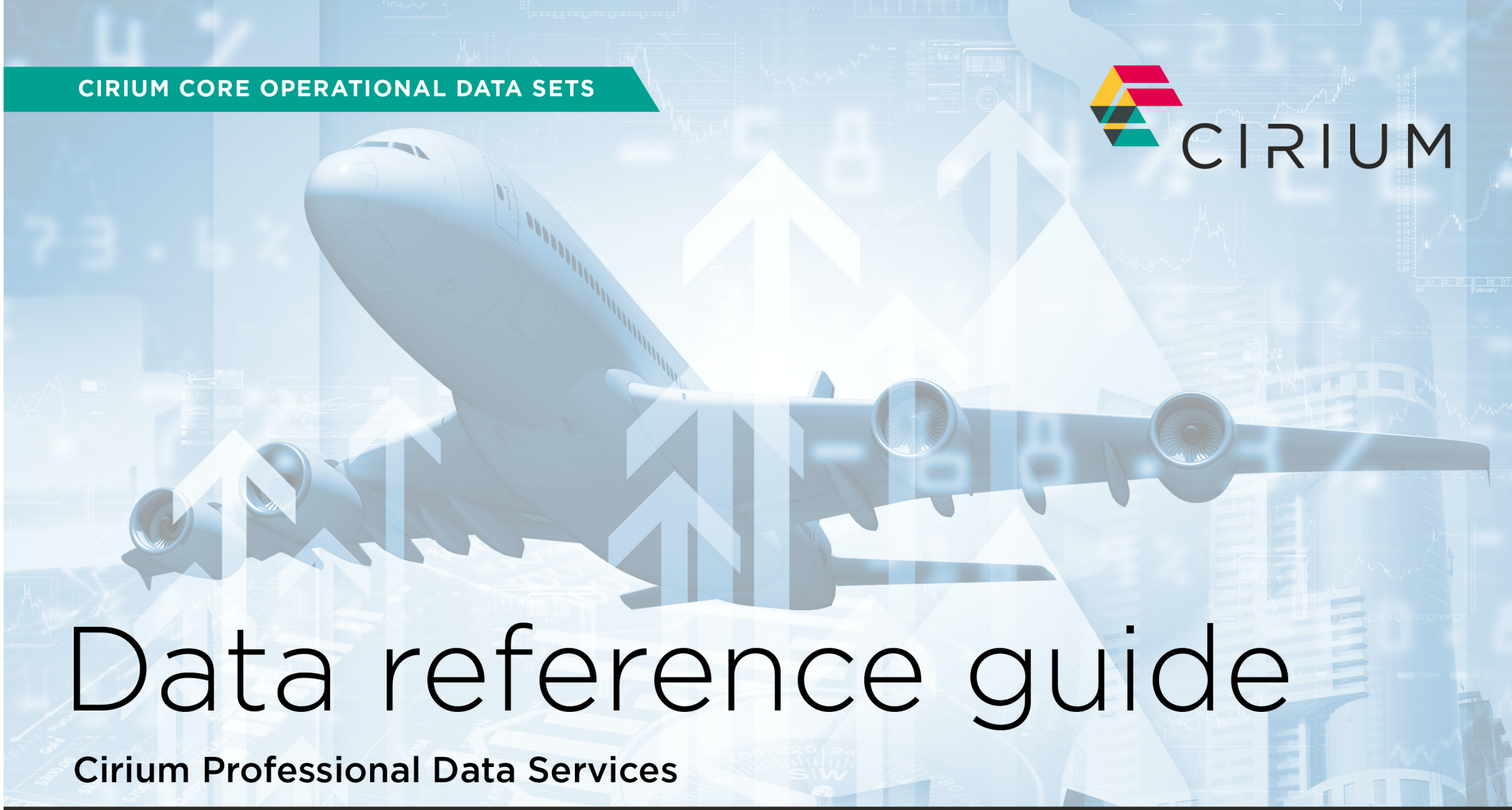
CIRIUM CORE OPERATIONAL DATA SETS



# Data reference guide

## Cirium Professional Data Services

This document provides information about specific datasets from The Cirium Core – the heart of Cirium which ingests millions of pieces of data every day from every corner of the air travel industry and transforms them for real-world use.

