D2.2 SOCFAI FRAMEWORK ARCHITECTURE AND DESIGNS

Secure Open Collaboration Framework powered by Artificial Intelligence

30 June 2024

Abstract

This document provides detailed information about the data standardization efforts carried out within the SOCFAI project.

Contents

Contents

1 . Introduction

The data standardization efforts have been designed to ensure that the project can be implemented across different airports and ports. Data extracted from airport and port systems has been prepared to align as closely as possible with the standards set by aviation and maritime authorities. Messaging protocols have been developed based on standardized data structures, which eliminates naming inconsistencies and facilitates clear communication between various stakeholders. This standardization ensures that the platform can easily adapt to different languages in the future.

This initiative represents the future of aviation and maritime operations, enabling the highest level of interaction among all operational stakeholders. During the project, the data structures of all subsystems were meticulously analyzed line by line, and integrations were built accordingly. This thorough approach ensures seamless compatibility and operational efficiency across different environments and systems.

2 . Airport Data Standardization

Each data standardization study includes the following key components:

- Data Source: Refers to the systems from which data is consumed and the content of the data itself.
- AOCC Static Tables: Represents static data that is loaded into the platform and used in transformations.
- Adapter Layer: A code block that integrates data from the data source, static tables, and other AOCC tables.
- AOCC Tables & Standards: Refers to the standardized tables.
- Other AOCC Dynamic Tables: Represents additional dynamic AOCC tables used in generating the standardized data.

As shown in Figure 1, there are mid-layer tables generated from the data sources. However, we cannot define these tables as standard because they will vary when subsystems differ from one airport to another. Therefore, these tables should be considered part of the adapter layer.

At İzmir Adnan Menderes Airport, the FMS, RMS, and CMS products operate together under a platform called TAMS. Data from all three systems can be retrieved using a single JSON format. The FMS optimizes flight operations by managing flight schedules, arrivals, departures, and turnaround processes. The RMS efficiently allocates airport resources such as gates, check-in counters, and baggage belts to streamline airport operations. The CMS manages and distributes content across various digital displays within the airport, ensuring accurate and timely information for passengers. These systems provide all the necessary information related to flights, resource planning, and service usage that other partners and project stakeholders require.

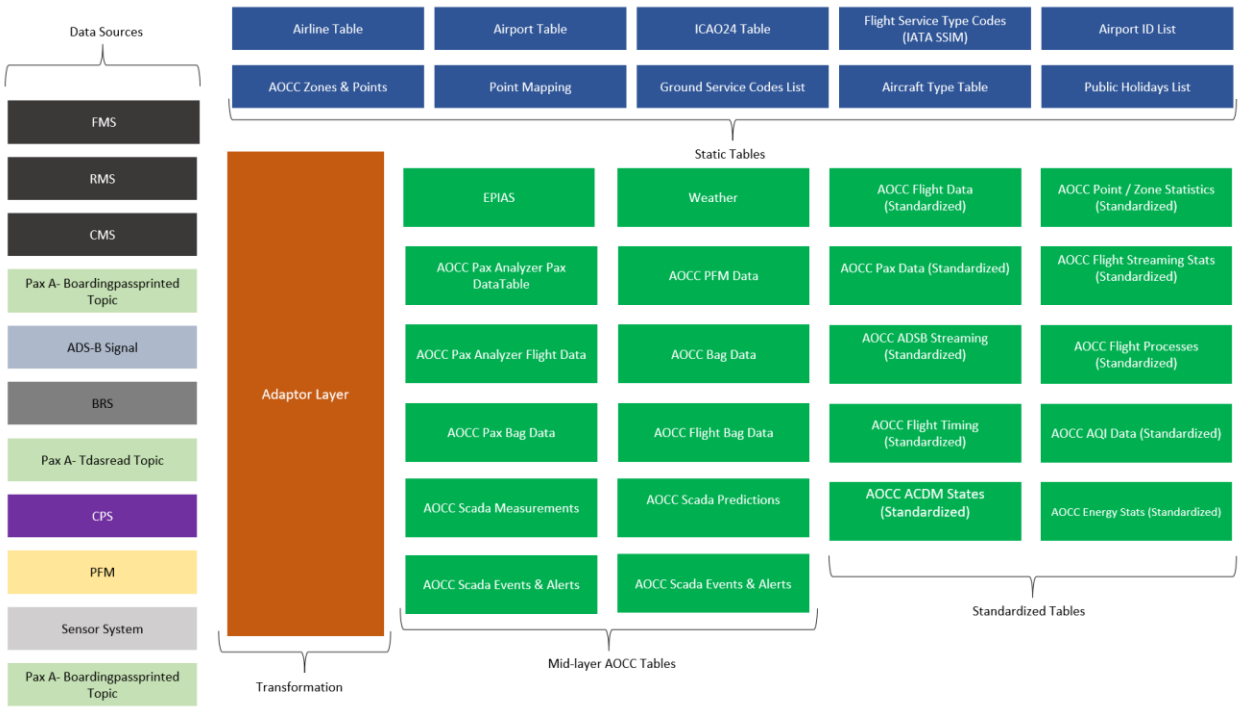

Figure 1 - Airport data tables overview

On the other hand, Pax Analyzer and PFM solutions are designed to monitor passenger flows and processes in the terminal. Pax Analyzer is a tool that provides real-time and historical data analysis of passenger flows, enabling airports to optimize queue management, predict passenger behavior, and improve overall efficiency. PFM, or Passenger Flow Management, complements this by offering detailed monitoring and control of passenger movement through various airport checkpoints, such as security, check-in, and boarding. Together, these tools enable airports to monitor passenger traffic and movements in a detailed way.

Another critical data source for the project is the broadcasted ADS-B data. With any ADS-B signal receiver, information such as the positions, speeds, directions, altitudes, and various other details of aircraft can be accessed, provided the aircraft is within a 200 km range of the receiver. However, the incoming data must be processed and interpreted to be meaningful. Deliverable 2.4 outlines the methodology for carrying out this process.

BRS, or Baggage Reconciliation System, is a system designed to track and manage baggage handling processes at airports. It ensures that each piece of luggage is accurately matched with the correct flight and passenger, reducing the risk of lost or mishandled baggage. The system integrates with various airport operations to provide real-time tracking and monitoring of baggage throughout the entire handling process, from check-in to loading onto the aircraft. Solution also helps us on detecting some passengers checkin events which is not possible for Pax Analyzer if there is no boarding pass printing event seen.

As the last data sources, Airport Energy SCADA with Sensor System is a specialized solution for monitoring and managing energy usage, air quality across airport facilities. This system integrates Supervisory Control

and Data Acquisition (SCADA) technology with a network of sensors to provide real-time data on energy consumption, equipment status, and environmental conditions. It enables airport operators to optimize energy efficiency, monitor critical systems, and quickly respond to any issues that may arise. By centralizing control and providing detailed insights, the system supports sustainable energy management and enhances the operational efficiency of airport infrastructure.

The Central Prediction Service is a system component designed to share prediction results for all timing-related fields. In this context, the predictions will be generated by the systems of the project stakeholders.

2.1 AOCC Flight Data (Standardized)

The provided table shared in Figure 2 is a detailed mapping document that outlines how various data fields related to airport operations are integrated into a standardized JSON format, which is essential for consistent data handling across different airport systems, such as the Flight Management System (FMS).

The table begins by defining general fields, which include identifiers and metadata such as id, version, and arrDep. These fields establish the basic identity and status of each flight record. There are also specific fields like linkedFlightID, divertedFlightId, and seasonalFlightId, which manage scenarios where flights are connected to other operations, such as diversions or seasonal scheduling. Timing information is a key component, with fields like stad (scheduled time), etad (estimated time), and atad (actual time) being mapped to track the flight's progress and performance.

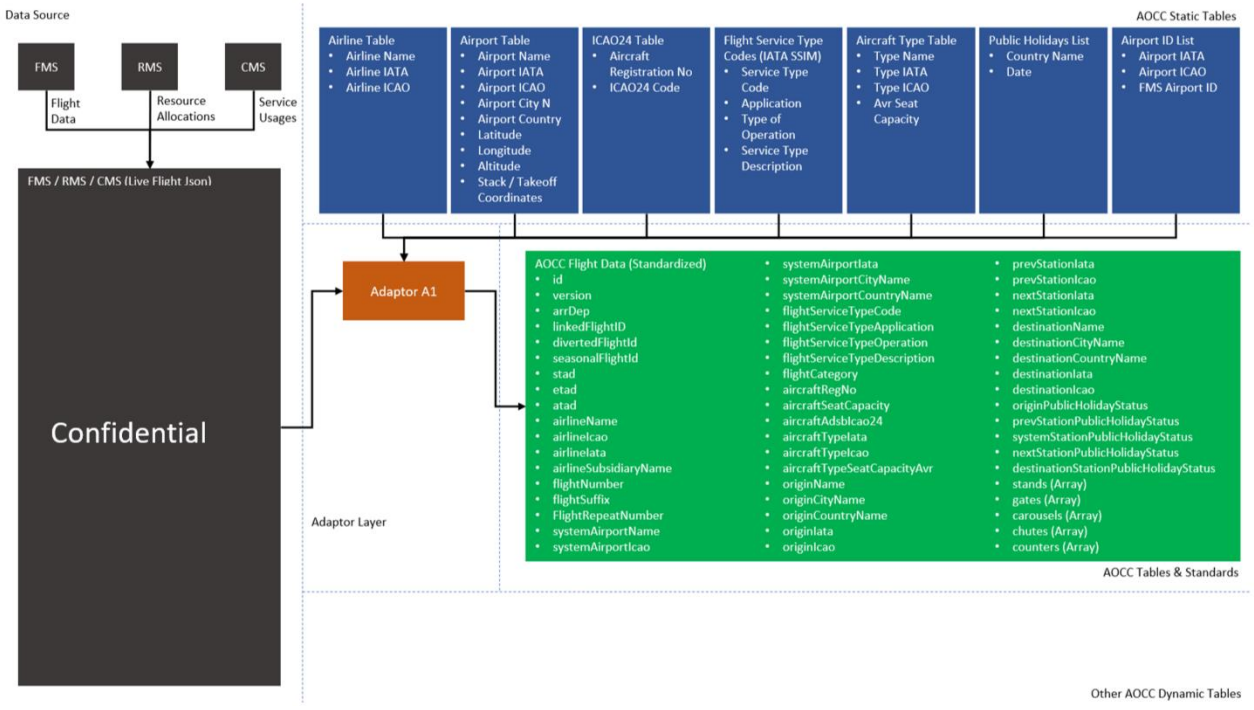

Figure 2 - AOCC Flight Data standarization and integration

Airline and aircraft data are also integrated into the table, with fields like airlineName, airlineIcao, aircraftRegNo, and aircraftType ensuring accurate identification and matching with the corresponding

tables in the Airport Operations Control Center (AOCC). These fields are essential for correctly identifying the operators and types of aircraft involved in each flight.

Service and operational descriptions are another critical area covered by the table. For instance, flightServiceTypeCode and flightServiceTypeApplication categorize the flight's operational nature, such as whether it is a passenger service, cargo, or charter, aligning with IATA standards. Additionally, the flightCategory field indicates whether a flight is international or domestic, a determination made by the AOCC system.

The table also addresses location information, with fields like systemAirportName, originName, and destinationName, along with their corresponding IATA/ICAO codes, linking each flight to specific airports. It details how previous, current, and next stations are tracked, particularly for multi-leg flights. Public holiday status fields, such as originPublicHolidayStatus and destinationPublicHolidayStatus, assess whether flights are impacted by public holidays at the origin or destination, which could affect operations.

Resource allocations for flights are another focus, with resources like stands, gates, carousels, chutes, and counters being dynamically tracked based on the information provided in the flight messages. The resource fields are defined dynamically and parsed from the resourceAllocations list in the FMS message, ensuring that all necessary resources are accurately accounted for and managed.

Finally, the table provides detailed instructions for adapting these fields to the AOCC system, specifying when to use certain codes, how to resolve conflicting information, and where to retrieve specific data from the appropriate tables. These adaptor settings are crucial for ensuring that the data integration process is smooth and that all relevant information is accurately represented in the AOCC system.

Field descriptions:

- id: A unique identifier for the flight record, used to track and manage individual flights within the airport's systems.
- version: Indicates the version of the flight record, useful for tracking updates and ensuring the most current data is being used.
- arrDep: Specifies whether the flight is an arrival or departure, helping to categorize and manage flights accordingly.
- linkedFlightID: References another flight that is linked to this one, such as a connecting flight, providing a way to associate related flights within the system.
- divertedFlightId: Identifies a flight that has been diverted, linking it to the new flight ID created after the diversion.
- seasonalFlightId: Represents a seasonal flight ID, used to group flights that are part of a seasonal schedule or pattern.
- stad: Scheduled Time of Arrival or Departure, indicating the planned time for the flight's arrival or departure.
- etad: Estimated Time of Arrival or Departure, providing an updated time based on current flight status and conditions.
- atad: Actual Time of Arrival or Departure, reflecting the real-time event as it occurs.
- airlineName: The name of the airline operating the flight, used for display and reference purposes.

- airlineIcao: The ICAO code of the airline, a four-letter code that uniquely identifies the airline in international aviation.
- airlineIata: The IATA code of the airline, a two-letter code commonly used in passenger and freight documentation.
- airlineSubsidiaryName: If the airline has a subsidiary operating the flight, this field contains the name of that subsidiary.
- flightNumber: The number assigned to the flight by the airline, used to identify and track the flight.
- flightSuffix: A suffix added to the flight number, typically used to distinguish between multiple legs or segments of a flight.
- FlightRepeatNumber: Indicates how many times a specific flight schedule is repeated within a certain period.
- systemAirportName: The name of the airport where the system is located, used to identify the airport within the system.
- systemAirportIcao: The ICAO code of the system airport, uniquely identifying it within international aviation databases.
- systemAirportIata: The IATA code of the system airport, commonly used in ticketing and baggage handling.
- systemAirportCityName: The city where the system airport is located, providing geographic context.
- systemAirportCountryName: The country where the system airport is located, used for further geographic context.
- flightServiceTypeCode: A code representing the type of service the flight provides, such as passenger, cargo, or charter.
- flightServiceTypeApplication: Describes the specific application or context in which the service type is used, like scheduled or non-scheduled service.
- flightServiceTypeOperation: Defines the operational nature of the flight, whether it's for passengers, cargo, mail, or other purposes.
- flightServiceTypeDescription: Provides a detailed description of the flight service type, explaining what the service entails.
- flightCategory: Classifies the flight as domestic, international, or other categories, helping to manage and organize flights.
- aircraftRegNo: The registration number of the aircraft, uniquely identifying the specific airplane used for the flight.
- aircraftSeatCapacity: The total number of seats available on the aircraft, used for passenger planning and load management.
- aircraftAdsbIcao24: The ADS-B ICAO24 code, a unique 24-bit identifier for the aircraft used in surveillance and tracking.
- aircraftTypeIata: The IATA code representing the type of aircraft, providing information about the aircraft model.
- aircraftTypeIcao: The ICAO code representing the type of aircraft, used for detailed flight and operational planning.

- aircraftTypeSeatCapacityAvr: The average seat capacity for the aircraft type, used for estimating passenger numbers and load factors.
- originName: The name of the origin airport where the flight is departing from.
- originCityName: The city where the origin airport is located, providing geographic context for the flight.
- originCountryName: The country where the origin airport is located, further specifying the flight's starting point.
- originIata: The IATA code of the origin airport, used in ticketing and baggage handling.
- originIcao: The ICAO code of the origin airport, used in flight planning and air traffic control.
- prevStationIata: The IATA code of the previous station or airport, indicating the last point of departure before the current segment.
- prevStationIcao: The ICAO code of the previous station or airport, used for tracking and operational purposes.
- nextStationIata: The IATA code of the next station or airport, indicating the upcoming destination after the current segment.
- nextStationIcao: The ICAO code of the next station or airport, used for flight planning and operational management.
- destinationName: The name of the destination airport where the flight is scheduled to land.
- destinationCityName: The city where the destination airport is located, providing geographic context for the flight.
- destinationCountryName: The country where the destination airport is located, giving additional geographic information.
- destinationIata: The IATA code of the destination airport, used in passenger and baggage processing.
- destinationIcao: The ICAO code of the destination airport, used for flight routing and air traffic control.
- originPublicHolidayStatus: Indicates whether the flight's origin is experiencing a public holiday, which could impact operations.
- prevStationPublicHolidayStatus: Shows whether the previous station or airport is observing a public holiday, potentially affecting flight schedules.
- systemStationPublicHolidayStatus: Indicates if the system airport is under a public holiday, which might influence airport operations.
- nextStationPublicHolidayStatus: Displays whether the next station or airport will be affected by a public holiday during the flight's schedule.
- destinationStationPublicHolidayStatus: Indicates whether the destination airport is observing a public holiday, which could affect the arrival process and passenger handling.

2.2 AOCC Pax Analyzer Pax Data Table & AOCC Pax Analyzer Flight Data

The AOCC Pax Analyzer Pax Data Table outlines the integration of passenger data from the Pax Analyzer Kafka Topics is matched and utilized within the system. This table is essential for mapping and aligning

passenger-related data for flights, ensuring that all relevant information is captured and used effectively in the airport's operational processes.

The Data Examples column provides sample data to illustrate what the actual data might look like. For example, flightID is represented by "13456", while flightKey is shown as "TK-2355-20240801-ADB-BJV", which includes the flight number, date, and airport codes. Together with these informations, pax data can be mapped with other systems data.

The Adaptor Settings column provides detailed instructions on how to process and match this data within the AOCC system. For example, it explains that the flightKey will be generated and used to collect results in the Pax Analyzer's passenger table specifically for the associated flight. It also describes how to match the ID of AOCC Flight Timings with Pax Analyzer Data, which is particularly important for departure flights, by using a combination of airline IATA code, flight number, date, and station codes.

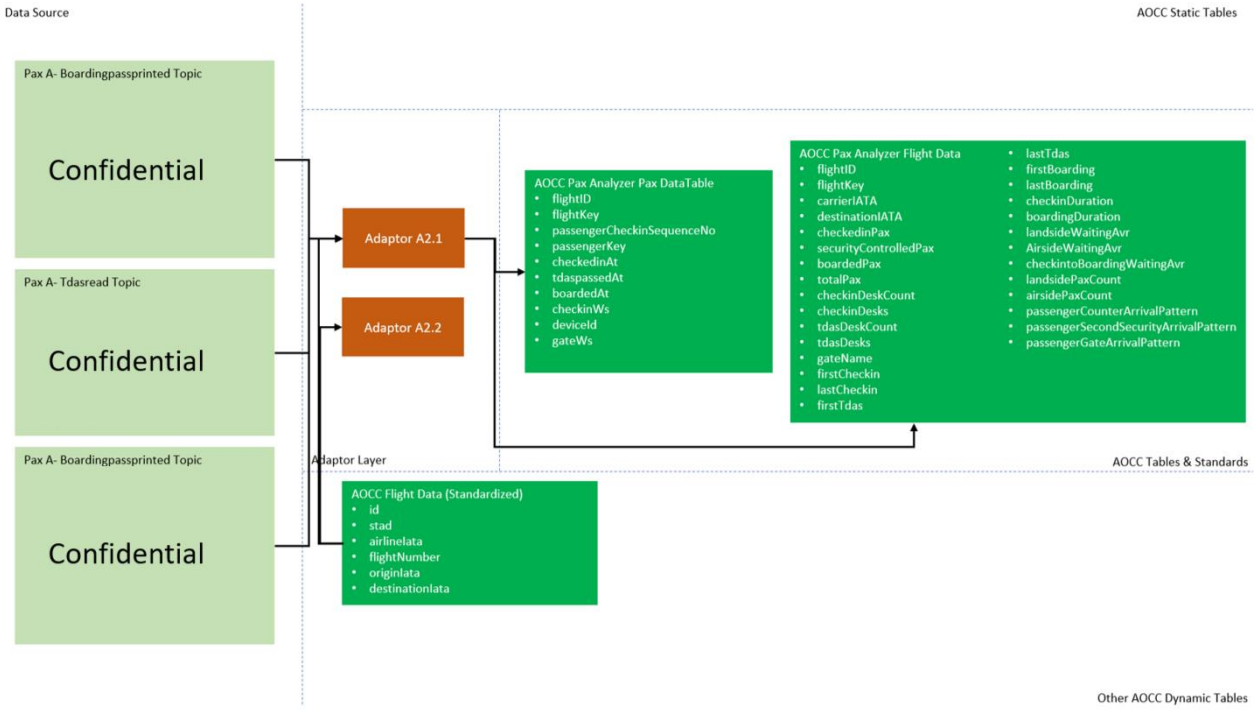

Figure 3 – AOCC Pax Analyzer data standardization and integration

Each data field in this table is carefully matched to ensure that all passenger-related events, from check-in to boarding, are accurately tracked and recorded.

On the other hand, AOCC Pax Analyzer Flight Data table merge passenger data of the same flight.

Field descriptions:

- flightID: This is a unique identifier assigned to each flight. It is used to track and manage the flight within the airport's operational systems.
- flightKey: The flightKey is a generated value that uniquely identifies a flight, combining information such as the flight number, date, and relevant airport codes. It is used to link specific passenger data to the corresponding flight.
- CarrierIATA: The carrierIATA refers to the IATA code of the airline operating the flight. This code is a two-character identifier assigned by the International Air Transport Association (IATA) to each airline.
- destinationIATA: The destinationIATA is the IATA code of the destination airport where the flight is scheduled to land. This three-character code uniquely identifies the airport in global aviation.
- checkedinPax: This field represents the number of passengers who have successfully checked in for the flight, meaning they have printed their boarding passes at the check-in desks.
- securityControlledPax: The securityControlledPax field indicates the number of passengers who have passed through security checks or passport control for the flight.
- boardedPax: This field captures the number of passengers who have boarded the flight, meaning they have passed the final boarding check at the gate.
- totalPax: The totalPax field reflects the total number of unique passengers who have been seen at least one processing point for the flight, such as check-in, security, or boarding.
- checkinDeskCount: This field shows the number of unique check-in workstations (desks) that have been used to check in passengers for the flight.
- checkinDesks: The checkinDesks field lists the specific IDs of the check-in workstations that processed passengers for the flight.
- tdasDeskCount: This field indicates the number of unique workstations used for passport control or security checks for passengers on the flight.
- tdasDesks: The tdasDesks field lists the specific IDs of the workstations used for passport control or security checks for passengers on the flight.
- gateName: This field provides the ID of the boarding gate workstation that was used for the flight, indicating where passengers passed the final boarding check.
- firstCheckin: The firstCheckin field records the earliest time a passenger was checked in for the flight, adjusted to the local time zone.
- lastCheckin: This field records the latest time a passenger was checked in for the flight, adjusted to the local time zone.
- firstTdas: The firstTdas field captures the earliest time a passenger passed through passport control or security for the flight, adjusted to the local time zone.
- lastTdas: This field records the latest time a passenger passed through passport control or security for the flight, adjusted to the local time zone.

- firstBoarding: The firstBoarding field indicates the earliest time a passenger was recorded as boarding the flight, adjusted to the local time zone.
- lastBoarding: This field captures the latest time a passenger was recorded as boarding the flight, adjusted to the local time zone.
- checkinDuration: The checkinDuration is calculated as the difference between the lastCheckin and firstCheckin times, providing the total duration of the check-in process for the flight.
- boardingDuration: This field is calculated as the difference between the lastBoarding and firstBoarding times, indicating the total duration of the boarding process for the flight.
- landsideWaitingAvr: The landsideWaitingAvr field represents the average duration passengers spent moving from the check-in desk to the security control or passport control desk for this flight, measured in minutes.
- AirsideWaitingAvr: This field indicates the average duration passengers spent moving from the security control or passport control desk to the boarding gate for this flight, measured in minutes.
- checkintoBoardingWaitingAvr: The checkintoBoardingWaitingAvr field provides the average duration from the check-in desk to the boarding gate for this flight, measured in minutes.
- landsidePaxCount: This field shows the number of passengers who have checked in at the desk but have not yet passed through security or passport control for the flight.
- airsidePaxCount: The airsidePaxCount field indicates the number of passengers who have passed through security or passport control but have not yet boarded the flight.
- passengerCounterArrivalPattern: This field tracks the number of flight passengers who arrived at the check-in counters within specific time intervals, providing a pattern of passenger arrival at check-in.
- passengerSecondSecurityArrivalPattern: This field tracks the number of flight passengers who arrived at the second security checkpoint within specific time intervals, providing a pattern of passenger arrival at the secondary security check.
- passengerGateArrivalPattern: This field captures the number of flight passengers who arrived at the gate desk within specific time intervals, providing a pattern of passenger arrival at the boarding gate.

2.3 AOCC ADSB Streaming (Standardized)

The data received from ADS-B signals does not directly indicate the phases of flights. To derive flight milestones from ADS-B signal data, certain processing steps are required. The details of these processes are explained in Deliverable 2.4.

