# ATM Twin: an Air Traffic Management Digital Twin for the Singapore FIR

Gabriel Mesquida Masana, François Balland
Thales Airspace Mobility Solutions, Singapore
gabriel.mesquida@thalesgroup.com, francois.balland.e@thalesdigital.io

***Abstract*—Creating new capabilities in Air Traffic Management *(ATM)* requires data extraction, prototyping and testing. Operational systems are constrained by safety requirements. Research processes tend to create bespoke non-reusable technical setups with reproducibility issues.**

**We present the *ATM Twin*, a *digital twin* of the Air Traffic Management System created for the Singapore Flight Information Region *(FIR)*. We explain its value proposition, requirements, and key design drivers. The main components are presented which include technical services, *ATM* domain services, and cloud-native infrastructure as well as the approach to enable a flexible instantiation and integration of diverse use cases.**

**Functionalities include traffic mirroring, traffic replay and synthetic traffic generation used to prototype new *ATM* domain capabilities, produce historical and live *Key Performance Indicators (KPIs)* and dashboards, new operational advisories for Air Traffic Control Officers *(ATCOs)*, and information sharing with third parties such as Research Institutions *(RIs)*. These are capabilities not available in existing ATM simulators. Domain prototyping includes green aviation, *Flight and Flow Information for a Collaborative Environment (FF-ICE)* services and *Trajectory Based Operations (TBO)*, covering a wide span of *Technology Readiness Levels (TRLs)*.**

**In 2023, the first live trials of *Continuous Descent Operations (CDO)* advisories took place with real traffic: the *ATM Twin* safely coexisted with the *Operational Air Traffic Management System (ATMS)* within the control room positions offering the Air Traffic Control Officers *(ATCOs)* the possibility to test *CDO* advisories in a real environment.**

**By using the *ATM Twin* as a platform, components and architectures can be reused to support research in varied topics and use cases of diverse *TRLs*. In practice, efficiencies have been achieved, and the research has been able to increase scale and maturity. Future work include using the *ATM Twin* to support *Artificial Intelligence (AI)* and *Machine Learning (ML)* development for *ATM*.**

***Keywords*—digital twin; air traffic management; cloud-native; data platform; prototyping; live trials; continuous descent operations; engineering**

## I. Introduction: The need for an ATM digital twin to support and augment ATM Research

*ATM* has lived within a turbulent environment in recent years. For a more efficient use of airspace, innovations are needed: operational, to support more efficient and sustainable procedures, and technical, to safely leverage new technologies.

Research processes require prototyping and testing of new capabilities. Examples of capabilities are advisories in *CDOs*, dynamic sectorisation, *Top of Descent Prediction (ToD)* and *ATCO* workload predictors as well as new experimental airspace configurations and new procedures. These are initially created at low *TRLs* [1], within lab-hosted demonstration platforms. Eventually, they mature to higher *TRLs* which require testing in real operational environments, with a test platform safely interacting with the operational system.

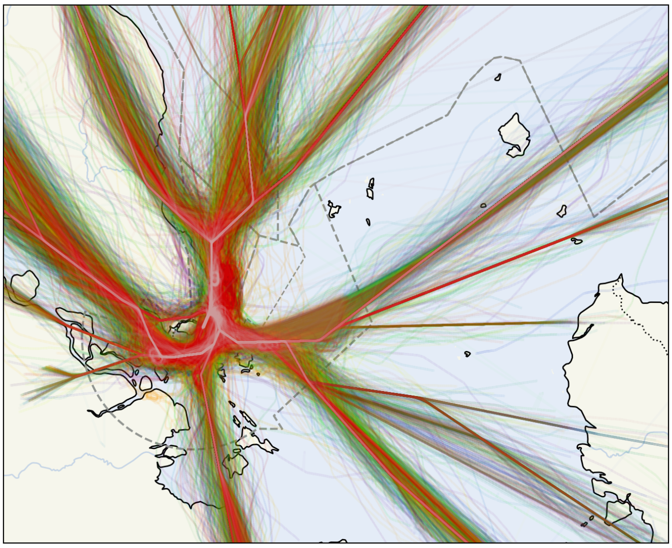

Figure 1. Sample *ATM Twin* data rendering arrival trajectories for 15k flights. The main *Standard Arrival Routes (STARs)* can be observed with the highest flight densities in grey. (Descents analysed in Figures 2 and 3.)

Traditionally, different research streams and *TRLs* have required diverse technical setups, bespoke in nature. Every team of researchers is used to selecting their own software, temporarily integrating solution, diverging in technical approach. With research projects being limited in time and budget, reusability criteria are not considered. Achieving the research goal is prioritised at the expense of future reproducibility.

We posited that a single platform could be built that would serve as shared technical enabler for different streams of research and maturity levels within the *ATM* domain. Such platform would account for common research needs whilst allowing for variability and prioritise re-usability of components and a standard and flexible approach to integration. The goal is to enable future research to leverage previous technical setups and demonstrators, allowing economies of scale to be achieved. More complex technical setups become feasible and lead to higher *TRLs* [2].

We named this platform the *ATM Twin*: an *Air Traffic Management digital twin*.

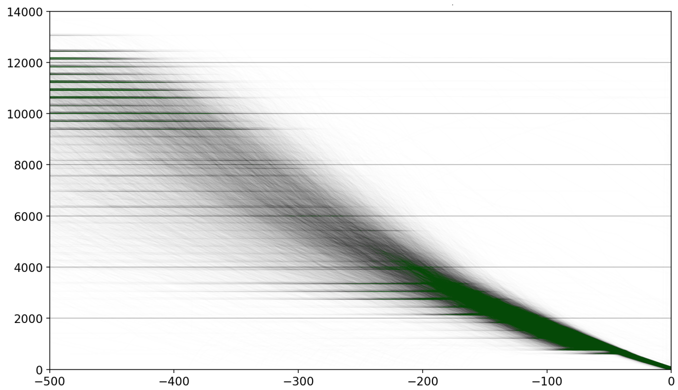

Figure 2. Sample *ATM Twin* descent profiles rendering for 15k flights. The cleared flight levels are visible as well as the levelling on descent. Aircraft levels on descent are rendered as density function in figure 3.

