform provides resource management capabilities for shared resources, as well as health monitoring and fault management capabilities to support the protection of shared resources. IMA systems may have multiple electrical power sources that are shared.

2.3.3 Robust Partitioning

Robust partitioning is a means for assuring the intended isolation of independent aircraft functions and applications residing in IMA shared resources in the presence of design errors and hardware failures that are unique to a partition or associated with application-specific hardware. If a (different) failure can lead to the loss of robust partitioning then it should be detectable and appropriate action taken. The objective of robust partitioning is to provide the same level of functional, if not physical, isolation and protection as a federated implementation. This means robust partitioning should support the cooperative coexistence of core software and applications hosted on a processor and using shared resources, while assuring unauthorized or unintended interference is prevented. Robust partitioning should meet the following information guidelines in ED-94/DO-248 (Ref. [5], Section 4.1.4.5):

- A software partition should not be allowed to contaminate the code, I/O, or data storage areas of another partition.
- A software partition should be allowed to consume shared processor resources only during its allocated time.
- A software partition should be allowed to consume only its allocation of shared I/O resources.
- Failures of hardware unique to a software partition should not cause adverse effects on other software partitions.

Robust partitioning is a means for assuring the intended isolation and independence in all circumstances (including hardware failures, hardware and software design errors, or anomalous behavior) of aircraft functions and hosted applications using shared resources. The objective of robust partitioning is to provide an equivalent level of functional isolation and independence as a federated system implementation (i.e., applications individually residing on separate Line Replaceable Units (LRU)). This means robust partitioning supports the cooperative coexistence of applications using shared resources, while assuring that any attempted unauthorized or unintended interaction is detected and mitigated.

The platform robust partitioning protection mechanisms are independent of any hosted applications, that is, applications cannot alter the partitioning protection provided by the platform.

2.3.4 Application Programming Interface (API)

An API defines the standard interfaces between the platform and the hosted applications and provides the means to communicate between applications and to use I/O capabilities. ARINC 653 (Ref. [13]) Parts 1 and 2 provide an avionics standards for an application programming interface and the related services. ARINC 653 Part 3 provides a standardized test specification to demonstrate compliance to ARINC 653.

2.3.5 Health Monitoring and Fault Management

Health monitoring and fault management (HM/FM) functions deserve special attention due to the integration of multiple applications and resource sharing. IMA systems manage platform faults, hardware failures, partitioning violations, and errors and anomalous behavior of hosted applications, including common mode faults and cascading failures. The methods used to manage platform faults are independent of the hosted applications. Fault management consists of detecting faults, failures and errors, correctly identifying them when they occur and performing the appropriate response. Guidance for fault management and fault reporting is included in Section [3.6](#).

The IMA platform provides health monitoring and fault management capabilities for the platform and hosted applications. The IMA system may have to provide higher level (aircraft function) health monitoring and fault management capabilities to support availability and integrity requirements.

2.4 STAKEHOLDERS

The assignment of roles and responsibilities is necessary, and should address the entire IMA system life cycle from conceptual design to retirement. These assignments may be based on the underlying IMA system architecture and should include the responsibility for selecting and providing tools. These assignments should be stated in the IMA system project plans, such as the IMA System Certification Plan and supporting lower-level plans.

This section identifies the typical data provided in support of the development, approval, and continued operational safety of IMA systems installed on aircraft. The responsibility for certain activities, data generation, and for demonstrating compliance should be coordinated, communicated, and resolved among the stakeholders as early as possible in the program. Data is accumulated to support incremental acceptance of IMA modules and hosted applications.

While a single group or organization may be able to perform all of the activities described (with the exception of the certification authority), it is anticipated that there will be multiple organizations involved. Certification applicants are strongly encouraged to coordinate their plans with the certification authority as early as possible and to maintain coordination throughout the IMA system development.

Where a failure or fault impacts common IMA components used across projects and aircraft products, there should be a process in place that provides information to be provided to all users of the common components. This process should have communications, not only to the users, but also to the certification authority and those responsible for continued operational safety of the aircraft product. There should be clear and well-defined communications paths and procedures to follow for recording the decisions made, the rectification path, and the agreement for any certification authority actions.

2.4.1 Certification Authority

The certification authority is the organization(s) granting approval on behalf of the state(s) responsible for aircraft and/or engine certification.

2.4.2 Certification Applicant

The applicant is responsible for demonstrating compliance to the applicable aviation regulations, and is seeking a Type Certificate (TC), Amended TC (ATC), Supplemental Type Certificate (STC) or Amended STC (ASTC). The applicant is responsible for generating and validating all aircraft-level requirements and their allocation to subsystems, providing installation instructions for the configured IMA system, integrating it with other aircraft systems, and, where appropriate, installing it on the aircraft. The applicant is ultimately responsible for ensuring that the installed IMA system and the aircraft comply with applicable regulations and are airworthy.

The applicant will perform the necessary development activities to define aircraft-level functions to be implemented and the necessary V&V activities on the system after it is installed in the aircraft. The applicant is also responsible for continued airworthiness (see [Chapter 6](#)).