The following diagram illustrates how ADSB data will be stored within the AOCC system and indicates which internal tables should be utilized.

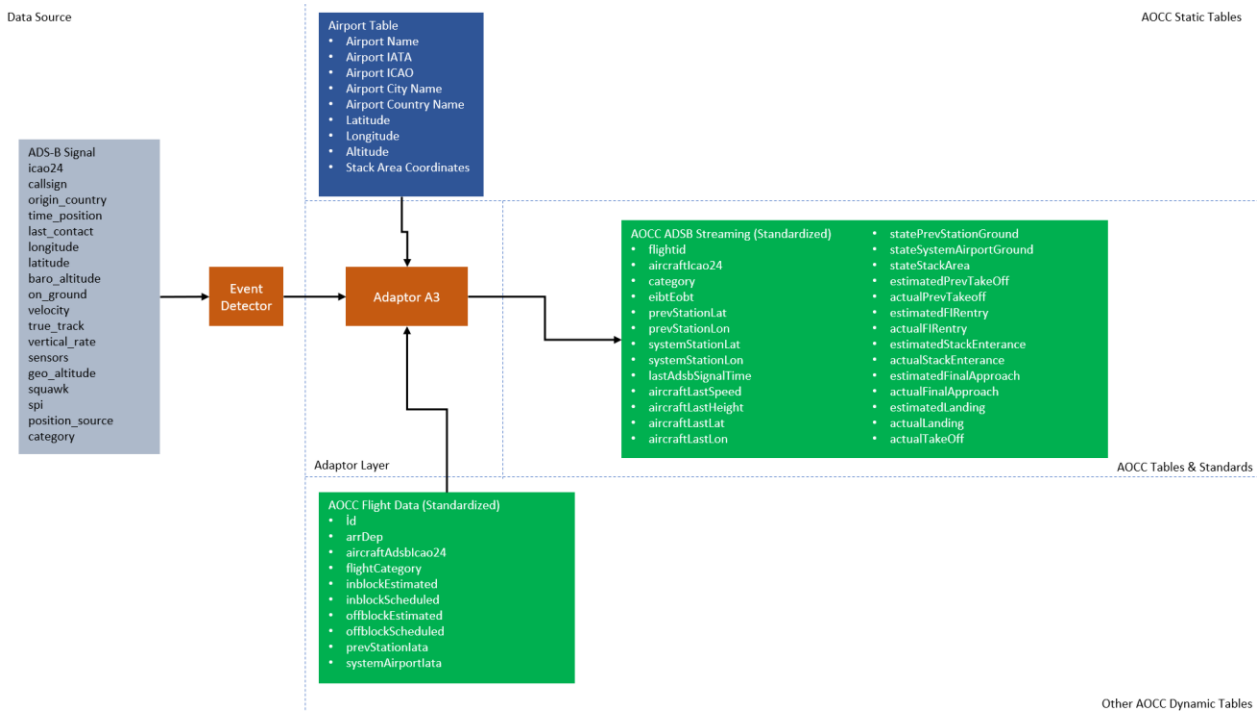

Figure 4 – AOCC ADS-B data standardization and integration

Field descriptions:

- flightid: This is a unique identifier for the flight, sourced from the AOCC Flight Data. It is essential for tracking and managing specific flight details across different systems.
- aircraftIcao24: This field captures the 24-bit ICAO address of the aircraft. This globally unique identifier is used in surveillance systems to track the aircraft and is sourced from the AOCC Flight Data.
- Category: The Category field indicates the type of flight operation, such as whether the flight is an arrival or departure. It is derived from the flight category data within the AOCC system.
- eibtEobt: This field represents the Estimated In-Block Time (for arrivals) or Estimated Off-Block Time (for departures). It is used to predict when a flight will arrive at or depart from the gate, based on either estimated or scheduled times from the AOCC Flight Timing Data.
- prevStationLat & prevStationLon: These fields provide the latitude and longitude of the previous station (airport) from which the flight arrived. The values are derived by retrieving the IATA code of the previous station from AOCC Flight Data and cross-referencing it with the Airport Table to get the corresponding geographic coordinates.
- systemStationLat & systemStationLon: These fields indicate the latitude and longitude of the system airport, i.e., the airport currently managing the flight. The system retrieves the IATA code of the airport and cross-references it with the Airport Table to determine the exact coordinates.
- lastAdsbSignalTime: This field records the last time an ADS-B signal was received from the aircraft. It is essential for tracking the aircraft's most recent known position and operational status.

- aircraftLastSpeed: This field shows the aircraft's last recorded speed, as detected by the ADS-B system. It is crucial for understanding the aircraft's velocity during its last known position.
- aircraftLastHeight: This field captures the last recorded altitude of the aircraft, as detected by the ADS-B system. This information is vital for monitoring the flight's altitude and ensuring safe navigation.
- aircraftLastLat & aircraftLastLon: These fields indicate the last known latitude and longitude of the aircraft, respectively. They are sourced from the ADS-B system and are crucial for real-time tracking of the aircraft's position.
- statePrevStationGround: This Boolean field (Yes/No) indicates whether the aircraft is on the ground near the previous station. It is evaluated based on the aircraft's speed, proximity to the previous airport, and altitude.
- stateSystemAirportGround: Similar to statePrevStationGround, this field indicates whether the aircraft is on the ground at the system airport. It is assessed using the aircraft's speed, location relative to the system airport, and altitude.
- stateStackArea: This field indicates whether the aircraft has entered the stack area near the system airport. The stack area is a holding pattern where aircraft may wait before landing. The system checks the aircraft's position and altitude to determine if it is within this area.
- estimatedPrevTakeOff: This field estimates the time the aircraft took off from the previous station. The calculation involves using the flight's expected or scheduled times and adjusting for the distances between airports and the aircraft's speed.
- actualPrevTakeoff: This field records the actual takeoff time from the previous station, determined by real-time monitoring of the ADS-B signal. The exact time is logged when the aircraft is detected leaving the ground.
- estimatedFIRentry: This field predicts when the aircraft will enter the Flight Information Region (FIR) boundary, particularly important for international flights. The calculation is based on the aircraft's current position and speed relative to the FIR boundary.
- actualFIRentry: This field records the exact time when the aircraft crosses into the FIR, using real-time data from the ADS-B system. It is especially relevant for monitoring the entry of international flights into national airspace.
- estimatedStackEnterance: This field estimates when the aircraft will enter the stack area near the system airport. The estimate is based on the aircraft's flight path and expected arrival time in the stack area.
- actualStackEnterance: This field logs the actual time the aircraft enters the stack area, confirmed by real-time ADS-B data. It provides a precise timestamp for when the aircraft begins holding in preparation for landing.
- estimatedFinalApproach: This field estimates when the aircraft will begin its final approach for landing. The estimate is based on the aircraft's position relative to the runway and its descent rate.
- actualFinalApproach: This field records the actual time the aircraft starts its final approach, determined by ADS-B data. It typically occurs when the aircraft descends to a specific altitude and aligns with the runway.

- estimatedLanding: This field calculates the expected landing time, considering the aircraft's final approach and descent. The estimation is adjusted based on the aircraft's approach speed and distance to the runway.
- actualLanding: This field records the actual landing time of the aircraft, confirmed by real-time ADS-B data when the aircraft touches down on the runway and switches to ground status.
- actualTakeOff: For departure flights, this field captures the exact takeoff time. It is determined by monitoring the ADS-B signal and confirming when the aircraft leaves the ground at the system airport.

2.4 AOCC Flight Timings (Standardized)

This table represents the timing data extracted from the FMS, RMS, and CMS systems. Each timing field has Scheduled, Estimated, Predicted, and Actual versions. The Predicted values are generated by the Central Prediction Service, while the Scheduled, Estimated, and Actual data are standardized values provided by airlines, aviation authorities, and ground handling companies. The table includes critical milestones for all processes related to the aircraft, passengers, and baggage. In some cases, data for the same field may come from multiple systems, and in such instances, a prioritization is applied to determine which data to use. The adapter settings may vary from one airport to another, but the AOCC Flight Timing table can be maintained as a standard. Although there is no subsystem at İzmir Adnan Menderes Airport that currently generates the following data, they are retained in the table to preserve the standard: atcFpa, ctotAllocation, ghStart, finalTobt, asrt, asat, deboardingStart, deboardingEnd, fuelLoadStart, fuelLoadEnd, wasteUnloadStart, wasteUnloadEnd, cleaningStart, cleaningEnd.

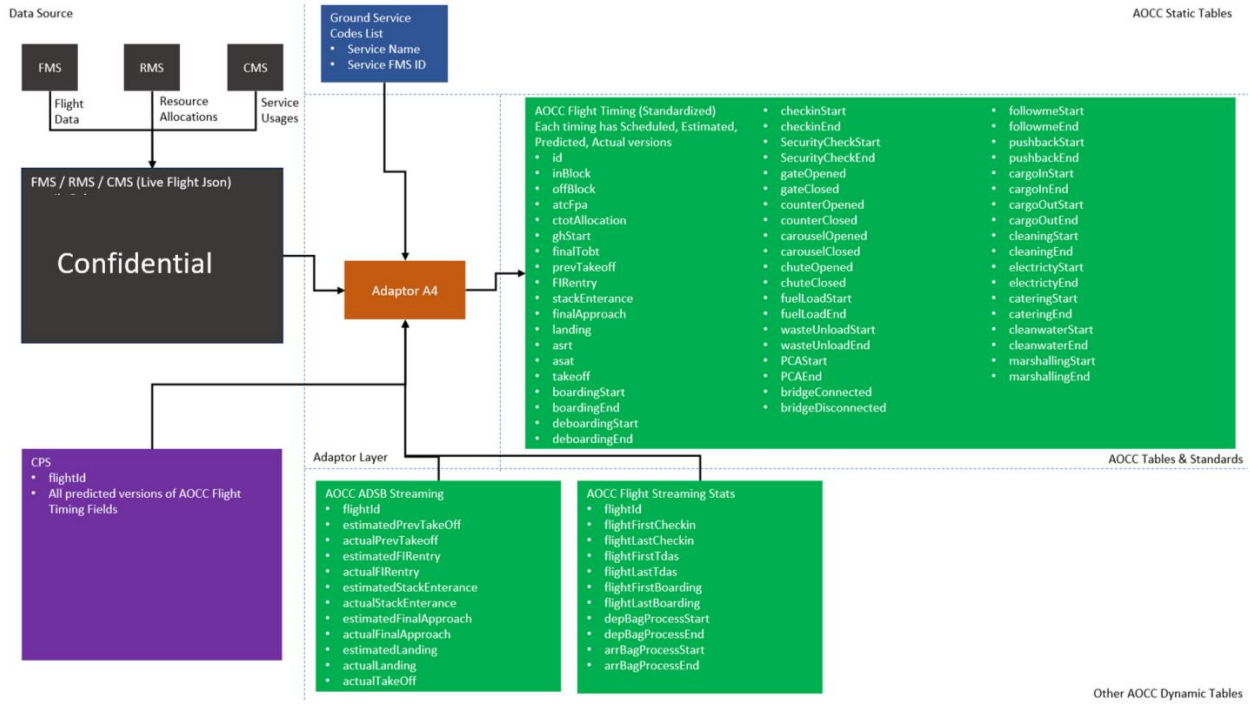

Figure 5 – AOCC Flight Timing data standardization and integration

Field descriptions:

- inBlock: The time when the aircraft reaches its parking position at the gate or stand and the wheels are chocked. This is an essential milestone indicating that the aircraft has safely arrived and is ready for the next stages of ground operations.
- offBlock: The time when the aircraft leaves its parking position at the gate or stand. This marks the beginning of the aircraft's taxiing phase toward the runway for takeoff.
- atcFpa: This stands for Air Traffic Control Final Pushback Approval. It is the time when ATC gives the final approval for the aircraft to push back from the gate, marking the start of its taxiing phase.
- ctotAllocation: Calculated Take-Off Time (CTOT) Allocation refers to the time slot allocated by air traffic control for the aircraft to take off. This is a critical timing to ensure smooth air traffic management and avoid congestion.
- ghStart: Ground Handling Start is the time when ground handling operations begin for the aircraft. This includes activities such as loading and unloading baggage, refueling, and other necessary services.
- finalTobt: Final Target Off-Block Time is the last agreed time when the aircraft is expected to depart from its parking position, often coordinated with air traffic control and ground services.
- prevTakeoff: This field represents the time the aircraft took off from the previous airport. It's important for tracking the aircraft's journey and ensuring accurate arrival predictions.
- FIRentry: Flight Information Region (FIR) Entry is the time when the aircraft enters a specific FIR, which is a defined area of airspace managed by a particular country's air traffic control.
- stackEnterance: The time when the aircraft enters a stack area near the destination airport. Stack areas are holding patterns where aircraft may circle before being cleared for final approach and landing.
- finalApproach: The time when the aircraft begins its final approach towards the runway, indicating that it is in the last phase of the landing sequence.
- landing: The time when the aircraft's wheels touch down on the runway, marking the completion of the flight.
- asrt: Actual Start of Runway Time, indicating when the aircraft begins its takeoff roll on the runway.
- asat: Actual Stop of Aircraft Time, marking when the aircraft comes to a complete stop after landing.
- takeoff: The moment when the aircraft leaves the ground and begins its flight. This is a critical point in the flight operation.
- boardingStart: The time when passengers start boarding the aircraft. This is an important part of the pre-departure process.
- boardingEnd: The time when the last passenger boards the aircraft, marking the completion of the boarding process.
- deboardingStart: The time when passengers begin to disembark from the aircraft after it has arrived at its destination.
- deboardingEnd: The time when the last passenger has exited the aircraft, completing the deboarding process.
- checkinStart: The time when the check-in counters open, allowing passengers to check in for their flight.

- checkinEnd: The time when the check-in counters close, marking the end of the check-in process.
- SecurityCheckStart: The time when security screening for passengers begins. This is a critical security measure before passengers proceed to the boarding gate.
- SecurityCheckEnd: The time when the last passenger has passed through security screening.
- gateOpened: The time when the boarding gate is opened for passengers to start boarding the aircraft.
- gateClosed: The time when the boarding gate is closed, typically signaling that boarding is complete and the aircraft is preparing for departure.
- counterOpened: The time when the check-in counters are opened for a flight, allowing passengers to check in and drop off baggage.
- counterClosed: The time when the check-in counters are closed, usually shortly before the flight's departure time.
- carouselOpened: The time when the baggage claim carousel is activated, signaling that passengers can start collecting their checked luggage.
- carouselClosed: The time when the baggage claim carousel is deactivated, typically after all baggage from the flight has been collected.
- chuteOpened: The time when the baggage chute, used for loading or unloading luggage, is opened.
- chuteClosed: The time when the baggage chute is closed after all luggage has been loaded or unloaded.
- fuelLoadStart: The time when refueling of the aircraft begins, a critical process to ensure the aircraft has enough fuel for its journey.
- fuelLoadEnd: The time when refueling is completed, and the aircraft is ready for departure.
- wasteUnloadStart: The time when waste unloading from the aircraft begins, part of the aircraft cleaning and maintenance process.
- wasteUnloadEnd: The time when waste unloading is completed.
- PCAStart: Pre-Conditioned Air Start refers to the time when ground-based air conditioning is connected to the aircraft to maintain cabin temperature while on the ground.
- PCAEnd: Pre-Conditioned Air End marks the time when the ground-based air conditioning is disconnected from the aircraft.
- bridgeConnected: The time when the passenger boarding bridge is connected to the aircraft, allowing passengers to board or deboard.
- bridgeDisconnected: The time when the passenger boarding bridge is disconnected from the aircraft, typically signaling that the aircraft is ready for departure.
- followmeStart: The time when the "follow-me" vehicle, used to guide the aircraft to the runway or gate, begins its operation.
- followmeEnd: The time when the "follow-me" vehicle completes its operation.
- pushbackStart: The time when the aircraft pushback process begins, moving the aircraft away from the gate for taxiing to the runway.
- pushbackEnd: The time when the pushback process is completed.
- cargoInStart: The time when cargo loading onto the aircraft begins.
- cargoInEnd: The time when cargo loading is completed.
- cargoOutStart: The time when cargo unloading from the aircraft begins.

- cargoOutEnd: The time when cargo unloading is completed.
- cleaningStart: The time when cleaning operations on the aircraft begin.
- cleaningEnd: The time when cleaning operations are completed.
- electrictyStart: The time when the aircraft is connected to ground power to supply electricity while on the ground.
- electrictyEnd: The time when the ground power connection is disconnected.
- cateringStart: The time when catering services begin loading food and beverages onto the aircraft.
- cateringEnd: The time when catering services are completed.
- cleanwaterStart: The time when clean water begins being supplied to the aircraft.
- cleanwaterEnd: The time when clean water supply operations are completed.
- marshallingStart: The time when the marshaller begins guiding the aircraft to its parking position.
- marshallingEnd: The time when the marshalling process is completed, and the aircraft is in its final parking position.

2.5 AOCC ACDM States (Standardized)

Airport Collaborative Decision Making (A-CDM) is a set of operational processes aimed at improving the efficiency of airport operations by enhancing the collaboration between various airport stakeholders, including airlines, air traffic control, ground handlers, and airport operators. A-CDM focuses on the sharing of accurate and timely information to optimize decision-making, reduce delays, and improve the overall flow of air traffic both on the ground and in the air.

The milestones in A-CDM are critical points in the airport and flight operation processes where specific actions or decisions are made. These milestones are used to ensure that all parties involved in airport operations have a clear understanding of the status of a flight at any given time, allowing for better coordination and resource allocation.

İzmir Airport is not fully operational as an A-CDM (Airport Collaborative Decision Making) airport, but its subsystems do allow for the monitoring of many A-CDM milestones. While the data structure has been designed with the A-CDM standard in mind, unfortunately, some fields cannot be populated for İzmir Airport.

The AOCC A-CDM States (Standardized) Table will utilize data from the following sources: AOCC Flight Streaming Stats, AOCC ADSB Streaming, Central Prediction Services, and FMS/RMS/CMS (Live Flight JSON).

These sources will provide the necessary information to standardize and monitor A-CDM states effectively within the AOCC framework.

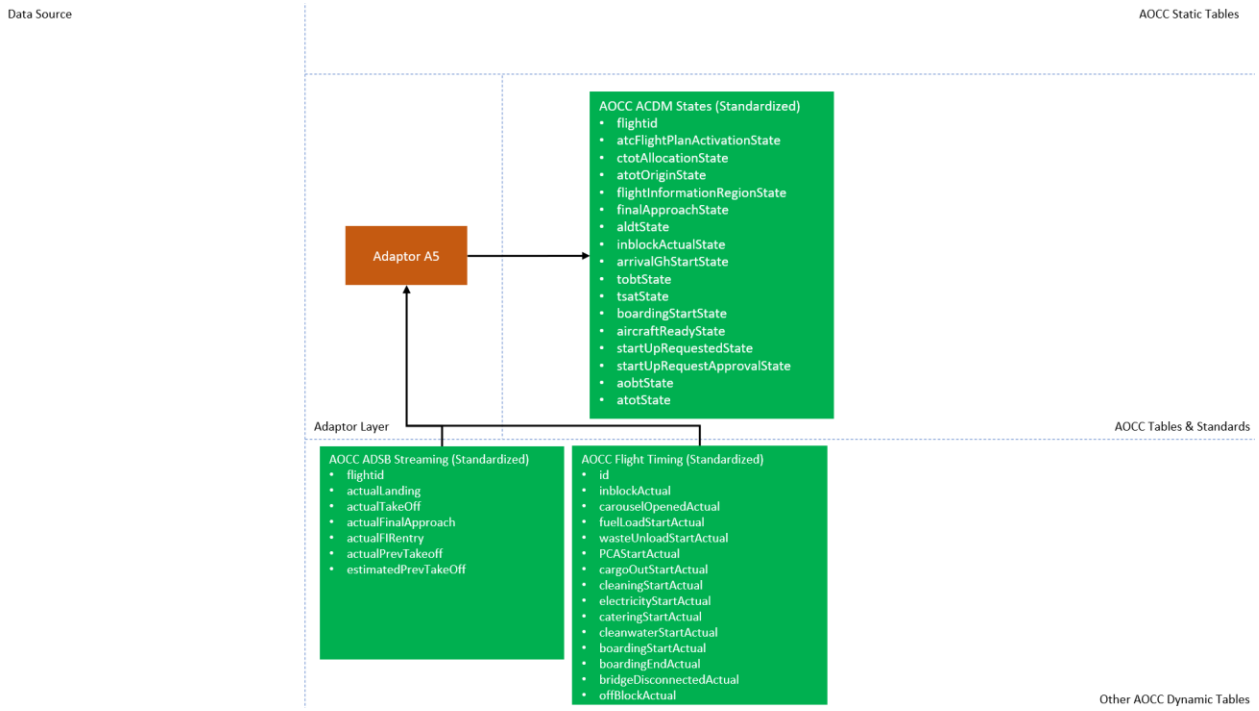

Figure 6 – AOCC ACDM States data standardization and integration

Adaptor settings:

Field	CDM Milestones	Value	Data Source	Calculation	Description
atcFlightPlanActivationState	ATC Flight Plan activation	Predicted Yes/ Predicted No/ Yes	ADSB	"Predicted Yes": if (current time > estimatedPrevTakeOff - 3h) and (actualPrevTakeoff is null) "Predicted No": if (current time < estimatedPrevTakeOff - 3h) and (actualPrevTakeoff is null) "Yes": if (actualPrevTakeoff is not null)	Milestone 1 Normally this takes place 3 hours before EOBT, however it may be later. In some cases a repetitive flight plan (RFPL) has been submitted, covering daily or weekly flights.
ctotAllocationState	EOBT - 2 hr	Predicted Yes/ Predicted No/ Yes	ADSB	"Predicted Yes": if (current time > estimatedPrevTakeOff - 2h) and (actualPrevTakeoff is null) "Predicted No": if (current time < estimatedPrevTakeOff - 2h) and (actualPrevTakeoff is null) "Yes": if (actualPrevTakeoff is not null)	Milestone 2 If the flight is regulated, a CTOT is issued at EOBT–2h
atotOriginState	Take off from outstation	Yes/Predicted Yes/Predicted No	ADSB	"Yes":if (actualPrevTakeoff is not null) "Predicted Yes": if (actualPrevTakeoff is null) and (current time > estimatedPrevTakeOff) "Predicted No": if (actualPrevTakeoff is null) and (current time < estimatedPrevTakeOff)	Milestone 3 The ATOT from the outstation (ADEP)
flightInformationRegionState	Local Radar Update	Yes/No	ADSB	"Yes": if (actualFIRentry in not null)	Milestone 4 The flight enters the FIR (Flight Information Region) or the local airspace of the destination airport.
finalApproachState	Final Approach	Yes/No	ADSB	"Yes": if (actualFinalApproach is not null)	Milestone 5 The flight enters the Final Approach phase at the system airport.

aldtState	Landed	Yes/No	ADSB	"Yes": if (actualLanding is not null)	Milestone 6 ALDT – Actual Landing Time. This is the time that an aircraft touches down on a runway. (Equivalent to ATC ATA – Actual Time of Arrival landing, ACARS=ON).
inblockActualState	Inblock Time	Yes/No	AOCC Flight Timings	"Yes": if (inblockActual is not null)	Milestone 7 AIBT - Actual In-Block Time. This is the time that an aircraft arrives in-blocks. (Equivalent to Airline /Handler ATA – Actual Time of Arrival, ACARS = IN)
arrivalGhStartState	Ground Handling Started	Yes/No	AOCC Flight Timings	"Yes": if any of the following field is not null carouselOpenedActual fuelLoadStartActual wasteUnloadStartActual PCAStartActual cargoOutStartActual cleaningStartActual electricityStartActual cateringStartActual cleanwaterStartActual	Milestone 8 Ground Handling Started
tobtState	Final Confirmation of the TOBT	Unknown			Milestone 9 The time at which the Aircraft Operator or Ground Handler provide their most accurate TOBT taking into account the operational situation
tsatState	The time ATC issues the Target Start Up Approval Time	Unknown			Milestone 10 The time ATC issues the Target Start Up Approval Time
boardingStartState	Boarding Start	Yes/No	AOCC Flight Timings	"Yes": if (boardingStartActual is not null)	Milestone 11 The gate is open for passengers to physically start boarding (independent of whether boarding takes place via an air-bridge/pier, aircraft steps or coaching to a stand).
aircraftReadyState	Aircraft Ready	Yes/No	AOCC Flight Timings	"Yes": if (boardingEndActual is not null and boardingEndActual < current time - 5 minutes) or (bridgeDisconnectedActual is not null)	Milestone 12 The time when all doors are closed, boarding bridge removed, push back vehicle connected, ready to taxi immediately upon reception of TWR instructions (ARDT)
startUpRequestedState	Start up Requested	Unknown			Milestone 13 The time that start up is requested (ASRT).
startUpRequestApprovalState	Start up Request Approved	Unknown			Milestone 14 ASAT - Actual start up Approval Time. This is the time that an aircraft receives its start up approval.
aobtState	AOBT – Actual Off-Block Time	Yes/No	AOCC Flight Timings, ADSB	"Yes": if (offBlockActual is not null)	Milestone 15 AOBT – Actual Off-Block Time. The time the aircraft pushes back/vacates the parking position (Equivalent to Airline/Handler ATD – Actual Time of Departure ACARS=OUT).
atotState	Takeoff	Yes/No	ADSB	"Yes": if (actualTakeOff is not null)	Milestone 16 Take off

2.6 AOCC PFM Data

Passenger Flow Management Platform constantly measures real-time KPIs and statistics such as: Passenger counts and queue lenghts in defined polygons, Passenger wait times according to user defined avr. processing speeds and active processing point counts, Passenger speed calculation for x-ray and gate processing points, Activeness of x-ray, gate, passport control, checkin counter desks, etc., Toilet occupancies. However, to align the PFM system data with the AOCC concept, some point and zone mappings are required. Additionally, PFM system data will provide input to the Flight Streaming, Point and Zone Statistics tables, alongside Pax Analyzer and BRS.