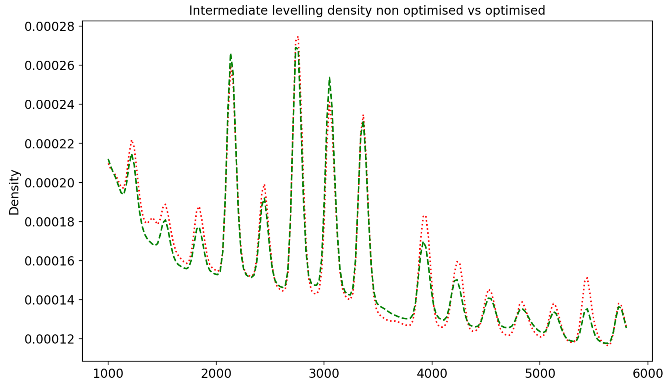


Figure 3. Density functions of aircraft arrivals. Two different functions are displayed in a *what-if* analysis: before (in red) and after (in green). An improvement can be detected as intermediate levelling reduction.

Ideated between Australia and France in 2020, the *ATM Twin* was designed to host the key components to simulate new operational and technical capabilities required for research and experimentation with flexible integration to support different use cases (Figure 4).

*Digital twin* is concept coined in 2015 in the context of *product lifecycle management* [3] [4]. In 2020 there were no references of *digital twin* implementations for the *ATM* domain. *The concept of a digital twin [was] not well-established in the transportation sector [and there was] an opportunity to take the term and use it to drive progress* [5].

Conceived as a virtual system connected with the real world, the *ATM Twin* provides researchers with a data platform replicating the aeronautical environment and providing real awareness, including the *ATMS* capabilities. It is fed with live data safely ingested from the operational system (Figure 1). Hence, research prototypes have access to the real airspace situation in real time, as well as historical data and synthetic data with simulation capabilities.

In 2021 the *ATM Twin* was implemented for the Singapore *FIR* by *AIR Lab*[1] Singapore, going live at the beginning of 2023. *CDO* prototyping live trials were conducted in Singapore Changi airport using the *ATM Twin*. The *ATM Twin* construction project was honoured with the *Minister's Innovation Distinguished Award for 2022 for Research and Development of ATM Twin*. The *ATM Twin* is currently supporting innovation and research works of whilst allowing collaboration with other *RIs*.

At the time of writing this paper, there are other relevant *digital twin* initiatives in the Airspace. We highlight *Project Bluebird* [6] [7] in the U.K. which use case is to train *ML*-based *ATCO* agents and the *Sector Performance Optimizer* [8] in Canada to optimise *ATCO* workloads.

[1]*AIR Lab* is a joint research lab by the *Civil Aviation Authority of Singapore (CAAS)*, and *Thales* with the support of the *National Research Foundation (NRF)* of Singapore.

## II. Safety Requirements in Operational Air Traffic Management Systems Restrict Their Use for Research, Data Analysis and Learning

*ATCOs* manage and deconflict air traffic prioritising safety and maximising efficiency. They provide separation between aircraft, sequencing arrivals and ensuring a continuous flow in departures, cruise, and arrival phases.

The role of *ATM* systems is to support *ATCOs* in their decision-making by providing accurate situational awareness, but the decisions are ultimately left to the *ATCOs*.

The safety critical *ATM* domain has a very strict set of safety requirements. Systems have a high level of redundancy and are, by design, deterministic. Safety cases are conducted during systems verification and validation processes.

Hence, research cannot directly use the *ATMS* for prototyping. And a direct implementation of a new capability within an *ATMS* is costly and long-lead. Unlike other domains in which *A/B testing* is possible, in the ATM domain research requires its own platform.

This limitation reinforces the need for an *ATM Twin*, not only to research, but also to complement the functions of the *ATMS* providing data analysis and learning capabilities in a more efficient and less costly manner outside the safety perimeter.

## III. Activities to be Supported and Requirements for an ATM Digital Twin

The key ATM research activities include operational research, extended data capabilities, learning and de-briefing support and collaboration with third parties:

- Operational research with prototyping of new operational capabilities and processes. Examples are *Continuous Descent Operations* prototyping applied to the Singapore FIR, *Tactical Controller Tool What-Else Probes (TCT-WEP)*, *Conflict Detection and Resolution (CDR)* and other advisories.
- New data capabilities to analyse the situational information such as live and historic KPIs and dashboards.

