3

SOA-Based Aeronautical Service Integration

Yifang Liu[1], Yongqiang Cheng[1], Yim Fun Hu[1],
Prashant Pillai[1] and Vincenzo Esposito[2]
[1]*School of Engineering, Design and Technology, University of Bradford*
[2]*SELEX Sistemi Integrati S.p.A.*
[1]*United Kingdom*
[2]*Italy*

1. Introduction

Over the past two decades, the air transport industry has experienced continuous growth. The demand for passenger air traffic is forecast to double the current level by about 2025 (European Organisation for the Safety of Air Navigation [EUROCONTROL], 2006). Small-to-medium sized low cost airlines in Europe such as EasyJet and Ryanair have observed a considerable percentage of passenger increase between 2008 and 2009 due to the growth in the number of regional airports and more choices offered on international destinations (EasyJet & Ryanair, 2009). To accommodate such growth and changes in new flight patterns and strategies, it is of paramount importance to ensure air transport communication systems around the globe be integrated to enable efficient air-to-ground and ground-to-ground communications for global air traffic management and coordination.

Traditional approaches for aeronautical system integration in the past impose a high level of system dependencies; a fixed connection is required to be set up every time a new application is added. Therefore, aviation companies are facing continuous investment increase every time a new connection is established. This situation discourages enterprises from fulfilling grater business values by adding interior constraints; it restricts the number of applications and services that can be integrated into the existing IT infrastructure. In safety-critical systems in the aeronautical context, overloaded complex system structure will increase the chances of operational failures and jeopardise passenger safety. Therefore, it is important to devise a suitable architecture which minimises system dependencies and allows new applications to be integrated easily with the lowest IT maintenance budget. A layer-extensible blueprint in a Service-Oriented Architecture (SOA) is considered as a solution in this case for the integration of future aeronautical communication systems. The proposed framework should allow consistent data capturing and sharing among all end users who are involved in the global aircraft operations in the 2020 timeframe and beyond.

In recent years, the SESAR SWIM (Single European Sky Air Traffic Management Research System Wide Information Management) concept has reflected the emerging needs and willingness of Air Traffic Management (ATM) organisations in transforming proprietary ATM systems into a standardised and interoperable information pool in the pan-European aeronautical network. As the challenge still exists where the ATM stakeholders today do not want to deal with the complexity of the lower communication layers, SOA is considered as one of the most effective emerging technologies to provide a scalable, flexible and interoperable system framework, according the adoption of the SOA paradigm in the global SWIM studies (Houdebert & Ayral, 2010; Luckenbaugh et al., 2007). However, SWIM focuses on ground-to-ground aeronautical services.

In extending the SWIM ideology to an airborne context, the EU FP7 project SANDRA (Seamless Aeronautical Networking through integration of Data-Links, Radios and Antennas) continues with the SOA notion in air-to-ground information exchange, service composition and integration to provide a complete and coherent set of communication services for Next Generation global Air Traffic Management.

Building on the investigation and analysis of the existing industrial programmes targeting aeronautical service integration, this chapter provides an introduction of the SOA-based future avionic systems. Section two outlines the middleware concept, the service-oriented architecture, its implementation technologies and the use of SOA design solutions forming an integrated ATM framework. Section three summarises the SOA-based future aeronautical communication referring to the SESAR and FAA (Federal Aviation Administration) SWIM approaches for information fusion and dissemination for ground-to-ground service integration. The SWIM ATM added-value services, data access services and technical services defined in Section three are used as a baseline for the definition of the SANDRA Airborne Middleware (SAM) through the utilisation of a set of airborne/ground data domain objects, as described in Section four. Finally, analysis of the service improvement methods and the technology options for future ATM system realisation is addressed at the end of this chapter.

2. SOA in aeronautical communication

SOA is an emerging middleware approach for linking various legacy systems into a standard platform to achieve a highly interoperable and collaborative communication infrastructure. It permits the separation of legacy system service interfaces from the underlying implementation, thus to reduce technology-dependent attributes of the system. SOA promotes service reusability and interoperability through a set of standardised data schemas used in the discovery and self-description of each course-grained, composed and loosely-coupled service.

The SOA capabilities are seen as an enabler for the realisation of the EUROCONTROL SESAR concept (EUROCONTROL, 2008), which explicitly states the focus on the global ATM interoperability with regard to the semantics of the data exchanged, the protocols and the overall quality of dialogue in terms of Communication, Navigation and Surveillance (CNS). According to the ICAO 2010 operational opportunity report (International Civil Aviation Organisation [ICAO], 2010), the European ATM Master Plan defines the “path” towards achieving performance goals by adopting the service-oriented architecture as agreed at the European Union ministerial level. The main targets are to:

- Enable a 10% reduction in CO2 emissions per flight
- Reduce ATM costs by 50%
- Enable a 3 times increase in capacity
- Improve safety by a factor of 10

Supported by state-of-the-art and innovative technologies designed to eliminate fragmentation in the future European ATM system, SOA-based middleware reflects the operational, technological and regulatory requirements of the future ATM infrastructure while serving for the improvement of system efficiency and interoperability.

2.1 Middleware and service-oriented architecture

2.1.1 Definition of middleware

The term middleware was first introduced in 1968 and had only gained its popularity until it was formally used as an integration platform to connect different systems and applications since the 1980s. The role of middleware varies in different domains. In the scope of enterprise applications integration, middleware is called plumbing because it connects two applications and passes data in between (Simon, 2003). For purpose such as data integration of heterogeneous networks across different geographical locations, especially in the Air Traffic Management context, middleware is a distributed software layer that sits above the operating system and below the application layer and abstract the heterogeneity of the underlying environment. Middleware provides an integrated distributed environment whose objective is to simplify the task of programming and managing distributed applications (Campbell et al., 1999).

The common types of middleware are Message-Oriented Middleware (MOM), adaptive and reflective middleware and transaction middleware. Middleware can be grouped according to different technologies, such as grid middleware, peer-to-peer middleware and real-time middleware concerning the Quality of Service, security, performance, Model-Driven Architecture, Service-Oriented Architecture and more (Mahmoud, 2004).

In aviation, the shift from proprietary air traffic control systems into a standardised and interoperable platform embracing the middleware approach facilitates the communication and integration of a wide range of ground-based and air-to-ground system applications operating across the networks. The middleware acts as an intermediary enabling direct communication with the legacy technology interfaces, to minimise system dependency.

2.1.2 Definition of service-oriented architecture

As an embodiment of the middleware concept, SOA is a paradigm for the integration of various applications running on heterogeneous platforms using common standards. It is designed to consolidate the system capabilities for the organisation and utilisation of data distribution managed by different ownership domains. The term "service" can be defined as a single or multiple operational functions offered by a software system for the fulfilment of business objectives, for example, flight plan filing, aircraft tracking, controller-pilot communication and passenger logistics management. The specification of services may be modified as the business objectives and operations change.

There are eight design principles that affect the design of services and SOA-based system integration (Erl, 2009):

- **Standard Service Contract** – Services within the same service inventory should have the same contract design standards.
- **Service Loose Coupling** – Service contracts ensure the service consumers are decoupled from their surrounding environment.
- **Service Abstraction** – Information contained in the service contracts are limited to what is published.
- **Service Reusability** – Services contain and express agnostic logic and can be reused as enterprise resources.
- **Service Autonomy** – Services exercise a high level of control over their underlying run-time execution environment.
- **Service Statelessness** – Minimised resource consumption by deferring the management of state information when necessary.

- **Service Discoverability** – Services described with metadata can be effectively discovered and interpreted.
- **Service Composability** – Services are effective composition participants.

The SOA concept as a recommendation for system integration is an emerging approach in the ATM development programmes in Europe and North America. It offers a uniform platform, which supports the registration, discovery and administration of individual business process with use capabilities to produce desired effects consistent with measurable preconditions and expectations in a short timeframe.

Rooted in the Business Process Management (BPM) notion, SOA is a holistic mechanism for the alignment and harmonisation of an enterprise and its IT development as:

- SOA encompasses the tools and methodologies for capturing business design, and uses that design information to help improve the business.
- SOA covers the programming model, tools, and techniques for implementing the business design in information systems.
- SOA contains the middleware infrastructure for hosting that implementation.
- SOA encompasses the management of that implementation to ensure availability to the business and efficient use of resources in the execution of that implementation.
- SOA encompasses the establishment of who has authority and the processes that are used to control changes in the business design and its implementation.

The SOA principles and standards highlight the significance of loosely coupled and reusable services in the software architecture perspective. Services are independent and are accessed via standardised interfaces as a frontend of the massive network resources. The advantage lies at the transparent communication SOA offers to end-systems (technology-agnostic), and hence, to effectively demonstrate the application of data-centric information sharing. The SWIM infrastructure provides the basis for information exchange between systems based on the principles of SOA.

2.2 SOA implementation technologies

The definition of SOA emphasises that the concept of service-orientation is a paradigm solely. SOA is remarkable for its flexibility allowing many types of system interactions to be performed based on a series of pre-defined architectural patterns. From the functional point of view, classification of business process and service interaction modelling are two dominant motives at system planning and design stage. Afterwards, the realisation of software services supporting these interactions requires the state-of-the-art technologies to be defined. Technology-independence is one of the most important criterions in terms of technology evaluation.

The paragraphs below provide a general overview on common technologies for the implementation of a service-oriented architecture of which are used in aeronautical communication.

2.2.1 BPM, BPMN and BPEL

In the past 30 years, the growing concept of **Business Process Management (BPM)** has shifted from the use of static business process flowcharts in unchanging organisations to dynamic corporate processes which can be accessed by collaborating partners in a more flexible and efficient way. A business process can be summarised as a collection of structured activity to produce a specific business service or product. BPM introduces sophisticated software and best practices targeting the modelling, simulation, automation and management of cooperative operations with dynamic business priorities.

Business Process Modelling Notation (BPMN) is a visual language with graphical notations for the modelling of business processes. It presents the business activities, tasks and their relations in a business process diagram (Juric et al, 2008). BPMN can be used in collaborative processes and internal business processes. The BPMN models in future aeronautical communication should appear as a mixture of both intra-business and inter-business flows reflecting the concept of Collaborative Decision Making (CDM).