NOTE: *The applicant is not required to perform the actual installation of the IMA system on the aircraft but is still responsible for the items described above.*

2.4.3 IMA System Integrator

The IMA system integrator performs the activities necessary to integrate the platform(s) and hosted applications to produce the IMA system. Typical data developed during IMA system integration includes:

- a. The system configuration consisting of the number, type, and specific versions of modules and hosted applications.
- b. The shared resource allocation and configuration tables for the integrated IMA system.
- c. Results of IMA system V&V, including performance data for the IMA system, consistent with the allocated requirements.

2.4.4 Platform and Module Suppliers

The IMA platform and module suppliers provide the processing hardware and software resources, including the core software. Development and configuration tools are provided to support use of the IMA platform or module. Typical data developed during platform and module development includes:

- a. Specification of the interfaces to the IMA platform and modules.
- b. Specification of the shared resource allocation and configuration tables.
- c. Required resources and configuration data for the IMA platform, including core software.
- d. Results of module and/or platform V&V, including performance data, consistent with allocated requirements.

2.4.5 Application Supplier

The application supplier develops the hosted application and verifies it on the IMA platform. The application supplier should ensure that any hardware or software resources that are unique to the hosted application meet the integrity and availability requirements consistent with the assigned failure condition classification, as determined by the aircraft safety assessment. Typical data developed during application development includes:

- a. Specification of external interfaces required by the application.
- b. Required resources and configuration data for the hosted application.
- c. Results of application/platform integration V&V, including performance data, consistent with allocated requirements.

2.4.6 Maintenance Organization

The maintenance organization follows the approved procedures received from the certification applicant to keep the IMA system and the aircraft in an airworthy condition. Additional information relating to continued airworthiness is provided in [Chapter 6](#).

CHAPTER 3

GENERAL DEVELOPMENT CONSIDERATIONS

The development of an IMA system is based on an IMA platform containing hardware and software that are common and can be shared by the hosted applications. Figure 3 shows an example of a typical design highlighting potential shared resources. The shaded areas identify the modules that may be shared.

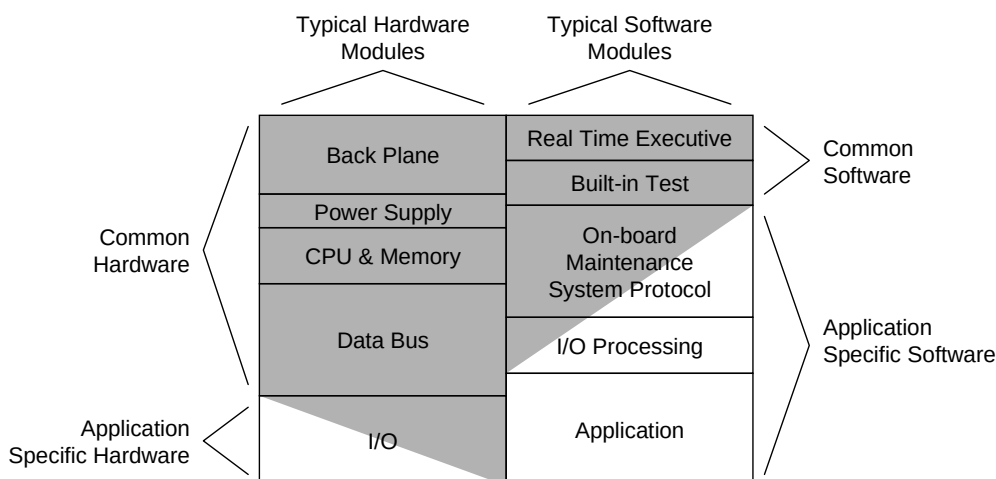


FIGURE 3 : EXAMPLE OF A TYPICAL DESIGN HIGHLIGHTING POTENTIAL SHARED RESOURCES

An IMA development process should ensure the following:

- Aircraft functions allocated to a specific IMA system are consistent with the design of the system (Section [3.2](#)).
- Aircraft safety and security requirements allocated to a specific IMA system are identified and have been satisfied by the IMA system design. This should include assignment of system development assurance, hardware design assurance, and software levels. These levels are determined by the aircraft-level safety assessment to support aircraft functions implemented by hosted applications and supporting availability and integrity requirements, as well as any requirements for tool assessment and qualification (Sections [3.3](#) and [3.4](#)).
- Behavior of any hosted application is prevented from adversely affecting the behavior of any other application or function by the design of the IMA platform. The platform has robust partitioning, resource management, and other protection means appropriate to the aircraft functions and hosted applications (Sections [3.5](#) and [3.8](#)).
- Health monitoring, failure reporting process, and fault management functions are provided for the platform to meet specified requirements of the hosted applications and IMA system (Section [3.6](#)).
- Configuration management for the IMA platform, applications, integrator, and certification applicant are established and maintained (Section [3.7](#)).
- IMA system dispatch requirements are implemented and verified. (Section [3.9](#))
- Human factors requirements pertaining to the IMA system are implemented and verified. (Section [3.10](#))

3.1 IMA SYSTEM DEVELOPMENT PROCESS

The overall development process for an IMA system should follow a structured process, such as ED-79/ARP 4754 (Ref. [7]). The IMA system development process should consider the primary characteristics of IMA: flexible, reusable, and interoperable. These characteristics influence the development process which should address, as a minimum:

- a. The IMA platform – Definition of reusable, sharable modules and resources (including application programming interfaces and health monitoring strategy).
- b. The hosted applications – Definition of the interfaces and system contracts to allow a given hosted application to reside on the given platform.
- c. The IMA system – Integration of the specific set of hosted applications onto a given IMA platform(s).