Cirium Professional Data Services offers a custom approach to data insights and in order to help interpret the data this document acts as a Data Dictionary. It includes information about how coverage, tracking and performance number are calculated. This data dictionary includes historical data including: operations, flight status, airline operator, airports, weather information (departure and arrival), equipment details and fleet data (engine and aircraft).

Version 3.1 Last Updated: January 22, 2021. Originally authored by Professional Data Services.



**Cirium**
99 Bishopsgate
3rd Floor
London
EC2M 3AL
United Kingdom
Tel: +44 207 911 1400

522 SW 5th Ave
Suite 200
Portland
OR 97204
USA
Tel: +1 503 274 0938

# CONFIDENTIALITY STATEMENT

This document contains the confidential and proprietary information of Cirium and its respective affiliates, subsidiaries, vendors and/or suppliers and is being provided to you for the sole and exclusive purpose of providing an understanding of the Cirium Core Operational Datasets and calculations.
Your receipt and review of this document constitutes your agreement to keep this document and its contents confidential, as well as your agreement to refrain from showing, reading , using or disclosing this document and/or its content to any third party (including affiliates) other than your or your employees on a need to know basis, without Cirium's prior written approval.

If for any reason you do not agree to or cannot keep the contents of this document confidential as stated above, please return this document to Cirium or destroy and purge this document (on any print or digital format) and any copies made thereof. In the event of destruction, please provide Cirium with a letter certifying destruction of this document. If Cirium does not receive such a written confirmation of your destruction of this document, Cirium will assume that you will continue to agree and abide by the confidentiality obligations stated above.

# DISCLAIMER OF LIABILITY

This document shall be used by you for the sole purpose of considering licensing of the Cirium's core operational datasets under a license agreement.

All intellectual property rights in this document remain vested in Cirium and any references to third party tradenames or trademarks in our document, save where expressly provided, is neither an assertion of ownership or representation of an association with the owners of such tradenames or trademarks.

The information and data contained in this document may have been obtained from public sources. Any analysis, forecasts, calculations projections and opinions are based on such information and Cirium have not verified the accuracy and completeness of the information or data. In no event shall Cirium be liable for any indirect, consequential, special, or incidental damages which may arise from the use of the information, even if advised of such possibility. **Whilst we endeavor to keep the information up-to-date or correct, Cirium makes no representation or warranties of any kind, express or implied about the completeness, accuracy, reliability, suitability or availability with respect to the information or data or related graphics contained therein for any purpose. Any reliance you place on such material is therefore strictly at your own risk. Cirium disclaims all warranties, express or implied, including but not limited to warranties of performance, merchantability or fitness for a particular purpose.**

This document is subject to contract and all warranties whether express or implied by statute, law or otherwise are hereby disclaimed and excluded to the extent permitted at law. Any rights and obligations that may arise under this document shall be governed by the laws of England and Wales and English courts shall have non-exclusive jurisdiction to adjudicate any disputes arising hereunder.

If you have any queries with regards to above or anything contained on this document, then please reach out directly to your Cirium contact person or email Professional Data Services email: pds-us@cirium.com

**Cirium**
99 Bishopsgate
3rd Floor
London
EC1M 3AL
United Kingdom
Tel: +44 207 911 1400

# FIELD AND TABLE NAMES

The tables documented in the following section include the Redshift table/field names and the table/field names typically used in the historical data exports. Note that the analyst working on the export can name the tables and fields as they wish. The customer can finalize these data fields names before signing the contract.

