Avionics standards, software and IMA
Brigitte Bauer

To cite this version:
Brigitte Bauer. Avionics standards, software and IMA. Conference ERTS’06, Jan 2006, Toulouse, France. hal-02270489

HAL Id: hal-02270489
https://hal.archives-ouvertes.fr/hal-02270489
Submitted on 25 Aug 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Avionics Standards, Software and IMA

Brigitte Bauer
THALES AVIONICS
Airborne Computer Solutions
18 Avenue du maréchal Juin, 92362 Meudon-La-Forêt
brigitte.bauer@fr.thalesgroup.com

Abstract

The paper covers the definition of Integrated Modular Avionics (IMA), the associated avionics standards and the impact on the Avionics Software. ARINC and RTCA/EUROCAE committees, in which all Avionic stakeholders are involved, developed these standards. 2005 is a key year for standardization: ARINC653 part1 supplement2 and part3 are ready for publishing, RTCA-SC200 / EUROCAE-WG60 is under ballot.

The concepts of IMA, the new architecture in Avionics, were defined in the late Eighties and published for the first time in the ARINC651 standard in 1991. The IMA concepts were firstly applied on Boeing 777, extended and used on Airbus A380 and now selected for the future Boeing 787. These concepts divide the avionic embedded domain into Platform (Hardware+Core Software) and Applications instead of Hardware and Software. Several applications of different criticality levels could reside on the same platform.

The consequences were the development of new standards and guidelines for supporting these concepts, e.g.:

- ARINC653 defines the API and the behavior of the Core Software services.
- DO-255/ED-96 contains the description of an Avionic Computing Resource (a platform separated from its hosted applications).
- DO-248B/ED-94B clarifies DO-178B/ED-12B and defines concepts like robust partitioning.
- SC200/WG60 (future ED-124) contains the IMA Development Guidance and Certification.
- SC205/WG71 has started. It reviews and extends DO-178B/ED-12B and DO-248B/ED-94B in regard of new technologies.

The paper describes the objectives and the results of these standardization committees. It focuses on ARINC653 and ED-124 standards and presents shortly the associated standards.

Keywords: ARINC653, IMA, Avionics Software, Avionics Standards, Certification.

1. Introduction

This paper tries to describe in an objective way the IMA standards. Therefore extracts of them are given in extenso to respect their spirit.

1.1 Standardization committees

RTCA

Organized in 1935 as the Radio Technical Commission for Aeronautics, RTCA [11] today includes roughly 250 government, industry and academic organizations from the United States and around the world. Member organizations represent all facets of the aviation community, including government organizations, airlines, airspace user and airport associations, labor unions, plus aviation service and equipment supplier.

RTCA, Incorporated is a not-for-profit organization formed to advance the art and science of aviation and aviation electronic systems for the benefit of the public. The organization functions as a Federal Advisory Committee and develops consensus-based recommendations on contemporary aviation issues. The organizations recommendations are often used as the basis for government and private sector decisions as well as the foundation for many Federal Aviation Administration (FAA) Technical Standard Orders (TSO).

EUROCAE

The European Organisation for Civil Aviation Equipment, EUROCAE, [12] was formed in 1963. At that time, there was no regular forum in Europe where administrations, airlines and industry could meet to discuss technical problems. EUROCAE was created to fill this gap.

The main European administrations, aircraft manufacturers, equipment manufacturers and service providers are members of EUROCAE, and they actively participate in the Working Groups.

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 worldwide levels.

EUROCAE documents are considered by Joint Aviation Authorities (JAA) as means of compliance to Joint Technical Standard Orders (JTSO) and other regulatory documents.

Joint RTCA and EUROCAE committees have developed their guidance documents related to Software and IMA.

ARINC/AEEC
Arinc/AEEC Aeronautical Radio Inc / Airlines Electronic Engineering Committee (ARINC/AEEC) [10] is a group of interest, funded by airlines. It is in charge of the definition of Aeronautical standards that ensure interchangeability and interoperability.

AEEC leads the development of avionics architectures and form, fit, function, and interface technical standards. The avionics installed in more than 10,000 aircraft around the world are based on these standards, known as ARINC Standards. The use of ARINC standards results in substantial benefits to airlines by allowing avionics interchangeability and commonality and reducing avionics cost by promoting competition.

Furthermore, for new aircraft and avionics installations, ARINC standards provide the starting point for avionics development and allow aircraft manufacturers to pre-wire aircraft, thus ensuring that cost-effective avionics for air transport aircraft are ready when needed.

AEEC is organized in committees, sub-committees and working groups.

