D4.1 DEFINITION OF DEPLOYMENT SCENARIOS

Secure Open Collaboration Framework powered by Artificial Intelligence

30 Jun 2023

Abstract

This document outlines a project aimed at developing a data platform to enhance airport management by integrating data from multiple stakeholders and employing modern technologies such as AI and predictive analytics. The platform focuses on improving operational efficiency, energy management, and security, utilizing advanced tools and secure data-sharing protocols to create a cohesive and adaptable system for all airport operations

Contents

1. Summary

Secure Open Collaboration Framework powered by Artificial Intelligence (a.k.a SOCFAI) is a project that aims to gather data from numerous stakeholders contributing to airport management and make the collected data accessible to other stakeholders. In addition to that, with the advancements of artificial intelligence and predictive solutions, the extended goal is to provide more accurate data to the stakeholders and, more possibly, to the passengers to foresee operational problems that may occur at the airport, energy consumption cost-saving, and lessening the operational delays.

2. Introduction

Airports are crucial facilities for cities, even for the countries and governments which is requiring the highest levels of both software and physical security altogether. Given the large number and complexity of operations, it is reasonable to have multiple stakeholders involved-in those operations. Stakeholders' different type of abilities and different areas of their expertises to manage operational zones effectively is a key component of smooth airport operations. However, with advancements at the field of technology and it's capabilities, requires gathering data from more sources in order to reduce, even, to eliminate the operation promlems. Long story short, the idea is to advance the managements of airports to a data-centric model.

Stakeholders and the software systems used to manage airports often consist of legacy-but-proven components, modern but not up-to-date components or totally contemporary ones. Due to the lack of a unified and standardized data sharing and saving protocol, different teams have to spend time for each integration, understand the dependant system and find hard-to-maintain solutions for its specific kind of reqirements, plus, agreeing on data sharing and storage methods, and choose the most suitable technologies and etc, lots of challenging problems and fragile solutions. Furhermore, it can degrade the motivation of analyzer and developer as they have to struggle with tons of non-sense issues instead of focusing on a better approach.

The platform we aim to develop is seeking to bring various stakeholders and experts which are taking role on management of the airport, in order to normalize the collected data in a contemporary way, store them on contemporary storages, to pull them in an easy and resilient way using a centric big data platform, and make it accessible to each others, and more.

This platform will employ modern security, encryption, and sharing methologies to achieve aforementioned topics. Additionally, by combining data obtained from these sources with modern technologies such as image processing, natural language processing, generative AI and so on to make airports more efficient, secure, and continuously operational without interruptions.

And the logistics platform at port is being achieved through convergence technology linked to ICT-based artificial intelligence, and related research and development is underway in many places. Thus, this Port platform in SOCFAI project is promoted as a process to promote the implementation of a digital port logistics system, through the development of solutions and services that enable data sharing and information flow throughout the port logistics supply chain. and optimize the efficiency of the entire port logistics supply chain flow.

3. Platform Security

As mentioned earlier, airports hold critical importance for nations. Therefore, all operations in these areas are developed using methods that prioritize security. For the platform we aim to develop, security is ,unavoidably, at top priority. Based-on the feedbacks collected from market and domain experts, security of the applications used to operate airport start from the network security and the security of the networks divided into two completely distinct layers:

- **Aviation Network:** This network hosts the most critical applications related to aviation, where essential operations concerning flights and aircraft are conducted. Access to this network is limited both physically and through software, and only authorized operators are permitted to access.
- **Airport Network:** This network hosts systems related to airport management and the software of stakeholders, not directly involved-in flight or aircraft management but also takes a crucial role for the hitchless airport operation like cleaning and climating systems and etc.

  Accessibility of those systems are handled separately based-on specific kind of requirements of the component. For example, climate control systems are positioned within this network as a closed sub-network with no external access, while the public WiFi network has been positioned is accessible to everyone, including passengers. In this layer, each system operates with individual security measurements according to its importance, instead of a fully isolated network-based security approach.