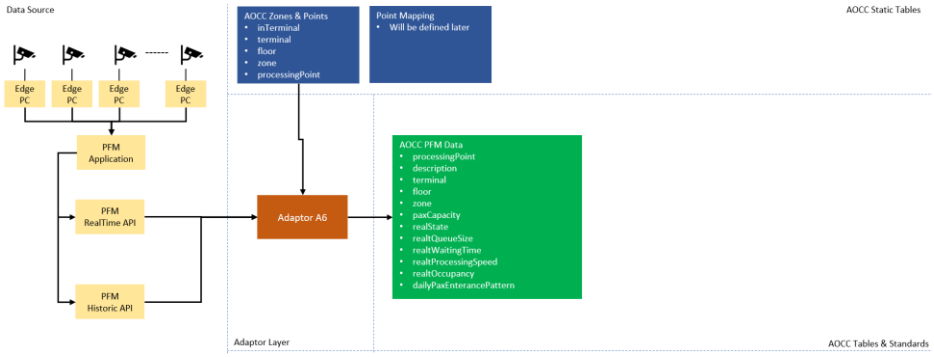

Figure 7 – AOCC PFM Data standardization and integration

Field descriptions:

- processingPoint: This column specifies the name of the specific checkpoint or processing area within the airport, such as "X-ray 01."
- Description: Provides a detailed description of the processing point.
- Terminal: Indicates which terminal the processing point is located in.
- Floor: Specifies the floor where the processing point is located.
- Zone: Refers to the specific zone within the terminal where the processing point is situated. It helps in defining the location more precisely. The zone is managed under AOCC (Airport Operations Control Center) Zones & Points standards.
- paxCapacity: Represents the passenger capacity that the processing point can handle at any given time. This capacity is crucial for assessing the flow and potential bottlenecks. Here, it's noted as 30 passengers.
- realtState: Indicates the real-time operational status of the processing point, such as whether it is "Active" or "Passive." The status is derived based on real-time data from an API, where it checks if the number of passing people is above a certain threshold.
- realtQueueSize: Shows the real-time queue size at the processing point. The data is sourced from an API, specifically from the "TotalWaitingPeopleCount" value.
- realtWaitingTime: Represents the estimated waiting time in minutes for passengers in the queue at the processing point. This time is calculated by multiplying the realtQueueSize by 0.75. The result gives a specific waiting time for that processing point in minutes.
- realtProcessingSpeed: Indicates the real-time processing speed at the processing point, expressed as the time taken to process each passenger (in seconds per passenger). This is calculated by dividing 1 by the "TotalPassingVelocity" value from the API.
- realtOccupancy: Reflects the occupancy level of the processing point, calculated as the ratio of the current queue size to the processing capacity (realtQueueSize / 30). In this case, it's 0.77, indicating that 77% of the capacity is being utilized.
- dailyPaxEnterancePattern: Provides a pattern of passenger entries throughout the day. This data is aggregated from the API and shows counts of passengers entering the processing point at different times. The pattern includes timestamps and counts, offering insight into peak times and flow trends.

2.7 AOCC Bag Data, AOCC Pax Bag Data, AOCC Flight Bag Data

The BRS (Baggage Reconciliation System) is a system that operates using baggage data from the Airline Departure Control System (DCS) and barcode scanning events performed by ground handling staff. Initially, airline DCS messages are sent to the SITA system, which then distributes them to all relevant stakeholders. If a passenger is not a transfer passenger for a particular airport and is departing, the baggage message for that passenger is distributed shortly after their check-in, indirectly providing information about when the check-in occurred. The Pax Analyzer system only accesses passenger information when the boarding pass is printed, so BRS's ability to provide additional data about passengers is particularly valuable. Within the BRS system, baggage can be scanned at two different stages for both arrival and departure flights. For departing flights, baggage is first scanned when it enters the sorter area and again when it is loaded onto the aircraft. However, this process can vary between airlines, with some airlines only scanning at a single point. For arriving flights, the first scan typically occurs under the aircraft when the baggage is unloaded, and the second scan happens when the baggage is placed on the carousel. This process also varies between airlines. The BRS system consolidates data from four different tables into a view called v_bag. All necessary data can be accessed through the v_bag table. By combining v_bag data with AOCC Flight Data (Standardized), three tables have been created: AOCC Bag, AOCC Pax Bag, and AOCC Flight Bag. The AOCC Bag table merges only baggage-related data with the AOCC flight structure, while AOCC Pax Bag consolidates all baggage messages for each passenger. The AOCC Flight Bag Data consolidates all baggage data related to a specific flight. Despite the different uses of these data, all of it is ultimately utilized in the AOCC Flight Streaming table.

V_bag field descriptions:

- SYSTEM_AIRPORT: This field indicates the airport system where the data is being recorded or processed. It identifies the specific airport involved in the baggage event.
- STATUS_INDICATOR: This field represents the current status or condition of the baggage or event. It could indicate whether the baggage is in transit, loaded, offloaded, or delivered.
- BIM_CREATE_DATE: This is the date when the Baggage Information Message (BIM) was created. It marks the timestamp when the baggage data was generated or updated in the system.
- AIRLINE_CODE: The unique code representing the airline operating the flight. This code is typically the IATA or ICAO code associated with the airline.
- FLIGHT_NUMBER: This field shows the specific flight number associated with the baggage event. It helps in identifying the exact flight on which the baggage is being transported.
- DEP_AP_CODE:This stands for Departure Airport Code, indicating the airport from which the flight is departing. It is usually an IATA or ICAO code.
- ARR_AP_CODE: This represents the Arrival Airport Code, which is the airport where the flight is scheduled to land and where the baggage is expected to be delivered.
- STD: Scheduled Time of Departure, indicating the planned departure time of the flight. This is crucial for tracking the timing of baggage events relative to the flight schedule.
- STA: Scheduled Time of Arrival, indicating the planned arrival time of the flight. This helps in coordinating baggage handling with the flight's arrival.
- SEQUENCE_NUMBER: This field represents a unique sequence number assigned to the baggage event. It helps in tracking and identifying specific events in a chronological order.

- TAG_NUMBER: The baggage tag number, which is a unique identifier attached to each piece of baggage. This number is used to track the baggage throughout its journey.
- BAGGAGE_EVENT: This field describes the specific event related to the baggage, such as loading, offloading, scanning, or delivery. It helps in identifying the stage of the baggage handling process.
- BAGGAGE_STATUS: The status of the baggage at the time of the event, such as "loaded," "offloaded," or "delivered." It provides information on the current state of the baggage.
- VERSION_SOURCE_INDICATOR: This field indicates the version of the data source or system generating the baggage event. It helps in tracking changes or updates to the data.
- DETERMINED_AUTHORITY: Identifies the authority or entity responsible for determining the baggage status or event, such as the airline, ground handler, or airport authority.
- ACTION_TIME_LOAD_BAG: The timestamp when the baggage was loaded onto the aircraft. This is critical for tracking the handling of the baggage from the point of loading.
- ACTION_TIME_OFF_LOAD_ARR_BAG: The timestamp when the baggage was offloaded from the aircraft upon arrival. This helps in monitoring the baggage handling process upon landing.
- ACTION_TIME_LOAD_TO_HOLD: The time when the baggage was loaded into the hold area, often used for baggage that is temporarily stored before being loaded onto the aircraft.
- ACTION_TIME_DELIVER_ARR_BAG: The timestamp when the baggage was delivered to the arrival area, such as the baggage carousel, for passenger pickup.
- COMPANY_CODE: The code representing the company or entity handling the baggage, such as the ground handling company or the airline.
- BAG_NUMBER: Another identifier for the baggage, which may be used internally within the system to track the piece of baggage. It may be the same as or different from the TAG_NUMBER.

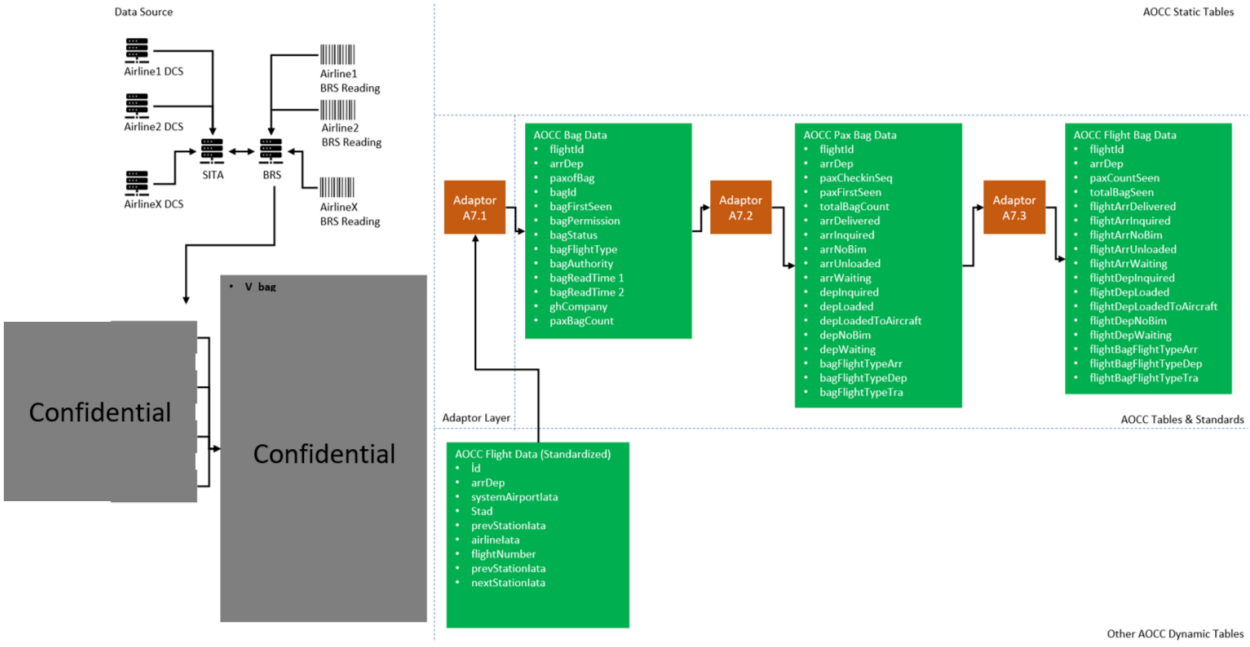

Figure 8 - AOCC Bag Data, Pax Bag Data, Flight Bag Data standardization and integration

AOCC Bag Data adaptor descriptions:

Fields	DataSource	AdaptorSettings
flightId	Confidential	
arrDep		
paxofBag		
bagId		
bagFirstSeen		
bagPermission		
bagStatus		
bagFlightType		
bagAuthority		
bagReadTime 1		
bagReadTime 2		
ghCompany		
paxBagCount		

AOCC Pax Bag Data adaptor settings:

Fields	Data Source	Adaptor Settings
flightId	AOCC Flight Data	For each flight in the list
arrDep	AOCC Flight Data	arrDep value for this flightId will be inserted here
paxCheckinSeq	AOCC Bag Data: paxofBag	For each uniquee pax checkin number in that flight
paxFirstSeen	AOCC Bag Data: bagFirstSeen	There could be many bag record for this passenger, earliest one will be inserted here
totalBagCount	AOCC Bag Data: paxBagCount	There could be many bag record for this passenger, in each of these bag records, paxBagCount should be same so just pick one of them
arrDelivered	AOCC Bag Data: bagStatus	if flight is departure, value will be null. İf flight is arrival, check all bag records of this passenger and count Delivered bags by looking at bagStatus fields of records.
arrInquired	AOCC Bag Data: bagStatus	if flight is departure, value will be null. İf flight is arrival, check all bag records of this passenger and count Inquired bags by looking at bagStatus fields of records.
arrNoBim	AOCC Bag Data: bagStatus	if flight is departure, value will be null. İf flight is arrival, check all bag records of this passenger and count NoBim bags by looking at bagStatus fields of records.
arrUnloaded	AOCC Bag Data: bagStatus	if flight is departure, value will be null. İf flight is arrival, check all bag records of this passenger and count Unloaded bags by looking at bagStatus fields of records.
arrWaiting	AOCC Bag Data: bagStatus	if flight is departure, value will be null. İf flight is arrival, check all bag records of this passenger and count Waiting bags by looking at bagStatus fields of records.
depInquired	AOCC Bag Data: bagStatus	if flight is arrival, value will be null. İf flight is departure, check all bag records of this passenger and count Inquired bags by looking at bagStatus fields of records.
depLoaded	AOCC Bag Data: bagStatus	if flight is arrival, value will be null. İf flight is departure, check all bag records of this passenger and count Loaded bags by looking at bagStatus fields of records.
depLoadedToAircraft	AOCC Bag Data: bagStatus	if flight is arrival, value will be null. İf flight is departure, check all bag records of this passenger and count Loaded To Aircraft bags by looking at bagStatus fields of records.
depNoBim	AOCC Bag Data: bagStatus	if flight is arrival, value will be null. İf flight is departure, check all bag records of this passenger and count No Bim bags by looking at bagStatus fields of records.
depWaiting	AOCC Bag Data: BAGGAGE_STATUS	if flight is arrival, value will be null. İf flight is departure, check all bag records of this passenger and count Waiting bags by looking at bagStatus fields of records.
bagFlightTypeArr	AOCC Bag Data: bagFlightType	if value from AOCC bag data is X then "YES". All bags records of this passenger for that flight must be same
bagFlightTypeDep	AOCC Bag Data: bagFlightType	if value from AOCC bag data is L then "YES". All bags records of this passenger for that flight must be same
bagFlightTypeTra	AOCC Bag Data: bagFlightType	if value from AOCC bag data is T then "YES". All bags records of this passenger for that flight must be same

AOCC Flight Bag Data adaptor settings:

Fields	Data Source	Adaptor Settings
flightId	AOCC flight Data	For each flight in the list
arrDep	AOCC flight Data	arrDep value for this flightId will be inserted here
paxCountSeen	AOCC Pax Bag Data: paxCheckinSeq	Total unique checkin sequence number for this flight.
totalBagSeen	AOCC Pax Bag Data: totalBagCount	Total bag count for this flight. Each total bag count for pax will be summed.
flightArrDelivered	AOCC Pax Bag Data: arrDelivered	if flight is departure, value will be null. Each count for flight pax will be summed.
flightArrInquired	AOCC Pax Bag Data: arrInquired	if flight is departure, value will be null. Each count for flight pax will be summed.
flightArrNoBim	AOCC Pax Bag Data: arrNoBim	if flight is departure, value will be null. Each count for flight pax will be summed.
flightArrUnloaded	AOCC Pax Bag Data: arrUnloaded	if flight is departure, value will be null. Each count for flight pax will be summed.
flightArrWaiting	AOCC Pax Bag Data: arrWaiting	if flight is departure, value will be null. Each count for flight pax will be summed.
flightDepInquired	AOCC Pax Bag Data: depInquired	if flight is arrival, value will be null. Each count for flight pax will be summed.
flightDepLoaded	AOCC Pax Bag Data: depLoaded	if flight is arrival, value will be null. Each count for flight pax will be summed.
flightDepLoadedToAircra	AOCC Pax Bag Data: depLoadedToAircraft	if flight is arrival, value will be null. Each count for flight pax will be summed.
flightDepNoBim	AOCC Pax Bag Data: depNoBim	if flight is arrival, value will be null. Each count for flight pax will be summed.
flightDepWaiting	AOCC Pax Bag Data: depWaiting	if flight is arrival, value will be null. Each count for flight pax will be summed.
flightBagFlightTypeArr	AOCC Pax Bag Data: bagFlightTypeArr	For each pax of flight, sum values
flightBagFlightTypeDep	AOCC Pax Bag Data: bagFlightTypeDep	For each pax of flight, sum values
flightBagFlightTypeTra	AOCC Pax Bag Data: bagFlightTypeTra	For each pax of flight, sum values

2.8 AOCC Pax Data (Standardized)

Specific passenger data is available within the Pax Analyzer and BRS systems. Under this standardized table, only specific passenger data has been merged. The data derived from this table has been utilized in various other tables within the system.

This table integrates data related to passengers and baggage from multiple sources, specifically AOCC Flight Data, AOCC Pax Analyzer Pax Data, and AOCC Pax Bag Data. The key function of this table is to consolidate data points for each flight, allowing the system to monitor and manage passenger movements and baggage handling efficiently. For each flight, the relevant flight ID is recorded, and the table captures whether the flight is an arrival or departure (arrDep). The check-in sequence data (checkinSeq) is used to determine whether the passenger is arriving or departing, with data being merged accordingly from the Pax Analyzer and Pax Bag systems. The table also tracks when the passenger was first seen at check-in (paxCheckinSeen) and when they passed through security (paxSecuritySeen) or boarded the aircraft (paxBoardedSeen). Additionally, it calculates the duration between key checkpoints such as from check-in to security, from check-in to boarding, and from security to boarding.

Workstations or devices where the passenger was processed during check-in, security, and at the gate are recorded (paxCheckinWs, paxTdasWs, paxGateWs). The table also identifies whether the passenger is in the landside or airside area based on the sequence of these events (paxLandsideStatus, paxAirsideStatus).

Baggage-related data is extensively covered, including total bag count (paxTotalBagCount), whether the baggage has been delivered, loaded, unloaded, or if there have been any inquiries (paxArrDelivered, paxArrInquired, paxDepLoaded, etc.). The table further categorizes the type of flight associated with the baggage—whether it's an arrival, departure, or transfer (paxBagFlightTypeArr, paxBagFlightTypeDep, paxBagFlightTypeTra).

This comprehensive data integration enables efficient tracking and management of both passengers and their baggage throughout the airport system, contributing to streamlined operations and better decision-making in the AOCC.

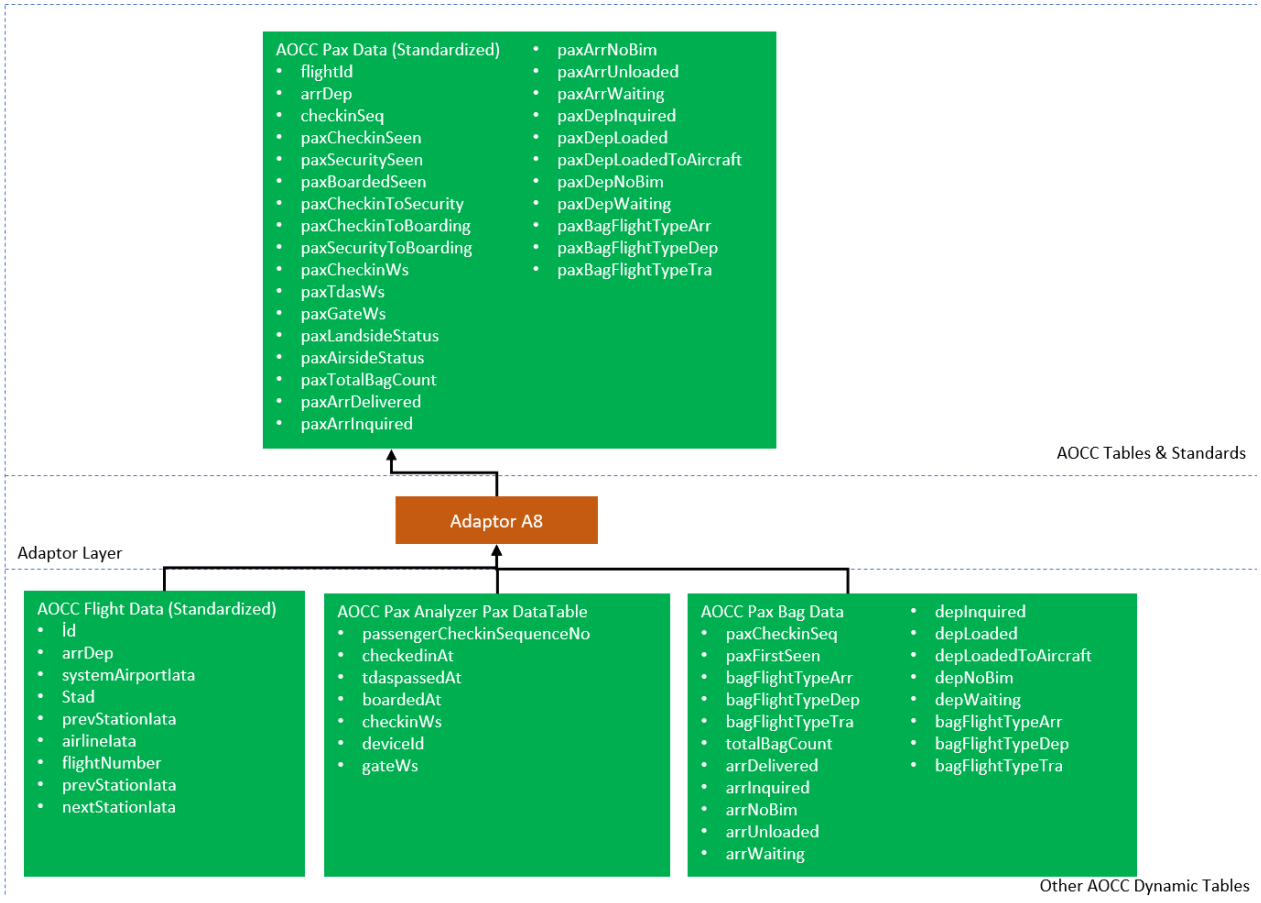

Figure 9 - Pax Data standardization and integration

Adaptor settings:

Fields	Data Source	Adaptor Settings
flightId	AOCC flight Data	For each flight in the list
arrDep	AOCC flight Data	arrDep value for this flightId will be inserted here
checkinSeq	AOCC Pax Analyzer Pax Data: passengerCheckinSequenceNo AOCC Pax Bag Data paxCheckinSeq	For this flight, if there is same checkin sequence data in both AOCC Pax Analyzer Data and AOCC Pax Bag Data so it means that pax is departing so pax analyzer data and bag related data will be considered together. For this flight, if there is a checkin sequence data only at AOCC Pax Bag so it means that pax is arriving so there will be no pax analyzer data for this checkin seq + flight
paxCheckinSeen	AOCC Pax Analyzer Pax Data: checkedinAt AOCC Pax Bag Data: paxFirstSeen bagFlightTypeArr bagFlightTypeDep bagFlightTypeTra	if there is checkedinAt data in AOCC Pax Analyzer Pax Data use it but if its empty, and bagFlightTypeDep is "Yes" and bagFlightTypeTra is "No" than use paxFirstSeen data from AOCC Pax Bag Data if checkedinAt data in AOCC Pax Analyzer Pax Data is empty and if bagFlightTypeArr or bagFlightTypeTra yes then paxCheckinSeen will be empty
paxSecuritySeen	AOCC Pax Analyzer Pax Data: tdaspassedAt	if tdaspassedAt is not empty, use it
paxBoardedSeen	AOCC Pax Analyzer Pax Data: boardedAt	if boardedAt is not empty, use it
paxCheckinToSecurity	paxSecuritySeen - paxCheckinSeen	Duration for one point to other
paxCheckinToBoarding	paxBoardedSeen - paxCheckinSeen	Duration for one point to other
paxSecurityToBoarding	paxBoardedSeen - paxSecuritySeen	Duration for one point to other
paxCheckinWs	AOCC Pax Analyzer Pax Data: checkinWs	if checkinWs is not empty, use it
paxTdasWs	AOCC Pax Analyzer Pax Data: deviceId	if deviceId is not empty, use it
paxGateWs	AOCC Pax Analyzer Pax Data: gateWs	if gateWs is not empty, use it
paxLandsideStatus	paxCheckinSeen paxSecuritySeen paxBoardedSeen	if paxCheckinSeen is not null but paxSecuritySeen and paxBoardedSeen is null than pax is in the landside and write YES
paxAirsideStatus	paxCheckinSeen paxSecuritySeen paxBoardedSeen	if paxSecuritySeen is not null but paxBoardedSeen is null than pax is in the airside and write YES
paxTotalBagCount	AOCC Pax Bag Data: totalBagCount	
paxArrDelivered	AOCC Pax Bag Data: arrDelivered	
paxArrInquired	AOCC Pax Bag Data: arrInquired	
paxArrNoBim	AOCC Pax Bag Data: arrNoBim	
paxArrUnloaded	AOCC Pax Bag Data: arrUnloaded	
paxArrWaiting	AOCC Pax Bag Data: arrWaiting	
paxDepInquired	AOCC Pax Bag Data: depInquired	
paxDepLoaded	AOCC Pax Bag Data: depLoaded	
paxDepLoadedToAircraft	AOCC Pax Bag Data: depLoadedToAircraft	
paxDepNoBim	AOCC Pax Bag Data: depNoBim	
paxDepWaiting	AOCC Pax Bag Data: depWaiting	
paxBagFlightTypeArr	AOCC Pax Bag Data: bagFlightTypeArr	
paxBagFlightTypeDep	AOCC Pax Bag Data: bagFlightTypeDep	
paxBagFlightTypeTra	AOCC Pax Bag Data: bagFlightTypeTra	

2.9 AOCC Flight Streaming Stats (standardized)

AOCC Flight Streaming Stats is one of the most important standardized table of the system. Within a single flight process, there are numerous points where passengers, baggage, and aircraft are handled. Throughout this process, various types of data emerge, requiring real-time monitoring across multiple areas. The AOCC Flight Streaming Stats table is designed to monitor all relevant streamable parameters related to a flight. Additionally, predictions made through the Central Prediction Services (CPS) are also consolidated within the same table for certain data points.