To implement the modelled business interactions, **Business Process Execution Language (BPEL)** enables the service orchestration for composed service and business processes reinforcing service reusability and loose coupling. BPEL conducts the orchestration of services. It is mapped from the BPMN diagram for service execution, and thus allowing integrated monitoring functions to be applied. The predominant standard for BPEL is the Business Process Execution Language for Web Services (BPEL4WS v1.1) in 2003 (Andrews et al, 2003), defined in the human-readable Extensive Mark-up Language.

Based on the BPM-related concepts, the SESAR SWIM Program has addressed the need to derive a common view on the ATM business processes for accessing SWIM ATM Data and ATM functionalities. The development of a formal business process model has been recommended and it is essential to follow standard IT industry practices through the use of enterprise architecture modelling techniques.

2.2.2 Web services

The Web Service infrastructure is one of the most common approaches for the realisation of SOA. It is recognised as a predominant technology framework in the avionic industry for the realisation of the SWIM infrastructure. The World Wide Web Consortium (W3C) defines Web services as a standard software system for interoperating different software applications running on a variety of platforms and/or frameworks (W3C, 2004). A Web service supports interoperable machine-to-machine interaction over a network based on the Web service stack illustrated in Fig. 1.

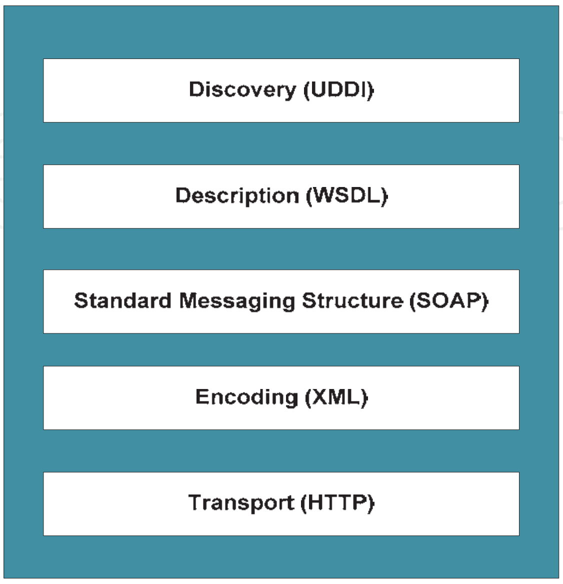


Fig. 1. Web Service Stack.

- **HTTP** – Hypertext Transfer Protocol (HTTP) located in the lowest level of the Web service stack supports the distribution of information across networks. Web applications usually use HTTP on port 80 and HTTPS on port 443 for security purposes. Web service uses SOAP as a messaging protocol over HTTP.
- **XML** –Extensible Mark-up Language (XML) data formatted with XML tags. Data exchanged between services could be described in XML Schema Definition (XSD) conforming to the SOAP as a messaging standard.
- **SOAP and transport protocols** – Simple Object Access Protocol (SOAP) as a standard messaging protocol, and other transport protocols such as HTTP, FTP, SMTP and JMS are common protocols for web service communication.
- **WSDL** – Web Service Description Language (WSDL) is an XML formatted file used to abstractly describe the network services corresponding to the endpoints. WSDL defines the service details (contracts) between service consumers and providers. Messages, port types for specific operations, bindings, services containing SOAP addresses are key elements of a WSDL file.

Web service stands out for its standardised languages and specifications. Nevertheless, the verbose structure and the long processing time of XML schema are two major constraints. Therefore, it is essential to define suitable mechanisms to improve the efficiency of XML applied in aeronautical communication where the performance level should be optimised in order to ensure the Quality of Service. As addressed in Section 4, the matching of Abstract Syntax Notation One (ASN.1) and XML and compression mechanisms can be considered as potential solutions.

2.2.3 Message-oriented middleware

Message-Oriented Middleware (MOM) is a software term that connects multiple systems in a network for data distribution, typically in a service-oriented architecture. It is inspired from the conventional client/server architecture nevertheless, with advanced features allowing both synchronous and asynchronous communication. Such feature is beneficial in providing a messaging platform accessed by global ATM systems.

The introduction of MOM enhances the level of quality attributes such as performance, scalability, flexibility and interoperability. MOM provides Java Message Service (JMS) as the standard Application Programming Interface (API) to perform the message broker function and to support client application.

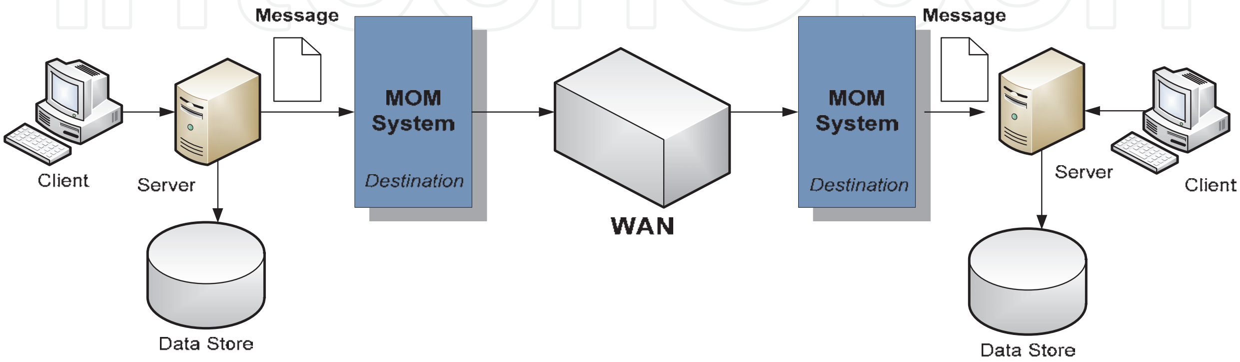


Fig. 2. MOM system architecture.

As indicated in Fig. 2, a MOM system can deliver message across networks via a centralised message server or it can distribute routing and delivery functions to each client machine. The client can continue requesting information from the data store or performing other operations while delivering messages to the JMS API.

2.2.4 Enterprise service bus

An Enterprise Service Bus (ESB) is a common implementation solution that allows services to be used in a productive system. An ESB provides coarse-grained interfaces with the purpose of sharing data asynchronously between applications. Such integration architecture pulls together applications and discrete integration components to create assemblies of services to form composite business processes, which in turn automate business functions in a real-time enterprise. The rise of multiple ESBs is a result of iterative SOA implementation approaches.

An ESB provides (but not limited to) the following functions:

- Messaging functions, such as transformation, delivery and routing
- Service registry and metadata management for the storage and discovery of services
- Adapter functions supporting various communication protocols
- Support to allow service composition/orchestration in business processes through WS-BPEL.
- Security management in order to provide authorization, authentication, and the creation of the policies
- Management monitoring and configuration of the management, components life cycle, logging and auditing

2.2.5 Data distribution service

Data Distribution Service (DDS) is a newly adopted middleware specification for distributed real-time applications introduced by Real-Time Innovations (RTI) in 2003. It is a standard implementation of Object Management Group (OMG) and is used in many time-critical and data-critical applications such as industrial automation, robotics, air traffic control and monitoring and transaction processing.

DDS is aimed at a diverse community of users requiring data-centric publish/subscribe with high flexibility, performance, portability and configurable data distribution management using the topic channels. Such publish/subscribe based communication model is used for sending and receiving data, events, and commands among the service nodes managed by different publishers (containing any number of DataWriters) and subscribers (containing any number of DataReaders) connected to the Global Data Space for resources sharing.

The development trend of DDS, e.g. the amalgamation with Web Services standards is driving DDS to a maturing SOA solution for future aeronautical service integration.

2.2.6 Common object request broker architecture

Common Object Request Broker Architecture, namely CORBA, is a specification defined by the OMG for system integration of aeronautical legacies. However, as the Web Service based solutions are gradually gaining recognition, CORBA is shifting to a legacy category.

2.3 SOA and air traffic management

The investigation and analysis of SESAR SWIM-SUIT (System-Wide Information Management SUpported by Innovative Technologies) and the FAA SWIM prototypes reveal substantive findings on service integration of ground-to-ground communication in a service-oriented architecture (EUROCONTROL, 2008; FAA, 2010). Implementation of the functional framework is carried out using Web Services, JMS, DDS and ESB to support one-to-one, one-to-many and event-based communication via request/reply and publish/subscribe message exchange patterns. It is envisaged that the SWIM-based SANDRA Airborne Middleware and future air traffic management infrastructure will also consider SOA as a baseline for seamless aeronautical communication in the next decades.

3. The SWIM SOA approach

3.1 SWIM overview

System Wide Information Management, namely SWIM, is an information sharing concept leading to the development of a number of technology programmes conducted by both the FAA, as the Next Generation Air Transportation System (NextGen), and EUROCONTROL to facilitate the information sharing and global situation awareness in the future aeronautical context. In the past, the state-of-the-art to connect different ATM systems required a fixed connection and application-level data interfaces to be set up individually between each system. SWIM is essentially introduced to reduce the high degree of system dependence by providing a Network Centric environment to enhance the flexibility of aeronautical system integration.

ICAO Global ATM Operational Concept definition of SWIM (SWIM-SUIT, 2008):

"System Wide Information management aims at integrating the ATM network in the information sense, not just in the systems sense. The fundamental change of paradigm forms the basis for the migration from the one-to-one message excha is geographically dispersed sources collaboratively updating the same piece of infornge concept of the past to the many-to-many information distribution model of the future, thatmation, with many geographically dispersed destinations needing to maintain situational awareness with regard to changes in that piece of information. Successfully managing the quality, integrity and accessibility of this complex, growing web of distributed, fast changing, shared ATM information, called the virtual information pool, can be considered as the main operational enabler for the operational concept."

FAA description of SWIM (SWIM-SUIT, 2008):

"To streamline the evolution and modernization process, SWIM concept is to support loosely coupled, many-to-many data exchange interfaces. When implemented, SWIM will allow information producers and consumers to exchange data in a secure, robust, standards-based, loosely coupled environment." […] "Exploitation of advancing technology that moves from an application centric to a data-centric paradigm – that is, providing users the ability to access applications and service through web services – an information environment comprised of interoperable computing and communications components."