The platform is planned to be located on the Airport Network for several reasons:

- Hosting the platform on the Aviation Network could pose a serious security vulnerability if unauthorized access occurs due to unforeseen circumstances.
- Since the platform will only collect data from all systems and could delegate the task of writing data to the discretion of the stakeholders using the platform, it will focus on the security of the data stored rather than ensuring the security of all systems.
- The evolving cloud solutions allow tremendous amounts of the GPUs (Graphics Processing Units) with different power and capacities for our use with competitive prices. It is not efficient in terms of time and cost to position all those hardwares at the airport. Therefore, making the platform accessible for solutions to be created in cloud environments by providing necessary security protocols makes the Airport Network the most reasonable option, given that the Aviation Network is a network where no access from the outside layer to the inside is allowed.
- Greater flexibility regarding changes and additions allows the platform to remain open to development.

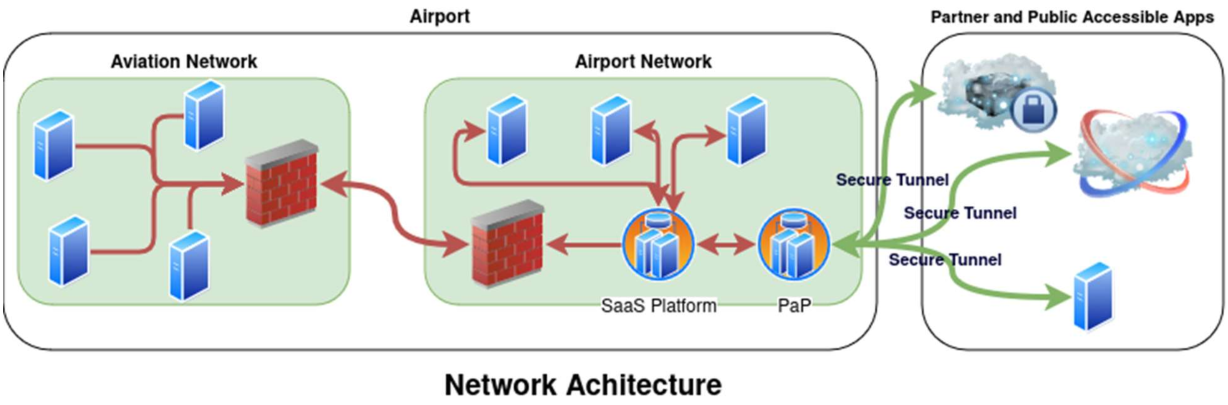


*Figure 1: Network Architecture of **SOCFAI** Platform*

For these reasons, the Airport Network layer seems the most suitable choice for positioning the platform.

3.1 Additional Component: Partner Access Platform (PaP)

Apart from the main platform, an additional component called the "Partner Access Platform (PaP)" will be added to allow stakeholders to connect and perform tests, and synchronize data with external cloud environments when necessary. The need for this platform arises from the following reasons:

- To eliminate the factor of individual error regarding authorized access since the platform, even in read-only mode, will have access to components on the Aviation Network.
- To prevent the overall system performance from being affected when a resource-intensive operation is running.
- To facilitate the secure return of data processing workflows carried out in cloud environments back to the platform.
- To allow stakeholders to safely perform final tests on the PaP system before deploying their applications onto the live system.

Considering these security requirements, the network architecture described above has been proposed. In Figure 1, a the large cluster on the left-hand side shown, labeled "Airport" is corresponds to all the applications from small to large, run by airport stakeholders. The right-hand side represents the applications needing cloud resources and facilities to use or global accessibility are positioned using secure tunnels. The direction of the arrows between the systems indicates whether access will be unidirectional or bidirectional.

4. Data Collection

As mentioned in the introduction, the applications we would like to collect data may have modern or outdated components. Unless data collection from modern applications is relatively more easy and reliable, collecting data from applications with outdated components are the most challenging part for the platform. Especially protocols like ARINC424, MATIP, TELEX, and X.25, which are defined for specific applications and requirements, make it difficult to normalize the date and integrate the relevant data into the platform under development. Furthermore, while modern methods such as streaming primarily aim to notify changes at the source of change and propagate it to the replication protocol to make data replication and integration easier, outdated methods aim to send data to more specialized targets, requiring special effort and processes. Considering that access to these systems may also be limited, the process involves many challenging factors. Therefore, a broad set of tools will play an important role in easing the data collection process.

