

# An Adaptive Software Architecture for Future CMS

Reza Ahmadi, Oliver Marquardt, Marc Riedlinger, and Reinhard Reichel  
University of Stuttgart

## ABSTRACT

Aircraft cabin systems, especially cabin management systems (CMS) have to cope with frequent cabin changes during their lifecycle. This includes not only layout rearrangements and technological upgrades during the service, but also extensive CMS customizations and product variations before aircraft delivery. Therefore it is inevitable for the CMS to be highly changeable and offer an easy and agile change process. Today's CMS solutions face this challenge with configurable system architectures. Although such architectures offer a vast change domain, they usually come with time consuming and error prone change processes.

This paper introduces an adaptive avionics software architecture that enables the CMS to cope with cabin changes highly automatically and with minimal human interactions. The adaptation is performed during an on ground organization phase, in which system changes are detected and evaluated by the CMS itself. Consequently the CMS instantiates and adapts the software within hardware modules and establishes the system overall communication.

The concept of adaptive CMS is based on an open service oriented software architecture with built in services, which abstract the communication, redundancy, fault-tolerance and the supervision of the system operation modes during the life-cycle. The software architecture contains also organization services, which enable the CMS to detect the system changes and adapt itself accordingly.

**CITATION:** Ahmadi, R., Marquardt, O., Riedlinger, M., and Reichel, R., "An Adaptive Software Architecture for Future CMS," *SAE Int. J. Aerosp.* 8(2):2015, doi:10.4271/2015-01-2545.

## INTRODUCTION

Avionic systems comprise numerous hardware modules, networking equipment and peripheral devices along the aircraft, and often carry out sophisticated functionalities with severe real-time and safety requirements. Moreover most avionics need to comply with intensive legal regulations regarding quality assurance. This leads to an expensive development process. Most of today's avionic systems are not designed to be changeable, and can be classified as rigid computer systems [1]. This may be acceptable for the avionics of many usage domains but not for systems within the cabin domain such as the CMS. The CMS functionality and devices are tightly coupled with cabin interiors, which are frequently subject to changes both before and after the entry into service [2].

Today's CMS face the challenge of changeability with configurable architectures. These consist of a generic core software that can be customized with configuration parameters. CMS solutions in modern commercial aircraft offer up to one million configuration parameters [3]. Due to such a vast change domain, today's CMS can be customized with almost every cabin layout. The drawback of this concept is the huge configuration effort. The configuration process for a complete aircraft cabin can even take more than 100 days [3]. More sophisticated CMS solutions improve the configuration process of CMS with automatic configuration generation based on minimum cabin models and supported cabin layout design tools [2], [4].

Although these solutions reduce the manual efforts for configuration generation noticeably [3], the automatically produced configuration still has to be transmitted to each hardware module individually. As every aircraft has hundreds of CMS hardware modules, this approach is still very time consuming and error-prone. This is not only a disadvantage during the aircraft production but is also a problem for the change and maintenance procedures.

This paper introduces an adaptive software architecture that enables the CMS to cope with cabin changes highly automatically with minimal human interactions. The adaptive CMS concept provides an exclusive lifecycle phase for system assembly and adaptation, which is called organization phase. During this phase, hardware modules, network equipment and new or already known types of peripheral devices can be added to the CMS without pre-configuration. Existing hardware can also be removed. New application software can be uploaded onto the CMS organization server to replace the previous ones or complement them. The CMS detects the hardware and software changes, and performs the system adaptation automatically. This includes communication establishment, as well as software (partition and service) instantiation and adaptation. All hardware modules come with a common firmware for organization that is responsible for performing all these tasks during the organization phase. After the organization phase, the CMS has no adaptive characteristics, and operates in accordance with classic avionics behavior, that is fault-tolerant and real-time execution of applications.

in the framework of redundant system architecture under various system operation modes during system life-cycle. This is achieved by built-in services in the software architecture for system management including communication, failure, health, redundancy and system operation management. System management services within the software architecture are generic and completely separated from CMS applications. This makes management issues transparent for applications and reduces their complexity significantly in comparison with today's avionics approaches where the system management is mainly done by the applications.

## CABIN MANAGEMENT SYSTEM

### *Functional Aspects*

The CMS is responsible for visual and audio intercommunication between crew members and with passengers. Moreover, it offers an interface for the cabin and the maintenance crew to control and monitor cabin related systems such as water/waste systems, air conditioning or inflight entertainment. Last but not least the CMS offers emergency and security functions including emergency signaling and cabin video observation. In a modern commercial aircraft the CMS can therefore comprise several hundreds of peripheral devices such as passenger and crew interfaces, illumination components and hardware modules or network equipment that are distributed along cabin decks and cargo compartments, as well as in cockpit, galleys, and lavatories.