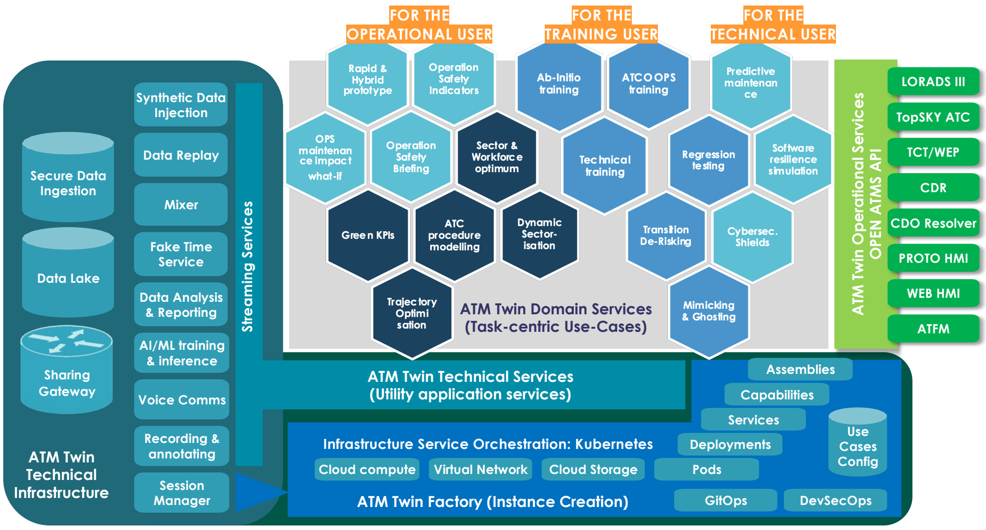


Figure 4. Conceptual view of the components of the *ATM Twin*. The figure is split into four different areas: *(i)* on the left, technical services which include data ingestion and storage, historical data reproduction, synthetic data generation via simulator, data analysis, and voice communication systems, *(ii)* on the right, *ATM* domain services such as tracking, coupling, flight plan and conflict detection, plus prototypes of new capabilities such as conflict resolution, what-else probes, or any other *ATCO* advisories, *(iii)* at the bottom, cloud-native, *Kubernetes*-based, infrastructure services including advanced abstractions for dynamic integration, and *(iv)* the central section containing use cases that are combinations of the above.

The *Air Navigation Service Provider (ANSP)* increases visibility of ongoing processes and their performance.

- Transformation of aeronautical data into feature datasets for *Machine Learning (ML)* model training (Figure 2). Examples are *ML* advisories such as *ToD*, *turn-to-final timeliness* and *VHF frequency-change* recommendations.
- *What-if* analysis for new features, processes, and configurations (Figure 3).
- Training and de-briefing capabilities that leverage the *ATM* situation awareness outside the control room: de-briefing by re-playing historical traffic remotely, learning from observation and training generated from real traffic data.
- Collaboration with third parties by safely and securely sharing live and historical information. Feeding the local *Small and Medium Enterprises (SMEs)* and *RIs* with data and jointly conducting new research. The collaboration between industry and *SME* & *RI* ecosystem achieves a wider *TRL* range.

The variability of use cases leads to dynamic integration requirements for an *ATM* research platform, creation of common architectural patterns such as stream-driven and microservice architectures, re-usability of common components or capabilities, re-use of pre-assembled blocks of capabilities. The underlying infrastructure must be flexible and scalable as any *ATM* research platform would require constant change.

### *A. A research platform free of safety requirements*

Since any impact on the *ATMS* is rejected, the *ATM* research platform must be completely isolated from the operational system.

A one-way stream of situational awareness data from the *ATMS* to the platform must be established through a uni-directional data stream.

We called the part of the platform to provide this safe link a *Secure Data Bridge (SDB)*. Uni-directionality was physically enforced by the implementation of Data Diodes based on laser beams. No feedback loops are possible from the *ATM Twin* into the *ATMS*.

From that point of the platform onwards, the safety restriction can be removed, allowing for additional flexibility in platform design. Additional exploration into new technological paradigms becomes possible. Cloud-native, data-centric architectures and *microservice* design patterns can be used within the platform.

A compromise must be found between restricting the technical stack and introducing additional diversity to foster new technical exploration. Further learnings can be achieved at the expense of economies of scale.

Not requiring safety is a key difference between the design of an *ATM Twin* and an *ATMS*.

### *B. Inefficiencies in one-off approaches to experimentation*

This approach of creating a common research platform contrasts with a *Minimum Viable Product (MVP)* approach. *MVPs* are commonly used in research, including the AIR Lab before the creation of the *ATM Twin*.

*MVPs* are project-driven, creating a one-off bespoke experiment for each, short, research project. The objective is to reduce the turnaround time to achieve a featured demonstrator. Agile methodologies are used and delivery speed of working code is prioritised over potential nugatory work if refactoring is needed later on. This leads to poor reusability and for different teams to use divergent software stacks and architectural patterns without much reflection.

When a set of *MVPs* is analysed holistically, an excessive software stack variance is observed. As a consequence, MVPs suffer reproducibility issues once the team is disbanded, and support functions of *RIs* end up collecting a large range of licences, open source tools, platforms and diverse hardware.

Knowledge transfer between teams is also limited by technology stack and integration pattern diversity. Learning curves become steeper for new members of the team.

Ultimately, diseconomies of scale are borne by *RIs*.

