White paper: digital transformation for ATM control centre applications

Providing seamless integration to boost efficiency, capacity and productivity

In all safety-critical industries, and in the air traffic control (ATC) domain in particular, the lack of integration between existing monolithic applications prevents controllers from reaching their full potential. The need to switch between multiple user interfaces and execute complex memorised workflows distracts operators from core tasks and delays decision making.

As air traffic continues to increase, ATC organisations want to enable existing personnel to safely manage more workload. However, the barriers between legacy applications are a significant obstacle. To achieve the required digital transformation, monolithic applications need to give way to flexible microservices that work together, backed by an ecosystem of services. This will lower integration costs, allow services to scale with growing flight traffic, and cut the cost of deploying and operating control centre environments.

Air Traffic Management

Overcoming integration challenges

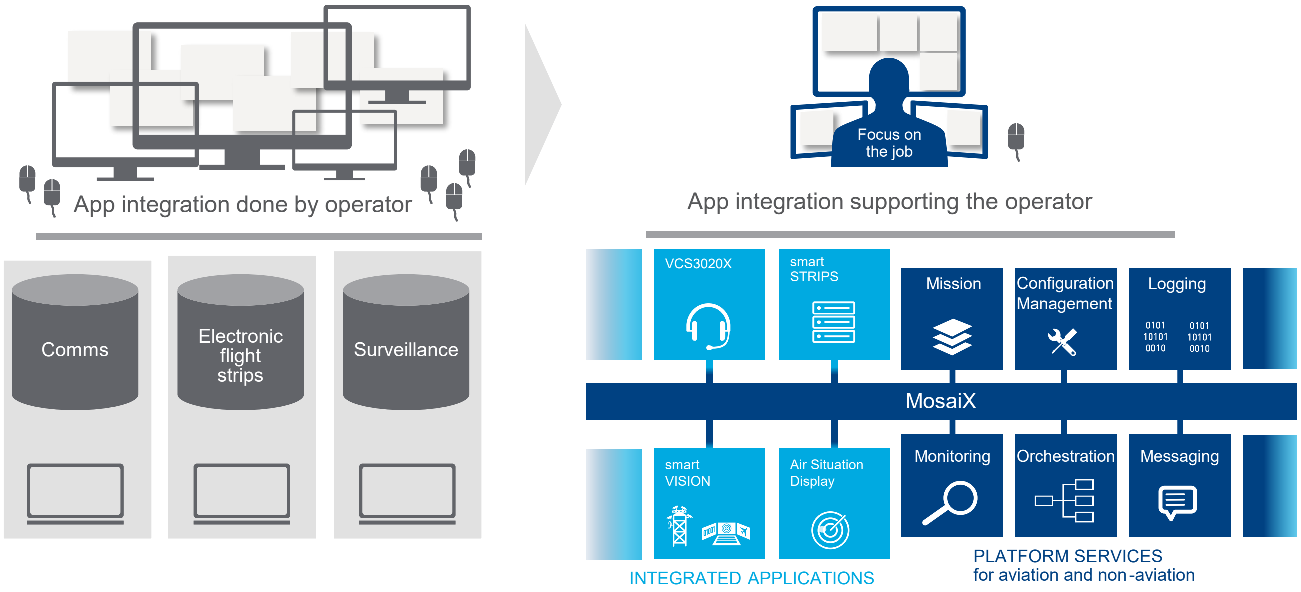
The lack of integration between existing systems deployed by ATC organisations is a common challenge. The root of the problem is the use of monolithic applications: self-contained software applications in which a single tier of code is responsible for all functions. Designed without any modularity, monolithic applications are inherently inflexible and hard to maintain, because changes to functionality or the user interface cannot be made in isolation. The underlying source code is effectively a single block of self-referential logic, so a change in one part may require multiple corresponding changes in other parts of the code. Application silos might be, for example, Communication, Electronic Flight Strips and Surveillance (see Figure 1).