The EUROCONTROL SESAR definition of SWIM (SWIM-SUIT, 2008):

"SWIM represents added value also in terms of facilitating general accessibility. Focus shifts from the producer of information to information itself and generalised access to information (as opposed of pre-packaged sets as is the case today) enables users to create their own applications which best suit their mission needs. In the ATM network, almost every participant is a producer as well as a consumer of information. It is not desirable to decide in advance who will need what information, from whom and when. The key issue is to decouple producer of information from the possible consumer in such a way that the number and nature of the consumers can evolve through time. On the contrary for what concerns the producers of information it is of the utmost importance to agree on the level of interoperability required with other ATM stakeholders that may have to contribute to the elaboration

of the consistent and consolidated view of the reference data. For that purpose, the SWIM participants have to share:

- *A reference Data and Services model,*
- *A set of agreed cooperation patterns (Rules, Roles and Responsibilities),*
- *A set of technical services necessary to support system interactions,*
- *An access to the SWIM physical network.*

In short, SWIM provides the mechanisms which support the partners in managing the Rules, Roles and Responsibilities (the 3Rs) of information sharing. This determines which kind of information is shared by whom, with whom, where, when, why, how, how much, how often, at which quality level, in what form, for which purpose, at which cost, under which liability, under which circumstances, security level of air traffic management. The 3Rs must also be properly addressed both in terms of institutional and Information Communication Technology (ICT) aspects.

3.2 SWIM in Europe

Since 1997, EUROCONTROL's Experimental Centre has been participating in the SWIM-SUIT Project, an initiative laying some of the foundations of SWIM. In 2008, the EUROCONTROL launched SWIM-SUIT as an underlying work package of SESAR with the aim to facilitate aeronautical information sharing with regarding to flight data, airport operational status, weather information and special use of airspace and restrictions. The information sharing targets a number of operators working with the ATM systems in aspects such as airline operation control, administration, air traffic services and passenger and logistics management.

Building on top of the SWIM concept, the SWIM-SUIT Project was designed to allow expedite and secure access and conveyance of vital information supported by the process of collaborative decision making with the adoption of the state-of-the art technologies, for achieving efficient and effective cross-domain operations.

3.2.1 Scope and timeline

The SESAR Concept of operations for the long-term time frame calls for an overall European ATM system (EATMS) that is fully interconnected via a SWIM network. As illustrated in Fig. 3, the connected systems include:

- Pan-European systems for managing Europe/network-wide information services;
- Civilian and Military ATC systems;
- Airspace users systems (Military, Scheduled and on-demand civilian operators);
- Airports;
- Aircraft.

Such a system (or rather a "system of systems") requires a move from point-to-point message exchange to the sharing of information within a common virtual information pool. The SWIM architecture will permit actors to focus upon the information itself, rather than the systems that produce / manage the information.

The SESAR SWIM environment, given its wide reaching scope, will require a progressive, iterative and constant implementation programme. The following diagram outlines the current SESAR view on the various developments streams (hence, the deployment begins progressively following the end of a development stream).

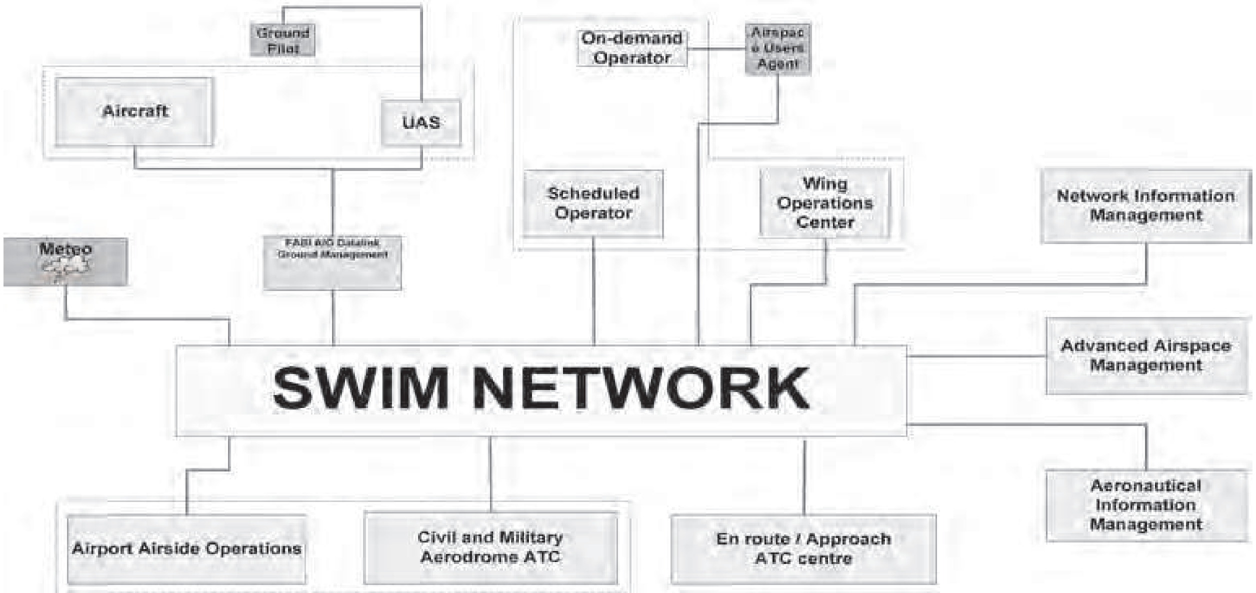


Fig. 3. SESAR SWIM architecture.

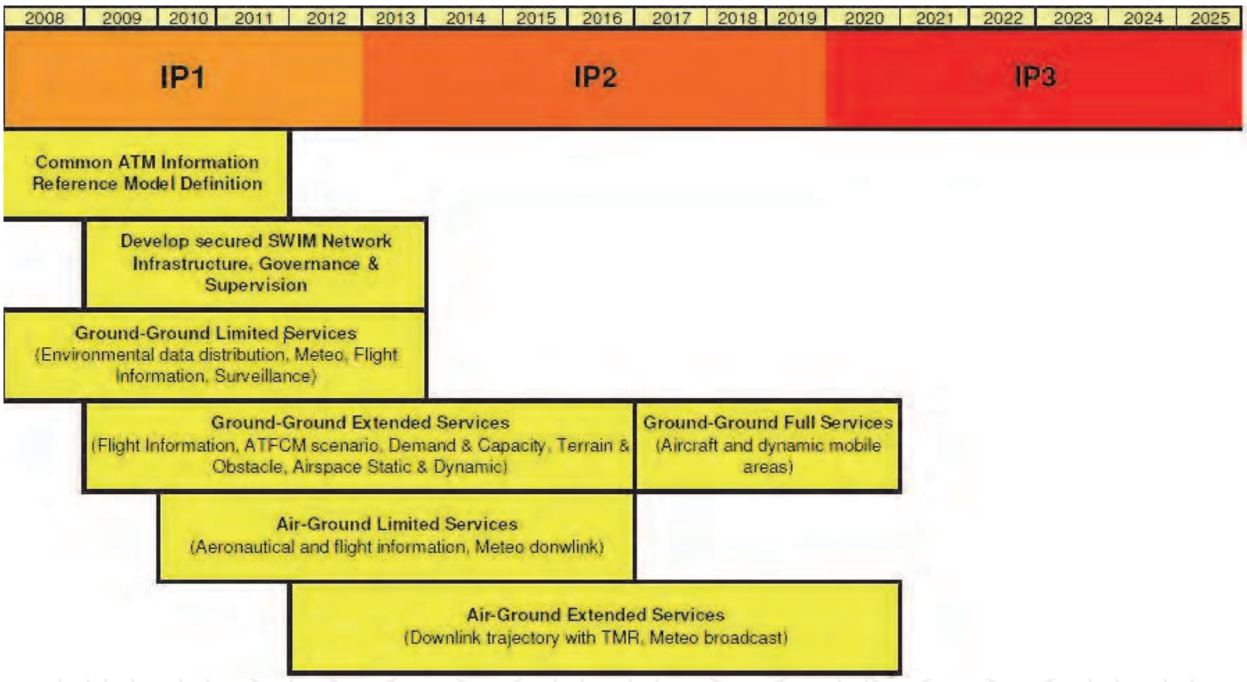


Fig. 4. SESAR SWIM implementation timeline.

3.2.2 Data sharing and services

The SESAR SWIM environment expects at least the following data to be shared across the SESAR-defined time frames, known as the short-term/medium planning, long-term planning, and the execution and post-execution phase:

- **Flight Data** - Flight Structure, Flight Script, Taxi Plan, Trajectory, What-If Flight and Context, and departure and arrival related data represented in the Flight Object Model (The European Organisation for Civil Aviation Equipment [EUROCAE], 2006)
- **Surveillance Data** - System Track, Sensor Descriptions, Aircraft Track, Traffic Advisory and ASAS Alert

- **Aeronautical Data** - Aerodrome Data, Heliport Data, Airspace Data, Navigation aids Data, Terrain and Obstacles Data and Aircraft Data with details described in (Aeronautical Information Exchange Model [AIXM], 2011)
- **Meteo Data** - Aerodrome Weather, Area Weather and Met Hazard with details described in (Weather Information Exchange Model [WXXM], 2011.)
- **Capacity and Demand Data** - Configuration Plan, Traffic and Airspace Demand, Traffic Load and Demand Capacity Balancing Measures
- **ATFCM Scenario Data** - Demand Capacity Balancing (DCB) Scenario, Flow Measures and DCB Scenario Catalogue

3.2.3 System architecture

Fig. 5 illustrates a layered conceptual view of the SWIM architecture:

- **SWIM network** – the physical pan-European network along with the essential technical IP building blocks (e.g. transport protocols, firewalls).
- **SWIM Technical Services** – the core technical services that SWIM will provide to all connected systems. These services are built, as far as possible, upon standard IT middleware technologies.
- **SWIM ATM Data Access Services** – this layer embodies the "SWIM virtual information pool". It provides access via defined services to the standardised SWIM ATM Data model. The data access services are typically be categorised into different domains (e.g. FlightDataAccess, MeteoDataAccess).
- **SWIM ATM Added-value services** – this layer contains services that provide access (perhaps distantly) to added-value ATM functionality beyond that of the "virtual information pool". Typically, this layer could contain CDM type applications/services.