The process outlined here supports the reuse of the IMA platform and hosted application development data on another aircraft program.

The following terms will be used when outlining the development process (these terms supplement those in Section [2.1](#)):

- **IMA platform** consists of the IMA modules and core software without any aircraft functions or applications installed.
- **IMA platform architecture** refers to the means of structuring, connecting and combining IMA modules to support the requirements of the hosted applications and aircraft functions.
- **IMA system** consists of the IMA platform(s) and the hosted applications.
- **IMA system architecture** consists of the IMA platform(s), connections and components required to meet the requirements of all of the hosted applications and aircraft functions.

3.1.1 IMA Platform Development Process

The IMA platform may be defined and developed separately from the specific aircraft functions and the hosted applications. If this can be achieved, it may be possible for the data developed for an IMA platform to be reusable with a different set of hosted applications.

One of the primary goals of IMA is to develop an IMA platform that can be reused on different aircraft and with different hosted applications. A reusable IMA platform should be developed using the process objectives described below. The module acceptance process is further described in [Chapter 4](#).

- a. Plan and define the IMA platform. The definition should include:
 1. The architecture definition, which contains the type and general function of the various IMA modules, resources and components, and how they will interact (e.g., distributed vs. centralized architecture).
 2. An approach for integrating hosted applications, both hardware and software, onto the IMA platform.
 3. An IMA platform acceptance approach.
 4. An IMA system certification approach that includes support for hosted applications and stakeholder roles and responsibilities for developing compliance data.
 5. A list of platform services to be provided to the hosted applications.
 6. The intended level of aircraft function availability and integrity needed, platform capabilities to support it, and methods for supporting it.

7. The health management and fault management approaches (excluding the actual external interfaces, as this will be dependent on the aircraft HM and crew alerting systems).
 8. The platform and IMA system configuration management approaches.
- b. Define the IMA platform requirements, including:
1. Safety capabilities:
 - i. identification of the top level platform failure events that could affect hosted applications.
 - ii. definition of the acceptable failure rates (reliability requirements of platform hardware modules) associated with each failure event.
 - iii. development of guidance for use of the above data in meeting the availability and integrity requirements of an aircraft and potential hosted functions.
 - iv. definition of the safety requirements, including robust partitioning, health monitoring, fault management, resource management, other safety features and other protection means.
 2. Performance capabilities.
 3. Configuration management approach.
 4. Environmental conditions under which the platform modules are intended to operate. Where modules share a common environment, such as a cabinet with common power or a common data bus, a definition of the conditions for the common environment should also be described.
 5. Fault management and reporting approach and requirements, including considerations for: fault tolerance, fault isolation to modules, and detection and isolation of single failures (for example, failures resulting in loss of capability, such as internal power source failures, redundant inter-module communication channels, and other similar resources should be considered).
 6. Detailed requirements for each aspect of the concept definition.
 7. IMA platform architecture which has been defined and verified to the required safety capabilities.
- c. Develop and implement the IMA platform design. The software and hardware development processes should follow ED-12/DO-178 (Ref. [2]) and ED-80/DO-254 (Ref. [6]), respectively, along with any regulatory supplemental documents, at the appropriate level to meet the required safety requirements. Additionally, common cause analysis (CCA) should be performed and qualitative failure analysis for the various top level events defined for the platform should be developed.
- d. Verify and validate the IMA platform addressing the following activities:
1. Perform environmental qualification testing to the specified environmental conditions.
 2. Perform a partitioning analysis and verification testing; verify other protection capabilities and safety features (i.e., resource management, health monitoring, fault management, built-in test (initial and continuous), etc.).
 3. Complete the CCA.
 4. Complete the numerical analysis showing that implementation meets the reliability requirements and capabilities.

5. Address modules sharing an environment and resources together. Non-IMA modules sharing the same environment should also be addressed. If there are only IMA modules sharing the environment and the configuration of modules is fixed, then the combined partial environmental qualification can be achieved prior to full integration on an aircraft.
- e. Obtain IMA platform acceptance using the module acceptance approach described in [Chapter 4](#). The module acceptance data described in [Chapter 4](#) and [Chapter 5](#) should be developed and submitted or made available. All IMA platform requirements should be validated and verified. Traceability between the requirements, implementation, and verification activities should be developed and maintained.