When an organisation operates several monolithic applications alongside each other, the challenge is compounded: patches or configuration changes may need to be applied in parallel in each separate application.

The microservices architectural style addresses these challenges by breaking down monolithic applications into a modular suite of services, each designed to execute a single task and to communicate with other services. Services can be written in different programming languages and run on different platforms, yet work together as a logical entity to fulfil the required business function. Existing services can be combined in new ways to fulfil other business functions, significantly reducing development effort. If demand on a particular service rises, new copies can be automatically deployed without the need to scale any of the other services. Similarly, resilience can be ensured by automatically starting replacement services in the event of a failure.

Since each microservice is entirely independent of every other microservice, they can be developed, updated and patched independently, such that the development team responsible for a given service is not constrained by the lifecycle demands of any other development team. Equally, in safety-critical industries such as ATC, software assurance is simplified because patches and code updates are limited in scope.

Figure 1: Application integration by the system for focus on operative challenges



Enhancing situational awareness

In all safety-critical industries, controllers work across multiple different systems, moving from application to application while following memorised workflows. Maintaining the required situational awareness is challenging in such an environment, because systems may present duplicated and / or inconsistent information. To ensure that controllers can clearly understand events and status in the outside world, organisations need to deliver on four objectives and shift from application integration done by the operator to integration done by the system paradigm (see Figure 1).

1. Single-screen focus

A typical controller working position in air traffic management (ATM) may have five screens and multiple input devices, requiring the controller to track information changes happening outside their field of vision. Organisations need to bring applications together on fewer screens, to improve focus on critical information and create a more ergonomic working environment.

2. Zero application barriers

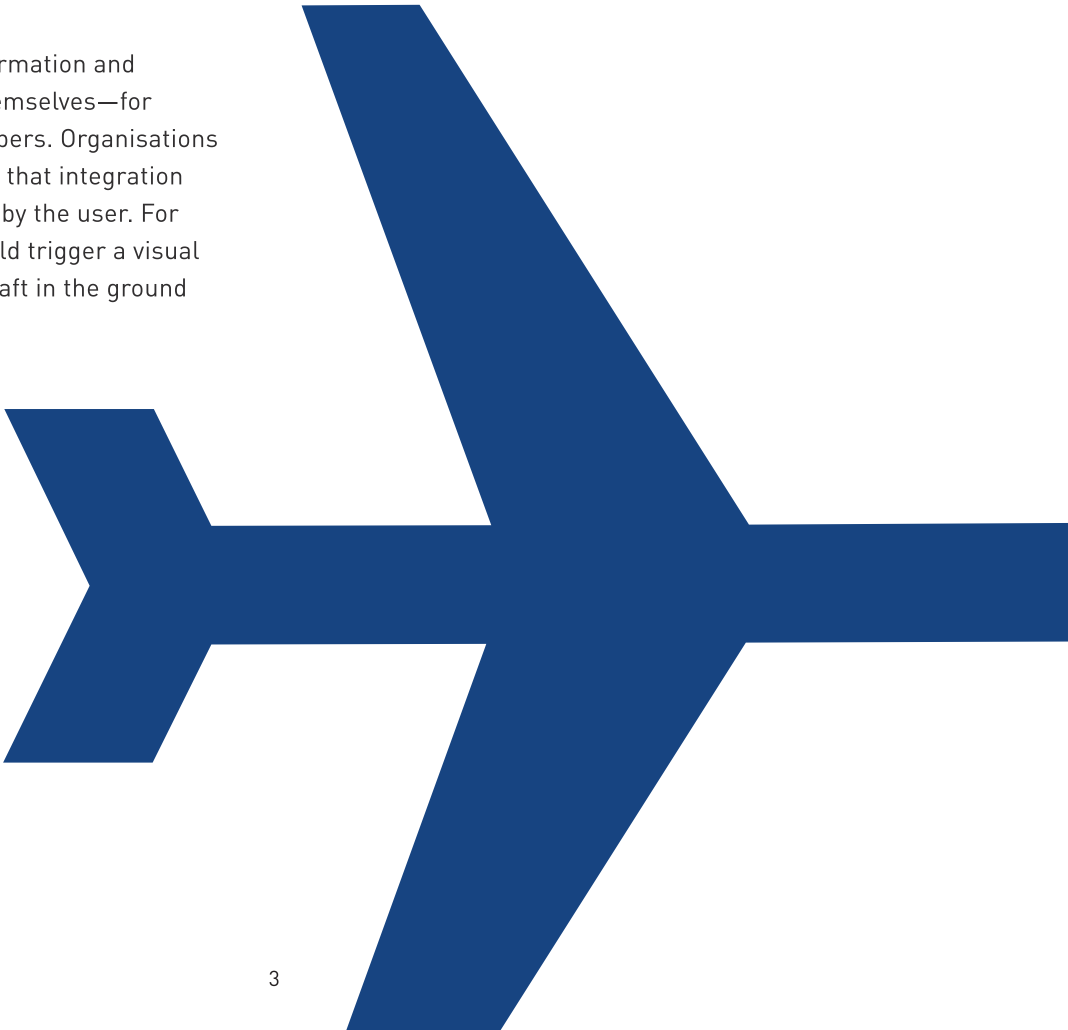
Controllers typically need to link information and processes across applications for themselves—for example, by searching for flight numbers. Organisations should assist controllers by ensuring that integration is handled by the system rather than by the user. For example, selecting a flight strip should trigger a visual indication of the position of that aircraft in the ground surveillance application.