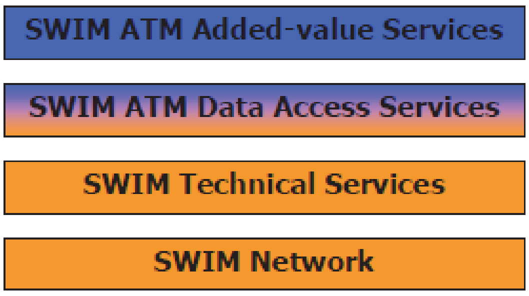


Fig. 5. SWIM layered architecture.

From a domain viewpoint, the layers can be divided into two main groups:

- **SWIM Infrastructure** – SWIM Network, SWIM Technical Services, and SWIM ATM Data Access Services (partial), which represent a common SWIM IT infrastructure, consisting of both turn-key solutions and toolkit/frameworks, upon which is built the SWIM ATM functionality.
- **SWIM ATM functionality** – SWIM ATM Added-value Services, SWIM ATM Data Access Services (partial), representing the domain specific functionality.

The SWIM infrastructure is spread out amongst the (legacy) ATM systems that form a part of the overall European ATM system (the "system of systems"). Each ATM system, typically composed of a number of major subsystems, implementing domain specific functionality will now include a "SWIM / IOP Management subsystem" which:

- Contains the (or parts of) SWIM Infrastructure functionality shown above;
- Connects the legacy system functionality to the SWIM environment;
- Translates standard SWIM ATM data structures into the appropriate legacy system data formats.

Services in SWIM are defined in a domain-specific manner providing a wide range of standard functional interfaces to support the communication and collaboration between participants connected to the SWIM network. Data collected from both the SWIM-enabled applications and ATM legacy systems, via the SWIM external adapter, is transmitted through the SWIM Ground/Ground gateways, namely the SWIM-BOXes, for the facilitation of data sharing.

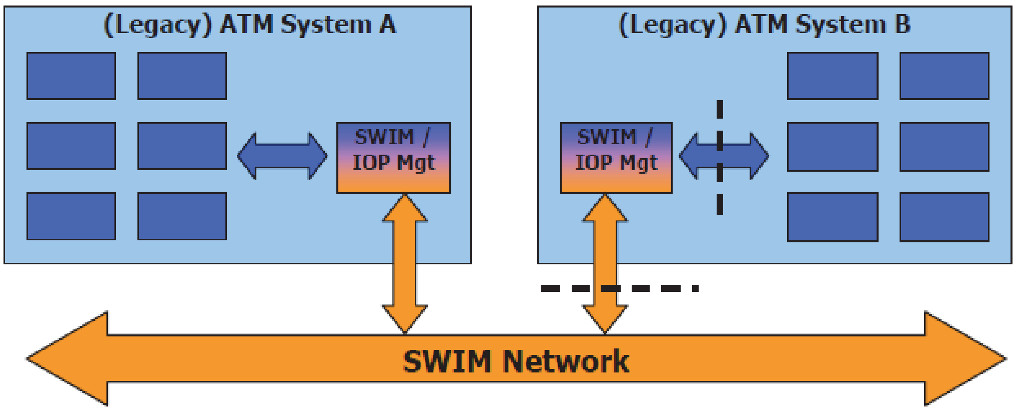


Fig. 6. SWIM/ATM system viewpoint.

3.3 SWIM in the U.S.

The SWIM Program in the United States was originated in 1997 as the EUROCONTROL initially presented the SWIM concept to FAA. The concept was under development ever since until the ICAO Global ATM Operational Concept adopted the SWIM concept in 2005. SWIM is now a part of development project in the United States NextGen framework for the development and integration of the National Air Space (NAS) systems for greater sharing of ATM system information on airport operational status, weather information, flight data, status of special use airspace, and NAS restrictions.

3.3.1 Scope and timeline

The FAA has established a notion called "Core Services" as a consistent capability existing at each node to provide a uniform mechanism for communicating among nodes. Fig. 7 illustrates the FAA view of how Core Services fit into the overall SWIM architecture.

The following system cores are included:

- En Route Automation Modernization(ERAM)
- Weather Message Switching Centre Replacement (WMSCR)
- Traffic Flow Management System (TFMS)
- FAA Telecommunications Infrastructure (FTI)
- Special Use Airspace Management System (SAMS)
- Central Processor (CP)
- National Airspace System (NAS)
- Electronic Flight Strip Terminal System (EFSTS)

- Airport Surface Detection Equipment –Model X (ASDE-X)
- Flight Data Input Output (FDIO)
- Terminal Data Link System (TDLS)
- Runway Visual Range (RVR)
- Air Route Traffic Control Centre (ARTCC)
- William J Hughes Technical Centre (WJHTC)

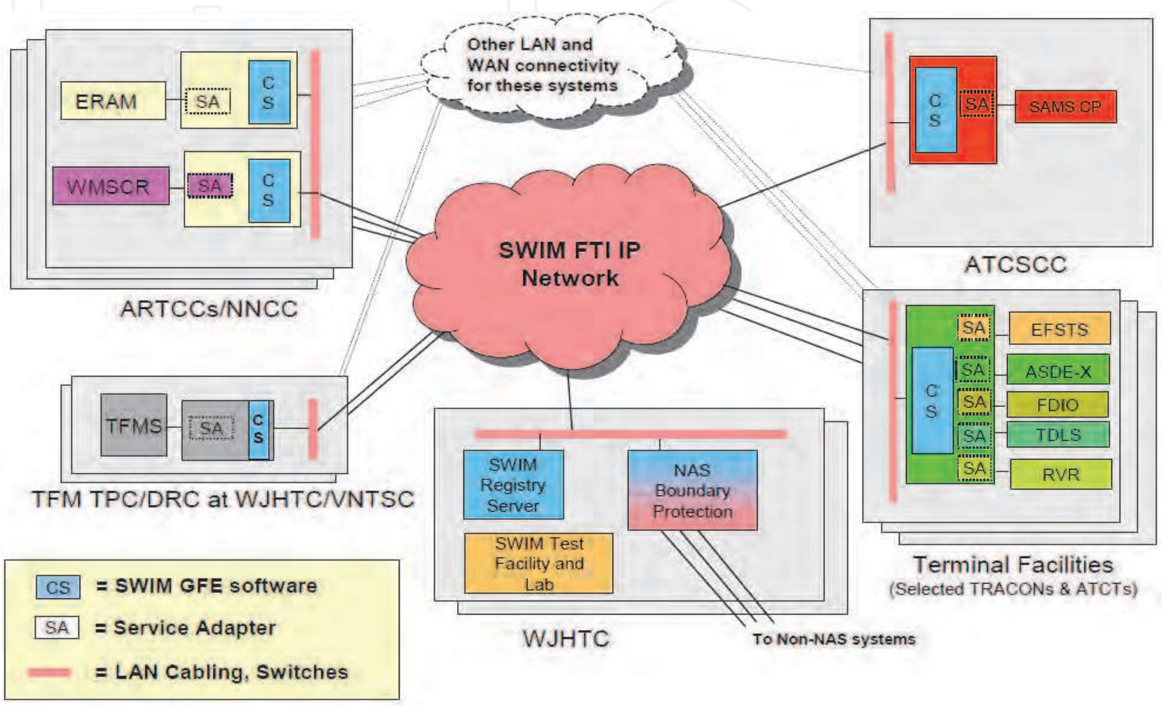


Fig. 7. FAA SWIM architecture with core services.

Fig. 7 raises the question of what specific capabilities are comprised in these Core Services. The FAA has proposed the core capabilities in Fig. 8 below.

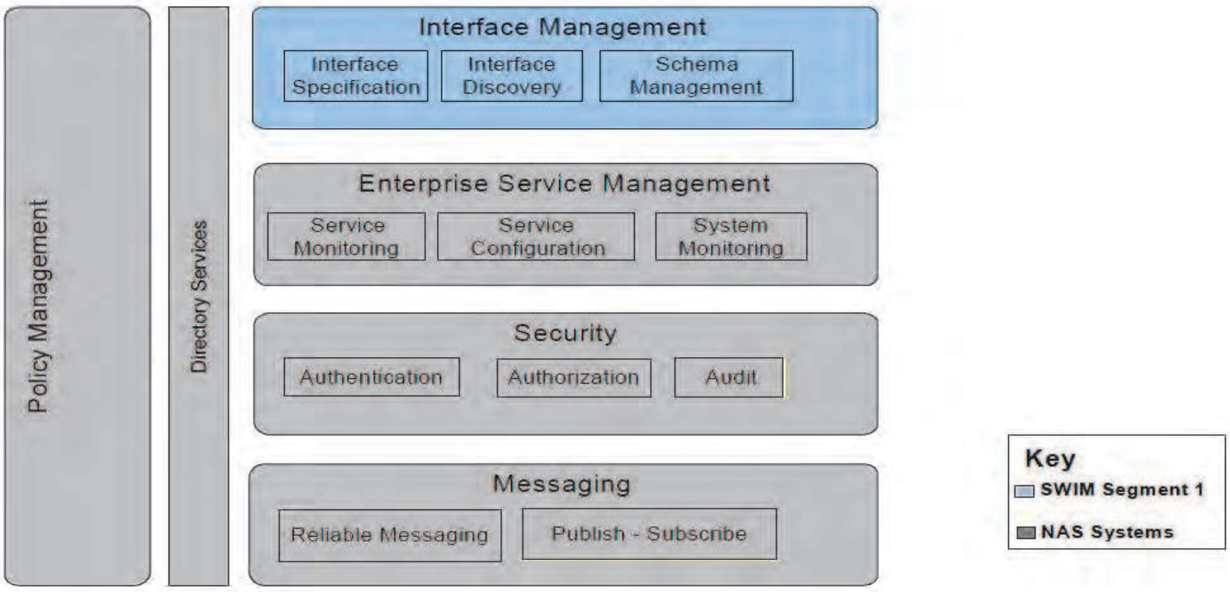


Fig. 8. SWIM Segment 1 Core Capabilities.

The FAA organizes SWIM's initial Core Services into four groups (SESAR shares the same Core Services grouping as FAA):

- Interface Management
- Messaging
- Security
- Enterprise Service Management

FAA SWIM will be deployed in segments (stages), with the first segment planned for the 2008-2012 timeframe though the NextGen has a long planning horizon (20+ years).