### *Real-Time Aspects*

Some CMS functions have severe time constraints. As an example the maximum acceptable delay for audio transmission in the cabin is between 100ms [5] and 200ms [6]. Illumination functions also require high time synchronicity [5]. On the other hand, some applications have to be run periodically while others have an event-triggered nature.

### *Integration Aspects*

In recent years the amount of functionalities within the cabin has grown enormously [7]. This trend seems to continue in the future [8]. Many of today's cabin functionalities are realized as separate systems with individual cabin networks [7]. Such a design philosophy may prevent the disadvantages of a highly integrated CMS at first sight. Though a deeper analysis makes clear that such a design increases not only the complexity, weight and production efforts, but also hinders the interoperability of functions [7] [8]. CMS functions often need to cooperate and interact with each other to deliver an optimal service. An audio announcement may stop the video streaming, and can itself be interrupted by an emergency call or alarm. The illumination color and intensity in a cabin room may vary according to the time of the day, flight phase and the status of cockpit or lavatory doors in the surrounding area. Such interactions between systems may seem simple, but many of them together result in high overall system complexity. An integrated CMS would not only ease the interoperability between functions, it would also use the hardware

resources optimally and ease the crew job with reducing the number of HMI. Therefore it seems to be inevitable for the future CMS to integrate as many cabin functions as possible [7].

### *Reliability*

The CMS may contain several safety critical functions such as emergency intercommunication, emergency lighting, emergency signaling and smoke detection functions. This requires not only a high level of design assurance [5] [9] for hardware and software, but also fail-operational behavior, and therefore fault tolerance and redundancy. Customer satisfaction is also a driving aspect for CMS reliability. Disturbances in comfort functions such as a flickering light or noisy audio communication during the flight may disquiet passengers as this can be interpreted as a hazardous situation. Loss of comfort functions in business and first class for a single passenger may even force the airline to replace the aircraft.

### *System Operation Modes*

The in-flight and on-ground operating environments of the CMS require specific system behavior in different lifecycle phases and modes. Each phase can also consist of sub-phases, and may require interactive operation with an operator. The relevant main phases for this work are "final assembly", "service" and "maintenance". The challenge here is to provide a generic system operation management in order to abstract most of these phases from the applications.

### *Change Demands on CMS*

Flexible deployment of the aircraft during its operational lifetime is an important commercial parameter. Airlines have to respond to dynamic market demands with reassigning aircraft into markets with increased demand, and/or away from markets with reduced demand [10]. Moreover a commercial aircraft is approximately resold 4 times in its lifetime [11]. Many new technologies or trends may also arise during the lifetime of an aircraft that have to be integrated in the CMS [12]. Each of these scenarios may result in extensive cabin and CMS changes. Accordingly a highly changeable CMS would improve the aircraft efficiency, as it would ease cabin layout rearrangements, retrofits and rebuilds.

Aircraft cabin and its systems are subject to changes even during their production time. More comfort and better service is the major parameter to legitimate higher ticket prices. Therefore airlines customize the cabin interior of their aircraft, to underline their corporate design and improve passenger perception [2]. On the other hand, aircraft and CMS manufacturer try to reuse the same cabin and CMS concept in as many aircraft types and families as possible. Consequently a highly changeable CMS concept would ease the process of cabin customization and save development costs for the manufacturer.

In order to comply with these changes, CMS should be able to allow comprehensive changes in its hardware, software and network. These can be grouped into topology and functional changes. Topology changes affect the network and hardware structure of a CMS including addition and removal of peripheral devices, hardware

modules, or network equipment. This covers skin-deep changes such as new illumination components or passenger interfaces, as well as fundamental changes such as network expansion with more switches, IOMs or changes in the wiring. Functional changes focus on changes in software components of the CMS. This can be the integration of new software applications into the CMS, adding the suitable communication driver for a new peripheral device, or modifying the existing applications.

## RELATED WORK

In the following paragraphs relevant system and software approaches as well as substantial definitions and standards are introduced. The focus lies on the field of automation, naval systems and avionics.

### Definition of Adaptivity

Adaptive systems and architectures have been a research subject in computer science and avionics for more than a decade. According to the most common and well-established definition [1] [13] [14], an adaptive (avionics) system requires two fundamental components. First of all it needs an underlying flexible architecture that allows structural changes in the system. Additionally it needs an internal change agent that is able to detect environmental changes, deduce suitable system changes and enforce them. An internal change agent works mostly automatically, and has as few human interactions as possible. This distinguishes an adaptive system from classical flexible or agile systems [13].