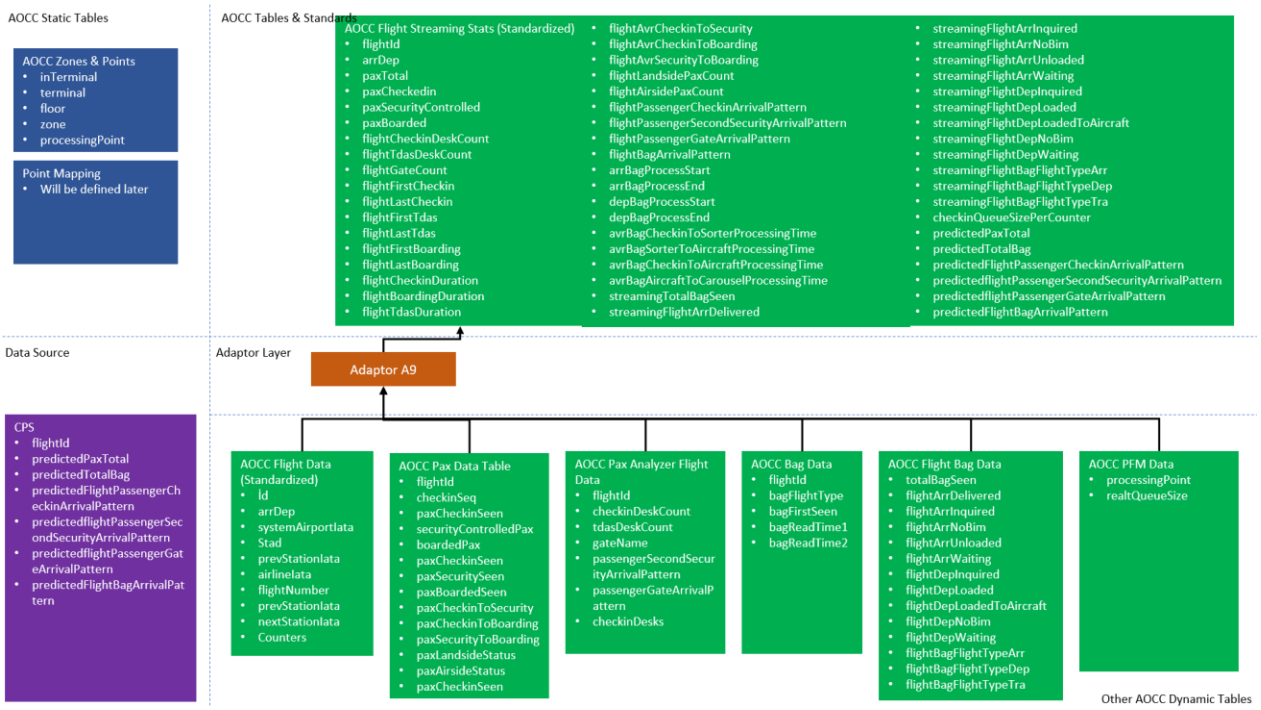

Figure 10 – AOCC Flight Streaming Stats data standardization and integration

The diagram below outlines how static data, such as terminal, floor, and zone information, is defined in AOCC Static Tables, which will later be used for point mapping within the system. Various data sources, including Central Prediction Services (CPS), AOCC Flight Data, Pax Analyzer Flight Data, Bag Data, and Passenger Flow Management (PFM) Data, feed into the system. CPS provides predicted passenger and baggage totals, as well as arrival patterns. AOCC Flight Data contains standardized flight information, including flight numbers, airport codes, and station details. Pax Analyzer Flight Data tracks passenger check-ins, security, and boarding information. Bag Data monitors the handling of baggage, such as scanning times and types. PFM Data manages passenger flow metrics, including queue sizes at various processing points.

The data from these sources is processed through an adaptor layer, labeled as Adaptor A9, which merges and transforms the information to create standardized AOCC tables. This adaptor layer ensures that all relevant data is aligned and ready for use in operational decision-making. The result is the generation of standardized tables within the AOCC system, which contain crucial operational statistics, including flight streaming stats, passenger check-in and boarding patterns, baggage processing times, and predicted

passenger flows. These standardized tables are essential for monitoring and optimizing airport operations in real-time.

Field descriptions:

- flightId: The unique identifier for a specific flight. This ID is used to track all related data points within the AOCC system.
- arrDep: Indicates whether the flight is an arrival (arr) or a departure (dep). This helps categorize the data and processes accordingly.
- paxTotal: The total number of passengers on the flight. This includes all passengers who have checked in and are expected to board.
- paxCheckedin: The number of passengers who have completed the check-in process. This figure is crucial for understanding how many passengers are ready to proceed to security and boarding.
- paxSecurityControlled: The number of passengers who have passed through security control. This data helps monitor the flow of passengers through security checkpoints.
- paxBoarded: The number of passengers who have boarded the aircraft. This is a key metric for determining the readiness of the flight for departure.
- flightCheckinDeskCount: The number of check-in desks that were used for the flight.
- flightTdasDeskCount: The number of TDAS (Traveler Document Authentication System) desks that were used for processing passengers
- flightGateCount: The number of gates used for the flight. Typically, this would be one gate, but in some cases, multiple gates may be used for large flights or special circumstances.
- flightFirstCheckin: The timestamp of the first check-in for the flight.
- flightLastCheckin: The timestamp of the last check-in for the flight.
- flightFirstTdas: The timestamp of the first document authentication for the flight.
- flightLastTdas: The timestamp of the last document authentication for the flight, indicating when the final passenger's documents were checked.
- flightFirstBoarding: The timestamp of the first passenger boarding the flight.
- flightLastBoarding: The timestamp of the last passenger boarding the flight.
- flightCheckinDuration: The total duration of the check-in process, calculated from the first check-in to the last check-in.
- flightBoardingDuration: The total duration of the boarding process, calculated from the first boarding to the last boarding.
- flightTdasDuration: The total duration of the TDAS processing for the flight, from the first document check to the last.
- flightAvrCheckinToSecurity: The average time it takes for passengers to go from check-in to security. This metric helps in analyzing passenger flow efficiency.
- flightAvrCheckinToBoarding: The average time it takes for passengers to go from check-in to boarding.
- flightAvrSecurityToBoarding: The average time it takes for passengers to go from security to boarding.
- flightLandsidePaxCount: The number of passengers who are still in the landside area (before security).

- flightAirsidePaxCount: The number of passengers who have moved to the airside area (after security).
- flightPassengerCheckinArrivalPattern: The pattern of passenger arrivals at check-in, showing the distribution of passengers over time.
- flightPassengerSecondSecurityArrivalPattern: The pattern of passenger arrivals at the second security checkpoint, useful for understanding how passengers progress through multiple security stages.
- flightPassengerGateArrivalPattern: The pattern of passenger arrivals at the gate, providing insights into how passengers move through the airport to reach their departure gate.
- flightBagArrivalPattern: The pattern of baggage arrival, showing how and when baggage is processed through the system.
- arrBagProcessStart: The timestamp indicating the start of the baggage processing for arriving flights.
- arrBagProcessEnd: The timestamp indicating the end of the baggage processing for arriving flights.
- depBagProcessStart: The timestamp indicating the start of the baggage processing for departing flights.
- depBagProcessEnd: The timestamp indicating the end of the baggage processing for departing flights.
- avrBagCheckinToSorterProcessingTime: The average time it takes for baggage to move from check-in to the sorter.
- avrBagSorterToAircraftProcessingTime: The average time it takes for baggage to move from the sorter to the aircraft.
- avrBagCheckinToAircraftProcessingTime: The average time it takes for baggage to go from check-in directly to the aircraft, bypassing intermediate stages.
- avrBagAircraftToCarouselProcessingTime: The average time it takes for baggage to move from the aircraft to the baggage carousel, important for passenger satisfaction.
- streamingTotalBagSeen: The total number of bags processed for the flight, tracked in real-time.
- streamingFlightArrDelivered: The number of bags delivered for arriving flights, tracked in real-time.
- streamingFlightArrInquired: The number of baggage inquiries made for arriving flights, indicating potential issues with baggage delivery.
- streamingFlightArrNoBim: The number of bags for arriving flights without a Baggage Information Message (BIM), which may indicate a lack of tracking information.
- streamingFlightArrUnloaded: The number of bags unloaded from the aircraft for arriving flights, tracked in real-time.
- streamingFlightArrWaiting: The number of bags waiting to be processed for arriving flights, providing insights into baggage handling delays.
- streamingFlightDepInquired: The number of baggage inquiries made for departing flights, indicating potential issues with baggage loading or tracking.
- streamingFlightDepLoaded: The number of bags loaded onto the aircraft for departing flights, tracked in real-time.

- streamingFlightDepLoadedToAircraft: The number of bags that have been loaded onto the aircraft for departing flights, ensuring all baggage is accounted for.
- streamingFlightDepNoBim: The number of bags for departing flights without a Baggage Information Message (BIM), which may indicate a lack of tracking information.
- streamingFlightDepWaiting: The number of bags waiting to be processed for departing flights, providing insights into potential baggage handling delays.
- streamingFlightBagFlightTypeArr: Indicates the type of flight (arrival) associated with the baggage.
- streamingFlightBagFlightTypeDep: Indicates the type of flight (departure) associated with the baggage.
- streamingFlightBagFlightTypeTra: Indicates the type of flight (transfer) associated with the baggage.
- checkinQueueSizePerCounter: The size of the queue at each check-in counter, providing insights into passenger flow and potential bottlenecks.
- predictedPaxTotal: The predicted total number of passengers for the flight, as calculated by the Central Prediction Services.
- predictedTotalBag: The predicted total number of bags for the flight, as calculated by the Central Prediction Services.
- predictedFlightPassengerCheckinArrivalPattern: The predicted pattern of passenger arrivals at check-in, based on historical data and predictions.
- predictedflightPassengerSecondSecurityArrivalPattern: The predicted pattern of passenger arrivals at the second security checkpoint, useful for resource planning.
- predictedflightPassengerGateArrivalPattern: The predicted pattern of passenger arrivals at the gate, helping to anticipate peak times and manage boarding efficiently.
- predictedFlightBagArrivalPattern: The predicted pattern of baggage arrival, which aids in planning and optimizing baggage handling processes.

Adaptor settings:

Fields	Data Source	Adaptor Settings
flightId	AOCC flight Data	For each flight in the list
arrDep	AOCC flight Data	arrDep value for this flightId will be inserted here
paxTotal	AOCC Pax Data: checkinSeq	Count uniquee checkinSeq for this flight at AOCC Pax Data
paxCheckedin	AOCC Pax Data: paxCheckinSeen	Count uniquee checkinSeq with paxCheckinSeen is not null
paxSecurityControlled	AOCC Pax Data: securityControlledPax	Count uniquee checkinSeq with securityControlledPax is not null
paxBoarded	AOCC Pax Data: boardedPax	Count uniquee checkinSeq with boardedPax is not null
flightCheckinDeskCount	AOCC Pax Analyzer Flight Data: checkinDeskCount	
flightTdasDeskCount	AOCC Pax Analyzer Flight Data: tdasDeskCount	
flightGateCount	AOCC Pax Analyzer Flight Data: gateName	
flightFirstCheckin	AOCC Pax Data: paxCheckinSeen	For this flight find the earliest paxCheckinSeen date and time

flightLastCheckin	AOCC Pax Data: paxCheckinSeen	For this flight find the latest paxCheckinSeen date and time
flightFirstTdas	AOCC Pax Data: paxSecuritySeen	For this flight find the earliest paxSecuritySeen date and time
flightLastTdas	AOCC Pax Data: paxSecuritySeen	For this flight find the latest paxSecuritySeen date and time
flightFirstBoarding	AOCC Pax Data: paxBoardedSeen	For this flight find the earliest paxBoardedSeen date and time
flightLastBoarding	AOCC Pax Data: paxBoardedSeen	For this flight find the latest paxBoardedSeen date and time
flightCheckinDuration	flightLastCheckin - flightLFirstCheckin	
flightBoardingDuration	flightLastBoarding - flightFirstBoarding	
flightTdasDuration	flightLastTdas - flightFirstTdas	
flightAvrCheckinToSecurity	AOCC Pax Data: paxCheckinToSecurity	For this flight, filter not null values of the AOCC Pax Data field and take the average
flightAvrCheckinToBoarding	AOCC Pax Data: paxCheckinToBoarding	For this flight, filter not null values of the AOCC Pax Data field and take the average
flightAvrSecurityToBoarding	AOCC Pax Data: paxSecurityToBoarding	For this flight, filter not null values of the AOCC Pax Data field and take the average
flightLandsidePaxCount	AOCC Pax Data: paxLandsideStatus	Count uniquee checkinSeq with paxLandsideStatus is YES
flightAirsidePaxCount	AOCC Pax Data: paxAirsideStatus	Count uniquee checkinSeq with paxAirsideStatus is YES
flightPassengerCheckinArrivalPattern	AOCC Pax Data: paxCheckinSeen	the number of flight passengers who have arrived checkin counters in specific time range. But for checkin calculation, we will not use AOCC Pax Analyzer Flight Data we will generate this table by looking at AOCC Pax Data because baggage related checkin date and time are included in AOCC Pax Data. Example 1: Array[5, 2024-08-02 07:00:00.0] 2: Array[15, 2024-08-02 08:00:00.0] 3: Array[17, 2024-08-02 09:00:00.0]
flightPassengerSecondSecurityArrivalPattern	AOCC Pax Analyzer Flight Data: passengerSecondSecurityArrivalPattern	the number of flight passengers who have arrived second security in specific time range
flightPassengerGateArrivalPattern	AOCC Pax Analyzer Flight Data: passengerGateArrivalPattern	the number of flight passengers who have arrived gate desk in specific time range
flightBagArrivalPattern	AOCC Bag Data: bagFlightType bagFirstSeen	For this flight, pattern will be calculated for only departing bags (bagFlightType: L) Example 1: Array[5, 2024-08-02 07:00:00.0] 2: Array[15, 2024-08-02 08:00:00.0] 3: Array[17, 2024-08-02 09:00:00.0]
arrBagProcessStart	AOCC Bag Data: bagReadTime 1	Only for arrival flights, for this flight, check all bagReadTime1 date and times and pick the earliest one. Represents the time of first barcode read under aircraft after flight arrived.
arrBagProcessEnd	AOCC Bag Data: bagReadTime 2	Only for arrival flights, for this flight, check all bagReadTime2 date and times and pick the latest one. Represents the time of last barcode read before putting it to carousel
depBagProcessStart	AOCC Bag Data: bagReadTime 1	Only for departure flights, for this flight, check all bagReadTime1 date and times and pick the earliest one. Represents the time of the first baggage barcode read in sorter area
depBagProcessEnd	AOCC Bag Data: bagReadTime 2	Only for departure flights, for this flight, check all bagReadTime2 date and times and pick the latest one. Represents the time of last barcode read before putting it into aircraft
avrBagCheckinToSorterProcessingTime	AOCC Bag Data: bagFirstSeen bagReadTime1	Only for departure flights, for bags of this flight, average duration of bags between bagReadTime1 and bagFirstSeen
avrBagSorterToAircraftProcessingTime	AOCC Bag Data: bagReadTime1 bagReadTime2	Only for departure flights, for bags of this flight, average duration of bags between bagReadTime2 and bagReadTime1
avrBagCheckinToAircraftProcessingTime	AOCC Bag Data: bagFirstSeen bagReadTime2	Only for departure flights, for bags of this flight, average duration of bags between bagReadTime2 and bagFirstSeen
avrBagAircraftToCarouselProcessingTime	AOCC Bag Data: bagReadTime1 bagReadTime2	Only for arrival flights, for bags of this flight, average duration of bags between bagReadTime2 and bagReadTime1

streamingTotalBagSeen	AOCC Flight Bag Data: totalBagSeen	value will be reflected here
streamingFlightArrDelivered	AOCC Flight Bag Data: flightArrDelivered	value will be reflected here
streamingFlightArrInquired	AOCC Flight Bag Data: flightArrInquired	value will be reflected here
streamingFlightArrNoBim	AOCC Flight Bag Data: flightArrNoBim	value will be reflected here
streamingFlightArrUnloaded	AOCC Flight Bag Data: flightArrUnloaded	value will be reflected here
streamingFlightArrWaiting	AOCC Flight Bag Data: flightArrWaiting	value will be reflected here
streamingFlightDepInquired	AOCC Flight Bag Data: flightDepInquired	value will be reflected here
streamingFlightDepLoaded	AOCC Flight Bag Data: flightDepLoaded	value will be reflected here
streamingFlightDepLoadedToAircraft	AOCC Flight Bag Data: flightDepLoadedToAircraft	value will be reflected here
streamingFlightDepNoBim	AOCC Flight Bag Data: flightDepNoBim	value will be reflected here
streamingFlightDepWaiting	AOCC Flight Bag Data: flightDepWaiting	value will be reflected here
streamingFlightBagFlightTypeArr	AOCC Flight Bag Data: flightBagFlightTypeArr	value will be reflected here
streamingFlightBagFlightTypeDep	AOCC Flight Bag Data: flightBagFlightTypeDep	value will be reflected here
streamingFlightBagFlightTypeTra	AOCC Flight Bag Data: flightBagFlightTypeTra	value will be reflected here
checkinQueueSizePerCounter	AOCC Flight Data: counters AOCC Pax Analyzer Flight Data: checkinDesks Zone Point Maps: PFM: realtQueueSize	counters from AOCC flight Data, checkinDesks from AOCC Pax Analyzer Flight Data will be gathered and for these counters by using Zone Point Map, points will be detected and PFM realtQueueSize of these counters will be summed and average value will be published here.
predictedPaxTotal	CPS	
predictedTotalBag	CPS	
predictedFlightPassengerCheckinArrivalPattern	CPS	
predictedflightPassengerSecondSecurityArrivalPattern	CPS	
predictedflightPassengerGateArrivalPattern	CPS	
predictedFlightBagArrivalPattern	CPS	

2.10 AOCC Flight Processes (Standardized)

This table is designed to share calculations related to the percentage of completion of general processes associated with the flight.

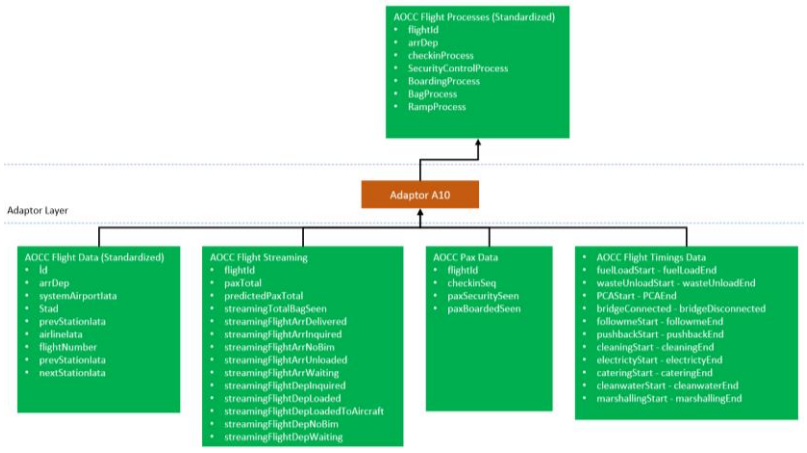

Figure 11 – AOCC Flight Process (Standardized)

Adaptor settings:

Fields	Data Source	Adaptor Settings
flightId	AOCC flight Data	For each flight in the list
arrDep	AOCC flight Data	arrDep value for this flightId will be inserted here
checkinProcess	AOCC Flight Data: atad AOCC Flight Streaming Stats: paxTotal / predictedPaxTotal	If atad is not null and flight is departure than value will 100%, if atad is null, paxTotal represents the seen passengers from BRS and Pax Analyzer systems so these passengers have at least completed their checkin even it is done in mobile or even they have not printed boarding pass in checkedin desk
SecurityControlProcess	AOCC Pax Data: checkinSeq paxSecuritySeen paxBoardedSeen AOCC Flight Streaming Stats: predictedPaxTotal	If atad is not null and flight is departure than value will 100%, if atad is null, for this flight, count uniquee checkinSeq numbers if passenger has one of these values paxSecuritySeen or paxBoardedSeen not null. So the value will be calculated: (pax with paxSecuritySeen or paxBoardedSeen) / predictedPaxTotal
BoardingProcess	AOCC Pax Data: PaxBoardedSeen AOCC Flight Streaming Stats: predictedPaxTotal	If atad is not null and flight is departure than value will 100%, if atad is null, for this flight, count uniquee checkinSeq numbers if passenger has paxBoardedSeen not null. So the value will be calculated: (pax with paxBoardedSeen) / predictedPaxTotal
BagProcess	AOCC Flight Streaming Stats: streamingTotalBagSeen streamingFlightArrDelivered streamingFlightArrInquired streamingFlightArrNoBim streamingFlightArrUnloaded streamingFlightArrWaiting streamingFlightDepInquired streamingFlightDepLoaded streamingFlightDepLoadedToAircraft streamingFlightDepNoBim streamingFlightDepWaiting	if flight is arrival: (streamingTotalBagSeen - streamingFlightArrWaiting) / streamingTotalBagSeen if flight is departure and atad is not null than 100%, otherwise: (streamingTotalBagSeen - streamingFlightDepWaiting) / streamingTotalBagSeen
RampProcess	AOCC Flight Timings: fuelLoadStart - fuelLoadEnd wasteUnloadStart - wasteUnloadEnd PCAStart - PCAEnd bridgeConnected - bridgeDisconnected followmeStart - followmeEnd pushbackStart - pushbackEnd cleaningStart - cleaningEnd electrictyStart - electrictyEnd cateringStart - cateringEnd cleanwaterStart - cleanwaterEnd marshallingStart - marshallingEnd	Only for departure flights, will be analyzed later.