3.3.2 Data sharing and services

The Segment 1 business services are defined in the SWIM Final Program Requirements. It identifies and describes the services in the following categories for example:

Flight and Flow Management

- Flight Data Publication
- Terminal Data Publication
- Flow Data Publication
- Runway Visual Range (RVR) Data Publication

Aeronautical Information Management

- Special Use Area (SUA) Data Publication
- Corridor Integrated Weather System (CIWS) Data Publication
- Integrated Terminal Weather Service (ITWS) Data Publication

3.3.3 System architecture

In response to the FAA SWIM program using SOA, Government Electronics & Information Technology Association (GEIA) which is a trade association that includes many industry partners who support the FAA, provides the industry solution, shown in following figure, both adheres to the overall SOA and to the FAA SWIM vision.

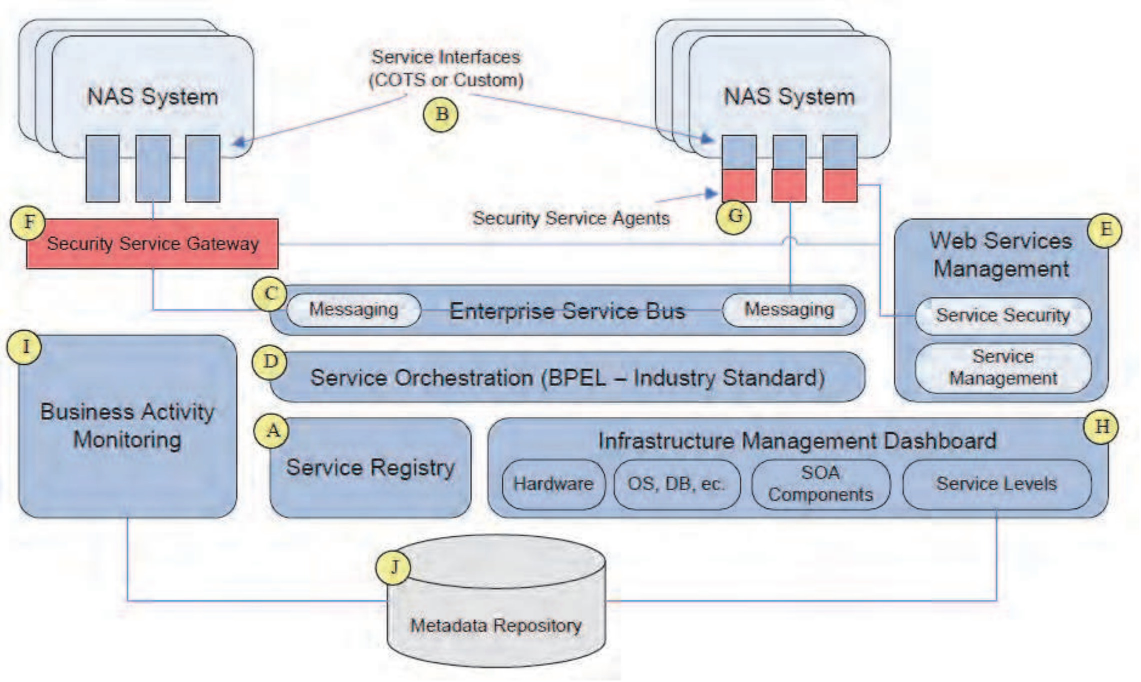


Fig. 9. GEIA SOA Architecture for SWIM (GEIA, 2008).

The GEIA architecture above supports the FAA Service groupings via the following subsystems, according to the "SOA Best Practices –Industry Input" (GEIA, 2008).

- Interface Management (A, B)
- Messaging (C, D)
- Security (E, F, G)
- Enterprise Service Management (H, I, J)

4. SANDRA – extending SWIM onboard

4.1 SANDRA overview

Building on the SESAR SWIM concept for information fusion and dissemination for ground-to-ground service integration, the EU FP7 project SANDRA (Seamless Aeronautical Networking through integration of Data-Links, Radios and Antennas) extends the ideology of SWIM to cover air-to-ground information exchange, service composition and integration to provide a complete and coherent set of communication services for future global Air Traffic Management in the 2020 timeframe. To ensure the integration of different service domains onboard legacy applications with very diverse requirements, the SANDRA communication system will represent a key enabler for the global provision of distributed services for Collaborative Decision Making based on the SWIM concept, and for meeting the high market demand for broadband passenger and enhanced cabin communication services. As a case study, the paragraphs below concentrate on the SANDRA Airborne Middleware design and how such architecture can be realised using the various SOA technologies.

Focusing on communications related aspects, the following high-level requirements are identified to allow future systems to be compatible with the expected air-traffic growth:

- Pilots situation awareness shall be improved
- Capacity at airports, as today's main limiting structural factors, shall be increased
- ATS shall be primarily based on highly reliable data communication
- AOC data traffic shall strongly increase for efficient airline operations
- Passengers and cabin communications systems shall be further developed
- Safety critical applications shall need diverse means to reach ground for global availability and higher reliability
- A simplification of on-board network architecture shall need convergence of protocols and interfaces

In order to satisfy the objectives (middleware aspect only), a possible airborne middleware architecture in SANDRA is defined aiming at the interoperation with the ground systems utilising SOA. To define the airborne middleware architecture, analysis of air/ground (A/G) information exchange is carried out focusing on the definition of the A/G data domain services and functional interfaces based on the SWIM information infrastructure. A combination of mechanisms to improve the A/G data exchange is proposed in dealing with the limited bandwidth available for over the SANDRA Data Link. The A/G integration architecture is defined containing a set of core infrastructural subsystems.

4.2 Analysis of air/ground information exchange

The SESAR SWIM environment expects at least the following data to be shared:

- Flight Data
- Surveillance Data
- Aeronautical Data

- Meteo Data
- Capacity and Demand Data
- ATFCM Scenario Data

4.2.1 SWIM ATM Information model

The ATM information to be exchanged via SWIM Network needs to be modelled explicitly, to allow a precise and concrete definition to be agreed. Previous work has already defined data models and required services within specific domains (e.g. Aeronautical Information and Flight Information), but this is not the case for all types of information. The services in support to SWIM are organized around a number of central themes called profiles.

An overall ATM Information Reference Model is required to define the semantics of all the ATM information to be exchanged. This model will form the master definition, subsets of which would be used in lower level models, supporting interoperability for each of the data-sharing domains.

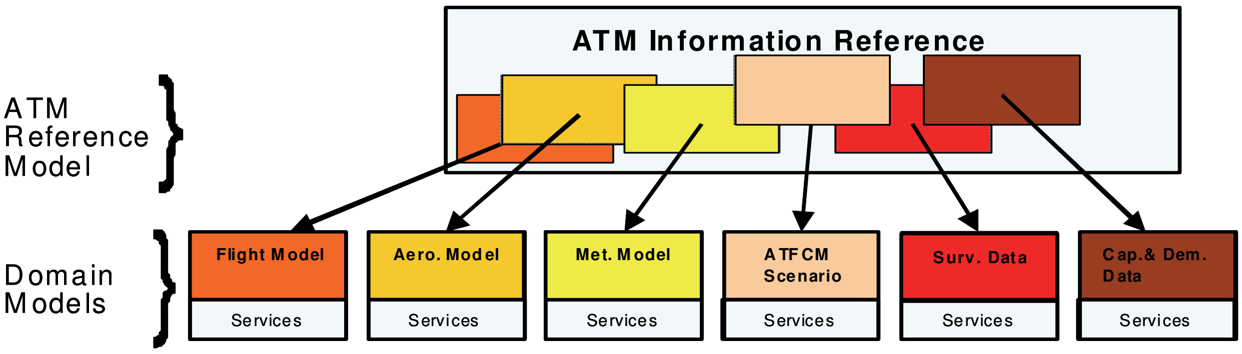


Fig. 10. SESAR ATM Reference Information Model.

The ATM Information reference would be a key asset in the ATM System design and would sit above a set of domain-specific, platform independent models which may overlap with each other, without being incompatible. The overall reference model and existing models such as OATA, FOIPS, OMEGA, AIXM etc., will need to be reconciled.

From these models, the specific SWIM exchange formats (i.e. on the wire exchange formats) can then be defined. SWIM assumes that the data handed over to SWIM Node services is already in the canonical ATM information model. There will be no mappings from/to the canonical data model in the data domain components. If such mappings are required, they have to be implemented in the SWIM Applications adapters. The data domain components perform data conversions for encryption/decryption and data filtering on the canonical data.

4.2.2 SWIM ATM information services

The ATM information services are all services that can be called on the northbound external interface of a SWIM Node. These are application services and data access services.

Fig. 11 shows a typical scenario: A simple request/reply service invocation results in a data change (on ATM System B) that is followed by a publish/subscribe exchange to distribute the changed data. In this case the ATM information service call is initially triggered by the SWIM Application Adapter on System A. System B is identified as the master of the data and the service call is forwarded to System B where the operation is performed. If the operation results in a data change, this is distributed to all relevant subscribers.

Only the initial service call originates outside of SWIM, all the other information flows are triggered by the SWIM infrastructure. This requires some sort of service orchestration inside of SWIM.

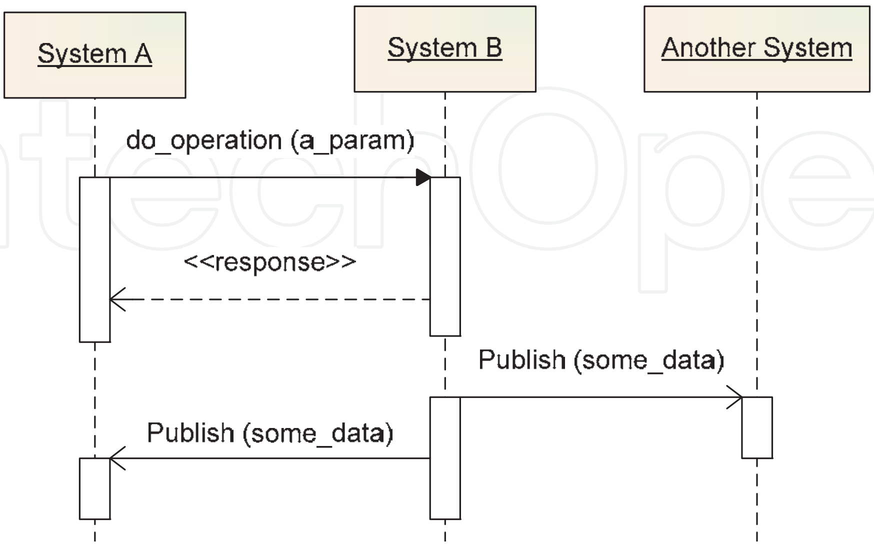


Fig. 11. Interaction Scenario.