3. One-touch role change

In traditional environments, each application has its own separate mechanism for mission management, so switching between missions requires an interaction in every single application. To avoid this distraction and additional workload for controllers, organisations should ensure that all applications share a common mission service. This means that selecting a new mission in one application will trigger the other applications to activate the same mission automatically.

4. Spotlight on essential information

At controller working positions, that depend on disparate systems, controllers see the same information multiple times across different screens. Organisations need to coordinate application displays, eliminating redundant information and helping users focus on the most important details. Equally, operators should be empowered to create an on-screen working environment that meets their precise needs.



Increasing business flexibility

In addition to improving the experience for controllers, organisations in safety-critical domains such as ATC are under pressure to increase flexibility and speed of response to new business requirements. At the same time, they are expected to remain within strict budgetary limits. To fulfil these objectives, organisations should ensure that their control centre systems meet the following criteria:

1. Modular scalability

To support for multiple centres—including remote and virtual towers—organisations should be able to deploy any chosen subsets of functional modules rapidly and at low cost. From central location, organisations should have the ability to deploy and manage multiple sites to face today's operational challenges, for example, for contingency or optimal network bandwidth utilisation.

2. Industrialised operation

To keep capital and operational costs low, organisations should choose solutions that run on common off-the-shelf (COTS) server and network equipment. By choosing centralised deployment of container-based microservices, organisations can radically simplify the roll-out and maintenance of distributed systems. Ongoing IT maintenance will be simpler and faster, and the use of industry-standard rather than proprietary technologies will make it easier to access the right technical skills.

3. Support for hot desking

It should be possible for any operator to access personalised role-based services through any working position, enabling users to move freely between locations within the same centre or even different centres. In addition to enabling greater flexibility for staff, such an approach eliminates local dependencies, thereby improving the speed of recovery in the event of needing to restore systems following a natural disaster or similar major outage.

Simplifying technical maintenance

The most significant element in the operational cost of control centre solutions is typically the ongoing monitoring and maintenance. With an eye on reducing long-term costs, organisations should consider deploying solutions with the following characteristics:

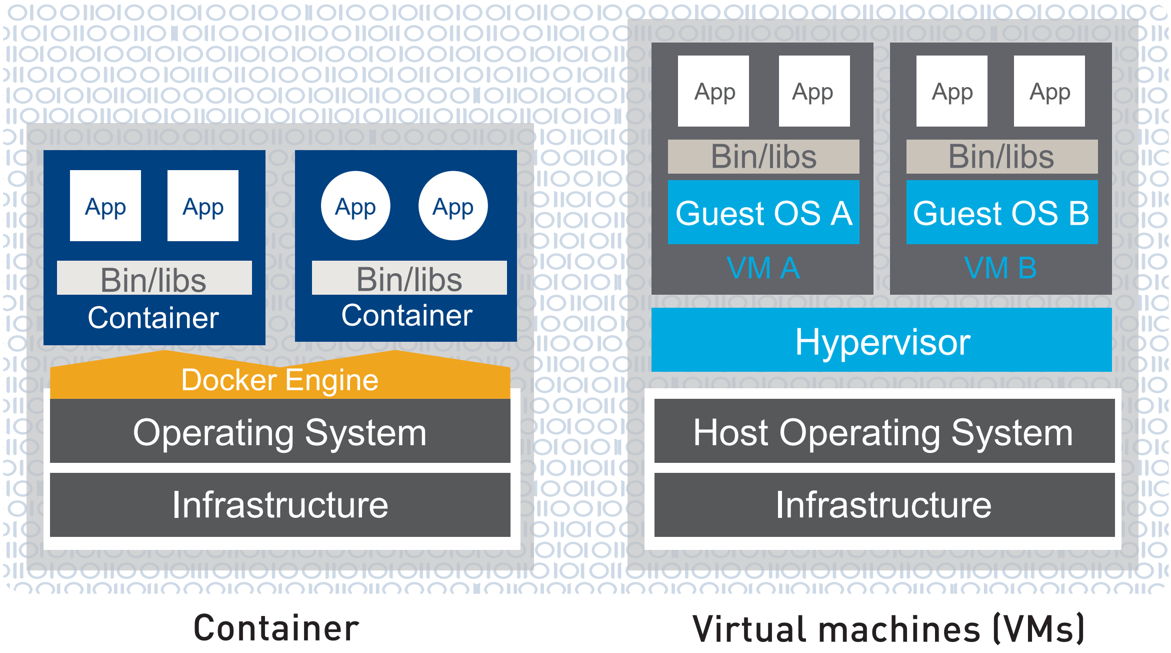
1. Central configuration and monitoring

There should be a single point of control for applications, with shared parameters between them. Web-based monitoring should provide at-a-glance views of the status of hardware and software services on a system schematic, with drill-down capabilities for deeper analysis. To simplify trouble-shooting, all logs should be collected and stored in a central logging system. Centralising software deployment should also help protect systems against vulnerabilities, by enabling rapid deployment of the latest patches.

2. Software container technology

By deploying solutions based on microservices running on Docker container technology, organisations can avoid the need to have each component hosted on a full virtualised operating system. Instead, the container for each application will include only the required elements, and the Docker engine will ensure process independence (see Figure 2 on page 5). This approach also helps accelerate software patching and updates, because organisations can selectively update just the changed containers. Containers are isolated from each other and their resource utilisation is strictly limited, so a problematic update in one container does not impact others.

From the software-assurance perspective, the container concept makes it possible to host software with different assurance levels on the same hardware, as the strong boundaries between containers ensure safety partitioning: the software in any given container can access only a limited subset of physical system resources.



Container

Virtual machines (VMs)

Figure 2: Compression of software layers used in container and virtual machine approach

Security

In safety-critical industries such as aviation, effective cyber security is naturally a vital consideration. Solutions must be designed from the ground up to support national, regional, and global standards in security and compliance. When planning security, organisations should pay close attention to the following topics:

1. System architecture supporting safety and security

Protection zones are defined as a collection of hardware, software and personnel with a common trust level.
In many cases, three different protection zones with sufficient isolation between them is adequate: The internal zone with no direct connections to other systems, the shared zone with connections to other "trusted" networks and the public zone with connections to a non-trusted environment (e.g. public network). In a simplified form, safety-critical functionality is located in the internal zone and security best practises focusing on safety are applied here, while functionality requiring high connectivity is located in the public zone and IT security best practises focusing on data protection are applied in this zone.