2.11 AOCC Air Sensor Data

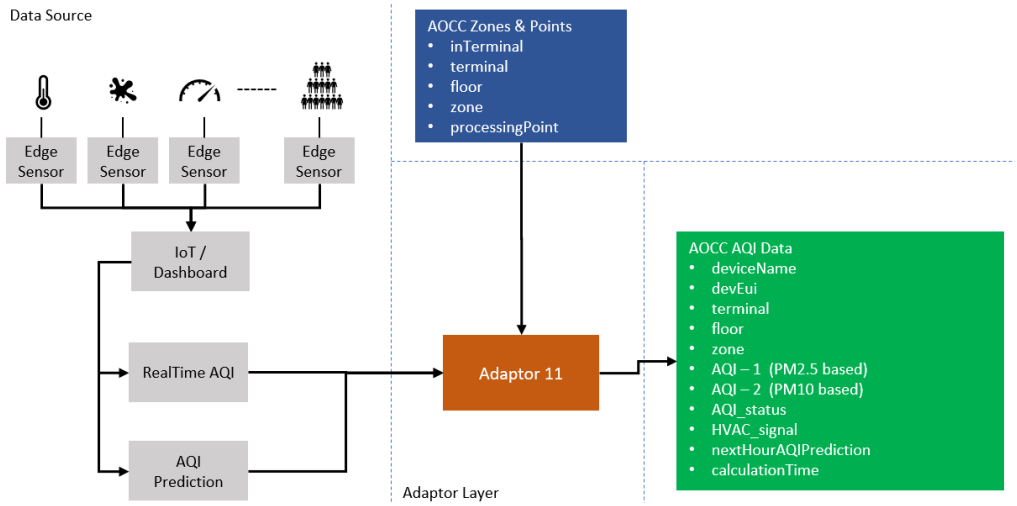

Figure 12 – AOCC Air Sensor data standardization and integration

Adaptor settings:

Field	Data Source	Example Value	Description
deviceName	Sensor System: deviceName	"9in1 IAQ Sensor AM319 HCHO IR Atölye"	Device Name
devEui	Sensor System: devEui	"24e124710c484479"	Uniquee Device Code
terminal	AOCCZones & Points		
floor	AOCCZones & Points		
zone	AOCCZones & Points		
humidity	Sensor System: humidity	43.5	Humidity refers to the amount of water vapor present in the air. It is expressed as a percentage. In this context, a humidity of 43.5% indicates that the air contains 43.5% of the maximum amount of moisture it can hold at the current temperature
light_level	Sensor System: light_level	1	Light level measures the intensity of light in the environment, often expressed in lux. A light level of 1.0 suggests a very low light intensity, which could indicate darkness or very dim lighting conditions.
pressure	Sensor System: pressure	916.2	Atmospheric pressure is the force exerted by the weight of the air above a given point. It is measured in units like millibars or hectopascals (hPa). A pressure of 916.2 hPa is relatively low and could indicate high altitude or a low-pressure weather system
temperature	Sensor System: temperature	25	Temperature measures how hot or cold the environment is, typically expressed in degrees Celsius (°C) or Fahrenheit (°F). A temperature of 25.0°C is considered warm and comfortable for most people
hcho	Sensor System: hcho	0.01	Formaldehyde is a colorless gas with a strong odor, often used in building materials and household products. It is measured in parts per million (ppm) or milligrams per cubic meter (mg/m³). An hcho level of 0.01 mg/m³ indicates a very low concentration of formaldehyde in the air
pm2_5	Sensor System: pm2_5	7	PM2.5 refers to fine particulate matter with a diameter of 2.5 micrometers or smaller. These particles can penetrate the respiratory system and cause health issues. A PM2.5 level of 7.0 µg/m³ is relatively low and generally considered safe for most people
co2	Sensor System: co2	574	Carbon dioxide is a colorless, odorless gas produced by burning carbon-based materials and by respiration. It is measured in parts per million (ppm). A CO2 level of 574.0 ppm is within the typical range for indoor environments but higher levels can indicate poor ventilation
tvoc	Sensor System: tvoc	100	TVOCs are a group of organic chemicals that can easily evaporate into the air. They are measured in micrograms per cubic meter (µg/m³). A TVOC level of 100.0 µg/m³ indicates the presence of various volatile organic compounds in the air, which can affect air quality
pm10	Sensor System: pm10	20	PM10 refers to particulate matter with a diameter of 10 micrometers or smaller. These particles are also inhalable and can affect health. A PM10 level of 20.0 µg/m³ is generally considered low to moderate
pir	Sensor System: pir	"idle"	A PIR sensor detects motion by measuring infrared radiation from objects in its field of view. The status "idle" indicates that no motion is currently detected by the sensor.
readTime	Sensor System: time	"2024-06-13T06:50:49.762767+00:00"	

2.12 AOCC Point and Zone Stats (Standardized)

The diagram provides an in-depth view of the data flow within the Airport Operations Control Center Point and Zone Stats (AOCC), showcasing how various subsystems and data sources are integrated to produce standardized operational metrics essential for managing airport activities. At the heart of the AOCC system are multiple data sources that each contribute vital information for airport operations. The Baggage Reconciliation System (BRS) tracks baggage throughout the airport, capturing data at different processing stages, such as during check-in, loading onto aircraft, and at baggage claim. This system is crucial for ensuring that baggage is correctly handled and delivered to the correct destination.

The Pax Analyzer system is responsible for gathering data on passenger movements within the airport. It tracks key stages of the passenger journey, including check-in, security clearance, and boarding, providing a detailed overview of passenger flow and helping to identify potential bottlenecks. Central Prediction Services (CPS) are integrated into the system to provide predictive analytics. This service uses historical data and current conditions to forecast operational metrics such as passenger flow, baggage handling, and aircraft turnaround times. The predictions generated by CPS are used to anticipate and mitigate potential disruptions, allowing for better planning and resource allocation.

Passenger Flow Management (PFM) systems monitor the movement of passengers through various airport zones. They provide real-time data on queue lengths, waiting times, and the distribution of passengers across terminals. This information is vital for maintaining smooth operations and ensuring that passengers move efficiently through the airport. FMS/RMS/CMS (Flight Management System/Resource Management System/Content Management System) are key systems for managing flight schedules, allocating resources like gates and staff, and overseeing other critical aspects of airport operations. These systems ensure that flights are coordinated and that resources are used efficiently.

The Energy System monitors and manages the airport's energy consumption, collecting data on current power usage, daily patterns of consumption, and predictions for future energy needs. This system is crucial for optimizing energy use and reducing costs while ensuring that all airport operations have the power they need. The Sensor System consists of various sensors placed throughout the airport, collecting data on environmental conditions such as temperature, humidity, and air quality. This information is used to maintain a comfortable and safe environment for both passengers and staff. The ADSB System provides real-time data on aircraft positions, speeds, and altitudes, which is essential for tracking aircraft movements within the airport's airspace. This data supports the safe and efficient management of flights as they arrive, depart, and taxi within the airport. All these data sources are processed through an adaptor layer, specifically referred to as Adaptor 12 in the diagram. The adaptor serves a critical role in standardizing and harmonizing the data from these diverse sources, ensuring that it is consistent, accurate, and usable within the AOCC's operational systems. This standardized data is then stored in the AOCC Point/Zone Statistics tables, which organize the information by attributes such as terminal, floor, zone, latitude, longitude, and altitude. Each point or zone within the airport is associated with specific systems, whether it be PFM, FMS/RMS/CMS, Pax Analyzer, BRS, Scada (Supervisory Control and Data Acquisition), Sensor System, or ADSB.

The final output is consolidated into AOCC Tables & Standards, which include a comprehensive range of operational metrics. These metrics cover various aspects of airport operations such as daily occupancy levels, patterns of passenger and baggage queues, processed passenger and baggage data, temperature and air quality levels, and power consumption statistics. This standardized information allows for the monitoring of real-time airport operations and provides predictive insights that help in planning and optimizing airport resources. The integration of these diverse data sources into a single, cohesive system ensures that the airport can operate efficiently, even during peak times, by providing a detailed and holistic view of all critical operational processes.

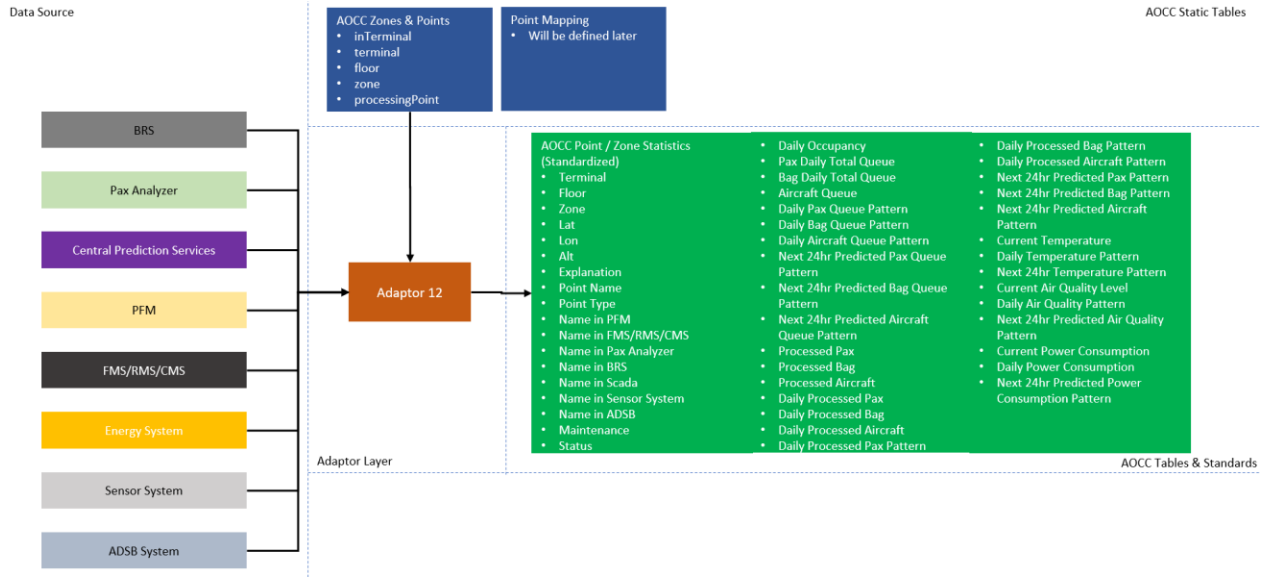

Figure 13 – AOCC Point and Zone Stats data standardization and integration

Field description:

- Terminal: This field represents the specific terminal within the airport where the point or zone is located.
- Floor: The floor field indicates the level within the terminal where the point or zone is situated, providing further granularity to the location data.
- Zone: This field designates the specific zone within the terminal or airport, further narrowing down the location to a particular area like a check-in area, baggage claim, or security checkpoint.
- Lat (Latitude): The latitude coordinate of the point or zone, which helps in mapping the exact geographic location of the area within the airport.
- Lon (Longitude): The longitude coordinate, paired with latitude, to pinpoint the specific geographic location of the point or zone within the airport.
- Alt (Altitude): The altitude or height above sea level of the point or zone, providing three-dimensional spatial information that might be relevant for certain operational tasks.
- Explanation: This field offers a detailed description or explanation of the specific point or zone, clarifying its role or importance within the airport's operational framework.

- Point Name: The name given to the specific point or zone within the airport, used for easy identification and reference within the AOCC system.
- Point Type: This field categorizes the point or zone by its function or type, such as a gate, security checkpoint, or baggage claim area.
- Name in PFM: The corresponding name of the point or zone within the Passenger Flow Management (PFM) system, ensuring consistency across different systems.
- Name in FMS/RMS/CMS: The name of the point or zone as it appears in the Flight Management System, Resource Management System, or Content Management System, facilitating cross-system integration.
- Name in Pax Analyzer: The point or zone name as it is recognized in the Pax Analyzer system, which focuses on passenger movement data.
- Name in BRS: This field provides the name of the point or zone within the Baggage Reconciliation System (BRS), linking baggage tracking data to specific locations in the airport.
- Name in Scada: The name of the point or zone in the Supervisory Control and Data Acquisition (Scada) system, which might monitor environmental or technical parameters.
- Name in Sensor System: The name used within the Sensor System for the point or zone, which may relate to sensors that track temperature, air quality, or other environmental conditions.
- Name in ADSB: The name of the point or zone as it is recorded in the Automatic Dependent Surveillance-Broadcast (ADSB) system, which tracks aircraft positions.
- Maintenance: This field indicates whether the point or zone requires or is currently undergoing maintenance, providing operational awareness.
- Status: The current operational status of the point or zone, such as active, inactive, under maintenance, etc.
- Daily Occupancy: This metric provides data on the occupancy levels within a specific point or zone on a daily basis, helping to monitor crowding and optimize space utilization.
- Pax Daily Total Queue: This field tracks the total number of passengers in queues across different zones of the airport on a daily basis, helping to manage wait times and improve passenger flow.
- Bag Daily Total Queue: This field records the total number of bags queued for processing each day, aiding in the efficient management of baggage handling operations.
- Aircraft Queue: This field tracks the number of aircraft waiting for processing at various points, such as gates or runways, within the airport.
- Daily Pax Queue Pattern: This metric provides patterns of passenger queues over the day, helping to identify peak times and optimize staffing and resource allocation.
- Daily Bag Queue Pattern: This metric tracks the patterns of baggage queues throughout the day, offering insights into when baggage handling operations are most and least efficient.
- Daily Aircraft Queue Pattern: This field tracks the pattern of aircraft queues throughout the day, helping to identify potential bottlenecks and streamline operations.
- Next 24hr Predicted Pax Queue Pattern: This field provides a prediction of passenger queue patterns for the next 24 hours, helping to anticipate and mitigate potential congestion.
- Next 24hr Predicted Bag Queue Pattern: This field forecasts the pattern of baggage queues for the next 24 hours, aiding in the preparation for peak times and ensuring smooth baggage handling.

- Next 24hr Predicted Aircraft Queue Pattern: This field predicts the pattern of aircraft queues over the next 24 hours, enabling better scheduling and resource allocation.
- Processed Pax: This metric provides the number of passengers that have been processed through various checkpoints or services within the airport.
- Processed Bag: This metric tracks the number of bags that have been processed through the airport's baggage handling system.
- Processed Aircraft: This metric counts the number of aircraft that have been processed through various stages, such as check-in, boarding, and takeoff.
- Daily Processed Pax: This field tracks the number of passengers processed each day, providing a clear picture of passenger throughput.
- Daily Processed Bag: This field records the number of bags processed each day, ensuring efficient baggage handling operations.
- Daily Processed Aircraft: This field tracks the number of aircraft processed on a daily basis, which is critical for managing flight schedules and airport capacity.
- Daily Processed Pax Pattern: This field provides a pattern of how passengers are processed throughout the day, helping to optimize operations and improve passenger experiences.
- Daily Processed Bag Pattern: This field provides data on the daily pattern of how bags are processed through the airport's baggage handling system. It helps to identify peak times and ensures that resources are allocated effectively to maintain smooth operations.
- Daily Processed Aircraft Pattern: This field tracks the daily pattern of aircraft processing activities, such as boarding, taxiing, and takeoff. It helps in understanding the flow of aircraft through the airport and optimizing runway and gate usage.
- Next 24hr Predicted Pax Pattern: This field forecasts the pattern of passenger processing over the next 24 hours. It helps in anticipating passenger flow and adjusting staffing levels and resources accordingly to manage potential congestion.
- Next 24hr Predicted Bag Pattern: This field predicts the pattern of baggage processing for the next 24 hours, allowing for proactive management of baggage handling operations to avoid delays or bottlenecks.
- Next 24hr Predicted Aircraft Pattern: This field provides a forecast of aircraft processing activities for the next 24 hours. It aids in scheduling and resource allocation to ensure that aircraft are processed efficiently, minimizing delays and optimizing airport operations.
- Current Temperature: This field captures the real-time temperature at the airport, which is essential for various operational aspects, including passenger comfort, equipment performance, and safety protocols.
- Daily Temperature Pattern: This field tracks the pattern of temperature changes throughout the day, providing insights into how temperature fluctuations might affect airport operations and helping to plan for any necessary adjustments.
- Next 24hr Temperature Pattern: This field predicts the temperature pattern over the next 24 hours, enabling the airport to prepare for any potential impacts on operations, such as extreme weather conditions.
- Current Air Quality Level: This field provides real-time data on the air quality at the airport, which is important for ensuring a safe and healthy environment for passengers and staff.

- Daily Air Quality Pattern: This field tracks the pattern of air quality throughout the day, helping to monitor environmental conditions and take action if air quality levels deteriorate.
- Next 24hr Predicted Air Quality Pattern: This field forecasts the air quality levels for the next 24 hours, allowing for proactive measures to be taken if poor air quality is anticipated.
- Current Power Consumption: This field captures the real-time power consumption at the airport, which is critical for managing energy resources and ensuring that there is enough power to support all airport operations.
- Daily Power Consumption: This field tracks the pattern of power consumption throughout the day, helping to identify peak usage times and optimize energy management strategies.
- Next 24hr Predicted Power Consumption Pattern: This field forecasts power consumption for the next 24 hours, enabling the airport to manage energy resources more efficiently and prepare for any potential power demands.

2.13 AOCC Energy Stats

Below image represents the data flow and integration of various systems and sources within the AOCC Energy (Airport Operations Control Center) framework, focusing on energy management, weather data, and flight timing. The data sources include multiple systems such as EPIAŞ for energy pricing, weather APIs for environmental conditions, energy assets, SCADA software, and advanced AI and optimization engines connected via the Enverse API. These sources feed into the AOCC system, contributing data such as energy prices, weather conditions, energy asset information, and SCADA measurements.

The AOCC system processes this data through an adaptor layer, integrating it into several standardized tables, such as the AOCC Energy Asset List, AOCC Energy Attribute List, AOCC Energy Message Types, AOCC Zones & Points, and others. These tables capture detailed attributes related to energy management, including asset configurations, measurement timestamps, energy trading details, and flight timing events such as takeoff, landing, and boarding times.

The processed data is then utilized within the AOCC for various functions, including monitoring and predicting energy usage, managing airport operations, and ensuring efficient resource allocation. The image depicts how the integrated data flows through the AOCC system, ensuring that all relevant parameters are tracked and utilized for optimal airport management.

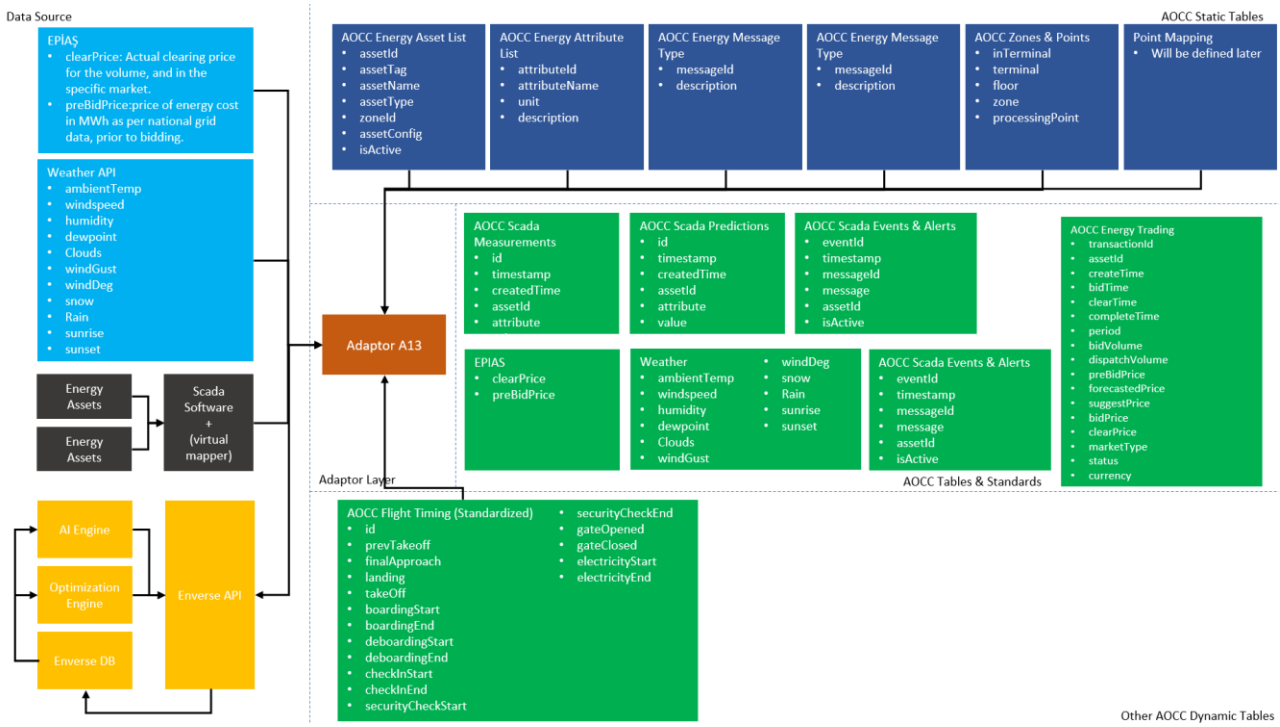

Figure 14 – AOCC Energy Data Flow standardization and integration

AOCC Energy Asset List:

Field	Data Type	Primary Key	Foreign Table	Foreign Key	Example 1	Example 2	Example 3	Explanation
assetId	string(8)	TRUE			ABCDE	FGHIJ	123AB	Randomly generated alphanumeric asset ID
assetTag	string(16)				TRIGEN1	SOLAR1	LOAD1	Asset Tag, defined by industrial purposes.
assetName	string(32)				Trigen 1	Solar Plant 1	Airport Load	Human readable asset name.
assetType	string(32)				generation	generation	load	Type of asset, acceptable value is "generation", "load", "storage", "convertion"
zoneId	string(16)				zone1	zone1	zone2	Zone of energy sources. This value is used to differentiate energy sources across different location and ensuring that they are pooled into the same zone, for the same type of energy.
assetConfig	JSON							Specific configuration for asset type.
isActive	bool				TRUE	FALSE	TRUE	If FALSE, do not display as API query

AOCC Energy Attribute List:

Field	Data Type	Primary Key	Foreign Table	Foreign Key	Example 1	Example 2	Example 3	Explanation
attributeId	string(16)	TRUE			voltageL1	voltageL2	powerL1	Unique Attribute ID. Refer to Enverse's SOCFAI specification for full list
attributeName	string(128)				Voltage L1	Voltage L2	Power L1	Human readable attribute name
unit	string(8)				V	V	kW	Unit measurement of the attribute
description	strign(256)				Phase 1 voltage of the asset	Phase 2 voltage of the asset	Phase 1 Power of the asset	Description of attribute

AOCC Scada Measurements:

Field	Data Type	Primary Key	Foreign Table	Foreign Key	Example 1	Example 2	Example 3	Explanation	Data Source
id	int	TRUE			1	2	3	Unique ID	AOCC
timestamp	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-	Airport SCADA
createdTime	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-	Airport SCADA
assetId	string(16)		Registered Assets	TRUE	ABCDE	FGHIJ	123AB		Airport SCADA
attribute	string(16)		Attributes	TRUE	voltageL1	voltageL2	powerL1		Airport SCADA
value	float				238.2	239.1	8012.2	Measured floating	Airport SCADA

AOCC Scada Predictions:

Field	Data Type	Primary Key	Foreign Table	Foreign Key	Example 1	Example 2	Example 3	Explanation
id	int	TRUE			1	2	3	Unique ID
timestamp	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS
createdTime	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS
assetId	string(16)		completeTime	TRUE	3004.2	3104.2	97.4	
attribute	string(16)		marketType	TRUE	0	0	0	
value	float				238.2	239.1	8012.2	Measured floating point of the attribute of the asset.

AOCC Scada Events & Alerts:

Field	Data Type	Primary Key	Foreign Table	Foreign Key	Example 1	Example 2	Example 3	Explanation
eventId	int	TRUE			2	52	13	Unique event Id
timestamp	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS
messageId	int		Message Type	TRUE	1111	1112	1113	Unique message id.
message	str(512)				Undervoltage detected.	Underfrequency detected	High energy consumption	Content of the message
assetId			Registered Assets	TRUE	ABCDE	FGHIJ	123AB	
isActive	bool							Default to active

AOCC Energy Trading:

Field	DataType	Primary Key	Foreign Table	Foreign Key	Example 1	Example 2	Example 3	Explanation	Data Source
transactionId	int	TRUE			12345678	12345679	12345680	Transaction id, recorded for each new transaction. Automatically created for future date.	AOCC
assetId	string(16)		Registered Assets	TRUE	ABCDE	FGHIJ	123AB		Enverse
createTime	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS	AOCC
bidTime	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS	AOCC
clearTime	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS	AOCC
completeTime	datetime				2024-01-01 01:01:01	2024-08-13 16:00:15	2024-08-13 16:00:15	Timestamp in YYYY-MM-DD HH:MM:SS	AOCC
period	int				1	15	22	Period of the transaction. For a half-hourly period, accepted value is 1-48, for hourly period, accepted value is 1-14	AOCC
bidVolume	float				3100	3100	100	Volume of energy bidded into the system.	Enverse
dispatchVolume	float				3004.2	3104.2	97.4	Volume of energy dispatched, as measured by the energy meter.	Enverse
preBidPrice	float				147	147	147	price of energy cost in MWh as per national grid data, prior to bidding.	EPİAŞ
forecastedPrice	float				148	149	148	Forecasted trading price.	Enverse
suggestPrice	float				148	149	148	Suggested price by Enverse	Enverse
bidPrice	float				145	145	145	Actual bidding price for the volume, and the specific market.	USER INPUT/AOCC
clearPrice	float				148	146	140	Actual clearing price for the volume, and in the specific market	EPİAŞ
marketType	string(16)				capital	capital	capital	Type of the market participated in. Specific to the target market	Enverse
status	string(8)				initial	bidded	completed	Status of the transaction, accepted values are "initial", "bidded", "accepted", "rejected", "completed".	Enverse/AOCC
currency	string(3)				TRL	USD	SGD	Currency of the transaction, 3 capital.	AOCC

3 . Port Data Standardization

Data standardization works for a Port will be performed in terms of the following key components:

- Data Source : refers to the system in which data is consumed and the content of the data itself.
- Static table : represents static data loaded into the platform and used for translation.
- Adapter layer : A code block that integrates data from data sources, static tables, and other tables.
- Tables and Standards : Refer to a standardized table.
- Other Dynamic Tables : Indicates additional dynamic tables used to generate standardized data.