4.2.3 SANDRA data domain concept

In order to respect the SWIM Data Domain Concept, described in SESAR as "Profiles", the SANDRA Airborne Middleware (SAM) provides a more generic approach to allow different kind of data sharing/services over SANDRA data links without any kind of restrictions, but just working on XML Optimised Representations of Shared Data.

Fig. 12 below clarifies the Data Domain Object (DDO) concept: as the Flight Object on SWIM Ground Side is described as composed by different Flight Object Clusters with an XML Representation, SANDRA DDO provides a more generic view of the same thing, in particular it is composed by different Data Object Clusters and, with this assertion, we can describe any kind of Data Domain (Meteo, Aeronautical, Flight, etc) using this kind of representation for specific entities.

For example, SANDRA supports the EUROCAE ED133 Flight Object services (EUROCAE, 2009) through the DataDomainObject.

4.3 Improving the air/ground data exchange services

SANDRA will provide a generic representation for the shared data over SAM, in particular, analysing the last image it is possible to discover two kinds of representations for the same information:

- **DataDomainObject (DDOject)**: it contains "n" DDObjectClusters of each with an Object Identifier and multiple Object Release Identifiers described in XML
- **DataObjectSummary**: it contains just one data object cluster the Distribution List cluster that contains information about the participants over the DDObject shared information. A release identifier, contained also in the DDObject, gives information about the actual releases of the single clusters contained into the DDObject.

These two representations are important in order to reduce the traffic between the involved stakeholders: the first one is the full data representation that each involved stakeholder has

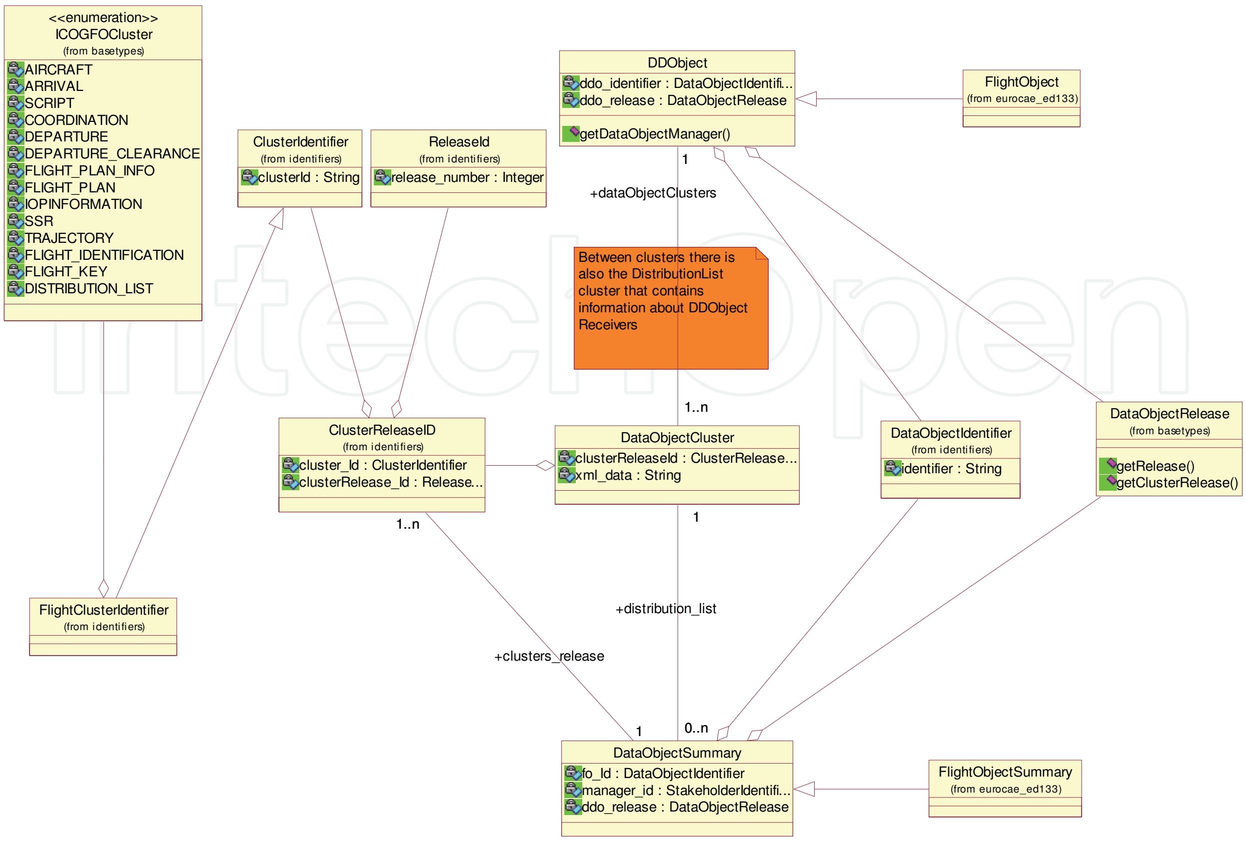


Fig. 12. SAM Data Domain Object.

to save locally; the usually shared information that informs which is the last "present" information is given by the DDObjectSummary.

Once the new AIRBORNE requests a subscription, it receives all the summaries of DDObject of interests (based on particular AIRBORNE StakeholderIdentifier). The SWIM Ground infrastructure may need to be extended in order to identify the objects exchanged to support SWIM A/G communication. For example, it may be useful to link the specific DDObjects (Meteo information, Flights over the trajectory, etc) to a Trajectory for distribution of the DDObjectSummaries of interest to subscribed aircrafts only.

SANDRA targets to reduce the traffic on the Air/Ground data link by designing a more schematic yet lightweight communication protocol. However, a publish operation for distributing the whole clusters information in XML format incurs a large overhead and high bandwidth usage. In order to reduce the amount of shared information, two important operations are applied to the XML format:

- Compression: A number of compression techniques can be used to reduce "drastically" the XML shared information size.
- Data Manipulation: the XML format is extremely schematic and there are techniques that can allow simple updates over an original data.

In considering the above, the requirements for data exchange optimisations are identified as follows:

- A common data representation to optimise the shared data is required
- An XML-based shared data content is required
- A complete knowledge of Shared Information by the ground system is required in order to share essential information starting from an airborne stored one.

To respect these few but extremely important requirements, the DDObject with its inner XML clusters are introduced. Concerning the Ground side DDObjects knowledge, a "DataRepository" component to store all the DDObject Cluster Releases of each DDObject is defined. The "Delta" concept for the differentiation and integration of XML data is applied using the information stored in the "DataRepository".

Based on the SESAR SWIM study, the "publish" and the "read" operations are the two most critical operations to integrate compression and data manipulation concepts into SAM:

- **Applying "Delta" Concept**: Once the DDObject Manager requires an update on an already shared DDObject, it requests a publish operation of the DDObjectSummary that contains the DDObject Identifier, DDObject Release and Distribution Cluster for such information to be received by airborne applications that have already been subscribed to and registered on the DDObject Domain, also given that the DDObject is created and updated. Under this premise, the airborne application knows the "stored" DDObject Release and the new Release Identifier that was just published. Knowing this information the application can request a "read" operation signalling the "stored" release to enable the ground side to compare the release values of the newly published DDObject and the one that is stored by the airborne application. Once the SWIM Ground locates the clusters identifier to update using a DDObject repository that contains all the DDObject Releases clusters, it can be possible to identify the differences between the XML file stored in the airborne application and that in the newly shared clusters. The difference in the XML contents, XML Diff, can then be sent as a reply message to the read operation signalled by the airborne application.
- **Applying Compression Concept**: Compression can be defined at different layers, in particular at the "message" layer and at the "parameter" layer. In order to design the SANDRA Middleware, Web Service technology and, in particular, the shared messages – SOAPMessages to realise the request/reply pattern are applied. Previous studies show that it is possible to compress SOAP messages before sending out the requests and to decompress them once received at the destination. In addition, with the "Delta" Concept, the SOAP message body will contain only a "read reply" message with just the differences between the "Airborne Stored DDObject Clusters" and "Ground Updated Clusters" as parameters. As a result, the SOAP Body contains only the XML Diff content. Applying XML Compression over XML Diff will further reduce the shared packet size.

The same compression can be however applied to the publish operation since all the DDObjects has the same data structure XML Based. Hence, it is possible to compress the publish topics clusters - the whole clusters for DDObjects and just the Distribution Cluster for the DDObjectSummaries before publishing.

4.4 The SANDRA airborne middleware architecture

The SANDRA Airborne Middleware architecture as shown in Fig. 13 has the scope to provide services to airborne applications and to the ground-based Air Ground Data Link Ground Management System (AGDLGMS) gateway over the SANDRA A/G Network. In order to establish such a connection, a set of external interfaces are defined for use in the ground side to communicate with SWIM Ground Gateway and airborne side to realise the communication with SWIM airborne applications.

A detailed SAM system architecture, as illustrated in Fig. 14, includes the AGDLGMS entity residing on the ground is communicating with the other SWIM G/G Gateways via the SWIM Ground Network; and also the SANDRA Airborne Middleware, via the SANDRA A/G Data Link. The SAM provides two kinds of external interfaces:

- **SAMUtilityService Interface:** this interface has to provide the utility services requests such as Subscription, Un-subscription and the set of connected Airborne status
- **SAMBusinessService Interface:** this interface has to provide the business services requests, in particular creation, update, handover, read, and so on.