1.2 IMA

Overview
ARINC651 “Design Guidance for IMA [1] introduces and defines the IMA concepts. It’s first publishing occurred in 1991: “IMA is formed around the concept of powerful computer(s) with an Operating System that allows independent processing of application software, while maintaining robust partitioning between software modules for even the most critical functions....”

A revised version with a few updates was published in 1997.

Objectives
IMA should benefit to the Airlines, Airframe Manufacturers, Equipment Manufacturers and Certification Authorities thanks to:
- Increased operational performance
- Reduced unscheduled maintenance
- Reduced spares requirement

- Simplified service life changes and additions to avionics
- Reducing development, certification and production costs
- Increasing market volume, longer production runs
- Independent component qualification of the application

This conducts to a need for modularity and interchangeability, portability of the applications, fault management. Therefore Standardization is a necessity.
Multimodule cabinets: Core, Gateway, I/O Modules and Multiapplication modules provide standardized resources.

A Standardised interface layer between Software and Hardware and an Operating system provide portability of the applications.

Software Architecture
Integration of many functions is encouraged by IMA. In ARINC651 the Software Architecture contains:

- The Application Software, which performs the avionics functions.
- The Core Software, which provides a standard and common environment in which applications software executes.
- APEX (APplication EXecutive) and COEX (COre EXecutive) interfaces.
2. ARINC653

2.1 APEX Working Group

APEX (APplication EXecutive Interface) is one of the AEEC working groups. The APEX working group started as an ad hoc for ARINC 651 (definition of IMA). It is in charge of the specification of the Avionics Application Software Standard Interface. The document that hold this specification has received the number “653”.


The scope of the Working Group is defined as:
- Assure the products of this working group meet the needs of safety critical, real-time applications.
- Ensure application portability.
- Ensure the means of communications between partitions is independent of physical transport mechanism.
- Maintain upward compatibility with the previous APEX.
- Ensure sufficient APEX services are provided to meet the needs of the using community.
- Assure consistency with the related activities.

The purpose of the Working Group is:

- To specify the baseline operating environment for application software used within IMA and traditional ARINC 700-series avionics.
- To define a general interface (APEX) between the Application Software and the O/S of an Avionics Computer Resource (i.e. IMA).
- To support one or more avionics applications of different levels of criticality per core module, independent execution of those applications in predetermined time slots (partitioning).

The result of the Working Group:
- Will improve the overall cost of avionics software development, modification and maintenance.
- Is a key part of Integrated Modular Avionics strategy.
- Allows independent development, modification, and deployment of software applications.
- Is supported by several development programs, including A380 and B787.

When the Working Group restarted in 2002, improvement of the existing standard and extensions were planned in 2 steps:
- Step 1: for Improvement, clarification, correction, required services only.
- Step 2: new services definition, which are optional services, and conformity test plan for the required services.

Now the status of the working Group is:
- Supplement 1 (October 16, 2003) and Part1 supplement 2 (adopted in October 2005) improve Application Software portability and concern only required services. The first implementation was made in A380.
- Part 2 - Extended Services adoption is planned for mid 2006; certain services like logbooks are also implemented in A380.
- Part 3 - Conformity Test Plan was adopted October 2005. The first draft came from the VICTORIA FP5 European Project.

2006 work program consists in:
- Completion of Part 2 (653 extensions) development of Part 1 Supplement 3.
- Starting Part3 Supplement 1 and Part2 Supplement1.
- Coordination with SC205/WG71.

2.2 ARINC 653 Specification

In the ARINC653 document, its primary objective is to define a general-purpose Application/Executive interface (APEX) between the Operating System (O/S) of an avionics computer resource and the application software.

Included within this specification, are the interface requirements between the application software and the O/S and the list of services, which allow the
application software to control the scheduling, communication, and status information of its internal processing elements. This Specification defines the data exchanged statically (via configuration) or dynamically (via services) as well as the behavior of services provided by the O/S and used by the application. The bulk of this document describes the runtime environment for embedded avionics software. This list of services identifies the minimum functionality provided to the application software, and is therefore the industry standard interface. It is intended for this interface to be as generic as possible, since an interface with too much complexity or too many system-specific features is normally not accepted over a variety of systems. The ARINC653 document is intended for use in a partitioned environment.