Systems operated by an end-user are usually composed of a number of subsystems from different vendors.
When Frequentis delivers a system, it usually comprises different protection zones with adequate isolation between them. When such a subsystem is integrated into the overall system on site, it is important to respect the defined protection zones and to connect networks and interfaces only as foreseen to trusted or non-trusted environments. Security needs to be ensured on a system level in order to ensure security also on the subsystem level and this is a responsibility of the system operator.

To enhance security of a subsystem inside the perimeters which are separating the different protection zones, the principle of "complete mediation" may be applied. This principle says, that "every access to every object must be checked for authority." When this principle is applied, not only users are authenticated and authorised when they log in, but authorisation also takes place inside the system itself between individual software services every time they exchange information with each other.

This concept is important if a system is distributed and it is hard to define reliable perimeters: the paradigm in this case would be: “trust no network”. To make this principle compatible with safety requirements—in case of a failure the reaction could be alerting only, or alerting plus blocking after a defined grace period, or alerting and immediate blocking. The applied mechanism depends on the protection zone.

2. Lifecycle-security as a process

State-of-the-art technical systems must be designed for safety and security from the beginning. A secure development lifecycle covers the phases design, development, integration, verification, and release. A security architecture is defined based on assumptions on the later operational environment of the system; security requirements are defined, implemented during development and integration, and tested. During the release phase, the responsibility for keeping the system secure is moving from the vendor to the operator of the system.

In the maintenance phase, the system operator needs to establish a security governance and security processes for keeping the system secure during its lifetime and system vendors provide the required support.

The activities to be done during operation can be broken down into four categories:

- Risk management and governance
- Protection
- Defence
- Resilience.

Flexible integration platform

By adopting a solution based on a microservices architecture, ATC organisations can deploy and manage applications independently of each other, and freely integrate existing or future applications—regardless of vendor—on the same platform. The goal is to ensure that applications are ready for industrialised data centre operation, so that deployment and maintenance can be radically simplified through centralised software distribution and monitoring, and shared configuration data.

Building on a flexible integration platform will transform ATC voice and automation applications into an intelligent digital ecosystem, enabling dynamic connections between people, processes, and things. In such an environment, adding a new application is as simple as deploying new container-based instances of microservices within the existing framework. The core framework provides the operating system, message bus, monitoring and control, logging, and configuration services—acting as the cement between the bricks of the domain-centric applications deployed on top.

With a containerised microservices approach that enables ATC functionality to be managed in a fully modular fashion, organisations are free to focus on supporting their controllers to work more efficiently and effectively.

SWIM and Unmanned Traffic Management (UTM) are now on the horizon. It is no longer just a matter of if stakeholders will need to support SWIM standards, but when they will need to do so. In addition to lowering integration cost and moving from point-to-point communications, using SWIM as the basis for digital transformation will help provide new services that leverage the latest technologies to deliver substantial increases in airspace efficiency.

In addition, unmanned aviation will become a major business factor, and the advent of an ecosystem of SWIM services and standardised, structured information is a key prerequisite for successful collaboration between ATC and UTM.

One thing is certain: traditional ATC systems are not able to manage UAS traffic, nor are traditional weather reports or flight plans able to fulfill the needs of unmanned aviation. UAS missions, microweather and UAV deconfliction are just some of the challenges that require mediation, communication and integration between two very different worlds. SWIM is the foundation for the successful integration of UTM and ATM.

Next-level cost efficiency

In safety-critical domains in general, and in ATC in particular, limited integration between existing applications distracts operators from core tasks and delays decision making. With air traffic on the rise, personnel need to be empowered to safely manage more workload. To achieve the required digital transformation, existing monolithic applications must be replaced by flexible shared microservices.