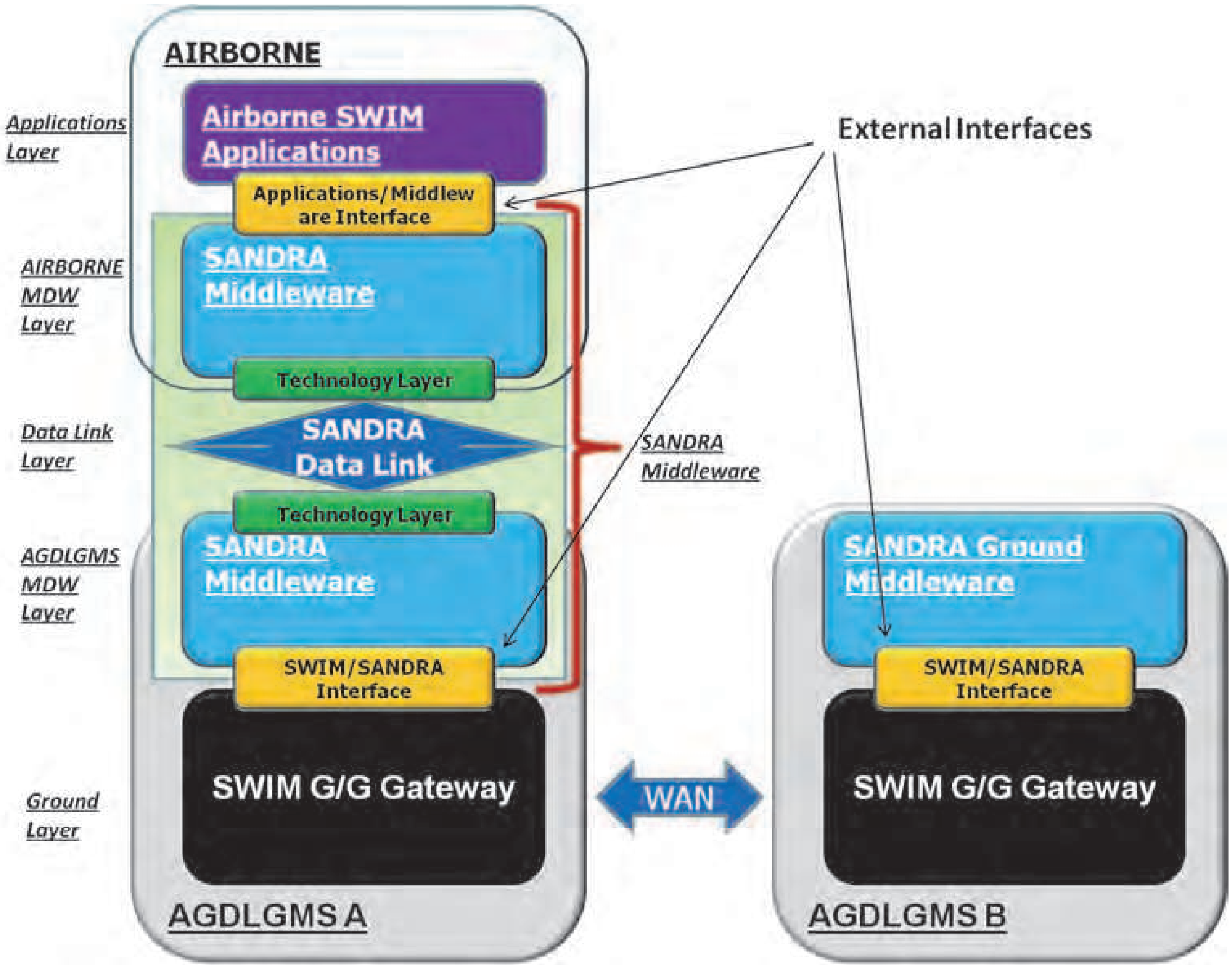


Fig. 13. SAM Architecture Overview.

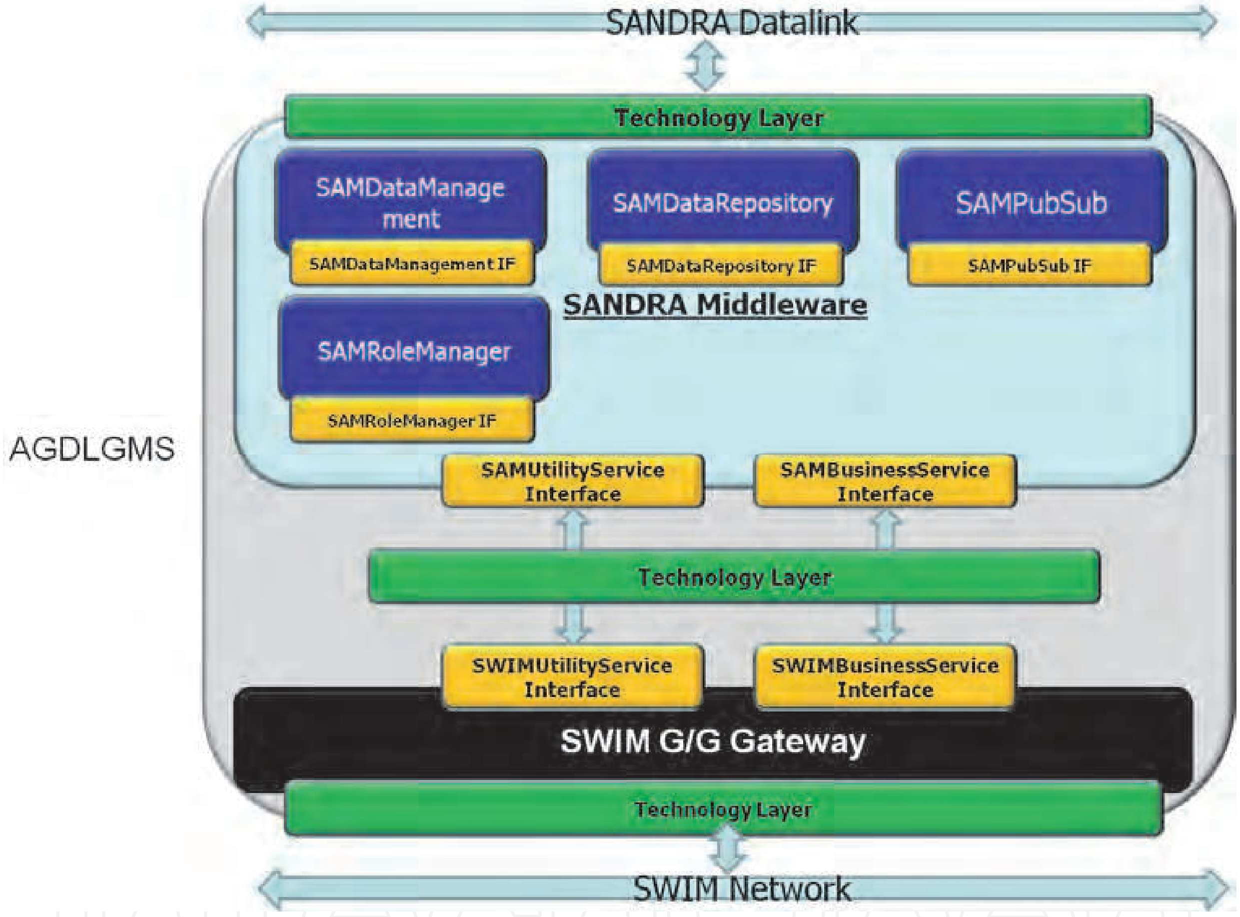


Fig. 14. SAM Architecture Details.

As described in the following paragraphs, both the airborne and ground SANDRA middleware segments should provide the core infrastructural functions to support the aeronautical operations via the defined SAM interfaces. Core services are defined in alignment with the EUROCONTROL and FAA SWIM approach to construct a coherent SWIM infrastructure around the globe.

4.4.1 Interface management

The interface management provides capabilities that enable service providers to publish information about services and service consumers to discover service information. In addition to the generic service registration/publication and discovery, user registration and subscription for notification are also associated supporting functionalities.

The interface management of the SANDRA system should support the manipulation of information from the service registry, shown as follows:

- **Service Registration**: the ability for a user to register a service to the system.
- **Service Lookup**: the ability for a user to look for (discover) a registered service.
- **Service Invocation**: the ability to invoke an identified and possibly distant service.

A service provider registers the service endpoints on a local registry. For routing purposes, the lookup service will be invoked if the service consumer does not know the endpoint of a specific service. However, this also depends on the message exchange pattern being applied. Both Flight Object key selection and content-based search support the service discovery.

4.4.2 Messaging services

The messaging services provided SAM the messaging mechanisms to enable the communication between service providers (publishers) and consumers (subscribers) in a loosely-coupled manner. However, these roles may vary in different scenarios. The decoupling feature of participants not only reduces chances of synchronisation, but also ensures the transparency and autonomy of services. Several mechanisms are introduced here to support a variety of services invocation styles and data exchange protocols:

- **Publish/Subscribe with notification**: allow publishers to check the subscribed users for the actual notification type and send them the subscribed notification.
- **Request/Response**: allows service client to request information from the server through for example, RPC-style client/server communication.

The adoption of Web Services and JMS allows both synchronous and asynchronous messaging for the realisation of various aeronautical operations across all data domains.

4.4.3 Security services

The security services offered by SAM focus on aspects such as user role management, access control, the encryption and decryption of messages as well as service security monitoring. The Lightweight Directory Access Protocol (LDAP) user registry is considered in the design phase to provide security functions.

A common approach is introduced to allow service consumers to gain restricted access only on specific tasks by building a standalone authentication and user management application that can be accessed as a service in a SOA framework. Operational services/ processes that require proof of identity will access this shared user account; thereby the shared function can be repurposed across all tasks. In this scenario, the shared user account service will be reusable for different business processes. This includes the following aspects:

- **Authentication**: this function provides password features for user login.
- **Authorisation**: this function provides user access control.
- **Logging**: this function stores the user actions in a security log and alerts unauthorised operations according to the security requirements (SANDRA, 2010).

4.4.4 Enterprise service management

Enterprise Service Management (ESM) includes the governance and monitoring of services. It provides passive and active management of services at run time while performing overall management operations of systems and applications.