The document describes also the software that resides on the hardware platform. It consists of:
- ARINC 653 Application partitions that are subject to robust space and time partitioning and are restricted to using only ARINC 653 calls to interface to the system.
- An O/S kernel that provides the API and behaviors defined within this specification, and supports a standard and common environment in which application software runs. This may include hardware interfaces such as device drivers and built-in-test functions.
- System partitions, partitions that require interfaces outside the scope of APEX services, yet constrained by robust spatial and temporal partitioning. These partitions may perform functions such as managing communication from hardware devices or fault management schemes. System partitions are optional and are specific to the core module implementation.

### 2.3 ARINC653 characteristics

#### Partition management
- A partition is a program unit of the application designed to satisfy the partitioning constraints. It contains multiple processes.
- O/S provides robust partitioning for time and space.
- Partitions of different criticality levels reside in the same core module.
- Partition scheduling is predetermined, repetitive, with fixed periodicity (no priority for a partition).
- Partition memory is in predetermined areas (no memory allocation services in APEX).

#### Process management
- A process is a software task, which executes concurrently with other processes of the same partition.
- Process scheduling is Priority Preemptive.
- Periodic and aperiodic processes are scheduled.
- Each process has a priority.
- Common O/S services like Suspend/Resume, Start/Stop, Lock preemption are provided.

#### Inter-partition communication
- The communication between partitions on the same or between different modules is generic and is not depending on the underlying media.
- A message is sent from a single source to one or more destinations. Sender and receivers do not know each other.
- Ports/channels/messages are the means for supporting queuing mode (no lost message) and sampling mode.

#### Intra-partition communication
- Synchronization between processes inside the same partition is done via semaphores and events.
- Communication between processes inside the same partition is done via buffers and blackboards.

#### Time management
The Time is managed by services and means controlling periodic and aperiodic processes and also controlling the duration of potential blocking services like timeouts, deadlines, timed and periodic waits, delayed start, replenishment.

#### Fault management
- The Health monitoring is responsible for monitoring and reporting faults and failures.
- The faults are managed at application, partition and module levels
- Configuration tables contain recovery action to respond to pre-defined faults of the partitions and the module.
- The error handler process, a specific process provided by the application, manages the application level errors.
2.4 Conclusion

ARINC Specification 653 is now organized into three parts, as follows:
Part 1 constitutes the basic requirements and guidance for ARINC 653, Required Services.
Part 2 constitutes extended requirements and guidance Extended Services.
Part 3 contains a compliance test procedure used to establish compliance to ARINC 653 Part 1.

The major improvements in the APEX Required Services are:
- addition of System Partition.
- The clarifications of partitioning requirements, Interpartition Communication and Health Monitoring.
- The representation of Time with 64-bit integer.
- The addition of a second Ada programming interface (Ada 95).
- The representation of configuration tables in XML language.

The APEX optional extensions would contain:
- The definition of a common Sampling Data format (under discussion).
- The creation of File Management Services.
- The creation of Non Volatile memory Services.
- The extension of partitions schedules and sampling ports services. The APEX Conformity Test Specification describes test assertions and responses necessary to demonstrate conformity to the software interface defined in Part 1, Required Services.

Real-Time Operating System (RTOS) software that claims compliance to this test plan would support all ARINC 653 interface services and data structures. The application base for ARINC 653 has increased substantially, and there is a recognized need to refine and augment the specification. In addition various organizations, including the Open Group [13] and military systems have shown an interest in ARINC 653 as the basis for a “secure” operating system.
Therefore the Working Group will continue to meet on a regular basis to review and to adapt ARINC653.

3. Future ED-124

This document, jointly prepared by EUROCAE Working Group 60 and RTCA Special Committee 200, [3] was officially approved by RTCA in November 2005, the approval from EUROCAE should occur before the end of 2005 and the document should become ED-124. The RTCA reference is not yet known.

3.1 Objectives

The SC200/WG60 document named IMA Development Guidance and Certification Considerations 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.

In the document, IMA system concepts are described, including the platform and modules, and their relationships to the hosted applications and avionics functions used in an aircraft installation. The six tasks defining the incremental acceptance of IMA systems in the certification process are presented.

IMA is defined 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.

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.

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.

An important aspect of the certification process for IMA systems is to obtain incremental acceptance of and certification credit for IMA platforms, modules, and/or hosted applications, cumulating in IMA system installation approval on an aircraft product, and resulting in issuance of the product certificate.

3.2 Terminology

A set of terms is defined in the document to assure understanding among all parties involved.

**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.

**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).

The figure below shows the relationships between many of these terms.