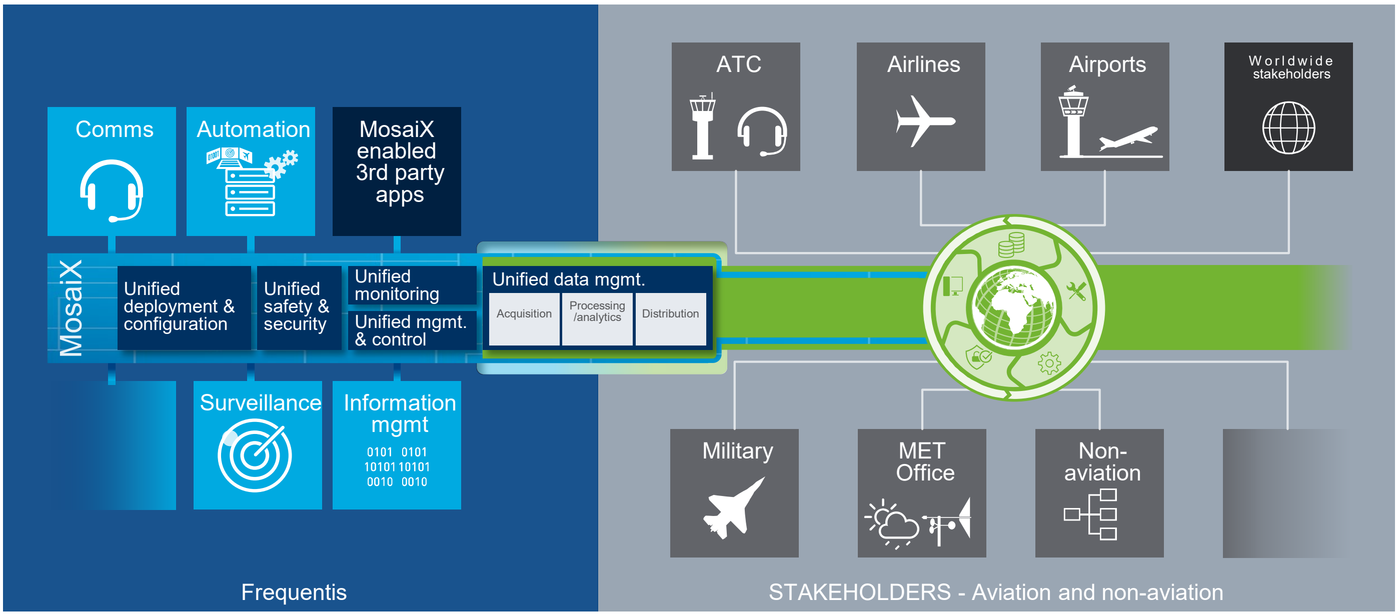
The MosaiX integration platform from Frequentis is designed to address these needs, supporting modular distributed systems with deep integration of voice and data applications for automated workflows.

The platform unlocks fresh value from the entire Frequentis portfolio, enabling faster and simpler deployment and management, rapid scalability to meet new demands, improved operational efficiency, increased standardisation, and reduced costs.
Based on its long history as a strategic supplier in the ATM domain, Frequentis fully understands the importance of safety and has built MosaiX adhering to EUROCAE ED-153 software assurance level 3 (SWAL3) and EUROCAE ED-109A assurance level 4 (AL4).

Initially launched for ATC applications, the platform can also bring benefits in Public Transport, Defence, Maritime, and other safety-critical industries.

With native support for VoIP and real-time voice processing capabilities, MosaiX enables integrated controller operations that empower organisations to better execute their core missions without adding headcount. Combining all control centre tasks on a single display is now a real possibility with Frequentis MosaiX, enabling organisations to step into the future today.

Figure 3: MosaiX — Frequentis digital ATM platform, ATM-grade availability — real-time enabled



FREQUENTIS AG
Innovationsstraße 1
1100 Vienna, Austria
Tel: +43-1-811 50-0
www.frequentis.com

The information contained in this publication is for general information purposes only. The technical specifications and requirements are correct at the time of publication. Frequentis accepts no liability for any error or omission. Typing and printing errors reserved. The information in this publication may not be used without the express written permission of the copyright holder.