As shown in Figure 15, there are tables generated from data sources, and each table shows the import and export operations of CFS(Container Freight Station) /CY(Container Yard) warehouses in the hinterland of the port as follows :

- container transport between port terminal and CFS/CY,
- Data optimization between the CFS/CY and the transportation system,
- Data prediction,
- data trust provisioning,
- Optimizing container placement,
- recommended container placement agent when brought in,
- Priority recommendation agent for container transport vehicles when taken out,
- Optimization of transport vehicle allocation based on digital twin simulation,
- Tables such as main data and metadata are managed for each system such as secure data sharing for logistics optimization based on blockchain.

At the port terminal, CFSMS(Container Freight Station management System) and TMS(Transport Management System) products operate in two environments, web and mobile, and data from a total of four systems are exchanged using a single JSON format.

CFSMS efficiently allocates port resources such as gates, congestion by CFS/CY, container status and location information, and container cargo operation status to streamline port operation. These systems provide all the necessary information related to transportation, resource planning, and service use that other partners and project stakeholders need.

TMS optimizes container transport by managing the transportation status of the transport vehicle's schedule, departure, arrival, and completion of transportation, as well as the cargo loading and unloading status of containers for each workplace warehouse.

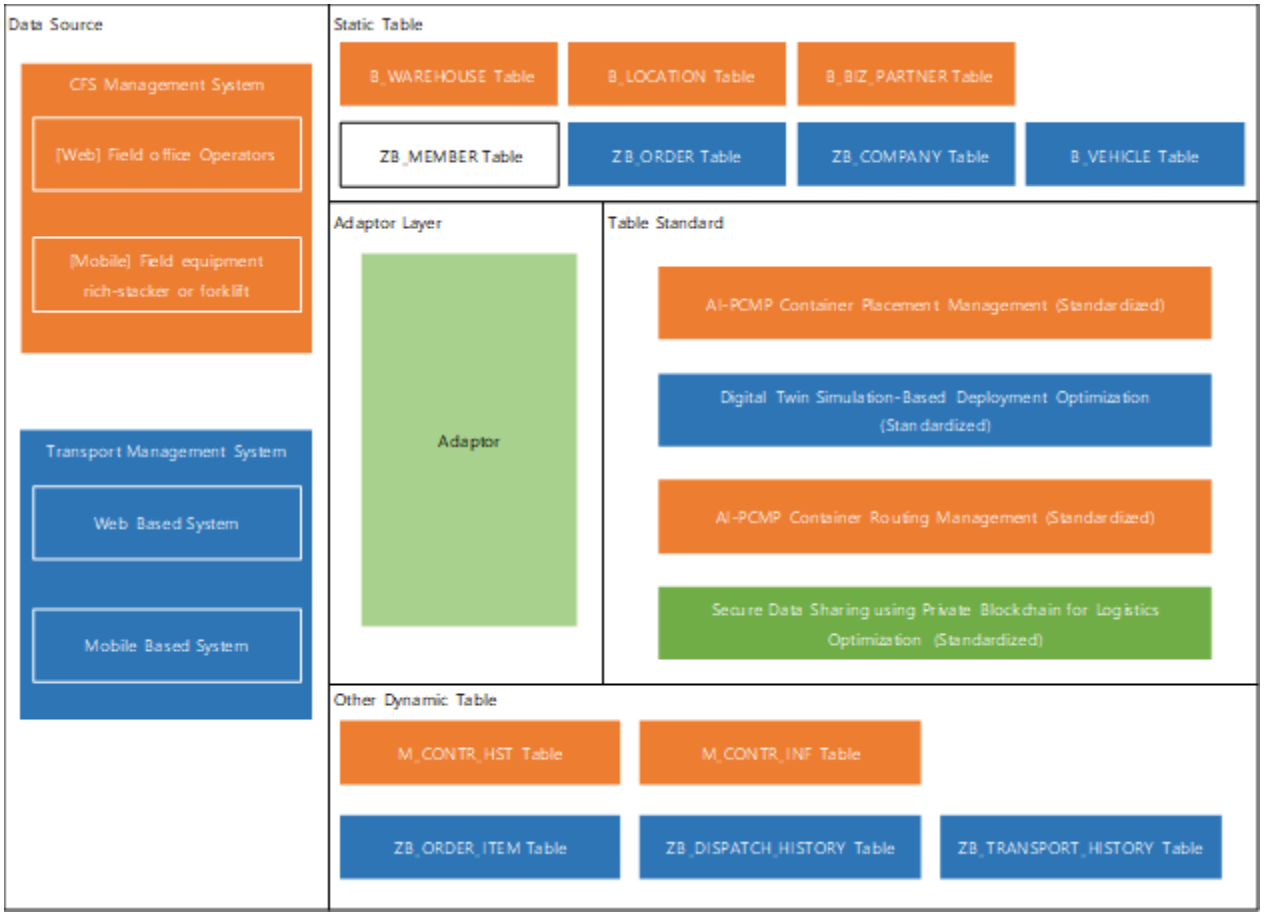

Figure 15 - port data tables overview

The CFSMS solution is designed to monitor the flow and process of container cargo in port terminals. CFSMS is a tool that provides real-time and historical data analysis of container flows, enabling optimization of queue management, behavioral prediction, and overall efficiency improvement for transport vehicles between port terminals and CFS/CY.

The TMS solution can monitor the traffic and movement of container transport vehicles between the port terminal and the CFS/CY through checkpoints that collect various port logistics data from container status information such as location information, departure point, destination, transport status information, expected arrival time, loading operation, and unloading operation, container entry/exit operation information, and container loading and unloading operation.

CFSMS and TMS data are important to support Data Collection Environment, and their approaches to collect data are as follows.

- Web-based CFSMS : On-site work management office generates and collects data such as work order registration and management, import/export work plan registration and management, import/export work instruction, container location inquiry, container inventory inquiry, etc.

- Mobile-based CFSMS : A mobile device mounted on a rich stacker or forklift, which is field work equipment, and the main device uses a tablet to generate and collect data when working on loading, unloading, and transferring containers.
- Web-based TMS : generates and collects data for each event through systems such as transport order registration and management, transportation history registration and management, customer registration and management such as origin, destination, business destination, registration and management of transportation warehouses, transportation vehicle dispatch processing, transport order dispatch status and transport status.
- Mobile-based TMS : A mobile device used by a transport driver or a mobile device loaded on a transport vehicle generates and collects data on each event, including changes in transport status such as dispatch acceptance, transport start, transport waiting, departure declaration, arrival declaration, completion declaration, and location information.

Various information such as the location, speed, departure point, destination, and expected arrival time of the transport vehicle can be accessed in accordance with the following considerations.

- If the fieldwork equipment of the CFS/CY, the rich-stacker and forklift, is within the working radius for each block of the CFS/CY, the estimated arrival time of the transport vehicle, the estimated arrival time, and the actual arrival time are managed.
- In order to predict the expected arrival time of the transport vehicle, various information such as the estimated arrival time may be accessed by using location information including latitude and longitude, data such as speed, departure point, and destination.
- In order to predict the expected arrival time of the transport vehicle, various information such as the expected arrival time may be accessed by using data such as the departure point and the destination of the past history data.
- The actual arrival time of the transport vehicle manages location information including latitude and longitude, speed, departure point, destination, and actual arrival time information.
- The collected data manages various information through data optimization, data prediction, prediction using historical data, and Data Trust Provisioning processes, and the methodology for performing this process is as shown in Deliverable 2.4.
- The container optimization arrangement and management system based on the location information of the CFS/CY distribution warehouse in the port hinterland complex, and the transportation management system specialized in shuttle transportation of import/export containers between port terminals and CFS/CY are based on the web and mobile environments.
- The system is integrated with the operation of CFS/CY distribution warehouses in various port hinterland complexes to track and monitor the status of loading and unloading operations and the location information of containers in real time.

The standardized data table is largely based on the following information:

- location and status information of containers loaded in the container yard in the CFS/CY logistics warehouse in the port rear complex,
- cargo loading/export operation status information of containers, expected time of container entry/exit,

- basic transport order information, transportation details,
- dispatch details,
- transport status information, and
- transportation status information required for the port terminal and the CFS/CY logistics warehouse transport management system in the port rear complex,

To support the above information in the Port platform, the standardized data are introduced to deliver well organized the status and situations in CFS and CY as shown below.

- B_LOCATION : The table registers and manages position control information by dividing the container yard loading section into blocks for each warehouse in CFS/CY.
- M_CONTR_HST : The table manages the Loading/Unloading/Transfer operation history of the container in blocks from the container yard of the CFS/CY
- M_CONTR_INF : The table manages the container's work history information block by block in the container yard of the CFS/CY
- ZB_ORDER: The table registers and manages basic transport order information such as dispatch status and container information.
- ZB_ORDER_ITEM : The table registers and manages detailed transportation order information such as transportation status, transportation destination information, transportation destination information, work place information, and transportation date.
- ZB_DISPATCH_HISTORY : The table manages the dispatch history of the transport order.
- ZB_TRANSPORT_HISTORY : The table manages the status and location information of the transport vehicle, such as the transportation status of the transport order and the dispatching vehicle information, and uses it as historical transportation data.
- B_WAREHOUSE : The table is registered and managed by warehouse of the client's workplace.
- ZB_COMPANY : Manage customer information and customer location information such as carry-out area, carry-in area, and CFS/CY.

Furthermore, in order to provide the forecasting service of the Port platform, the system will be designed to share the forecasting results for container work history information, shipping status and shipping history, in this context the forecast is generated by the project stakeholder system.

3.1 Container Import/Export Logistics - carry in container Order Data (Standardized)

The table in Figure 16 shows a detailed mapping procedures of the process of integrating various data fields related to the operation of the CFS/CY distribution warehouse in the port terminal and the port hinterland complex in a standardized format. These are order registration and management between the port terminal and the CFS/CY workplace of the port hinterland complex, import and export business orders on incoming and outgoing containers, and container status information change.

Consistent data processing throughout the container location management system in CFSMS is essential, and the consistent data processing is performed to support checking order information and processing work requests for equipment such as rich stackers and forklifts in the workplace.

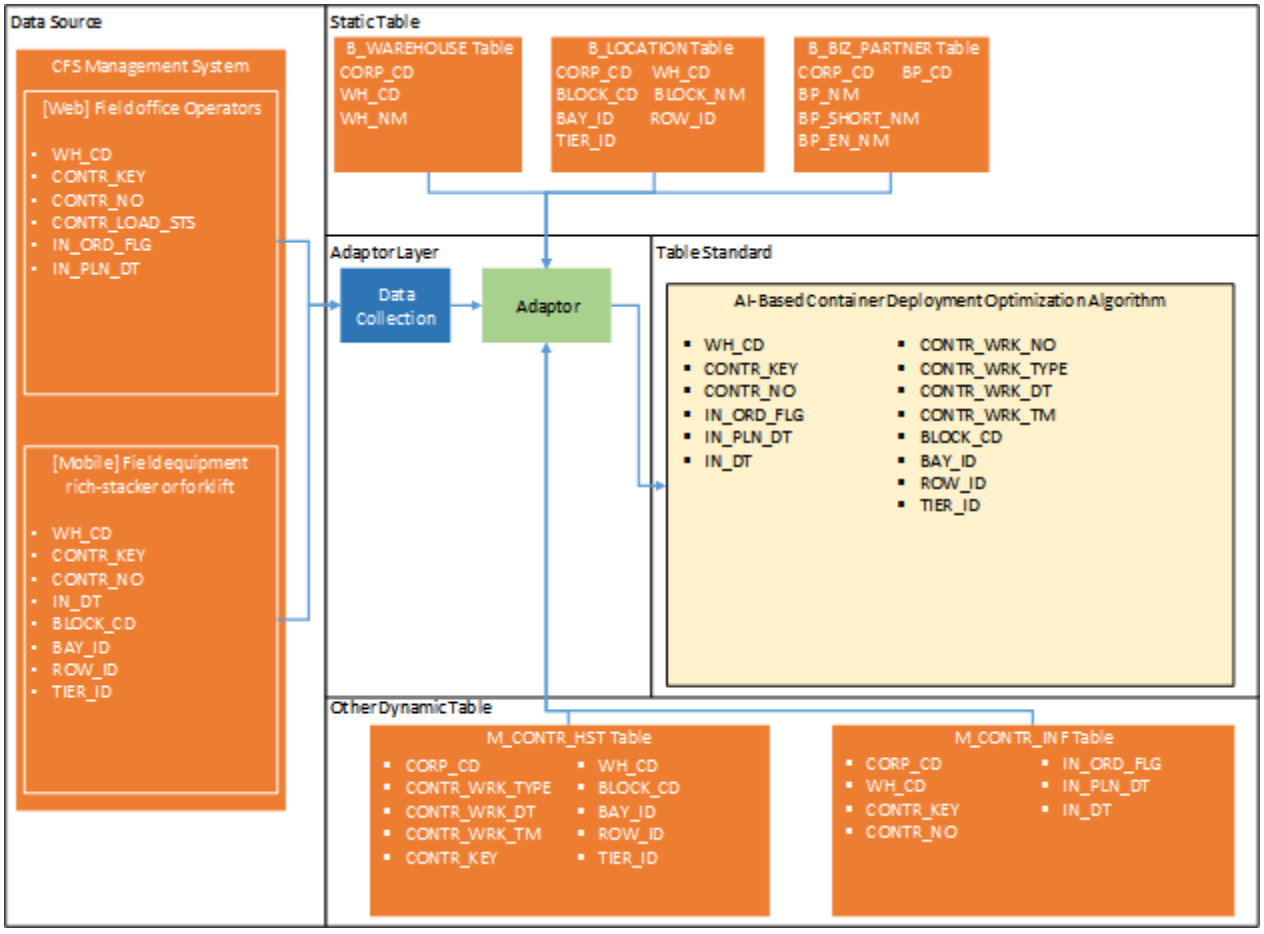

Figure 16 - Container Import/Export Logistics - carry in container Order Data

The IMPORT_EXPORT field represents the export or export of port logistics, which is determined by the registration of the import/export operational command of the CFSMS.

- B_WAREHOUSE: The table provides fields and account code information related to a specific CFS/CY, such as CORP_CD, WH_CD, WH_NM.
- B_LOCATION: The table provides location control information by dividing the container yards of each warehouse into units such as BLOCK_CD, BAY_ID, ROW_ID, TIER_ID, etc.
- M_CONTR_INF : Table manages information such as carry-in/carry-out request information and carry-in/carry-out processing information.

These adapter settings are critical to ensure that the data integration process is smooth and all relevant information is displayed correctly on the system.

Field Description:

- WH_CD: Logistics warehouse code being managed by CFS/CY
- CONTR_KEY : Unique value of container to be carry-in of CFS/CY
- CONTR_NO : Container number to be carry-in of CFS/CY
- IN_ORD_FLG : Status value of the container to be carry-in to CFS/CY
- IN_PLN_DT: Expected date and time of container carry-in to CFS/CY
- IN_DT: Date and time of container unloading operation
- CONTR_WRK_NO: Operation instruction number
- CONTR_WRK_TYPE : When carrying into CFS/CY, work type (M : transfer work, L : Loading work, 7 : unloading work is displayed)
- CONTR_WRK_DT: Working Date
- CONTR_WRK_TM : Working time
- BLOCK_CD: Location of unloading block in containers brought into CFS/CY
- BAY_ID : Location of unloading bays in containers brought into CFS/CY
- ROW_ID : Location of unloading row in containers brought into CFS/CY
- TIER_ID : Location of unloading tiers in containers brought into CFS/CY

Data for confirming the expected date and time of arrival of ships at port terminals during import logistics operations

- BL_NO (BILLOFLADING_NO) :

  - Securities issued by the shipper under the contract of carriage in Port Import Logistics,

  - Information required to receive the shipment listed on the securities

- DO_NO (DELIVEL_ORDER_NO) :

  - documents issued to the ship's captain or bonded warehouse manager,

  - Information needed to receive the cargo

Data for confirmation of port terminal import deadlines in export logistics operations

- BK_NO (BOOKING_NO):
  - Confirmation and reservation of loading space and schedule of ships operated by shippers
  - Maritime shipping company accepted the request and confirmed the booking

3.2 Container Import/Export Logistics - carry in Container Transport Order Data (Standardized)

The table in Figure 17 is a detailed mapping document showing various data fields related to the operation of port terminals in the port hinterland complex and how CFS/CY logistics warehouses are integrated in a standardized format in port logistics.

Port logistics requires data for order registration and management between port terminals and CFS/CY sites, import and export business orders such as incoming and outgoing containers, and container status information change.

Container Location Control System in CFSMS supports two functions of checking work plan information for equipment such as rich stackers and forklifts at the work site and processing work requests.

Container Transport Management System (TMS) provides the management functions as shown below.

- registering and managing transport orders between port terminals and CFS/CY sites,
- processing transport orders,
- handling transportation status of transport orders and location information of transport vehicles,
- forecasting expected arrival time of transport vehicles using historical data, actual arrival time information, and confirmation of transport order information.

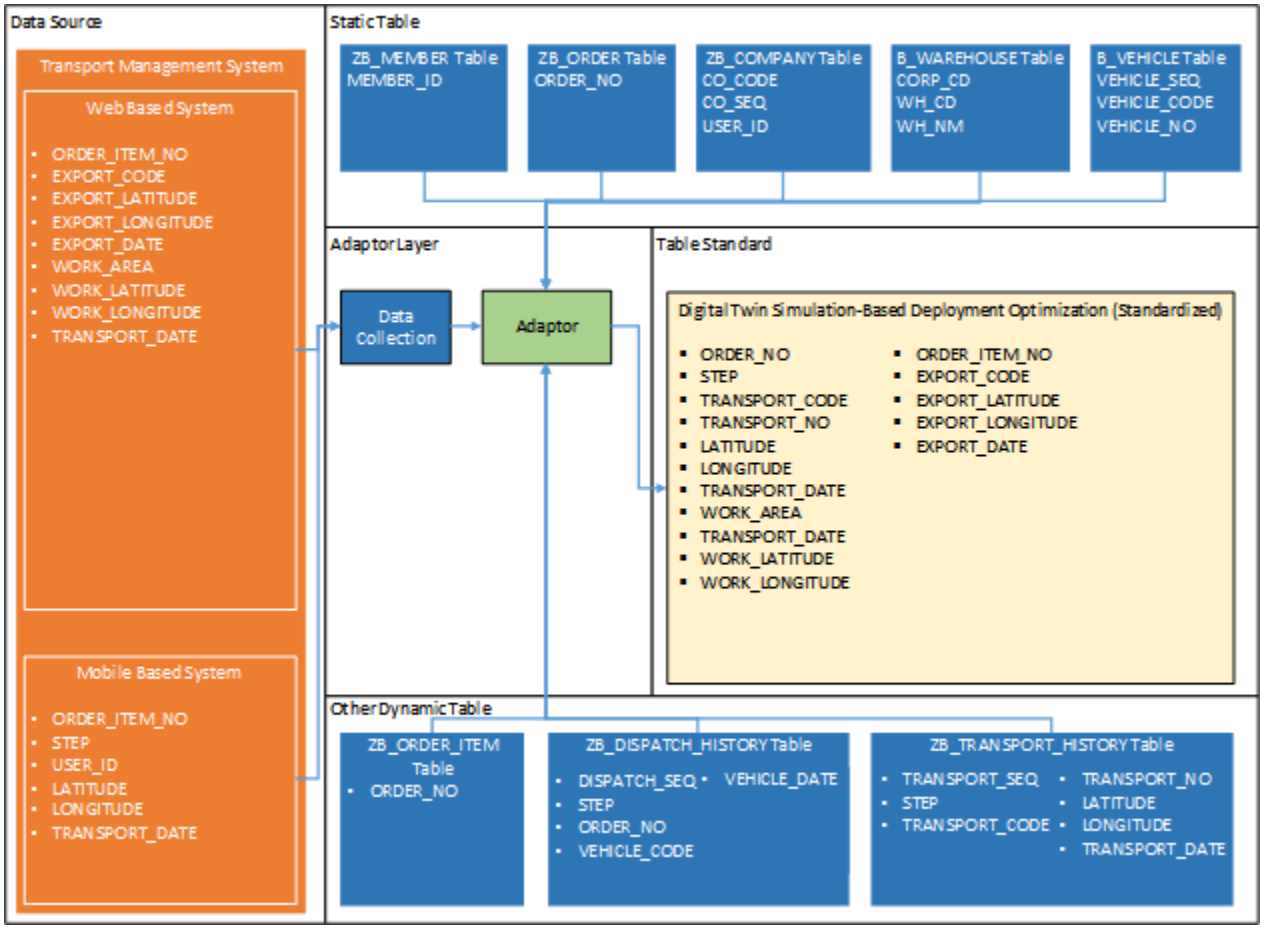

Figure 17 – Container Import/Export Logistics - carry in Container Transport Order Data (Standardized)

IMPORT_EXPORT field represents the export or export of port logistics, which is determined by the registration of the import/export operational command of the CFSMS.

B_WAREHOUSE: The table provides fields and account code information related to a specific CFS/CY, such as CORP_CD, WH_CD, WH_NM.

B_LOCATION: The table provides location control information by dividing the container yards of each warehouse into units such as BLOCK_CD, BAY_ID, ROW_ID, TIER_ID, etc.

M_CONTR_INF : Table manages information such as carry-in/carry-out request information and carry-in/carry-out processing information.

ZB_ORDER: The table registers and manages basic transport order information such as dispatch status and container information.

ZB_ORDER_ITEM: The table registers and manages detailed transportation order information such as transportation status, transportation destination information, transportation destination information, work place information, and transportation date.

ZB_DISPATCH_HISTORY: The table manages the dispatch history of the transport order.

ZB_TRANSPORT_HISTORY: The table manages the status and location information of the transport vehicle, such as the transportation status of the transport order and the dispatching vehicle information, and uses it as historical transportation data.

B_WAREHOUSE: The table is registered and managed by warehouse of the client's workplace.

ZB_COMPANY: Manage customer information and customer location information such as carry-out area, carry-in area, and CFS/CY.

The order data information and order item data describe information about the previous carry-out area, the work area CFS/CY information, and the carry-in area information.

Transport resource allocation dynamically tracks resources such as the availability and expected arrival time and expected arrival time prediction of the transport vehicle based on the transport vehicle's port terminal or CFS/CY's workstation point information.

These adapter settings are critical to ensuring that the data integration process is smooth and all relevant information is displayed correctly on the system.

Field Description:

- WH_CD : Logistics warehouse code being managed by CFS/CY
- CONTR_KEY : Indicates the unique sequence number assigned at the time of container registration.
- CONTR_NO : Container number to be carry-in of CFS/CY
- IN_ORD_FLG : Status value of the container to be carry-in to CFS/CY

- IN_PLN_DT : Expected date and time of container carry-in to CFS/CY
- IN_DT : Date and time of container unloading operation
- CONTR_WRK_NO : Operation instruction number
- CONTR_WRK_TYPE : When carrying into CFS/CY, work type (M : transfer work, L : Loading work, 7 : unloading work is displayed)
- CONTR_WRK_DT : Indicates job date information.
- CONTR_WRK_TM : Indicates operation time information.
- BLOCK_CD : Location of unloading block in containers brought into CFS/CY
- BAY_ID : Location of unloading bays in containers brought into CFS/CY
- ROW_ID : Location of unloading row in containers brought into CFS/CY
- TIER_ID : Location of unloading tiers in containers brought into CFS/CY
- ORDER_NO : Import/Export Container Transport Order Number
- STEP : The dispatch status value of the transport order (0: wait for dispatch, 1: complete dispatch, 2: cancel dispatch)
- VEHICLE_CODE : Transport Vehicle Code
- VEHICLE_NO : Transport Vehicle Number
- LATITUDE : Transport vehicle Latitude location information
- LONGITUDE : Transport vehicle Longitude location information
- TRANSPORT_DATE : Date and time by transport status
- WORK_AREA : Workshop information on CFS/CY Logistics Warehouse
- WORK_LATITUDE : Latitude information for CFS/CY Logistics Depot
- WORK_LONGITUDE : Longitude Information of CFS/CY Logistics Warehouse
- ORDER_ITEM_NO : Indicates the unique sequence number assigned to the transport details.
- ORDER_STEP : Transportation status stage value of transport order (10 : Transport waiting, 20 : Transport start, 30 : Departure report, 40 : In transit, 50 : Arrival report, 60 : Complete report)
- EXPORT_CODE : Code information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_LATITUITY: Latitude information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_LONGITY: Hardness information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_DATE : Scheduled date of export of port terminals

Data for confirming the expected date and time of arrival of ships at port terminals during import logistics operations

- BL_NO (BILLOFLADING_NO) :

  - Securities issued by the shipper under the contract of carriage in Port Import Logistics,

  - Information required to receive the shipment listed on the securities

- DO_NO (DELIVEL_ORDER_NO) :

- documents issued to the ship's captain or bonded warehouse manager,

- Information needed to receive the cargo

Data for confirmation of port terminal import deadlines in export logistics operations

- BK_NO (BOOKING_NO):
  - Confirmation and reservation of loading space and schedule of ships operated by shippers
  - Maritime shipping company accepted the request and confirmed the booking

3.3 Container Import/Export Logistics - Container bring out Order Data (Standardized)

The table shown in Figure 18 contains a detailed mapping document showing the process of integrating various data fields related to the operation of the CFS/CY distribution warehouse in the Port terminal and the port hinterland complex in a standardized format, such as order registration and management between the port terminal and the CFS/CY workplace of the port hinterland complex, import and export business orders such as incoming and outgoing containers, and container status information change.