3.1.2 Hosted Application Development Process

Development of hosted applications follow the same development processes as used in non-IMA systems, but should address these objectives:

- a. Identify IMA platform resources to be used (part of interface definition).
- b. Quantify required IMA platform resources (part of interface definition).
- c. Map hosted application safety assessment to IMA platform safety assessment and capabilities (i.e., Preliminary System Safety Assessment (PSSA), Functional Hazard Assessment (FHA), and Common Cause Analysis (CCA)).
- d. Define HM/FM requirements for the hosted application and define interactions with IMA platform HM/FM functions.
- e. Identify dedicated resources peripheral to the IMA platform (for example, application-specific hardware).
- f. Specify environmental qualification level for dedicated resources.
- g. Integrate application onto the platform and perform software/platform integration testing.
- h. Assess human factors requirements against IMA platform performance.

3.1.3 IMA System Development Process

The IMA system development process should address the following objectives:

- a. Identify aircraft functions, including functional, performance, safety, availability, and integrity requirements.
- b. Allocate IMA platform resources to the aircraft functions considering the aircraft-level FHA, resource requirements (interface specifications), safety capabilities of the IMA platform, and MMEL considerations. Determine what hosted applications or aircraft functions need isolation and/or protection from other hosted applications and functions and other protection mechanisms or safety features needed.
- c. Develop the IMA system architecture, addressing the following aspects:
 1. Develop IMA System Certification Plan (Section [4.4.3](#)) based on aircraft requirements, hosted applications, and the IMA system certification approach.
 2. Determine the quantity, quality, and type of IMA platform modules and resources needed to meet all application requirements, including functional, performance, safety, availability, integrity, and redundancy requirements.

3. Determine any aircraft function requirements driven by the capabilities of the IMA platform modules, for example:
 1. Availability requirements beyond those available from a single IMA module or platform which could drive the application to be hosted on multiple modules and/or platforms.
 2. Applications using multiple modules should determine application redundancy management requirements.
 3. Integrity requirements beyond those available from an IMA module or platform that could drive the hosting of multiple instances of the application and/or data that may be compared to achieve the necessary integrity.
4. Perform a Preliminary System Safety Assessment (PSSA) for each hosted application using the IMA platform safety requirements.
5. Evaluate the aircraft effects from the combination of platform, hosted applications, and shared resource failures.
6. Identify changes required to the allocation of IMA platform resources to correct any issues identified from the individual and combined PSSA activities.
- d. Implement the IMA system, including the following activities:
 1. Develop the applications and perform partial verification.
 2. Integrate all applications onto the platform and perform IMA system validation and verification activities.
 3. Develop initial IMA system failure analysis using IMA platform top-level events as basic events for the hosted application failure analyses.
 4. Evaluate the combination of IMA platform component failures affecting hosted applications which could lead to aircraft level effects, and adjust the allocation and/or application implementation as necessary. It should be noted that IMA platform component failures should have a unique top level event.
 5. Perform aircraft ground and flight testing to validate assumptions in the SSA, requirements and environmental definitions.
- e. Integrate, validate, verify, and obtain acceptance of the IMA system (off aircraft). Specific configuration of applications in the IMA system should be shown to meet their requirements (including performance, redundancy management, and IMA platform interface requirements). Analyses for each hosted application should be developed to show it complies with its FHA. Additionally, the hosted application analyses should be combined into an IMA system hardware quantitative analysis that shows that the combined events satisfy the aircraft level safety and reliability requirements.
- f. Where appropriate, integrate, validate, verify, and obtain acceptance of the IMA system installed on the aircraft.

The integration activities and the relationship to the acceptance tasks are shown in TABLE 3. The activities are general and can be scaled appropriately to align with the size of development. The acceptance process of IMA is described by six tasks (also refer to [Chapter 4](#)).

TABLE 3 : RELATIONSHIP AMONG INTEGRATION ACTIVITIES AND ACCEPTANCE TASKS

Integration Activity	Acceptance Tasks	
Integrate components and/or modules to form a platform	Task 1	Module and/or platform acceptance
Integrate a single application with the platform	Task 2	Application acceptance (software and/or hardware)
Integrate multiple applications with the platform(s) and one another	Task 3	IMA system acceptance
Integrate IMA system with aircraft and its systems	Task 4	Aircraft integration
Identify changes and their impacts, and need for re-verification	Task 5 ¹	Change
Identify and use IMA components on other IMA systems and installations	Task 6 ¹	Reuse

3.2

IMA SYSTEM RESOURCE ALLOCATION ACTIVITIES

The aircraft functional and performance requirements influence the allocation of IMA hosted functions. In addition to the specific aircraft functions, several issues may need to be addressed including provisions for computing resource availability, application-specific I/O resources, network bandwidth, and meeting the safety, integrity and reliability requirements. Platform services performed on behalf of a specific application should occur within the resource constraints of that application.