SAM should also provide core functions to collect the Service Level Agreement (SLA) and policy-related metrics to ensure QoS. An SLA refers to a set of performance figures that are indicated in the service contract shared by a service provider and the consumer(s) of that service. Guaranteed operation time, i.e. system uptime, downtime and response time, as well as capacity of the system are the most common figures. The topics covered are:

- **Fault monitoring and reporting**: with this function the administrator can monitor and manage the service faults.
- **Performance monitoring and reporting**: with this function the administrator can monitor the performance level of the services.
- **Enterprise configuration**: with this function the administrator can perform system configuration (e.g. service setup and software configuration) through GUI.
- **SANDRA service management**: with this function the administrator can monitor the quality level of services, according to the QoS requirements (SANDRA, 2010).

4.5 Technology choices

A number of technologies are proposed for the realisation of the SANDRA Airborne Middleware in a service-oriented architecture. Evaluation of the suitable technologies is performed based on the three following steps:

- The propose of candidate technologies
- A list of evaluation criteria taking into account the system requirements and communication patterns of SWIM, SWIM-SUIT and SANDRA Airborne Middleware (SAM)
- Technology selection based on the rating of each technology against the quality attributes to define a technology selection matrix

In summary, it is envisaged that Web Service is the one of the most appropriate technologies in future aeronautical communication considering its standardisation, interoperability and compatibility with the ground SWIM infrastructure. It is also suitable in the airborne context for its maturity and integration features. Web services provide the support for the realisation of a SOAP-based SOA infrastructure incorporating the MOM JMS API for providing pragmatic request/reply and publish/subscribe messaging mechanisms. The Enterprise Service Bus is considered as an alternative approach for system implementation in addition to Web Service and MOM/JMS. DDS outstands for its excellent network performance. Therefore, it is envisioned as an emerging candidate technology for high-efficient and technology-independent data distribution along the refinement of its specification. The SWIM-SUIT project has also concluded with the same prospect and adopted DDS for the SWIM-BOX implementation on ground.

The evaluation of the implementation software indicates that Apache and JBoss frameworks are respectively the most and second-most appropriate technologies. The scores are obtained based on the aggregation of a Web service infrastructure, a messaging backbone (MOM) and an ESB for service composition and system integration. Here, the following sets of technology options are proposed:

- Apache CXF, Apache ActiveMQ, Apache Camel, and Apache ServiceMix or Apache Synapse - Apache CXF and ActiveMQ compose a lightweight system framework for onboard systems. Apache Camel can be used for advanced routing and mediation functions if required. Apache Synapse can be used as a replacement of ServiceMix to achieve a more lightweight system. System bridging is required for achieving full interoperability with SWIM-SUIT.
- JBoss Application Server/JBoss ESB, and HornetQ - direct integration with SWIM-SUIT, thus to achieve full interoperability. HornetQ is claimed to have better performance among all message brokers available.
- XOperator or XMLUnit - used for comparing, summarising and synchronising XML documents for the realisation of the "Delta" data sharing in the SWIM context, i.e. to support operations such as "getDataDiff", "makeDataDiff" and "integrateDataDiff" in the "SAMDataRepository" and "SAMDataManagement".
- FastInfoset and Fast Web Service - used for converting test-based XML into binary-based ASN.1 FastInfoset provided by the JAX-WS interface within Java framework while remaining the capability to allow message transmission using SOAP. No additional tool is required for this approach. Although the support for Fast Web Service has yet become sufficient, the approach is envisioned for SAM development.

5. Conclusion

With an appreciation of envisioning the future aeronautical communication system in a service-oriented architecture, this chapter summarises the ongoing development of avionic service integration and the utilisation prospects of the middleware and SOA concepts in the 2020 timeframe and beyond. SWIM, as an architectural blueprint, is envisaged as a baseline solution in dealing with the growth of airline passenger and the increasing complexity of aeronautical operations. The state-of-the-art SWIM system architectures proposed by world-leading aviation organisations, e.g. EUROCONTROL and FAA allow Air Traffic Management stakeholders around the globe to obtain coherent and persistent knowledge of relevant system data regarding the flights in operation.

To continue with the ground-based SWIM programmes, this chapter proposes a possible SANDRA Airborne Middleware approach for the integration of airborne/ground Air Traffic Management systems for SOA-based future aeronautical service integration. Analysis of airborne/ground information exchange targeting on the domain-based operations and interactions are defined according to the added-value services, data access services and technical services in the SWIM context. An architectural overview of the SANDRA Airborne Middleware is illustrated featuring the connections between the onboard middleware and the SWIM Ground Gateways with its underlying system infrastructure. The "Delta" concept and compression mechanisms are proposed to optimise system performance and message exchange based on the selected technologies.

6. Acknowledgment

The research leading to these results has been partially funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement n° 233679. The SANDRA project is a Large Scale Integrating Project for the FP7 Topic AAT.2008.4.4.2 (Integrated approach to network centric aircraft communications for global aircraft operations). The project has 31 partners and started on 1st October 2009.

In addition, as an acknowledgement from the authors, the chapter represents only the opinion of the authors but not the whole SANDRA Consortium.

7. References

AIXM. (2011) AIXM Aeronautical Information Exchange Model, 12.12.2010, Available from http://www.aixm.aero/public/subsite_homepage/homepage.html

Andrews, T.; Curbera, F.; Dholakia, H.; Goland, Y.; Klein, J.; Leymann, F.; Liu, K.; Roller, D.; Smith, D.; Thatte, S.; Trickovic,I. & Weerawarana, S. (5th May 2003). Business Process Execution Language for Web Services Version 1.1, 24.11.2010, Available from http://xml.coverpages.org/BPELv11-May052003Final.pdf

Campbell, A.T.; Coulson, G. & Kounavis, M.E. (1999). Managing Complexity: Middleware Explained, *IEEE Computer Society IT Professional*, vol. 1, September | October 1999, pp. 22-28.

EasyJet & Ryanair. (2009). *Ryanair and EasyJet continue strong growth, British Airways' struggles continue in Oct-2009*.

Erl, T. (2009). *SOA design patterns (Ed. 1)*, Prentice Hall PTR, ISBN: 9780136135166, New Jersey, USA.

EUROCAE. (2006). *D7 Flight Object Interoperability Proposed Standard (FOIPS) Study. Issue 1.4.*

EUROCAE. (2009) *ED-133 Flight Object Interoperability Specification.*

EUROCONTROL. (2006). *Long-Term Forecast Flight Movements (2006 - 2025) v1.0*, EUROCONTROL.

EUROCONTROL. (2008). What is SESAR WP 14 SWIM technical architecture, 21.10.2010, Available from http://www.swim-suit.aero/swimsuit/project.php

EUROCONTROL. (2011). Weather Information Exchange Model, 12.12.2010, Available from http://www.eurocontrol.int/aim/public/standard_page/met_wie.html

FAA. (28th April 2010). SWIM Documents, 11.11.2010, Available from http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/atc_comms_services/swim/documentation/

GEIA. (March 2008) FAA *SWIM Program SOA Best Practices - Industry Input*

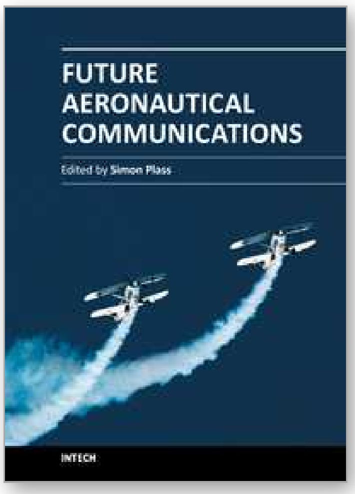
Houdebert, R. & Ayral, B. (2010). Making SWIM Interoperable between US and Europe, *Integrated Communications Navigation and Surveillance (ICNS) Conference Proceedings of IEEE*, Herndon, Virginia, USA, May 11-13, 2010.

ICAO. (2010). Operational Opportunity. *ICAO Environmental Report 2010*, Chapter 3, pp.96-125

Luckenbaugh, G.; Landriau. S.; Dehn, J. & Rudolph, S. (2007). Service Oriented Architecture for the Next Generation Air Transportation System, *Integrated Communications Navigation and Surveillance (ICNS) Conference Proceedings of IEEE*, Herndon, Virginia, USA, May 1-3, 2007.

Mahmoud, Q. H. (2004) *Middleware for Communications (Ed. 1)*, John Wiley & Sons, Inc., ISBN: 0470862068, New York, USA.

SANDRA. (June 2010) *Deliverable D3.1.1 SANDRA Detailed Network Requirements v1.0*.
Simon, R. (2003) Middleware, In: *The Internet Encyclopedia*, Bidgoli, H., vol. 1, 603-613, John Wiley & Sons, Inc. ISBN: 978-0-471-22202-6, New Jersey, USA.
SWIM-SUIT. (June 2008) *Deliverable D1.5.1 Overall SWIM Users Requirements v01*.
W3C. (2004) Web Services Architecture, 11.02.2004, Available from http://www.w3.org/TR/ws-arch/



Future Aeronautical Communications
Edited by Dr. Simon Plass

ISBN 978-953-307-625-6
Hard cover, 378 pages
Publisher InTech
Published online 26, September, 2011
Published in print edition September, 2011

There are well-founded concerns that current air transportation systems will not be able to cope with their expected growth. Current processes, procedures and technologies in aeronautical communications do not provide the flexibility needed to meet the growing demands. Aeronautical communications is seen as a major bottleneck stressing capacity limits in air transportation. Ongoing research projects are developing the fundamental methods, concepts and technologies for future aeronautical communications that are required to enable higher capacities in air transportation. The aim of this book is to edit the ensemble of newest contributions and research results in the field of future aeronautical communications. The book gives the readers the opportunity to deepen and broaden their knowledge of this field. Todayâ€™s and tomorrowâ€™s problems / methods in the field of aeronautical communications are treated: current trends are identified; IPv6 aeronautical network aspect are covered; challenges for the satellite component are illustrated; AeroMACS and LDACS as future data links are investigated and visions for aeronautical communications are formulated.

Yifang Liu, Yongqiang Cheng, Yim Fun Hu, Prashant Pillai and Vincenzo Esposito (2011). SOA-Based Aeronautical Service Integration, Future Aeronautical Communications, Dr. Simon Plass (Ed.), ISBN: 978-953-307-625-6, InTech, Available from: http://www.intechopen.com/books/future-aeronautical-communications/soa-based-aeronautical-service-integration

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821