### SOA Approaches

Service Oriented Architecture (SOA) is the summation of well-known and practice-proven design patterns that can be used as a roadmap for building changeable computer systems [15]. The focus of SOA is more on the flexibility of architecture and not on change mechanisms or agents. Meanwhile SOA is widely used as the ultimate benchmark to measure the changeability and reusability of software architectures. SOA is also widely used as a roadmap for changeable time deterministic device level architectures (SODA) in the automation domain [16] [17]. Recent research try to expand the application domain of SODA into driver assistance systems (DAS) [18]. Although presented solutions are highly innovative, they are usually domain specific and not suitable for avionics. Nevertheless SOA is used as an architectural roadmap for our adaptive CMS architecture.

### DDS Approaches

Data Distribution Service (DDS) is a standard from Object Management Group (OMG) for net-centric communication adaptation (Pub/Sub). Moreover it offers time-awareness and data reconstruction services to obtain real-time characteristics. These enhanced real-time services distinguish DDS from older Pub/Sub standards such as DCOM, CORBA Notification Service or JMS [19], and make it a suitable choice for building adaptive real-time systems. In combination with SOA as an architectural standard, DDS became the leading technology for Real-Time SOA soon after its standardization [20]. DDS was even explicitly

recommended as the standard adaptive middleware for marine applications in OACE (Open Architecture Computing Environment), the leading standard for open architectures from US Navy [21]. Meanwhile some implementations of DDS have also been certified with DO-178C level-A design assurance level, and are ready for service in IMA in combination with VxWorks operating system [22]. This increases flexibility of the communication layer of IMA significantly. Nevertheless, even this advanced IMA would not really offer powerful services for a generic system management, as proposed in this paper, and in consequence will not provide any comparable adaptability.

### PnP Approaches

Plug and play (PnP) technologies such as PCI and USB are originally protocols to detect and integrate new devices, and aim to ease hardware integration mainly on PCs. Similar technologies are widely used in automotive and embedded systems. As an example IEEE-1451 represents a very important standard for embedded plug and play transducers, and is meanwhile used in fault-tolerant avionics [23]. The potential of plug and play systems were also soon recognized in space avionics, where systems have an extreme long life time and very challenging maintenance procedures. In [24] an initial concept is described for adaptive space avionics with elements of IMA and IEEE 1451. Recent standard for space plug-and-play architecture (SPA) from Air Force Research Laboratory combines adaptive behaviors similar to IEEE-1451 and DDS (Component Detection and Publish/Subscribe services) with a standard architectural framework, and is also used by the NASA in the area of space avionics [25]. SPA offers abstract services for adaptation, fault-tolerance and other functionalities, however not sufficient for all avionic domains. More sophisticated management services such as service level degradation, system reconfiguration or system operation management need to be implemented by the application. Moreover SPA management services have to be orchestrated by the application. This results in orchestration complexity in the application. The enhanced generic services for management in the adaptive CMS architecture solve this problem.

## ADAPTIVE CMS ARCHITECTURE

### System and Network Architecture

A centralized processing architecture is assumed in this work. Though this is inspired by classic CMS solutions and is not mandatory for the adaptive CMS concept. The main module types of adaptive CMS are CPMs (Core Processing Modules) and IOMs (Input Output Modules). Core application services as well as main supervisors are located in CPMs. IOMs in contrast act as interfaces between CMS network and peripheral devices e.g. passenger interfaces, smoke detectors, or illumination components. The distribution of functions on IOMs and CPMs is also a well-established practice in avionics and is used in Boeing B-777 and Airbus A380 [26] [27]. CMS network connects all modules (CPMs and IOMs) with each other. It is built up of adaptive network switches (ANS), which are full duplex IP switches with specialized communication protocol and adaptation firmware. ANS perform message routing between CPMs and IOMs as well as support reconfiguration on demand. In order to comply with safety

requirements for commercial aircraft, at least two redundant CPMs and a fault tolerant network are assumed. ANS are connected in parallel daisy chains along the cabin and are cross wired at their ends. Figure 1 illustrates a single deck double aisle cabin, as well as a double deck single aisle cabin with this network architecture.