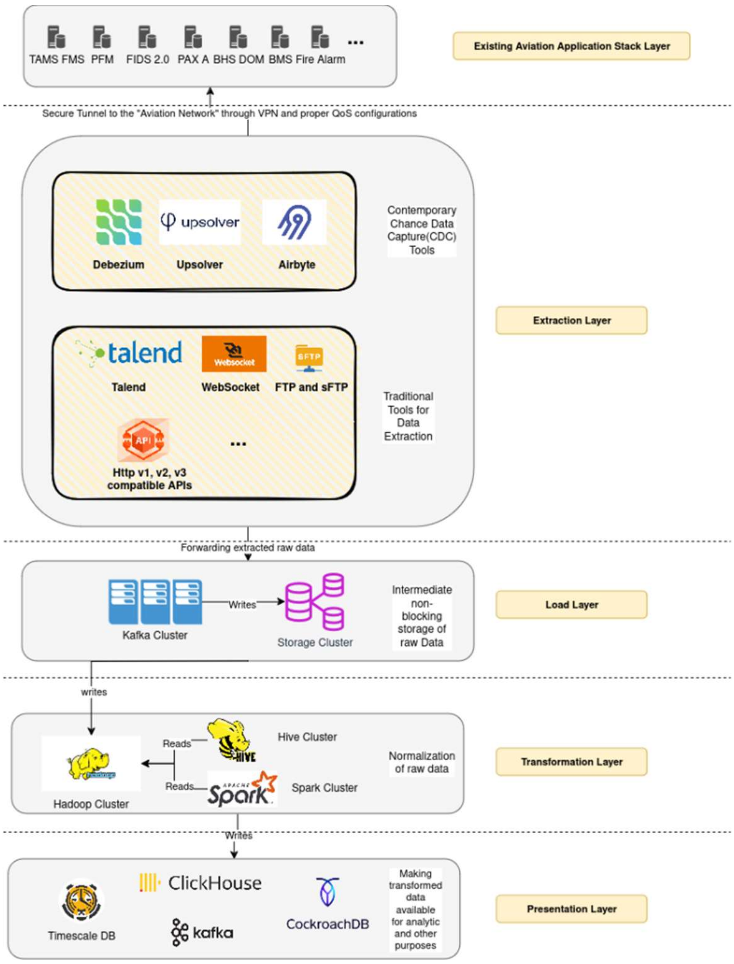
Data Collection Tools and Techniques:

- **Change Data Capture (CDC)**: Given method, known as capturing data changes, is constructed upon the observer design pattern principle that a data source system notifies its subscribers who want to capture changes when a change occurs in the data. For instance, consider a traditional RDBMS database. Databases that support this method transmit the changes to a replication stream when an application connects and performs data manipulation (DML). Once the RDBMS received a commit message after a couple of DML operation, it drops messages to its short-lived replication protocol it order to let the subcribers based-on their change-interest which results in replication of the data.