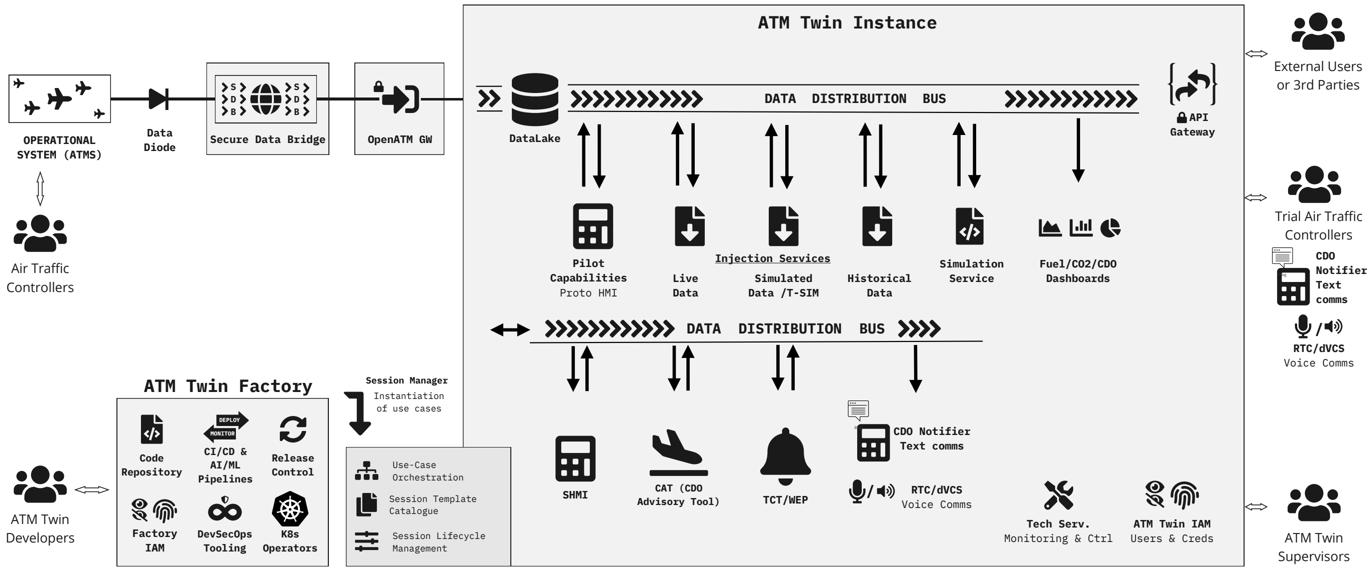


Figure 5. Sample component diagram for the *ATM Twin*. The instance depicted is implementing a *CDO prototyping* use case supporting actual live trials. The main components, arranged in a data streaming pattern, are: *(i)* a *Secure Data Bridge (SDB)* that safely extracts live information from the operational ATM system, *(ii)* a simulation service plus pilot HMI capabilities for testing specific scenarios, *(iii)* a *CDO* advisory tool to create *CDO* recommendations, relying on fuel consumption calculation and conflict detection services, *(iv)* an *HMI* to observe the recommendation along with the air situation in the lab for refinement, it can be seen in Figure 10, *(v) CDO* notifiers which offer recommendations to the *ATCOs* (Figures 11 and 12), and *(vi)* common elements to different use cases such as the data distribution bus, the *data lake*, the shared infrastructure, the *ATM Twin factory* or the *session manager*. Other instances may be live with different instantiation of the same components, integrated differently in order to serve other use cases. Note that some elements, such as the *SDB* or *data lake*, are singleton components not replicated in instances.

## IV. Methodology: Key Design Drivers to build an ATM digital twin

We briefly review within this section the key design drivers for an *ATM digital twin* and how they were resolved for the *ATM Twin* for the Singapore FIR.

### A. Cloud-native platform and infrastructure-as-code

It was our requirement to have a cloud-native infrastructure that allowed for dynamic integration, full flexibility and scalability.

Our approach was to use distributed computing within a cloud supplier whilst providing a cloud-agnostic abstraction layer. We chose *Kubernetes* [9] as platform. An open source platform, initially contributed by *Google*, which has become the dominant platform in distributed computing.

*Kubernetes* leverages container-based architectures to provide abstractions such as *pods*, or groups of containers, *deployments*, or groups of *pods* to ensure a specific service level, and *services*, to ensure the delivery of *services* as supported by a *deployment* back end, as well as many other useful, standardised, abstractions.

*Kubernetes* is a *declarative* system in which the state of the system is described and *Kubernetes* deploys as per desired state. If any element fails, the divergence is detected and an automated healing system restores the declared desired state.

This approach is contrasted to an *imperative* approach, usually script-based, where specific building instructions are given. In an imperative approach, in the case of failure, either the specific correction needs to be scripted or the system needs to be rebuilt from scratch. For the *ATM Twin* design, it was a key requirement to avoid this complexity.

The abstraction in the infrastructure: the fact that compute, storage and network allocation are orchestrated through *Kubernetes*, makes the *ATM Twin* cloud-agnostic. Although currently running mostly in the cloud, it can run in a hybrid environment between cloud and on-prem. The *SDB* is implemented on-premises and bridges the on-prem and cloud deployments.

### B. Modular and composable capabilities as higher microservice abstractions

A key capability of *Kubernetes* is the creation of *Customer Resource Definitions (CRDs)* [10], or extensions of the Kubernetes API as defined by the user. We leveraged those to create services, microservices, or groups of them, that we named *capabilities*.

Containers package code. *Kubernetes* groups containers into *pods*. They are exposed as *services*. Using *Kubernetes*' *CRDs*, we create *capabilities* as domain-specific services.

Examples of *capability* is a *Human-Machine Interface (HMI)* providing situational awareness (Figure 10), simulation or audio communications. Deeper within the technical domain, can providing a data-streaming bus to articulate the integration around the flow of information or data injectors from a live feed or from historical recordings within the *data lake*.

*Capabilities* encapsulate complex technical solutions making the code reusable for diverse use cases. Several *capabilities* compose a *session*, which is defined as a specific implementation of a use case within the *ATM Twin*. Capabilities contain *services*, *deployments*, *statefulsets*, *jobs* and *pods*, which are

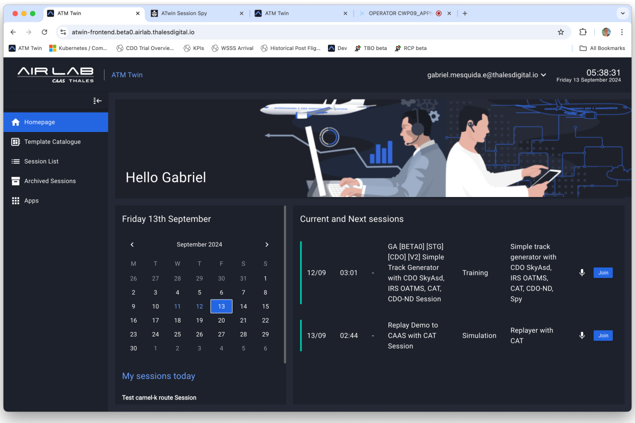


Figure 6. *ATM Twin session manager*: welcome page after user logon. The page reminds the user of active sessions and future activities scheduled as well as offering access to *ATM Twin* system status.