### 3.3 Key characteristics

Platforms and hosted applications key characteristics are:
- Platform resources are shared by multiple applications.
- An IMA platform provides robust partitioning of shared resources.
- An IMA platform only allows hosted applications to interact with the platform and other hosted applications through well-defined interfaces.
- Shared IMA platform resources are configurable.
- An application may be designed independent of other applications and obtain incremental acceptance on the IMA platform independently of other applications.
- Applications can be integrated onto a platform without unintended interactions with other hosted applications.
- Applications may be reusable.
- Applications are independently modifiable.

Shared resources :IMA systems may host several applications that share resources.

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

An Application Programming Interface (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. ARINC653 provides an avionics standard for an application programming interface and the related services.

IMA systems manage platform faults, hardware failures, partitioning violations, errors and anomalous behavior of hosted applications, including common mode faults and cascading failures.

The assignment of roles and responsibilities is necessary, and should address the entire IMA system life cycle from conceptual design to retirement.

The identified stakeholders are Certification Authority, Certification Applicant, IMA System Integrator, Platform and Module Suppliers, Application Supplier, Maintenance organization.

3.4 Certification tasks

Typical development processes are divided into six tasks that define the incremental acceptance activities for the certification process for IMA systems:

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

3.5 Integral processes

Integral processes for IMA development are described:

- For Safety Assessment, due to the high level of integration inherent in IMA systems, it is recommended that applicants use ARP 4754/ED-79 [8] and ARP 4761 [9] or acceptable alternatives. The process should consider a minimum list of activities described in the document.
- For System Development Assurance, a section describes key areas of system development assurance, which include software guidance, hardware design assurance, shared design assurance, IMA system configuration management, and environmental qualification.
- The verification process should ensure that the implementations of specified requirements for the IMA system have been met.
- For Configuration management, a section addresses configuration management (CM) of the IMA system life cycle data and life cycle environment.
- A Quality Assurance process in accordance with ARP-4754/ED-79 should be implemented for both the IMA system and the modules within the system.
- The certification liaison process establishes communication and understanding between the applicant and the certification authority throughout the IMA system life cycle to assist in the acceptance and certification process.

3.6 Objectives tables

The objectives are used to plan and ensure that the installed IMA system complies with the guidance of this document. Objectives should be addressed during the planning process of the aircraft certification program (e.g., Aircraft and/or IMA System Certification Plans, Platform Acceptance Plan, Module Acceptance Plan, Hosted Application Acceptance Plans). Since multiple stakeholders will typically be involved in the acceptance and certification efforts, the plans should clearly delineate who is responsible for achieving each objective. Plans should also indicate roles and responsibilities.

3.7 IMA system design examples

This annex contains examples of various system designs. Each example shows how some subset of the IMA characteristics may be implemented in a typical design.

Single LRU platform, distributed IMA platform, distributed complex IMA System, Software designed radio are the examples described in the document.

4 Other Associated Standards

4.1 DO-255/ED-96

DO-255/ED-96, Requirement specifications for an avionics computer resource (ACR) [4], defines the Avionic Computing Resource, identifies the characteristics of a platform separated from its hosted applications and identifies the first features of incremental certification.

This document contains the Minimum Operational Performance Standards (MOPS) for an ACR. The
ACR provides shared computation resources, core software and signal conditioning necessary to interface with a variety of aircraft systems. By complying with the requirements specified by the MOPS, the ACR and the software applications may be independently developed. Together the ACR and installed software applications perform one or more aircraft functions. The aircraft functions can be hosted on multiple ACR platforms that can be provided by multiple vendors.

Compliance with the MOPS ensures a complete specification of the Application Programming Interface (API) provided by the ACR, its capacity and capability, and its performance. Based on this ACR specification, application software may be independently developed.

This document is intended to assist manufacturers, software application developers and installers, in achieving an ACR and installed software applications, which will perform their intended functions satisfactorily under all conditions normally encountered in routine aeronautical operations. This document stipulates that the ACR manufacturer provides a suite of tests and analyses to assure the operation, performance and capacity of the ACR. Although specific test procedures are not included, assurance objectives are established for each ACR attribute.

In developing an ACR as specified by the requirements in these MOPS, the ACR manufacturer is required to specify the development environment required only to develop the ACR. The ACR manufacturer is also obligated to specify the integration and configuration tools and environment necessary such that requirements for the addition or deletion of applications to the ACR will be adhered to at the time they are integrated.

Approval of the ACR ensures that the defined API services, partitioning and health monitoring perform correctly over the full range of environmental conditions encountered in aeronautical operations. The DO-255/ED-96 document contains the ARINC 653 API as an example.

The IMA concepts are introduced in this document that was published in June 2000. DO-255/ED-96 was a major input of the Joint SC200/WG60 committee, which defines guidance for IMA systems.