- a. Processing throughput for an IMA system should address the required execution time of the applications, the context switch times, the platform overhead, and the overall processing requirements. For example, any service performed on behalf of an application should be executed in its allocated time and not during the time allocated to another application. This would include services provided by the platform, such as resource management, health monitoring, fault management, partitioning enforcement, and other active protection means.
- b. The amount of computing resource and network bandwidth should address overhead (including partitioning enforcement, processing, and data bus jitter and traffic) associated with context switching, data scheduling, and other constraints.
- c. Satisfying the safety requirements for the aircraft, the IMA platform and hosted applications define the integrity and availability requirements of the functions allocated and the configuration of the IMA system. The IMA system architecture should be capable of supporting the highest level of integrity and availability for the aircraft functions and hosted applications.

¹ Integration aspects of Tasks 5 and 6 are performed at the aircraft or system-level and may include Task 1 - 4 activities

3.3 AIRCRAFT SAFETY AND SECURITY

Safety requirements should be addressed in the IMA system requirements. These requirements drive the system configuration and the allocation of functions and hosted applications to IMA resources, and establish the independence, availability, and integrity requirements for those hosted applications contributing to the aircraft functions. Additional safety considerations are addressed in the safety assessment process described in Section 5.1. Requirements derived from the safety assessment and the security analysis are distinguished from functional requirements.

Security requirements should be addressed at the aircraft-level safety assessment (see Section 5.1.5.8). IMA mechanisms may be used to address security concerns.

3.4 DEVELOPMENT ASSURANCE AND TOOL ASSURANCE

The IMA system and components should be designed and developed to the highest assurance levels needed to support the safety, integrity, and availability requirements of the aircraft functions and hosted applications intended for the IMA system as determined by the IMA system safety assessment. This may be difficult to determine for a “general purpose” IMA platform intended to be used on multiple aircraft types with unknown sets of aircraft functions and hosted applications. Therefore, to reduce the risk for a specific aircraft type and configuration of hosted applications, the IMA system developer may want to develop their system to a specific assurance level.

Both the IMA platform user’s guide for hosted applications and the interface specification, describing the IMA platforms interface requirements with external systems, should be complete and unambiguous. Since the aircraft functions and hosted applications on an IMA system may originate from multiple suppliers, the IMA platform user’s guide should be available to all application suppliers.

Tools provided by the IMA platform developer and used to develop and/or verify compliance to the interface specifications may need to be qualified. For example, certain tools used to generate or verify configuration tables or verify robust partitioning may need to be qualified by the IMA platform developer and/or system integrator, but should not need to be qualified by the application developers.

3.5 PARTITIONING AND RESOURCE MANAGEMENT ACTIVITIES

A primary goal of IMA is to integrate and host multiple aircraft functions and applications on one or more computing platform(s) while ensuring that any one aircraft function is not able to affect another function in an undefined or unacceptable way. Furthermore, failures of the mechanisms that enforce and maintain this independence should be detectable and mitigated to ensure the required levels of safety and integrity.

The criteria for “unacceptable” can be determined only through safety assessment. This requires analyzable mechanisms for behavioral containment and isolation between the aircraft functions and hosted applications. One such technique is commonly known as partitioning and is defined in ED-12/DO-178 (Ref. [2]) as follows:

“Partitioning is a technique for providing isolation between functionally independent software components to contain and/or isolate faults and potentially reduce the effort of the software verification process.”

Partitioning within an IMA system should allow independent applications to share resources without any unintended interactions. While the concept of partitioning is used in many commercial computer-based systems, the usual implementations would not provide the degree of protection needed in safety-related systems. Thus, “robust partitioning” is defined to distinguish partitioning for IMA systems.

The characteristics of robust partitioning are:

- a. The partitioning services should provide adequate separation and isolation of the aircraft functions and hosted applications sharing platform resources. Partitioning services are the services provided by the platform that define and maintain the independence and separation between partitions. These services ensure that the behavior of functions or applications within a partition cannot unacceptably affect the behavior of functions or applications in any other partition. These services should prevent any adverse effects on the aircraft of the simultaneous undetected corruption of all the functions and applications' partitions sharing the affected resources.
- b. The ability to determine in real-time, with an appropriate level of confidence, that the partitioning services are performing as specified consistent with the defined level of safety.
- c. Partitioning services should not rely on any required behavior of any aircraft function or hosted application. This implies that all protection mechanisms required to establish and maintain partitioning are provided by the IMA platform.

NOTE: *Other protection mechanisms (e.g., higher level fault detection and resolution, application-specific safety monitoring, or error detection and correction capabilities) may be IMA hosted as part of a hosted application, or may be hosted on other aircraft systems.*

These characteristics should be compatible with the required reliability and integrity levels identified in the aircraft safety assessment. The analysis of these characteristics may be done at the aircraft platform or IMA system integration level.

Dedicated resources are resources allocated by design exclusively to a given application or application function within a partition. An important characteristic of dedicated resources is that the failure of a dedicated resource only affects the application or partition to which it is dedicated. Typical examples of dedicated resources are dedicated memory, specialized hardware, and software logical resources, such as dedicated buffers.