    Change Data Capture (CDC) was first introduced in 2015 through an academic study and has quickly been adapted into existing systems by software development communities and companies. Most of the modern database management systems have already been integrated this design pattern into their applications, or it is on the way as an experimental process. CDC is widely used in data warehousing, real-time analytics, and data integration solutions, enabling efficient synchronization across multiple systems.
- **Specialized CDC Plugins**: Software versions that support CDC functionality in their modern versions can also provide this functionality to older versions that do not support CDC by using existing plugin development interfaces. Most database management systems either officially support this or open-source communities develop these plugins. Some of these projects, which are not mature enough, may need further development. Being open-source is a significant advantage in this regard. For example, two different CDC plugins for Elasticsearch that work with CDC and Long-Pooling design patterns, and a plugin working with CDC for Ignite, have been developed by the TAV Technology R&D team.
- **ETL/ELT Tools**: These are tools planned to be used for database management systems where the above methods cannot be applied. These methods are applications used to take data from the source, transform it, and then move it to the target, originating from periods when modern concepts such as "Big Data," "Data Warehouse," and "Data Lake" were not part of our lives. Examples of such applications include Talend Open Studio, Apache NiFi, Apache Airflow, and Fivetran. ETL (Extract, Transform, Load) and ELT (Extract, Load, Transform) are processes used for integrating and analyzing data from various sources. ETL tools extract data from different data sources, transform it into a usable format, and load it into a data warehouse or data lake for further analysis. ELT, on the other hand, loads raw data directly into the target system and performs transformations as needed. These tools are critical in big data environments for their flexibility and wide range of data source compatibility. One of the significant challenges with traditional ETL/ELT processes is the time and resource intensity, especially when dealing with large data volumes.

Incremental Backup Restore and Cluster Aggregation Hashing are often used to minimize the impact on the source server by capturing only the changed data blocks rather than processing the entire data set.

- **Customized Tools:** This method involves writing custom applications in order to extract data from the source based on special requirements of the source. For example, data is read from the source and transferred to the target by reading/listening, such as, the file system events, intercepting network, or unix socket through by routing changes into an intermediary proxy application for intervening and etc. Custom tools and adapters are often used when standard data integration methods are insufficient due to unique data formats or proprietary protocols of data source. These tools provide a tailored approach to data extraction, ensuring compatibility and efficiency. However, developing and maintaining such tools can be resource-intensive and may require specific technical expertise.



P2: In the above image, you can see the flow diagram of our data collection and processing methods related to the Proof of Concept (PoC) work conducted at our pilot airport, İzmir Adnan Menderes(Iata ADB) Airport.

5. System Architecture

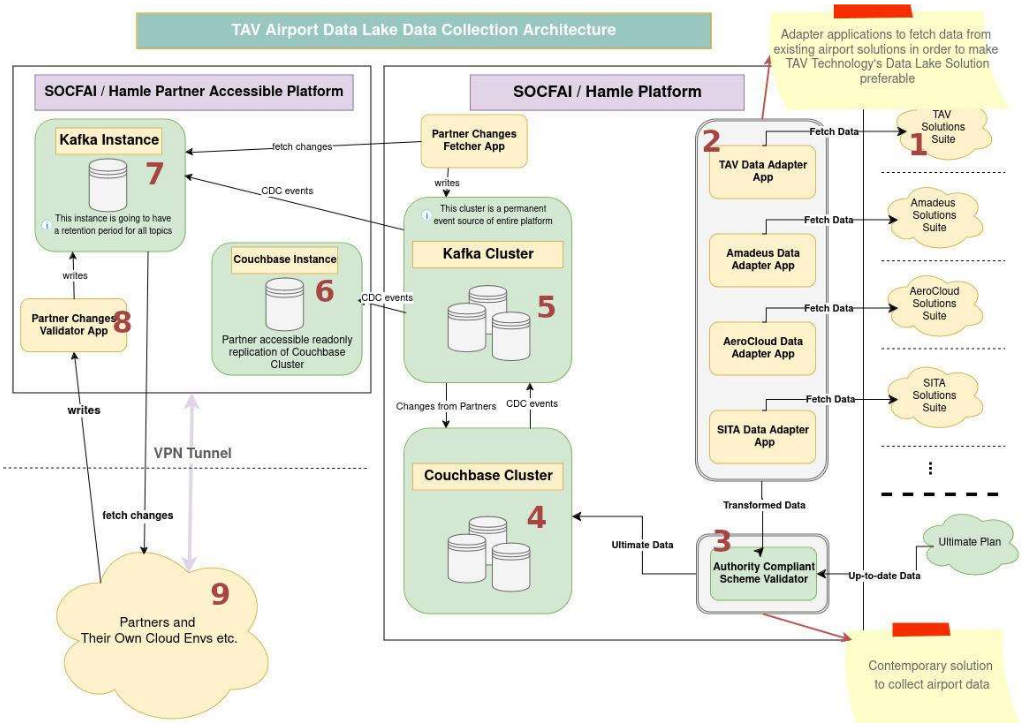


Figure 3: Given the planned "Network Architecture," the system architecture for data collection is visualized in Figure 2, with the main components numbered from 1 to 9.