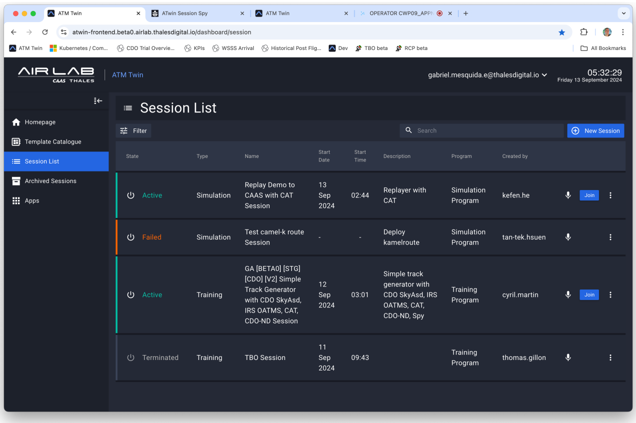


Figure 8. *ATM Twin session manager*: sample sessions list. The ongoing sessions are listed along with their status. New sessions can be created by re-using and configuring templates in a *no-code* simple approach.

native *Kubernetes* abstractions and can be complex technical artefacts (Figure 7). By using *capabilities* there is no need to worry about the internal details, just to provide the key configuration parameters.

Composability of *capabilities* provides additional flexibility over container re-use, which is the usual approach to microservices architectural patterns. In the case of *capabilities* the service does not have to be a microservice, as in *micro-sized*, but can instead be a fully fledged service such as a simulator or a *digital Voice Communication System (dVCS)*.

Additionally, thanks to the *declarative* approach, *capabilities* and *sessions* are collections of components with their interactions declared so that they can be instantiated and re-built on demand without following scripts.

In the *ATM Twin*, capabilities can be composed and configured easily using a *no-code* interface (Figure 9).

## C. Maximising the use of open source components

It is the belief of the authors of this paper that proprietary code is better avoided, hence we choose to maximise the number of open source components and avoid creating code, maximising the efficiency of the resources available.

Hence, it was a matter of principle to rely on the open source offerings, and get their support where appropriate.

Open-source tools used include the *Elastic* [11] stack, *Kafka* [12] as event streaming platform, or *Kubernetes* [9] as distributed computing orchestrator.

## D. Open system fostering data sharing and collaboration

*Open ATM APIs*, a series of *Application Programming Interfaces (APIs)* that act as a contract to retrieve aeronautical information were built and open-sourced by *AIR Lab* to facilitate information serving. This was an essential first step to create effective information exchange internally within *MVP* components at first, then *ATM Twin* components. Also externally, facilitating collaborative research with third parties.

The *APIs* are compatible with the aeronautical industry standards for information interchange, this includes *Flight Information Exchange Model (FIXM)* [13], *System Wide Information Management (SWIM)* [14], and *Flight and Flow Information for a Collaborative Environment (FF-ICE)* [15].

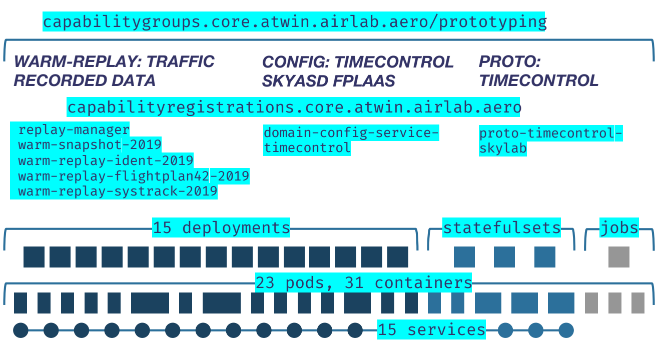


Figure 7. Diagram showing the different *Kubernetes* abstractions involved in generating a simple historical traffic replay use-case. A use case is an assembly or capability group, made from a selection of *capabilities* that the user has selected. These are underpinned by native *Kubernetes* abstractions such as *services*, *deployments* and *pods* containing standard software containers.

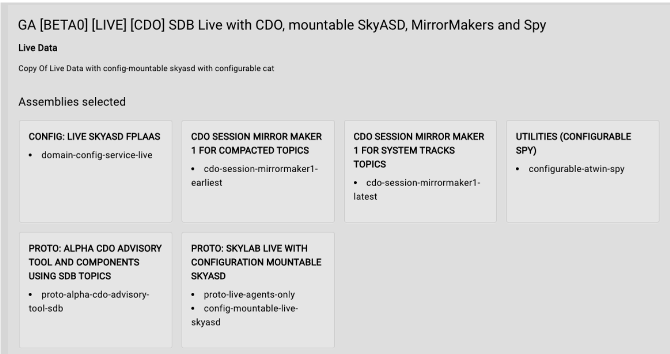


Figure 9. Creating a new *assembly* to support a use case with the *session manager* by selecting *capabilities* in a *no-code* user interface. Configurable parameters can be identified and altered. An *assembly* gets saved as a template and can be instantiated or modified at any time. The technical components underpinning the *capabilities* are automatically included.