Shared resources are resources that are used by multiple partitions. Therefore, independence or isolation of the application functions in one partition from those in another is needed (e.g., protection of one function from failures or anomalous behavior of another). An important characteristic of a shared resource is that a failure of that resource affects all the applications and partitions using that resource. Different architectures may designate the same resource as either dedicated or shared.

Partitioning analysis and design should conform to two principles. First, the dedicated resources assigned to one partition can never be affected by or affect the operation of an application or application function in any other partition. Second, the use of shared resources by one partition cannot unacceptably affect, as shown by a safety assessment, the operation of an application or application function in any other partition sharing that resource.

An IMA execution environment should ensure that all hosted applications operate in a manner equivalent to operation in a federated system architecture. Each hosted application should have total independence with respect to any other application being co-hosted with it. The platform should guarantee the allocation of necessary resources to all hosted applications, regardless of the operation or errors in other hosted applications.

The design of an IMA system and its associated architecture is determined by many factors (e.g., system issues requiring redundancy, common cause failure considerations, and so on). This section addresses only those aspects of design and architecture related to partitioning. The inability of the IMA system to guarantee partitioning in the presence of hardware failures or software errors may require specific flight crew action or maintenance action. Such actions can be determined only by considering the actual aircraft functions and hosted applications and are beyond the scope of this document. Nevertheless, any mechanism potentially involved

in failures of the IMA system to guarantee partitioning and the effects of those failures on aircraft functions and hosted applications should be identified and provided to the system development and safety assessment processes. In some cases, an IMA system developer may provide specific safety properties independent of any application, such as fail-passive or fail-operational architectures, providing additional constraints and protection that could simplify the overall system safety assessment, verification of robust partitioning and other protection features, and demonstration of compliance.

3.5.1 Design for Robust Partitioning

The design for partitioning in an IMA platform is an iterative process. The proposed design should be analyzed to ensure that the criteria identified in Section 3.5 have been satisfied. If deficiencies are detected, then the architecture should be revised until the criteria can be shown to be satisfied. While the approaches used will be dependent on the specific implementation, some generic guidelines are provided within this section.

All of the dedicated and shared resources should be identified. All propagation paths to those resources where any unintended interaction may occur should be determined. These propagation paths may be the result of hardware failures, hardware design errors, or software design errors. They may also be the result of normal execution. Once propagation paths have been determined, containment boundaries should be established and validated to prevent undesirable interaction between partitions through these propagation paths. Robust partitioning services should provide the protection of the dedicated and shared resources. Failure of these partition services may lead to the generation of unintended failure propagation paths. Example propagation paths are:

- a. A faulty partition allows an application to write to a memory location to which another partition assumes it has exclusive access.
- b. A common shared communication channel is caused by a faulty partition to deny service to another partition.
- c. Processor execution time is denied by one partition to another.
- d. A shared flash memory file system is corrupted by a faulty partition.
- e. Cache memory on a CPU is not flushed on context switch to a new partition.
- f. A faulty partition causes an event to occur at a time, or in a sequence, which differs from what is expected by another partition.

In some cases, more than one containment boundary may be needed for a given propagation path. In others, a single containment boundary may protect more than one propagation path.

The allowed interactions and interfaces between partitioned functions as supplied by the partitioning services should be specified completely. A complete interface definition will facilitate the analysis of propagation paths. Strict adherence and enforcement of the interface specifications between partitions should apply to IMA platforms to avoid unintended behavior.

Partitioning mechanisms should be consistent with IMA system integrity, availability and reliability requirements. IMA system design should satisfy the highest levels of these requirements while recognizing the implications of the combination of hosted applications and aircraft functions.

3.5.2 Partitioning Analysis

A partitioning analysis should demonstrate that no application or sub-function in a partition could affect the behavior of a sub-function or application in any other partition in an adverse manner. All propagation paths between partitions should be identified. The effects of each propagation path should be documented. The mitigation of unacceptable interactions at containment boundaries should be identified. Fault tree analysis, formal methods, and other techniques may be employed. The partitioning analysis should be addressed in the context of the aircraft system safety assessment to include any probability of multiple failure requirements. Alternatively, the analysis may make assumptions about these requirements. Partitioning analysis should address plans, procedures and requirements for how the partitioning and other protection schemes will be verified and validated.

The following sections describe what a partitioning analysis should contain.

3.5.2.1 Top-level Partitioning Property

The requirements allocated to the top-level partitioning properties should be based on the aircraft safety assessment. This part of the analysis should state the main property or properties that are needed to establish that robust partitioning has been achieved and maintained. These may be stated as either a positive property or a property that prevents an undesirable effect.

3.5.2.2 Decomposition of Partitioning Properties

This part of the analysis is a decomposition of the top-level properties to lower level properties that must be achieved to satisfy the top-level properties. The lower level properties may be further decomposed until the lowest level property is reached that can be shown to be fully satisfied by one or more design features of the platform.