The component labeled 1 represents the current software components used in airports, such as software for managing flights, baggage information, counters, information displays and etc. Different airports could been chosen different type of software vendors to manage such kind of operations, or even more, there could be different versions of software from the same software vendor because of different type of requirements and so on.

The idea for component 2 arose from the need to create an independent component for each solution we plan to integrate to avoid this complexity. The underlying concept is similar to the "Adapter Design Pattern," which we often see in everyday life, such as power adapters or Type-C to USB 2.0 converters. Adapter Design Pattern allows incompatible systems to work together by converting the interface of a class into another interface expected by the clients. This is especially useful in software systems with diverse and evolving components, providing a means to ensure interoperability without requiring extensive changes to the underlying systems.

By considering all these requirements, the following tools are recommended to be used in the sequence listed, ensuring optimal results in terms of platform sustainability and cost:

- Change Data Capture (CDC)
- Talend Open Studio for Big Data
- Customized Tools

6. CI/CD Processes

To enable project stakeholders to develop their solutions, integrate with their systems, and deploy new applications on the platform using the data stored on it, it is crucial to develop codebase management and Continuous Integration/Continuous Deployment (CI/CD) processes within the platform.

Applications for managing these processes will be accessible to partners from a location open to their access, making it necessary to implement them on the platform referred to as PaP. However, since this auxiliary platform will have relatively limited resources and no access to existing systems, it will not be suitable for running applications. Therefore, the PaP environment will only serve as a delivery point where stakeholders will submit their developments.

The Git version control application Gitea, an open-source solution, will be used for submitting their code, tracking issues related to the developed applications, and following new requests, allowing documentation in AsciiDoc, Markdown, or HTML formats. Partners will develop using new Git branches or forks if they prefer and complete Code Review, Unit/Integration testing, and Quality Assurance processes before delivery. Applications that complete these processes will be deployed on the main platform with the help of the Jenkins application, which will manage the CI/CD processes and ensure that they fulfill their functions.

By incorporating modern tools and techniques, this project seeks to create a secure, efficient, and data-driven platform that enhances airport operations while providing stakeholders with the flexibility and resources to innovate and improve continuously.

7. Port Platform Capabilities

The implementation of digital logistics at port is being achieved through convergence technology linked to ICT-based artificial intelligence, and related research and development is underway in many places. This Port SOCFAI project is promoted as a process to promote the implementation of a digital port logistics system, through which customer-centered port logistics services are possible through the development of solutions and services that enable data sharing and information flow throughout the port logistics supply chain. and optimize the efficiency of the entire port logistics supply chain flow.

In particular, PORT SOCFAI seeks to develop an optimized solution for the logistics system in the import and export process at CFS/CY (Container Freight Station/Container Yard). The main priorities in Port SOCFAI's Use Case 4 are as follows.

The developments of an AI based CFS/CY management functions and blockchain capabilities that collects and manages Port logistics information based on SOCFAI platform are as follows.

- Container import and export order management and container transportation request processing capabilities

- Integrate with the mother ship's schedule for better planning and coordination of container loading/unloading and transfer management
- AI based on managing work history through mobile devices, managing container import and export performance, and managing warehouse cargo inventory to match system records with actual inventory for real-time updates on container operations.
- AI based management of transportation orders, establishment of an allocation plan for transportation vehicles, and accurate reporting of the departure and arrival of vehicles
- Contribution to WP3 (Development and Data analysis) in terms of data analysis and related functions development of Port Logistics.

The following figure shows the system model to be implemented in Use Case 4 to deploy AI based CFS/CY management functions and blockchain capabilities.