Consistent data processing throughout the container location management system (CFSMS) system is essential, such as checking order information and processing work requests for equipment such as rich stackers and forklifts in the workplace.

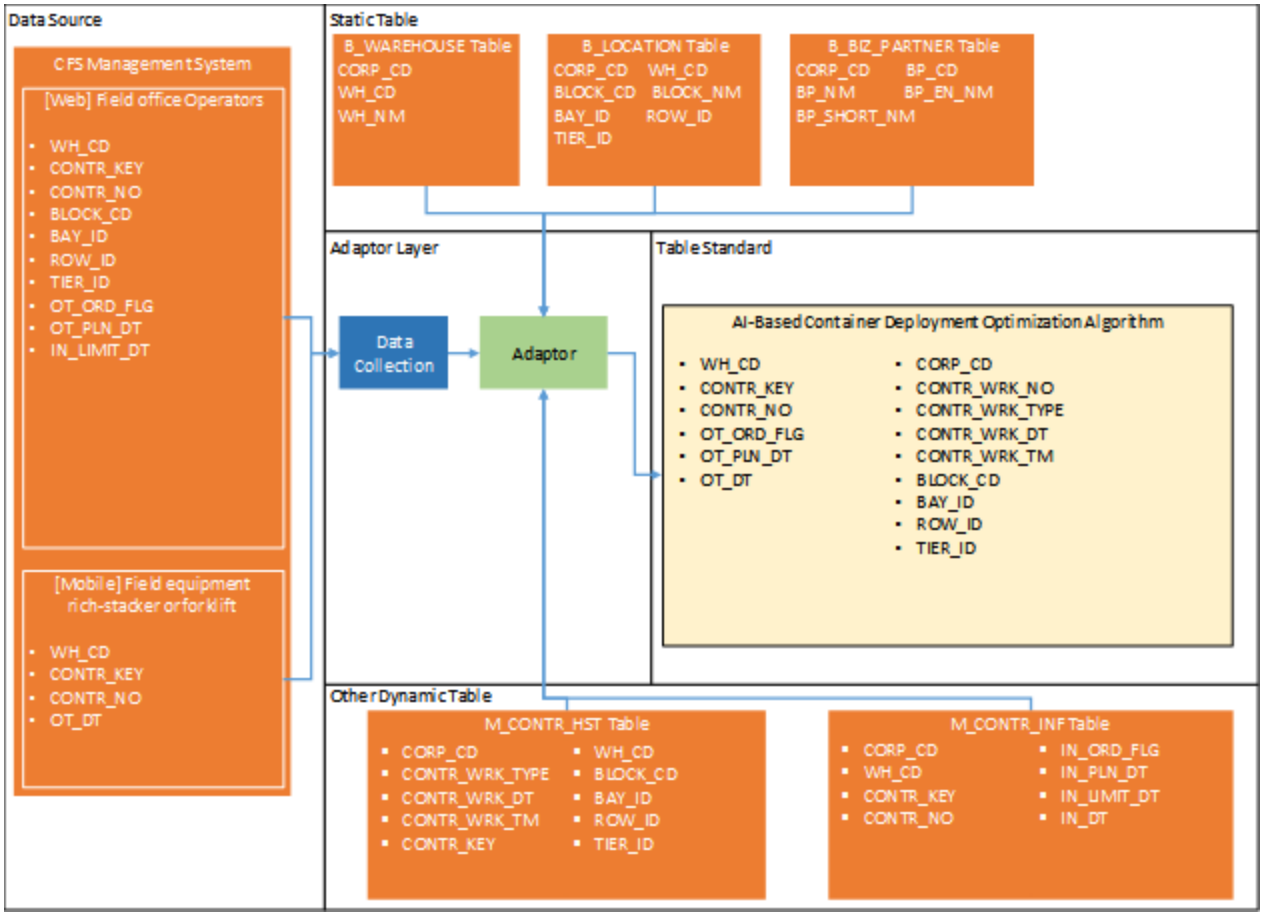

Figure 18 – Container Import/Export Logistics - Container bring out Order Data (Standardized)

The IMPORT_EXPORT field represents the import or export of port logistics, which is determined by the registration of the import/export operational command of the CFSMS.

B_WAREHOUSE: The table provides fields and account code information related to a specific CFS/CY, such as CORP_CD, WH_CD, WH_NM.

B_LOCATION: The table provides location control information by dividing the container yards of each warehouse into units such as BLOCK_CD, BAY_ID, ROW_ID, TIER_ID, etc.

M_CONTR_INF : Table manages information such as carry-in/carry-out request information and carry-in/carry-out processing information.

These adapter settings are critical to ensuring that the data integration process is smooth and all relevant information is displayed correctly on the system.

Field Description:

- WH_CD: Logistics warehouse code being managed by CFS/CY
- CONTROR_KEY : Indicates the unique sequence number assigned at the time of container registration.
- CONTROR_NO : Container number
- OT_ORD_FLG : container carry-out request status value
- OT_PLN_DT : Container carry-out reservation date and time
- OT_DT : Container loading operation date and time
- CONTROR_WRK_NO: Indicates the unique sequence number assigned to the task history of CFS/CY.
- CONTROR_WRK_TYPE : Container operation type (M: previous operation, L: loading operation, 7: unloading operation)
- CONTROR_WRK_DT: Indicates job date information.
- CONTROR_WRK_TM : Indicates operation time information.
- BLOCK_CD: block unit location information for the container.
- BAY_ID : Bay location information of the container.
- ROW_ID : Container row location information.
- TIER_ID : container tier location information.

Data for confirming the expected date and time of arrival of ships at port terminals during import logistics operations

- BL_NO (BILLOFLADING_NO) :

  - Securities issued by the shipper under the contract of carriage in Port Import Logistics,

  - Information required to receive the shipment listed on the securities

- DO_NO (DELIVEL_ORDER_NO) :

  - documents issued to the ship's captain or bonded warehouse manager,

  - Information needed to receive the cargo

Data for confirmation of port terminal import deadlines in export logistics operations

- BK_NO (BOOKING_NO):
  - Confirmation and reservation of loading space and schedule of ships operated by shippers
  - Maritime shipping company accepted the request and confirmed the booking

3.4 Container Import/Export Logistics - Container bring out Operation Transport Order Data (Standardized)

The table in Figure 19 is a detailed mapping document showing various data fields related to the operation of port terminals in the port hinterland complex and how CFS/CY logistics warehouses are integrated in a standardized format in port logistics.

Port logistics requires data for order registration and management between port terminals and CFS/CY sites, import and export business orders such as incoming and outgoing containers, and container status information change.

Container Location Control System (CFSMS) supports a checking work plan information for equipment such as rich stackers and forklifts at the work site and processing work requests.

Container Transport Management System (TMS) supports the following capabilities to perform the followings.

- registering and managing transport orders between port terminals and CFS/CY sites,
- processing transport orders,
- handling transportation status of transport orders and location information of transport vehicles,
- forecasting expected arrival time of transport vehicles using historical data, actual arrival time information, and confirmation of transport order information.

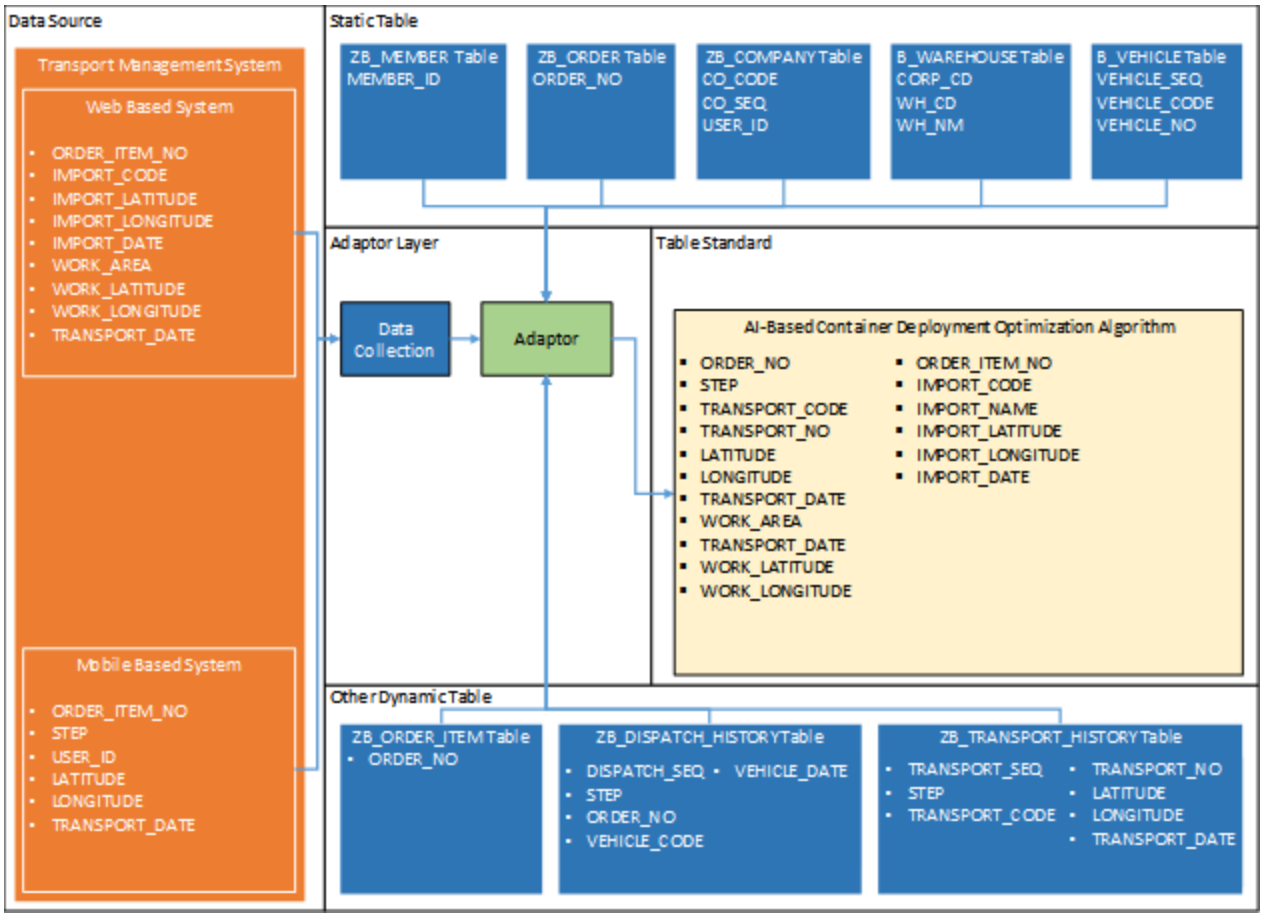

Figure 19 – Container Import/Export Logistics - Import/export Container Transport Order Data (Standardized)

The IMPORT_EXPORT field represents the export or export of port logistics, which is determined by the registration of the import/export operational command of the CFSMS.

B_WAREHOUSE: The table provides fields and account code information related to a specific CFS/CY, such as CORP_CD, WH_CD, WH_NM.

B_LOCATION: The table provides location control information by dividing the container yards of each warehouse into units such as BLOCK_CD, BAY_ID, ROW_ID, TIER_ID, etc.

M_CONTR_INF : Table manages information such as carry-in/carry-out request information and carry-in/carry-out processing information.

ZB_ORDER: The table registers and manages basic transport order information such as dispatch status and container information.

ZB_ORDER_ITEM: The table registers and manages detailed transportation order information such as transportation status, transportation destination information, transportation destination information, work place information, and transportation date.

ZB_DISPATCH_HISTORY: The table manages the dispatch history of the transport order.

ZB_TRANSPORT_HISTORY: The table manages the status and location information of the transport vehicle, such as the transportation status of the transport order and the dispatching vehicle information, and uses it as historical transportation data.

B_WAREHOUSE: The table is registered and managed by warehouse of the client's workplace.

ZB_COMPANY: Manage customer information and customer location information such as carry-out area, carry-in area, and CFS/CY.

The order data information and order_item data describe information about the previous carry-out area, the work area CFS/CY information, and the carry-in area information.

Transport resource allocation dynamically tracks resources such as the availability and expected arrival time and expected arrival time prediction of the transport vehicle based on the transport vehicle's port terminal or CFS/CY's workstation point information.

These adapter settings are applied to ensure that the data integration process is smooth and all relevant information is displayed correctly on the system.

Field Description:

- WH_CD : Logistics warehouse code being managed by CFS/CY
- CONTROR_KEY : Indicates the unique sequence number assigned at the time of container registration.
- CONTROR_NO : Container number
- OT_ORD_FLG : container carry-out request status value

- OT_PLN_DT : Container carry-out reservation date and time
- OT_DT : Container loading operation date and time
- CONTROR_WRK_NO : Indicates the unique sequence number assigned to the task history of CFS/CY.
- CONTROR_WRK_TYPE : Container operation type (M: previous operation, L: loading operation, 7: unloading operation)
- CONTROR_WRK_DT : Indicates job date information.
- CONTROR_WRK_TM : Indicates operation time information.
- BLOCK_CD : block unit location information for the container.
- BAY_ID : Bay location information of the container.
- ROW_ID : Container row location information.
- TIER_ID : container tier location information.
- ORDER_NO : Import/Export Container Transport Order Number
- STEP : Vehicle dispatch status value of transport order (0: dispatch waiting, 1: dispatch complete, 2: dispatch cancellation)
- VEHICLE_CODE : Transport Vehicle Code
- VEHICLE_NO : Transport Vehicle Number
- LATITUDE : Transport vehicle Latitude location information
- LONGITUDE : Transport vehicle Longitude location information
- TRANSPORT_DATE : Date and time by transport status
- WORK_AREA : Workshop information on CFS/CY Logistics Warehouse
- WORK_LATITUDE : Latitude information for CFS/CY Logistics Depot
- WORK_LONGITUDE : Longitude Information of CFS/CY Logistics Warehouse
- ORDER_ITEM_NO : Indicates the unique sequence number assigned to the transport details.
- ORDER_STEP : Transportation status stage value of transport order (10 : Transport waiting, 20 : Transport start, 30 : Departure report, 40 : In transit, 50 : Arrival report, 60 : Complete report)
- IMPORT_CODE : Code information of port terminals and CFS/CY warehouses where containers are carry in
- IMPORT_LATITUITY: Latitude information of port terminals and CFS/CY warehouses where containers are carry in
- IMPORT_LONGITY: Hardness information of port terminals and CFS/CY warehouses where containers are carry in
- IMPORT_DATE : Scheduled date of carry in of port terminals

Data for confirming the expected date and time of arrival of ships at port terminals during import logistics operations

- BL_NO (BILLOFLADING_NO) :

  - Securities issued by the shipper under the contract of carriage in Port Import Logistics,

  - Information required to receive the shipment listed on the securities

- DO_NO (DELIVEL_ORDER_NO) :

  - documents issued to the ship's captain or bonded warehouse manager,

  - Information needed to receive the cargo

Data for confirmation of port terminal import deadlines in export logistics operations

- BK_NO (BOOKING_NO):
  - Confirmation and reservation of loading space and schedule of ships operated by shippers
  - Maritime shipping company accepted the request and confirmed the booking

3.5 AI-Based Container Placement Management (Standardized)

Solves the problem of optimizing container relocation for intelligent container yard management solutions based on AI of CFS/CY distribution warehouses in the port hinterland complex.

With regard to 'scheduling container tasks' and 'managing automatic container placement', container relocation can result in additional energy consumption and unexpected delays during the process, minimizing container relocation and proactively optimizing container placement to reduce energy consumption and improve port logistics operational efficiency.

For container relocation optimization, AI based container placement management function is based on a Markov decision process and a reinforcement learning model in accordance with proximal policy optimization (PPO).

The required data is as follows :

Main data : Port Logistics data Between Port Terminals Collected From CFS/CY Distribution Warehouse. And CFS/CY Distribution Warehouse In Port Hinterland Complex, and Transport Data Between Port Terminals And CFS/CY Distribution Warehouse

- Container ID coming into the port (for example, CONTRO_KEY)
- Estimated time to bring in containers (e.g. IN_PLN_DT)
- Estimated container delivery time (e.g. OT_PLN_DT)
- Container yard status (e.g. location information of containers stored in container yard; bay, row, tier)

Metadata : The above data outputs the following data by entering the input of the development solution.

- Destination container ID (for example, the ID of the moving container determined by the solution)
- Target container departure location (e.g. departure location for moving containers determined by solution; bay, row, tier)
- Destination container arrival location (e.g. the arrival location of the mobile container as determined by the solution; bay, row, tier)
- Target container move ranking (e.g., determine container move order based on solution)
- Number of container location errors (e.g., number of containers placed under low priority despite high priority as a means of verifying optimization within container yards)

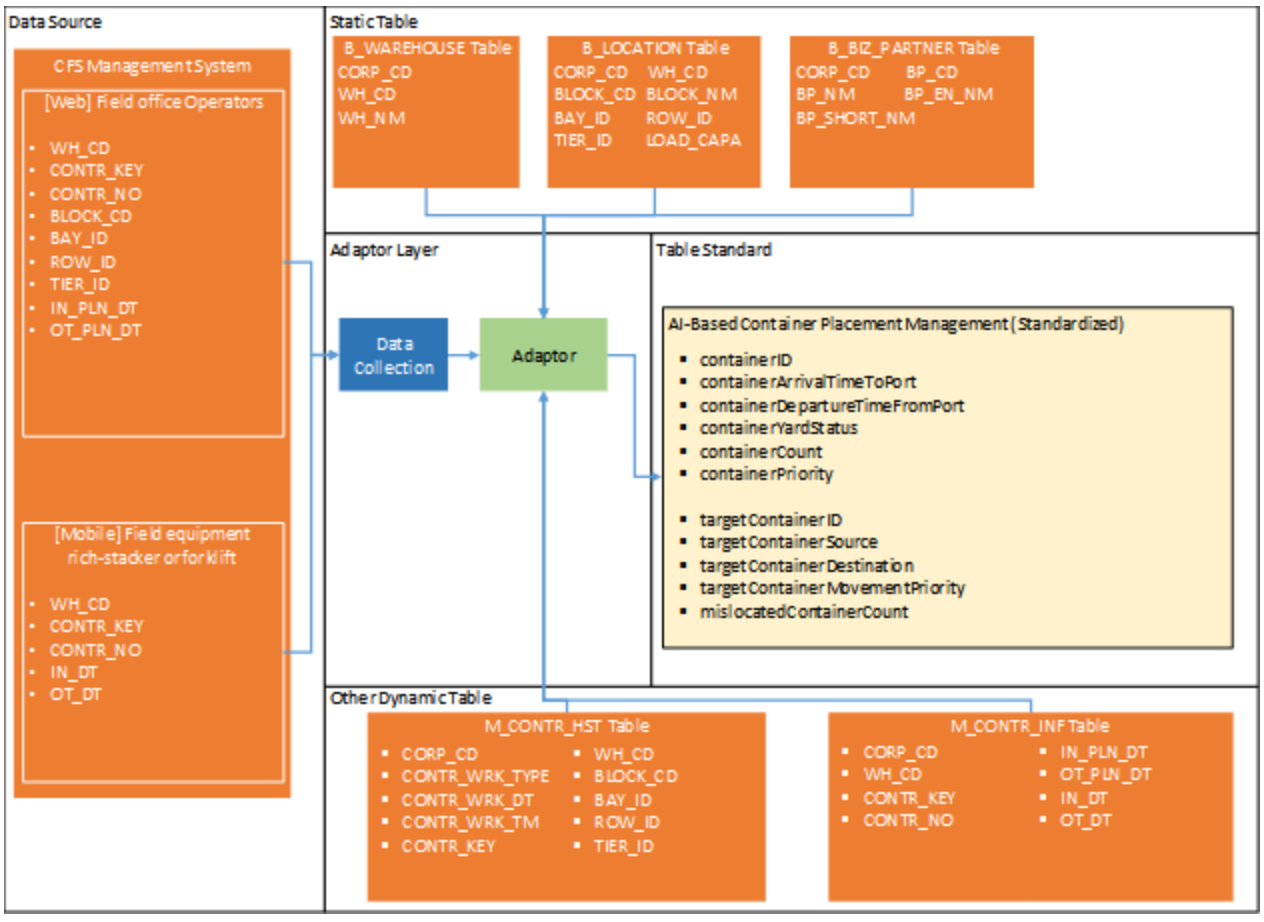

Figure 20 – AI-Based Container Placement Management (Standardized)

Field Description:

- containerID : container's unique identifier ID
- ContainerArrivalTimeToPort : Estimated arrival time of container at port terminal (Arrival at port terminal : Imported logistics)

- ContainerDepartmentTimeFromPort : Estimated time to dock the port terminal of the container ship (Port terminal departure: Export logistics)
- containerYardInfo : container yard information (ex. block, bay, row, and end)
- containerYardStatus : container yard current status (ex. container loading status)
- ContainerCount : Total number of containers stored in the container yard
- ContainerPriority : Processing priority of containers stored in container yards

- targetContainerID : TargetContainerID whose movement has been determined through the CRP algorithm
- targetContainerSource : departure location of the target container (ex. block, bay, row, but)
- targetContainerDestination : Arrival location of target container (ex. block, bay, row, but)
- targetContainerMovementPriority : Priority of target container movement (sequence)
- MislocatedContainerCount : Number of containers mislocated (in reverse order) within the container yard

3.6 AI-Based Container Routing Management (Standardized)

Container Routing Management (Standardized) is the name of a solution that optimizes the storage location of incoming containers among various container yard (CFS) environments, and the direction of development is currently being planned and has not yet been specified (Contents of the 3rd and 4th year development research)

According to the plan so far, the data you need is as follows :

Main data: Port logistics data between the port terminal including location information of containers collected from the CFS/CY logistics warehouse and the CFS/CY logistics warehouse in the port background complex, and transport data between the port terminal and the CFS/CY logistics warehouse

- ID of containers entering the port (ex. CONTR_KEY)
- Expected time to bring in the container (ex. IN_PLN_DT)
- Estimated time to ship out containers (ex. OT_PLN_DT)
- Maximum container yard capacity (ex. maximum number of containers a container yard can hold)
- LOAD_CAPA bay,row,tier
- Container yard status (ex. Location information of containers held in container yards; bay, row, tier)
- Vehicle Departure Time and Arrival Time

  ※ Data requirements related to the 3rd year's project will be specified later

Metadata: The above data enters the input value of the development solution and outputs the following data.