NOTE: *An example of decomposition of the top-level property (no application affects the behavior of an application in any other partition in an adverse manner) could be as follows: No application function in one partition can*

- a. access memory of any another partition in an adverse manner,*
- b. affect the timing of any another partition in an adverse manner, or*
- c. adversely affect the resources used by any other partition.*

The above three properties satisfy the top-level property. Other means could be used. In addition, the memory protection can be considered a lowest-level property if there is a design feature that uses a Memory Management Unit (MMU) and a set of configuration files for setting up the MMU that guarantee that the memory allocated to a partitioned function is only accessed by that function.

3.5.2.3 Life Cycle Data

For every design feature related to the lowest level property traceability to life cycle data and associated verification data should be provided. If design features and associated life cycle data are part of another verification process, then only traceability to the design feature and associated requirement is needed.

3.5.2.4 Partitioning Vulnerability Assessment

For each partitioning property a vulnerability assessment should be conducted which examines all potential actions of external system and human interfaces as well as the effects of hardware failures when required by the safety assessment or hazard analysis.

NOTE: *Because the configuration files may be outside of the control of the platform designer, one vulnerability would be that an error in the configuration files could result, for example, in overlapping memory regions between partitioned functions. This would require a mitigation that would result in a requirement for the IMA system integrator to provide verification data that there are no overlapping memory regions.*

3.5.2.5 Potential Sources of Error

The potential sources of errors that should be included in a partitioning analysis are unique to each system analyzed. While there is no comprehensive list of issues that should be addressed, the following list represents some potential sources of design errors that could impact the partitioning analysis.

- a. Interrupts and interrupt inhibits (software and hardware).
- b. Loops (for example, infinite loops or indirect non-terminating call loops).
- c. Real-time correspondence (for example, frame overrun, interference with real-time clock, counter/timer corruption, pipeline and caching, deterministic scheduling).
- d. Control flow (for example, incorrect branching into a partitioned or protected area, corruption of a jump table, corruption of the processor sequence control, corruption of return addresses, unrecoverable hardware state corruption (for example, mask and halt)).
- e. Memory, input, and/or output contention.
- f. Sharing of data flags.
- g. Software traps (for example, divide by zero, unimplemented instruction, specific software interrupt instructions, unrecognized instruction, and recursion termination).
- h. Hold-up commands (e.g., performance hedges).
- i. Loss of input or output data.
- j. Corruption of input or output data.
- k. Corruption of internal data (for example, direct or indirect memory writes, table overrun, incorrect linking, calculations involving time, corrupted cache memory).
- l. Delayed data.
- m. Program overlays.
- n. Buffer sequence.
- o. External device interaction (for example, loss of data, delayed data, incorrect data, protocol halts).

There is an iterative process between the partitioning analysis and the different design steps for the system. The analysis is complete when all partitioning properties have been satisfied and verified, and all identified vulnerabilities have been mitigated and have associated verification data.

3.6 HEALTH MONITORING AND FAULT MANAGEMENT

Health monitoring and fault management (HM/FM) functions should be provided with recognition that the IMA platform may support a collection of diverse aircraft functions. The IMA platform should provide basic health monitoring and fault management capabilities for IMA platform modules. These capabilities should be independent of the hosted applications.

- a. Health monitoring should address both operational and maintenance concerns.
- b. Strategies for the following example issues may be developed:
- c. Identification of components and aspects to be monitored.
- d. Health determination of each application.
- e. Determination of the health of the IMA system as a whole.
- f. Response to each type of failure or anomalous behavior.
- g. Flight crew annunciation and messaging.
- h. Control of maintenance actions and reporting.
- i. Redundancy management.
- j. Single event upsets.

3.6.1 Components and Aspects to be Monitored

Health monitoring is that part of the IMA platform responsible for detecting, isolating, containing, and reporting failures that could adversely effect applications using the IMA platform resources, or the resources themselves. This function should detect faults in the shared resources used by the hosted applications. However, some resources dedicated to applications may also be monitored by the IMA platform. For example, memory dedicated to an application partition may be monitored by the IMA platform rather than the application.

Faults in the shared resources could adversely impact all of the applications using those resources. The system architecture should be designed to promote the detection of faults at the lowest possible architectural level, ideally at the component level. This will reduce the potential for ambiguity. This can be done with self-monitoring, platform monitoring, or a combination thereof. Faults are generally detected by their symptoms potentially leaving some ambiguity about which component actually failed. The smaller the ambiguity group is for a fault, the easier it is to define policies for resolving it safely at the IMA system level. Sound architectural choices can frequently reduce the ambiguity to a single isolation region. This is an important consideration when determining responsibility for the detection of faults in the IMA platform. This is even more important with respect to the IMA platform robust partitioning services. Failures of these services can directly affect the ability to maintain that separation and independence. If detection of failures in these services requires hosted application participation, it becomes nearly impossible to isolate these failures to the partitioning mechanisms. For this reason, the health monitoring of the partitioning services should be performed by the IMA platform.

The IMA platform should monitor the health of its services and interfaces.

3.6.2 Health Determination of Each Application

The application supplier should identify potential failure modes of the application. In particular, any application failure modes which require action by the platform should be identified. For example, this may include partition restart, shutdown, or other platform-specific actions.