4.2 DO-248B/ED-94B

The purpose of DO-248B/ED-94B, Final Report for Clarification of DO-178B/ED-12B [5], is to provide clarification to the guidance material in DO-178B/ED-12B.

It contains Errata, Frequently Asked Questions, (FAQ) and Discussion Papers (DP).

- The purpose of Errata is to provide a means of correcting errors in clarify DO-178B/ED-12B.
- The purpose of a FAQ is to provide short and concise responses that are frequently asked by the industry concerning the material of DO-178B/ED-12B. These questions are frequently asked to certification authorities or others who provide interpretation of DO-178B/ED-12B.
- The purpose of a Discussion Paper is to provide clarification for certain sections of DO-178B/ED-12B when clarification requires more than a short answer to a question.

The document contains 12 Errata, 75 FAQ and 15 DP.

One of the Discussion Paper, DP14 concerns partitioning in DO-178B/ED-12. Partitioning rigor and guidelines for robust partitioning mechanisms are described. Robust partitioning definition was reused in ARINC653 and SC200/WG60 documents. The DO-248B/ED-94B document is the third and final report of a Joint RTCA/EUROCAE committee. It was published in October 2001.

4.3 SC205/WG71

Joint RTCA Special Committee 205 and EUROCAE Working Group 71, Software Considerations in Aeronautical Systems, was formed to work on potential updates and improvements to DO-178B/ED-12B [6], and associated guidelines and clarification, DO-248B/ED-94B and DO-278/ED-109 [7]. These documents provide guidance and guidelines for the production of software for airborne, CNS/ATM systems and equipment with a level of confidence in safety, which complies with airworthiness requirements.

DO-178B/ED-12B was released in 1992. The document is seen as reasonably complete in its guidance and remains effective. However it does not efficiently address all current and emerging software technology issues (e.g. the use of a model-based development process, the application of formal mathematical methods etc.). The issues raised by a set of experts, need additional or revised guidance for industry and the certification authorities.

The Objectives of the Joint Committee are:

- To promote safe implementation of aeronautical software.
- To provide clear and consistent ties with the systems and safety processes.
- To address emerging software trends and technologies.
- To implement an approach that can change with the technology.
Revision C will be provided for DO-178B/ED-12B, and DO-248B/ED-94B. The technology independent nature of the DO-178B/ED-12B and the associated DO-248B/ED-94B will be maintained. Supplements to document technology-specific or method-specific guidance and guidelines will be developed if necessary.

The kick-off meeting of the Joint Committee occurred in January 2005. 2 plenary meetings are planned per year. The third one will occur in April 2006. The goal is that in end of 2008, the material produced should be ready for approval by EUROCAE and RTCA. Additional ad hoc meetings are planned depending on the sub group activities. A large number of persons participated to each meeting. Therefore the committee is split in 7 sub groups dedicated to editorial, management and technical aspects. 5 technology or method oriented SG will clarify the use of formal methods, tools, model based design, Object Oriented techniques, other safety issues and CNS/ATM.

5. Conclusion

Thanks to IMA, today Avionics Software contains Application Software dedicated to the avionics functions and Core Software dedicated to the management of the platform resources. The independence between Application Software and Avionic Platform is achieved.

Hardware isolation and standard interfaces are provided with the IMA platform to the function suppliers and ARINC653 is confirmed as the avionics standard Application Programming Interface. Furthermore technical standards and guidance material are available to the IMA stakeholders.

Some IMA concepts were applied in the B777, which flew for the first time in 1995, but the full application was done in A380, which flew for the first time in 2005. During 15 years a succession of programs and studies on the both sides of Atlantic Ocean improved the IMA concepts and demonstrated their maturity.

To definitively confirm this new way of Applications development is to port them, from one IMA platform to another or from one aircraft to another.

All commercial or military aircrafts as well as the Airbus and Boeing pioneers will then follow these standards.

6. References

[10] Arinc.com
[12] EUROCAE.org
[13] opengroup.org

7. Glossary

ACR: Avionic Computer Resource
AEEC: Airlines Electronic Engineering Committee
APEX: Application Executive Interface
API: Application Programming Interface
ARINC: Aeronautical Radio Incorporated
CNS/ATM: Communication Navigation Surveillance / Air Traffic Management
EUROCAE: European Organisation for Civil Aviation Equipment
FAA: Federal Aviation Administration
IMA: Integrated Modular Avionics
JAA: Joint Aviation Authorities
RTCA: Radio Technical Commission for Aeronautics