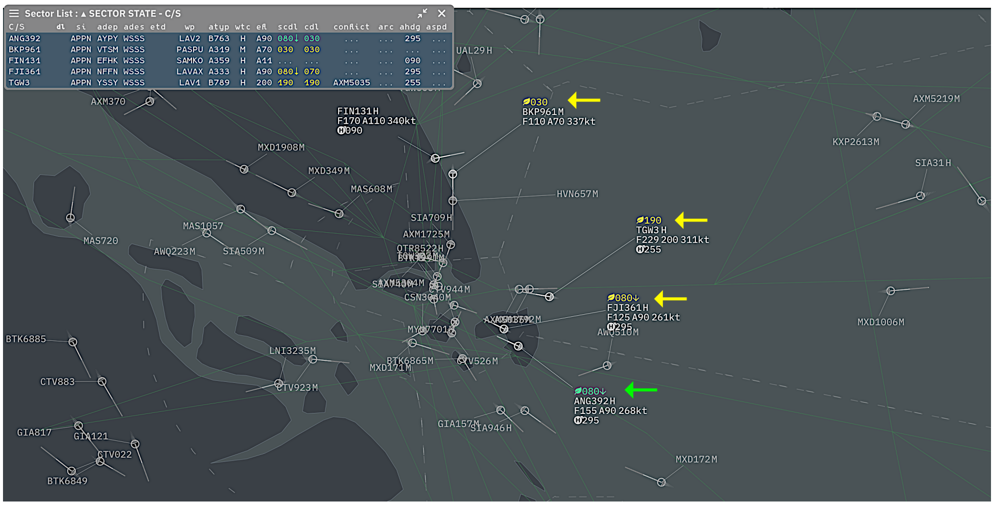


Figure 10. *HMI capability* providing situational awareness that can be combined with other capabilities to integrate instances of diverse use cases. In the figure the *HMI capability* is combined with a simulator and a *CDO* advisory prototype. In the *HMI*, four flights are highlighted: three in yellow, awaiting action from the *ATCO* in order to facilitate a *CDO*, and one in green as the *CDO* is fully cleared. (Arrows have been added for readability)

A key element is the *gateway*, which is built on top of *Kubernetes* with the *ingress* abstraction and contributes to a hardened cybersecurity profile to ensure the safe interchange of information with third parties and data-loss prevention.

## V. Results and Discussion

Ideated in 2020, the first release of the *ATM Twin* was built between 2021 and 2022 with detailed design and delivery phases. A conceptual view of the design is depicted in Figure 4 with the main conceptual component blocks. A component diagram of the set of *capabilities* needed in order to implement a specific use case: a *CDO prototyping session*, is illustrated in Figure 5.

The basic features from the fist release included the *ATM Twin factory (GitOPS)*, the cloud-native platform, prototyping capabilities, the *session manager* (Figures 6 and 8), a *data lake*, *streaming* capabilities and *digital voice communications*.

The first release already supported data analysis use cases for *ATCO* workload estimation, live *KPI* generation and dashboards, cloud-hosted air traffic simulation services to support training plus online briefing capabilities. Historical data was available for prototyping, traffic replay and simulator-based training exercise generation. Note that *simulators* are natural components in any *digital twin*, supporting synthetic data generation to be used in more sophisticated use cases. Within the *ATM Twin*, two different simulators from different generations of *ATMS* are included as *capabilities*.

Worth mentioning was the creation of *Post Flight Reports*, driven by a request from *ATCOs*, producing flight reports data tailored to their needs without any *ATMS* modification. At that point the *ATM Twin* was already providing data analysis capabilities further to the existing *ATMS*.

In the second year of the *ATM Twin*, up to mid-2023, it was used in several research streams. They included topics as diverse as *FF-ICE* [15] and *TBO* [16], as well as *CDO* [17] [18] [19] within the scope of green aviation research. They used a combination of historical data and simulator-generated synthetic data dynamically plugging existing components and new prototypes. The flexibility and re-use was already higher than any experiment-based bespoke platform in previous *MVPs*.

In 2023 the *ATM Twin* was connected to the *ATMS*, streaming live data through the *SDB*. At that point it became able to mirror live traffic, and to provide a live replica environment of the Singapore *FIR*, including real situational awareness.

As live data became accessible, live prototyping of new features driven by the *ATM Twin* became possible. As an example of live prototyping in a live and real environment we highlight the *CDO* live trials in the Singapore *FIR* in 2023.

*Continuous Descent Operations* aim to facilitate *Continuous Descent Approaches*, optimising the descent profile using low or idle thrust, avoiding levelling in between *ToD* levels and the levels restricted by *STARs* in order to minimise fuel consumption and hence environmental impact with noise and emissions reduction. The *CDO* advisory prototype is capable of identifying candidate flights, notifying them to the ATCOs though the *notification display* shown in Figure 12.

The trials lasted 6 weeks. *ATCOs* were able to experiment with a *CDO* advisory tool within the control room. Figure 11 shows a sample temporary *ATCO* control position setup to

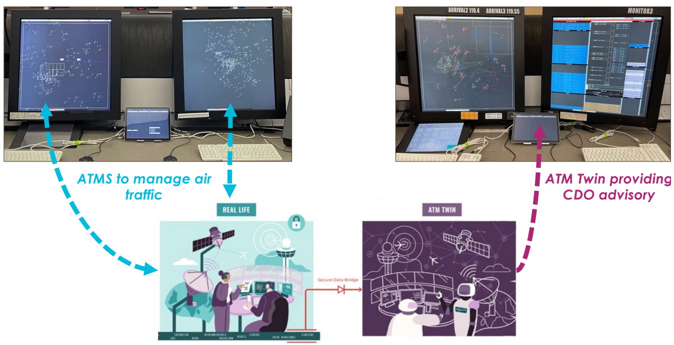


Figure 11. Sample *ATCO* temporary position setup for *CDO* prototyping trial. The *ATM Twin* provides a live *CDO* advisory to the *ATCO* whilst keeping the *ATMS* completely isolated from the trial.