- Target container ID (ex. ID of the carry-in container you want to move)
- Target container arrival CFS ID (ex. final storage CFS of target container determined by solution)
- Estimated arrival time for target containers (ex. Estimated arrival time until final storage CFS of target containers determined by solution)

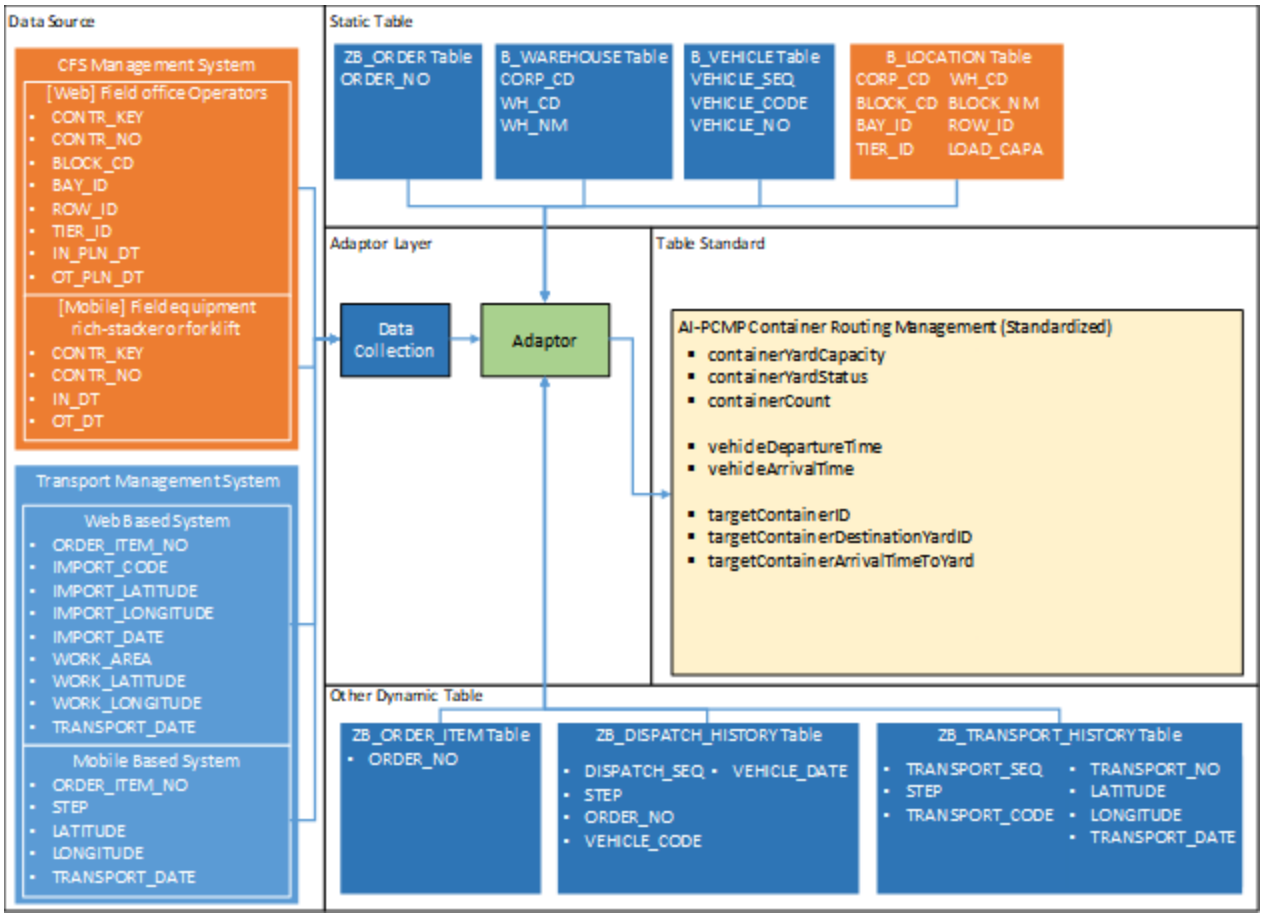

Figure 21 – AI-Based Container Routing Management (Standardized)

Field Description:

- containerID : container's unique identifier ID
- containerYardInfo : container yard information (ex. using block, bay, row, tier units)
- Container Yard Capacity : Container yard capacity (ex. maximum container capacity)
- containerYardStatus : container yard current status (ex. container loading status)
- ContainerCount : Total number of containers stored in the container yard

- VehicleDepartureTime : Estimated time to leave the transport vehicle (current location)
- Vehicle Arrival Time : Estimated time of arrival of transport vehicle (port)
- targetContainerID : TargetContainerID whose movement has been determined through the CRP algorithm

- targetContainerDestinationYardID : Unique identifier of the container yard where the target container arrives
- targetContainerArrivalTimeToYard : Destination Container Estimated Time of Arrival (Container Yards)

3.7 Shuttle transportation optimization data through digital twin simulation (Standardized)

Digital twin simulation-based optimal dispatching system for container transport between CFS/CY and port terminals in port hinterland complex.

Develop services to optimize transportation schedules between terminals and CFSs by utilizing containers and transport operators, which are cargo data participating in port logistics between port terminals and CFS/CY.

It identifies the optimization model and data of the system required for operation, and stores the simulation and optimization results.

It reflects real data on digital twins and provides optimization model development and container transport schedule optimization services.

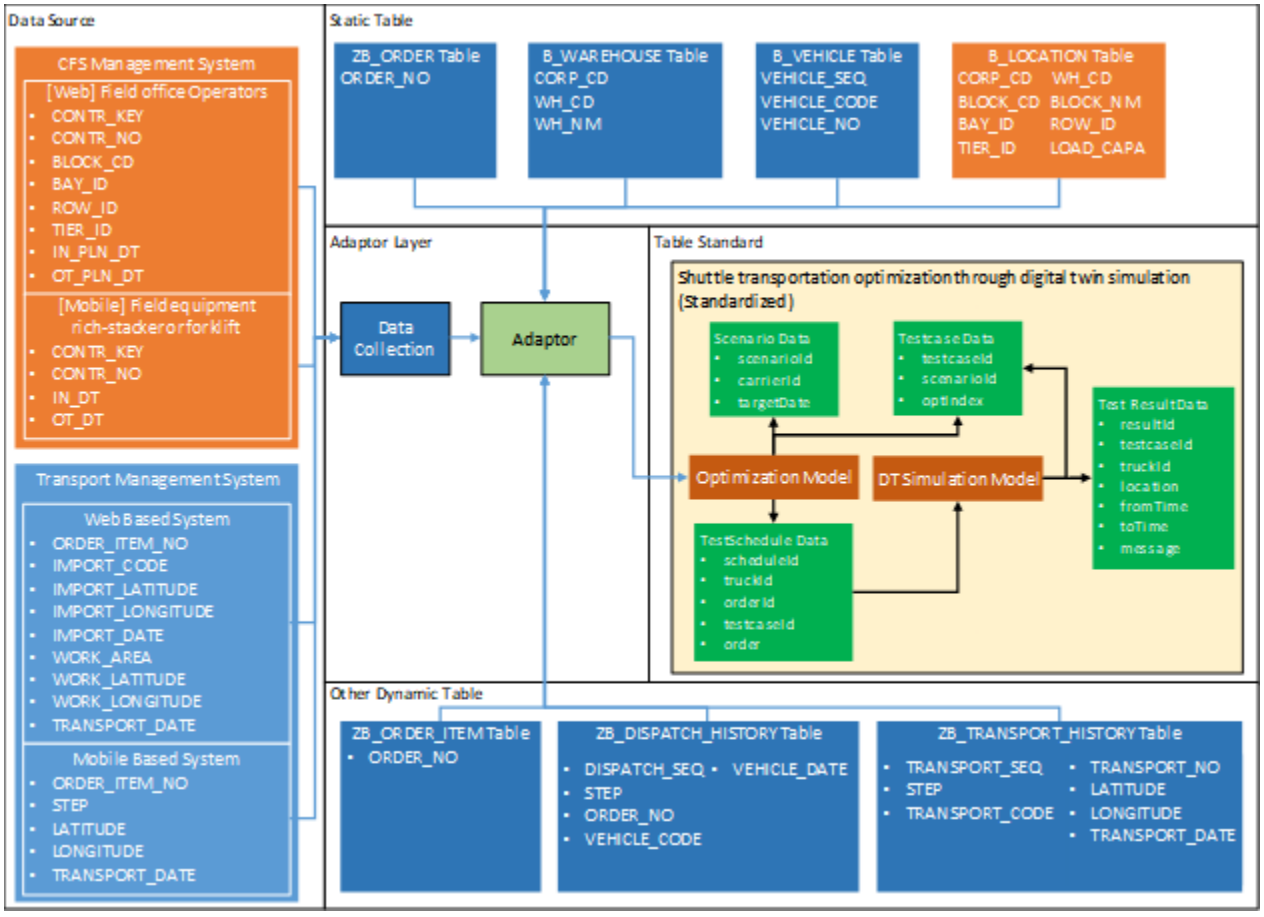

Figure 22 – Shuttle transportation optimization data through digital twin simulation (Standardized)

The service consists of physical system data, optimization algorithm, and digital twins as follows:

Actual system data:

- Various data generated in the process of creating and executing a schedule, such as operating environment, contract information, and logistics information, are included.

Optimization Algorithm applied:

- Create and execute transportation schedules such as operating environment, contract information, and logistics information.
- Minimize costs by evaluating transportation schedules using a variety of data generated during execution.

Digital Twin:

- Analyze the collected data and apply it to attributes and behaviors.
- An optimization index is generated by simulating a given transport schedule.

The optimization index can be calculated by simulating a given transport schedule. Digital twins are implemented with various objects and processes that simulate real-world workflows.

It is designed to identify the data of the optimization model and the actual system data needed for digital twin operations and to store simulation and optimization results.

Simulation results can be applied to the platform by verifying the model's behavior, calculating the optimization index, and selecting the results of the optimized transport schedule optimization service among the simulation scenarios.

The data needed for optimization models and digital twin operations are as follows :

Main data: Port logistics data between the port terminal including location information of containers collected from the CFS/CY logistics warehouse and the CFS/CY logistics warehouse in the port background complex, and transport data between the port terminal and the CFS/CY logistics warehouse

Field Description:

- WH_CD : Logistics warehouse code being managed by CFS/CY
- CONTRO_KEY : Indicates the unique sequence number assigned at the time of container registration.
- CONR_NO : container number of CFS/CY
- IN_ORD_FLG : Container carry request status value

- IN_PLN_DT : Expected date and time to bring in the container
- IN_DT : Date and time of container unloading operation
- OT_ORD_FLG: container export request status value
- OT_PLN_DT : Date and time of container export reservation
- OT_DT : Date and time of container loading operation
- CONR_WRK_NO: Operation instruction number
- CONR_WRK_TYPE: Type of work during carry-in/out operation (M: transfer operation, L: loading operation, 7: unloading operation)
- CONTR_WRK_DT : Indicates job date information.
- CONTR_WRK_TM : Indicates operation time information.
- BLOCK_CD : Block position of container
- BY_ID : the bay location of the container
- ROW_ID : row location of container
- TIER_ID : Tier location of container
- ORDER_NO : Import/Export Container Transport Order Number
- Step: The transport vehicle dispatch status value of the transport order (0: Standby for dispatch, 1: Complete dispatch, 2: Cancel dispatch)
- VEHICLE_CODE : Transport Vehicle Code
- VEHICLE_NO : Transport Vehicle Number
- LATITUDE : Transport vehicle Latitude location information
- LONGITUDE : Transport vehicle Longitude location information
- TRANSPORT_DATE : Date and time by transport status
- WORK_AREA : Workshop information on CFS/CY Logistics Warehouse
- WORK_LATITUDE : Latitude information for CFS/CY Logistics Depot
- WORK_LONGITUDE : Longitude Information of CFS/CY Logistics Warehouse
- ORDER_ITEM_NO : Indicates the unique sequence number assigned to the transport details.
- ORDER_STEP : Transportation status value of transport order (10 : Waiting for transportation, 20 : Starting transportation, 30 : Report of departure, 40 : In transit, 50 : Report of arrival, 60 : Report of completion)
- EXPORT_CODE: Code information for port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_LATITUITY : Latitude information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_LONGITY : Hardness information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_DATE : Estimated date of export to port terminal
- IMPORT_CODE : Code information for port terminals and CFS/CY warehouses where containers are brought in
- Import_LATITUITY : Latitude information for port terminals and CFS/CY warehouses where containers are brought in
- Import_LONGITY : Hardness information of port terminals and CFS/CY warehouses where containers are brought in

- IMPORT_DATE : Scheduled date of carry in of port terminals

Metadata : The main data above enters the input value of the development solution, identifies the data needed for the optimization model and the digital twin operation simulation model, and the data output from the optimization and simulation results are as follows.

Data Required for Optimization Model

Scenario Data

- ScenarioId : ID for optimization simulation scenario
- carrierId : Target transport company ID for optimization
- Target Date : Target date for optimization

Test Case Data

- testcaseId : ID for each optimization test according to each scenario.
- ScenarioId : ID for optimization simulation scenario.
- optIndex : Optimization Index of testcase.

Data Required for Digital Twin Simulation Models

Test Schedule Data

- scheduleId : ID for tasks scheduled in the simulation.
- truckId : Transport vehicle ID
- OrderId : Transport Order ID
- testcaseId : ID for each optimization test according to each scenario
- Order : Order of operations

Test Results Data

- Result ID : ID for simulation result record
- testcaseId : ID for each optimization test according to each scenario
- truckId : Transport vehicle ID
- Location : Location (port terminal to be brought in, location information of work area, port terminal to be brought in)
- From Time : Start date and time of the task
- toTime : End date and time of the task
- Message : Remarks or notes

3.8 Secure Data Sharing using Private Blockchain for Logistics Optimization (Standardized)

Figure 23 provides information necessary for CFS/CY operation, dispatch processing, and transport vehicle management using blockchain technology between the port terminal and the CFS/CY distribution warehouse in the port hinterland complex as follows:

- Import and Export Order Information,
- Import and Export Transportation Order Information,
- Import/export transportation order transportation vehicle dispatch request information,
- Check the dispatch order information of the import and export order,
- Change of transport status of import/export transportation order and transfer of location information,
- Completion of transportation of import/export vehicles

Consistent data processing across the CFS/CY Operational Management System (CFSMS) and Traffic Management System (TMS) systems is entered through the smart contract of the development solution,

Order information for equipment such as rich-stackers and forklifts at the work site, processing work requests, checking the dispatch information of transport vehicles, and checking the transportation status.

Important information such as personal information and corporate information is encrypted as a hash value through an encryption module, and the input value is input through a smart contract.

Securely share data between CFS/CY Operations Manager, Transport Vehicle Distribution Manager, Transport Vehicle Distribution Manager and Transport Vehicle via smart contracts on the blockchain, including Transport Order Vehicle Distribution Request, Transport Order Vehicle Distribution Information, Transport Vehicle Location and Transport Status Information.

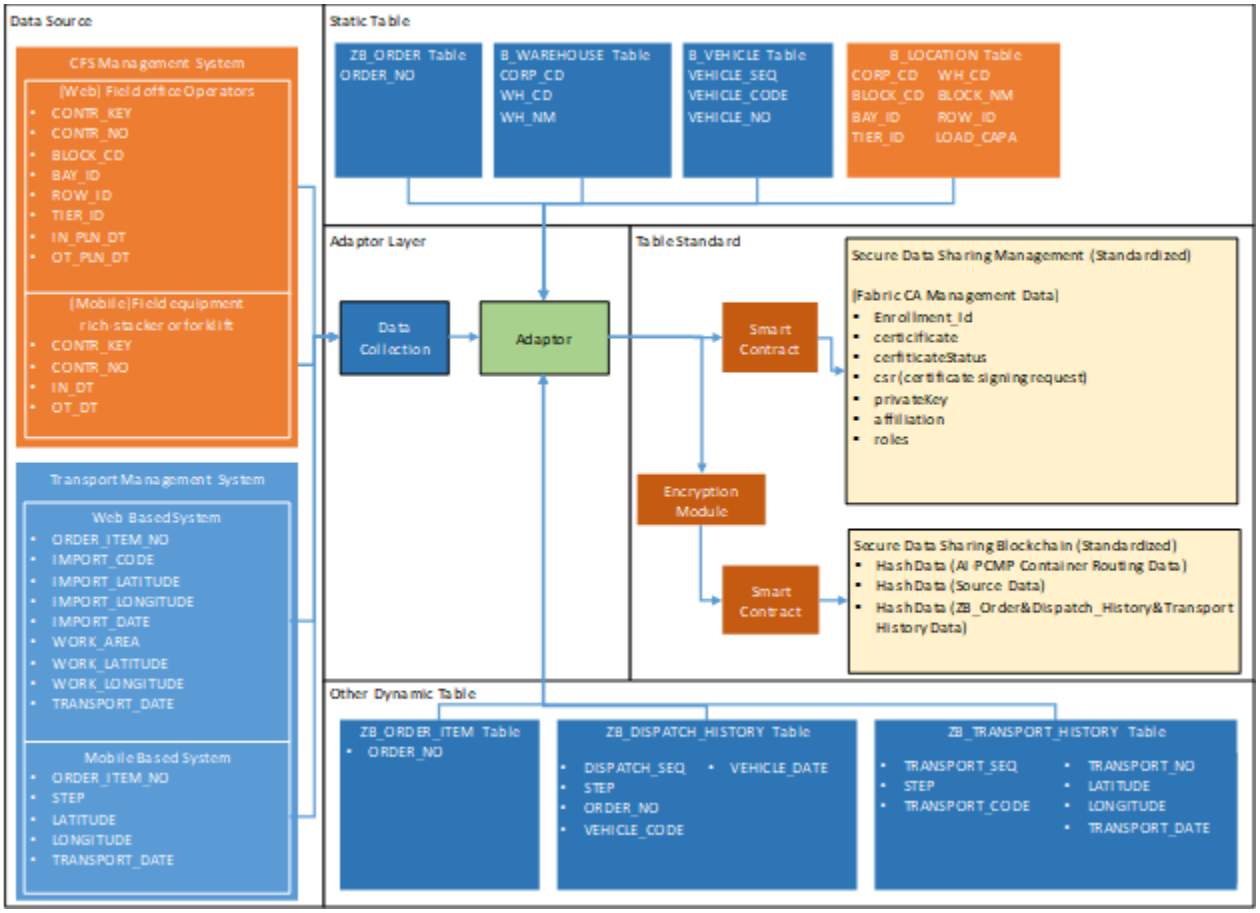

Figure 23 – Secure Data Sharing using Private Blockchain for Logistics Optimization (Standardized)

The data required for CFS/CY operation, transport vehicle dispatch processing and transport vehicle management using blockchain technology are as follows:

Main data : Port logistics data between the port terminal including location information of containers collected from the CFS/CY logistics warehouse and the CFS/CY logistics warehouse in the port background complex, and transport data between the port terminal and the CFS/CY logistics warehouse

Field Description:

- WH_CD : Logistics warehouse code being managed by CFS/CY
- CONTRO_KEY : Indicates the unique sequence number assigned at the time of container registration.
- CONR_NO : container number of CFS/CY
- IN_ORD_FLG : Container carry request status value
- IN_PLN_DT : Expected date and time to bring in the container
- IN_DT : Date and time of container unloading operation
- OT_ORD_FLG: container export request status value

- OT_PLN_DT : Date and time of container export reservation
- OT_DT : Date and time of container loading operation
- CONR_WRK_NO: Operation instruction number
- CONR_WRK_TYPE: Type of work during carry-in/out operation (M: transfer operation, L: loading operation, 7: unloading operation)
- CONTR_WRK_DT : Indicates job date information.
- CONTR_WRK_TM : Indicates operation time information.
- BLOCK_CD : Block position of container
- BY_ID : the bay location of the container
- ROW_ID : row location of container
- TIER_ID : Tier location of container
- ORDER_NO : Import/Export Container Transport Order Number
- Step: The transport vehicle dispatch status value of the transport order (0: Standby for dispatch, 1: Complete dispatch, 2: Cancel dispatch)
- VEHICLE_CODE : Transport Vehicle Code
- VEHICLE_NO : Transport Vehicle Number
- LATITUDE : Transport vehicle Latitude location information
- LONGITUDE : Transport vehicle Longitude location information
- TRANSPORT_DATE : Date and time by transport status
- WORK_AREA : Workshop information on CFS/CY Logistics Warehouse
- WORK_LATITUDE : Latitude information for CFS/CY Logistics Depot
- WORK_LONGITUDE : Longitude Information of CFS/CY Logistics Warehouse
- ORDER_ITEM_NO : Indicates the unique sequence number assigned to the transport details.
- ORDER_STEP : Transportation status value of transport order (10 : Waiting for transportation, 20 : Starting transportation, 30 : Report of departure, 40 : In transit, 50 : Report of arrival, 60 : Report of completion)
- EXPORT_CODE: Code information for port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_LATITUITY : Latitude information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_LONGITY : Hardness information of port terminals and CFS/CY warehouses where containers are taken out
- EXPORT_DATE : Estimated date of export to port terminal
- IMPORT_CODE : Code information for port terminals and CFS/CY warehouses where containers are brought in
- Import_LATITUITY : Latitude information for port terminals and CFS/CY warehouses where containers are brought in
- Import_LONGITY : Hardness information of port terminals and CFS/CY warehouses where containers are brought in
- IMPORT_DATE : Scheduled date of carry in of port terminals

Metadata : The above key data is entered through Smart Contact in the development solution, Important information such as personal information and corporate information is encrypted as a hash value through the Encryption Module, and the input value is input through Smart Contact. Securely share data between CFS/CY operation managers, dispatch managers, and transportation vehicles through the blockchain, such as requesting dispatch of transportation orders, checking transportation order dispatching information, and transportation vehicle location and transportation status information. The data shared is as follows.

Fabric CA Management Data

- enrollmentId: The ID issued when the user is registered.
- certificate: A certificate issued by Fabric CA, proving the user's identity.
- certificateStatus: The status of the issued certificate (e.g., VALID, REVOKED), managing the validity of the certificate.
- csr (Certificate Signing Request): A certificate signing request used when a client requests a certificate from Fabric CA.
- privateKey: The user's private key, used in conjunction with the certificate for secure communication.
- affiliation: The organization or department the user belongs to.
- roles: The user's role, determining their permissions within the network.

Secure data sharing blockchain

- Hash data (import/export container order information data): The hash value of the import/export information data of the import/export container.
- Hash data (source data): The hash value of the source data.
- Hash data (transport order information, dispatch history information, transport status information):
  - Transport order information for containers
  - Transport vehicle dispatch history information
  - the state of transportation of a transport vehicle
  - Location information data

3.9 Column Definition Form.

Column definition among data standardization forms used in CFS/CY logistics warehouses and port terminals and container import/export delivery order management systems in the port hinterland complex.

The instructions used in the form are as shown in Figure 26 as shown below:

- Document format is used with Excel documents to provide filtering functions such as database name, table name, field name, field name, field abbreviation, description, data type, data size, data format, use, privacy, encryption, and more.
- Using a database and table-specific field management ledger, such as adding and modifying fields.

* Attached is the Excel file, which is a data standardization form.

Column Definition (컬럼 정의서)

no	DATABASE	TABLE	Field	Field (Initials Case)	DESCRIPTION (ENG)	DESCRIPTION (KOR)	DATA TYPE	DATA SIZE
1	CFSMS	M_CONTR_INF	CORP_CD	CORP_CD	Company classification code	업체 구분코드	varchar	10
2	CFSMS	M_CONTR_INF	WH_CD	WH_CD	Warehouse code	창고코드	varchar	10
3	CFSMS	M_CONTR_INF	CONTR_KEY	CONTR_KEY	Classification number	구분번호	int	4
4	CFSMS	M_CONTR_INF	STOCK_FLG	STOCK_FLG	Inventory status	재고상태	varchar	1
5	CFSMS	M_CONTR_INF	CONTR_NO	CONTR_NO	Container Number	컨테이너 번호	varchar	20
6	CFSMS	M_CONTR_INF	CONTR_SIZE	CONTR_SIZE	Container Specification	컨테이너 규격	varchar	3
7	CFSMS	M_CONTR_INF	CONTR_TYPE_CD	CONTR_TYPE_CD	Container Type	컨테이너 타입	varchar	10
8	CFSMS	M_CONTR_INF	CONTR_LOAD_STS	CONTR_LOAD_STS	Loaded status	적재상태	varchar	1
9	CFSMS	M_CONTR_INF	CONTR_STS	CONTR_STS	Container Status	컨테이너 상태	varchar	3
10	CFSMS	M_CONTR_INF	SEAL_NO	SEAL_NO	Container room number	컨테이너 실 번호	varchar	30

Figure 24 Column Definition Form 1.

DATA FORMAT	Not Null Y/N	PK Y/N	FK Y/N	AK Y/N	Constraints terms	USE Y/N	PERSONAL Y/N	ENCRYPT Y/N	OPEN/CLOSE Y/N
KULSNC	Y	Y							
KNC10									
34073	Y	Y							
Y									
MRSU0219470									
20	Y								
GP									
F					E:빈 컨테이너, F:화물이적입된 컨테이너				
S					P:계획, S:재고, O:출고				
KR0730916									

Figure 25 Column Definition Form 2.

Item	Item Definition and Creation Guidelines
DATABASE	Write the name of the database to which the column belongs
TABLE	Write the name of the table to which the column belongs
Field	The physical English name of the column
COLUMN (Initials Case)	Use abbreviations as the physical English name of the column
DESCRIPTION (ENG)	Additional description of column
RELATED ENTITY (ENG)	Write the logical data element 'entity name' that the column represents
DATA TYPE	Write the physical DBMS datatype of the column value
DATA SIZE	Write the length of the physical DBMS data of the column value
DATA FORMAT	The format of the data to represent the values in that column
Not Null Y/N	Indicates whether a column value must exist at the time the data is generated (insert)
Primary Key Y/N	If a column participates in a PK (default key), mark it using the "PK" and numeric order of participation and omit it if you do not participate in a PK

Alternate Key Y/N	If you are a column participating in an AK (alternative key), mark it using the "AK" and numeric order of participation, and if you are not participating in an AK, skip it
Foreign Key Y/N	Write the relevant table name and column name by a period (.) only if the column is one that participates in an FK (external key) constraint
Constraints terms	In addition to the characteristics of the column values described in the corresponding value area, additional constraints to be specified for the column (such as allowable range, delimitation, and default values) are described
USE Y/N	Write whether the column is used or not
PERSONAL Y/N	Whether the column values contain personal information ("Personal Information De-identification Action Guidelines" Identifier Action Criteria)
ENCRYPT Y/N	Write whether the column is encrypted for privacy reasons, etc
OPEN/CLOSE Y/N	Whether the metadata and source data information in the column are disclosed or not disclosed (in the case of non-disclosure, the reason for non-disclosure is stated)

Figure 26 Item Definition and Creation Guidelines