HM facilities to record, monitor, and manage failures affecting the platform should be provided. This includes the facilities for applications to record and announce maintenance conditions and failures. The application supplier may also provide application-level health monitoring capability specific to their application. This information, independent of the approach used, should be identified and integrated into the overall IMA system health monitoring strategy.

3.6.3 Health Determination of the IMA System as a Whole

The health determination for the IMA system should address IMA platform-level failure conditions and the potential for partitioning failures and application failure modes that are not addressed within the application. An integrated HM strategy should be defined and documented. This strategy should be coordinated with the IMA system safety assessment.

The integrated HM strategy should also define:

- a. The means to identify systemic failures and report the conditions to the hosted applications and other platform services.
- b. Rules governing partition restart and shutdown.
- c. Rules for degraded operations, if applicable.
- d. Guidelines to support the MMEL.
- e. IMA system health status reporting, content and frequency.

3.6.4 Response to Each Type of Failure

In addition to detection and isolation of faults, the ability to report and contain faults is needed. Reporting refers to internal logging, indicating to hosted applications and platform services, indicating to the flight crew, and indicating to the maintenance crew the existence of the fault or failure. Fault containment refers to the response taken to keep a failure from affecting anything external to the containment area. This subsection focuses on the containment of faults. Reporting of faults will be covered later.

All detectable faults within an IMA platform should have a response determined. Failures within an application which do not affect IMA platform resources are the responsibility of the application. The application should provide the correct response to these failures including providing health status of the application. Any attempts by the application to violate platform partitioning services should be monitored and detected by the IMA platform and appropriate action taken by the platform. The IMA platform response to these faults may be configurable.

The ability to contain faults to an isolation zone greatly facilitates the integration of hosted applications on an IMA platform. This assumes that partitions are all fault containment zones and that platform services are also fault containment zones.

3.6.5 Flight Crew Annunciation and Messaging

In general, IMA systems should be treated no differently than traditional federated systems with respect to flight crew annunciation and messaging. Within the framework of flight alerts and maintenance message systems, the focus for the flight crew is function availability. Although maintenance systems provide information on aircraft maintenance aspects and aircraft dispatch (MMEL) with a degree of correlation to improve the efficiency and efficacy of the maintenance reports, it is the responsibility of the flight alert/warning systems to provide the primary source of information to

assist the flight crew in their decision-making process. This is achieved through systems monitoring with respect to a centralized alerting and warning function, including messages and aural alerts. The flight alert/warning systems, in conjunction with items such as the MMEL, provide the flight crew and maintenance personnel with the information essential to determine the dispatch status of the aircraft.

However, given the status of the IMA system as a whole, and the individual components and hosted applications, certain types of status may need to be annunciated to the crew with respect to system degradation. Flight alert/warning systems may selectively inhibit some warnings to reduce the effects from cascading failures; typically, however, all failures and system degradation and reconfiguration actions are reported.

The appropriate faults, failures, and errors to be logged, reported, or displayed to the flight crew should be determined. These notifications may be restricted to failures from which the system cannot recover (e.g., a hardware failure), failures resulting in degraded performance or a degraded mode, failures that can reduce the safety margins of the aircraft or system (e.g., loss of redundancy or integrity), potential unsafe conditions or notification of loss of functionality. The IMA system fault messages displayed to the flight crew should be analyzed to ensure they are unambiguous and have appropriate priorities assigned to them. Displayed messages and aural alerts should be provided in a clear and prioritized manner to the crew.

3.6.6 Control of Maintenance Actions and Reporting

Maintenance personnel need to be able to analyze health monitoring data, identify system components that have failed or exhibited anomalous behavior, and determine the most appropriate maintenance actions (e.g., defer, test, repair, replace, etc.).

The level of integration, interdependency, and complexity of an IMA system may result in unique and more numerous potential failure modes (for example, all functions affected by the loss of a single shared resource) than federated system architectures. Power supply transients, cascading failures, or failures affecting shared resources can result in multiple functions and hosted applications failing simultaneously, and multiple failures, warning, alert or caution messages being logged and annunciated to the flight crew. Although many of the features described can be attributed to traditional federated systems, particular emphasis should be placed on fault correlation with respect to Built-In Test Equipment (BITE) fault reporting within IMA systems to reduce fault propagation and distinguish between platform-related versus application-related issues.

In general, the maintainability of airborne systems should also be considered in conjunction with the safety, reliability, and supportability objectives for the aircraft. Analysis of these interrelated objectives may also identify unique aspects of preventative maintenance actions.

3.6.7 Redundancy Management

Redundancy is used to improve both the reliability and availability of an aircraft function. An IMA platform and/or system may include mechanisms to support management of redundancy for hosted applications. Redundancy management policies should be established for the IMA system. Where possible, redundancy should be managed independent of the hosted applications to support separate analysis and development of applications and platforms.

Some hosted applications may require application-specific redundancy management. For these the IMA system should provide correct and consistent information about the health of the platform resources, as well as the health of other modules with which the application interacts.