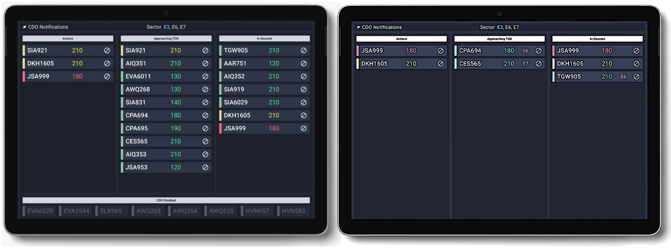

Figure 12. Sample *Notification Display* for *CDO* advisory as provided by the *ATM Twin*. The advisory is generated in real time from the incoming live data stream. A recommendation is provided to the *ATCO*, but the final decision is always made by the officer.

include the *CDO* advisory within a control room for live trials.

The system also allows to continuously track recommendation status (Figure 10) and to delve into the calculation details to assess the quality of the recommendation and to further refine system and configurations.

During the six weeks of the live trials, all sorts of data was collected and analysed, creating dashboards and providing useful operational information on how to increase the level of *CDO* in Changi. The data will inform for further industrialisation of the prototype. A live prototyping activity can be classified as *TRL* 6 [1] [2].

The *ATM Twin* has proved its applicability in a wide span of research activities. Other examples include:

- Training of *ML* models for *ToD* prediction with the *Air Traffic Management Research Institute (ATMRI)* of the *Nanyang Technological University Singapore (NTU)*.
- *FF-ICE* [15] services prototyping including *FF-ICE/R1 Flight Plan Filing Service*, *Flight Data Request Service*, *Trial Service* and *Planning Service*. They are supported by a *Flight plan validation* service which checks the content of an *ICAO 2012 Flight Plan (FPL)* message and a *Flight plan route extension* service which creates a 2D route based on *field #15* of *ICAO* message.
- Technical research in future *ATMS* platforms and architectures. This includes the use of *cloud-native* platforms such as *Kubernetes*, *microservice* and *stream-based* architectures and *open-source* software.

Ongoing and future works include additional developments of the *ATM Twin* in order to support *ML* processes. As the *ATM Twin* was connected to the live feed coming from the *ATMS*, The *data lake* started to store the data received. With an additional transformation into features, it can support *ML* processes. A *Machine Learning Operations support stack (MLOps toolchain)* has been created with a compatible software stack. This enables the *ATM Twin* to support the development of *AI* for the *ATM* domain. *ML* models can now be trained with historical data, tested with unseen live traffic, incorporated into live *ATCO* advisories. Data drifts, where the underlying distributions of the features change over time, will be able to be monitored. (Figure 13)

## VI. Conclusions

The *ATM Twin* construction project has demonstrated the feasibility of using a *digital twin* of the ATM environment as data platform enabling different research streams with diverse topics and *TRLs*.

Compared to previous *MVP* experiences with specific bespoke technical setups, the use of a single flexible research platform has greatly increased systematisation of components and architecture. Subsequently, reproducibility and reusability have also improved. Technical setups have become more standardised and easier to maintain.

This has contributed to larger projects with higher real life impact to become increasingly feasible. The *ATM Twin* has been instrumental for research to be able to reach *TRL* 6, prototype demonstration in a relevant environment [1], with successful *CDO* live trials in the Singapore FIR in 2023. We believe it can be applied to a wide range of *TRLs* (Figure 14).

The creation of the *ATM Twin* has also impacted the structure of the research team within by increasing profile and role specialisation. Focusing the software stack has increased knowledge retention and use of more advanced features.

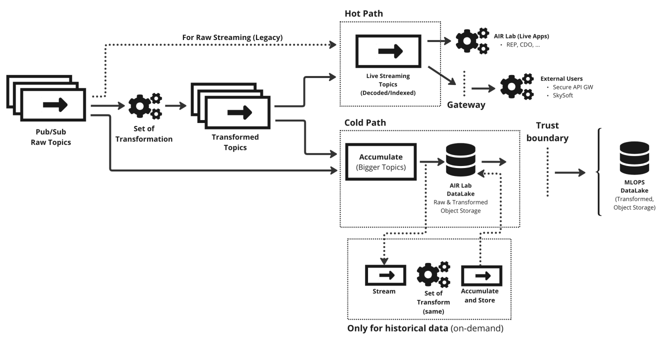


Figure 13. Sample design for future work of integration of the *ATM Twin* with the *MLOps toolchain* by forking the incoming data stream into a *hot and cold ingestion path* architectural pattern. The *cold path* stores incoming data in the *data lake* and provides objects containing batches of data for *ML* training and data drift checks. The *hot path* provides a stream of live data for real time prediction in order to power *ML* advisories within the *ATM Twin*.