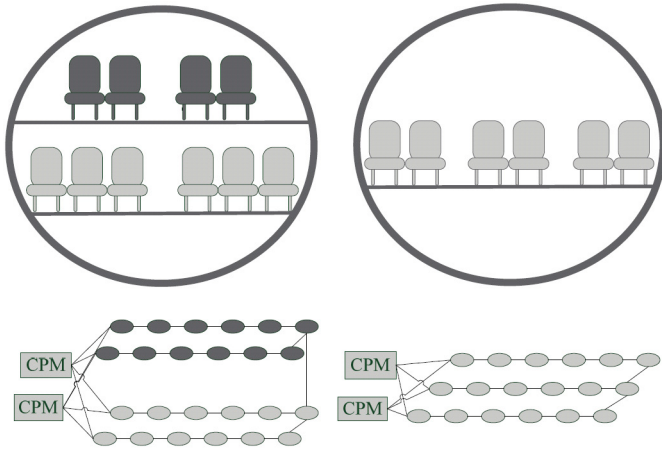


Figure 1. Adaptive network in two exemplary cabins

Although last ANS in all chains are wired with each other, they terminate the chain routing and do not allow ring communication. This produces a virtual break in the network and makes the routing in each chain definite. However in case of failure, network routing can be reconfigured to allow an alternative communication path. For this purpose the virtual break point can be duplicated or its position can be changed in the network. The upper and lower graphics in figure 2 show the communication and virtual break points in the topology respectively in failure free and failure cases.

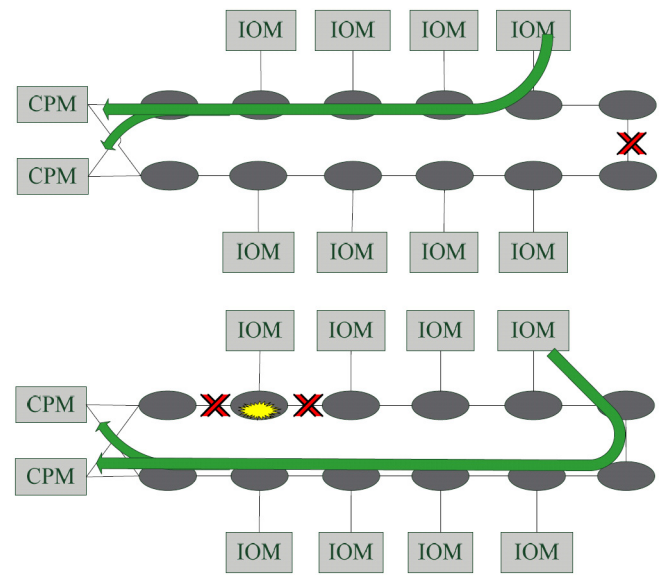


Figure 2. Virtual break in failure-free and failure case

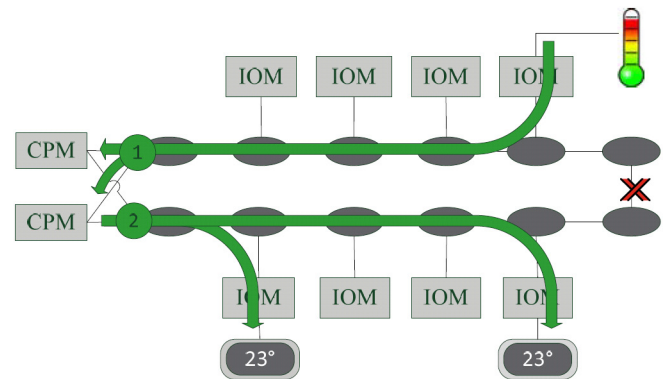


Figure 3. Communication in adaptive CMS

During normal operation, data is transmitted via multicast messages. IOMs gather data from peripheral devices, pack it inside messages and send them to CPMs via network. While all CPMs receive and process these messages, only the master CPM produces reaction messages, which it sends to one or more IOMs. As all messages have multicast addresses, CPMs, which are not master, receive all messages and process them silently. Management information and commands are also transmitted as multicast messages between CPMs and IOMs, CPMs and ANS or between several CPMs. This allows the realization of the data centric communication paradigm and enhanced failure reactions.

Video streaming data is the only exception to this rule, as it is directly transmitted between IOMs without the participation of CPMs. All other data messages are exchanged between IOMs and CPMs.

In figure 3 the communication in adaptive CMS network is demonstrated, where an IOM receives the temperature data from the sensor, packs it into message 1 and sends it to CPMs. Current master CPM packs the received temperature value into message 2 and sends it to IOMs where it can be shown to the crew. Other CPM receives messages simultaneously and processes them silently.

## Software Architecture

### Fundamentals

All bare CMS hardware modules (CPMs and IOMs) contain a common pre-loaded software. This includes an operating system for the memory, process, and network interface management as well as an organization firmware for communication and software adaptation.

During the organization phase all hardware modules receive the proper application and management software packages automatically according to their required tasks. The main task of an IOM is the communication with its peripheral devices. CPMs in contrast host core applications. In order to allow the implementation of software packages of different DAL and to avoid negative side effects during incremental system integration, software services in IOMs and CPMs are segregated in partitions. The operating system assures each partition its required minimum system resources including memory, computing time and network access. Applications are assigned to the so called "application partitions".