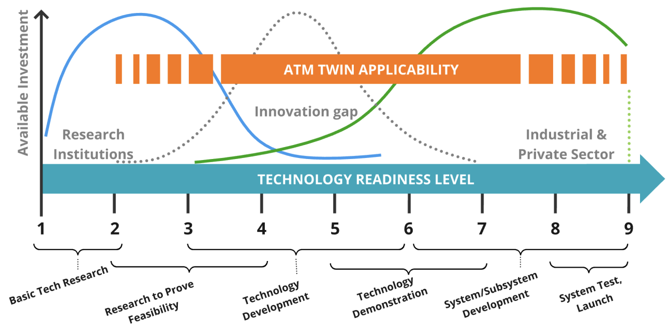


Figure 14. Applicability of *ATM Twin* as a research platform throughout *TRLs*: it can support and cover the innovation gap between academic *RIs* and industry by supporting industrialisation research (depicted as grey curve in the middle of the figure) usually referenced as *innovation gap* in the literature [20] [21]. We believe that *ATM Twin* can cater for a wide range of *TRLs* supporting an *Open Platform* approach [22].

Challenges for future platform development include finding the optimal balance the introduction of new components and architectural patterns with regard to reuse of standardised and well known existing components. Work remains in the area of increasing cloud use efficiency, standardisation of key integration artefacts and *API* multiple versions coexistence.

Future planned work includes the incorporation of *AI agents* within the *ATM Twin* to interact with the user in prototyping and training use cases. *ML-based Digital Pilots* will potentially respond to *ATCO* clearances providing additional *what-if* capabilities, and future *Digital ATCO agents* will deconflict traffic in high-speed traffic simulations for efficient fast-time testing of *ATM* prototypes and configurations.

## ACKNOWLEDGMENT

This research is supported by the *National Research Foundation*, Singapore, and the *Civil Aviation Authority of Singapore*, under the *Aviation Transformation Programme*. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of *National Research Foundation*, Singapore and the *Civil Aviation Authority of Singapore*.

## REFERENCES

[1] International Standards Organization (ISO). ISO 16290:2013 Space systems: Definition of the technology readiness levels (TRLs) and their criteria of assessment. https://www.iso.org/obp/ui/en/#iso:std:iso:16290:ed-1:v1:en, 2013. Accessed: August 2024.

[2] Ben Hicks, Andreas Larsson, S.J. Culley, and Tobias Larsson. A methodology for evaluating technology readiness during product development. volume 3, pages 157–168, January 2009.

[3] Michael Grieves. Digital twin: manufacturing excellence through virtual factory replication. *White paper*, 1(2014):1–7, 2014. https://www.3ds.com/fileadmin/PRODUCTS-SERVICES/DELMIA/PDF/Whitepaper/DELMIA-APRISO-Digital-Twin-Whitepaper.pdf. Accessed: August 2024.

[4] Michael Grieves and John Vickers. *Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems*, pages 85–113. Springer International Publishing, Cham, 2017.

[5] Research ARUP Foresight and Innovation. Digital twin: towards a meaningful framework. Technical report, ARUP, London, United Kingdom, November 2019. www.arup.com/digitaltwinreport. Accessed: August 2024.

[6] The Alan Turing Institute. Project Bluebird. https://www.turing.ac.uk/research/research-programmes/project-bluebird, 2023. Accessed: August 2024.

[7] NATS. Project Bluebird. https://www.nats.aero/about-us/research/n/project-bluebird/, 2021. Accessed: August 2024.

[8] Shavin Fernando. Sector performance optimizer. https://www.eurocontrol.int/sites/default/files/2024-04/20240429-flyai-forum-session-3-fernando.pdf, April 2023. Accessed: August 2024.

[9] The Kubernetes Authors. Kubernetes components. https://kubernetes.io/docs/concepts/overview/components/, 2024. Accessed: August 2024.

[10] Kubernetes Custom Resources. Kubernetes components. https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/, 2024. Accessed: August 2024.

[11] Elasticsearch B.V. Elastic stack. https://www.elastic.co/elastic-stack, 2024. Accessed: August 2024.

[12] Apache Software Foundation. Apache kafka. https://kafka.apache.org/, 2024. Accessed: August 2024.

[13] FIXM Change Control Board. Flight information exchange model (FIXM). https://fixm.aero, 2022. International Civil Aviation Organization (ICAO). Accessed: August 2024.

[14] ICAO. Manual on system wide information management (SWIM) concept. Technical Report Doc 10039 AN/511, International Civil Aviation Organization (ICAO), Montreal, Quebec, Canada, 2015.

[15] ICAO. Manual on flight and flow information for a collaborative environment (FF-ICE). Technical Report Doc 9965 AN/483, International Civil Aviation Organization (ICAO), Montreal, Quebec, Canada, 2012.

[16] ICAO. Global trajectory-based operations (TBO) concept. https://www.icao.int/airnavigation/tbo/Pages/Why-Global-TBO-Concept.aspx, 2018. International Civil Aviation Organization (ICAO). Accessed: August 2024.

[17] ICAO. Continuous descent operations (CDO) manual. Technical Report Doc 9931 AN/476, International Civil Aviation Organization (ICAO), Montreal, Quebec, Canada, 2010.

[18] EUROCONTROL. Continuous climb and descent operations (CCO/CDO). https://www.eurocontrol.int/concept/continuous-climb-and-descent-operations, 2024. Accessed: August 2024.

[19] European CCO/CDO Task Force. European continuous climb and descent operations action plan. Technical report, EUROCONTROL, Brussels, Belgium, November 2020. https://www.eurocontrol.int/publication/european-cco-cdo-action-plan. Accessed: August 2024.

[20] Henry Chesbrough and James Euchner. The evolution of open innovation: An interview with Henry Chesbrough. *Research-Technology Management*, 54, 09 2011.

[21] Henry Chesbrough. *A Reconsideration of Open Innovation After 20 Years*, pages 3–18. Oxford University Press, 02 2024.

[22] Geoffrey Parker, Georgios Petropoulos, Marshall Van Alstyne, and Joel West. *Driving Open Innovation Through Open Platforms*, pages 387–404. Oxford University Press, 02 2024.