This reference guide describes the following tables:

1. Flight History Table
2. Marketing Airline Table
3. Operating Airline Table
4. Equipment Table
5. Arrival Airport Table
6. Departure Airport Table
7. Fleet Aircraft Table
8. Fleet Seating Table
9. Metar by Flight Table

# DATA DICTIONARY – CIRIUM PROFESSIONAL DATA SERVICES

## Flight History Table (flight_history_prod)

This is the primary table of flight information with each row in the database representing a single, non-stop flight.

| Database Field | Description | Data Type |
|---|---|---|
| **flight_history_id** | Primary key for the flight record created when the flight record is first created. | Integer |
| **marketing_airline** | Marketing airline code, typically the IATA or ICAO code. Duplicate codes append an asterisk (*). | String |
| **marketing_flight_number** | Flight number of the marketing flight. | String |
| **operating_airline** | Operating airline code, typically the IATA or ICAO code. Duplicate codes append an asterisk (*). | String |
| **operating_flight_number** | Flight number of the operating flight which is often the same as the marketing flight number. | String |
| **departure_airport** | Departure airport code, typically IATA or ICAO code. | String |
| **arrival_airport** | Arrival airport code, typically IATA or ICAO code. | String |
| **departure_date** | The scheduled departure date. | Date |
| **diverted_airport** | Diverted airport code, typically IATA or ICAO code. Duplicate codes append an asterisk (*). | String |
| **arrival_date** | The scheduled arrival date. | Date |
| **departure_gate** | Departure gate assignment. | String |
| **departure_terminal** | Departure terminal assignment. | String |
| **arrival_gate** | Arrival gate assignment. | String |
| **arrival_terminal** | Arrival terminal assignment. | String |
| **baggage_claim** | Baggage claim assignment. | String |
| **aircraft_type** | The IATA equipment code. This field is used to join the equipment table. | String |
| **tail_number** | Tail number of the aircraft. | String |
| **seats** | The total number of seats (irrespective of cabin) | Integer |
| **Status** | The Flight History status codes which come directly from the Flight History table in production.<br>• A (Active)<br>• C (Cancelled)<br>• D (Diverted) | String |

| | | |
|---|---|---|
| | • L (Landed)<br>• NO (Not Operational)<br>• R (Redirected)<br>• S (Scheduled)<br>• U (Unknown) | |
| **scheduled_arrival** | Scheduled arrival time in local airport time. | Date/Time |
| **scheduled_departure** | Scheduled departure time in local airport time. | Date/Time |
| **scheduled_runway_departure** | Scheduled runway departure time in local airport time. | Date/Time |
| **scheduled_runway_arrival** | Scheduled runway arrival time in local airport time. | Date/Time |
| **scheduled_air_time** | Scheduled air time (wheels up to wheels down) in minutes. Null if not available. | Integer |
| **scheduled_block_time** | Scheduled block time (gate to gate) in minutes. Null if not available. | Integer |
| **actual_air_time** | Actual air time (wheels up to wheels down) in minutes. Null if not available. | Integer |
| **actual_block_time** | Actual block time (gate to gate) in minutes. Null if not available. | Integer |
| **taxi_in** | Arrival taxi time in minutes (Null if either gate or runway time is not available). | Integer |
| **taxi_out** | Departure taxi time in minutes (Null if either gate or runway time is not available). | Integer |
| **codeshare** | Codeshare flag. A 1 indicates that a flight is a duplicate codeshare flight. Typically, a filter is used to limit flight records to those where codeshare=0. | Integer |
| **wetlease** | Set to a 1 if there is a wetlease agreement | Integer |
| **actual_gate_departure** | Actual gate departure time. | Date/Time |
| **actual_runway_departure** | Actual runway departure time. | Date/Time |
| **actual_runway_arrival** | Actual runway arrival time. | Date/Time |
| **actual_gate_arrival** | Actual gate arrival time. | Date/Time |
| **estimated_gate_departure** | For flights that have not departed, this field may include an estimated gate departure time. | Date/Time |
| **estimated_runway_departure** | For flights that have not departed, this field may include an estimated runway departure time. | Date/Time |
| **estimated_runway_arrival** | For flights that have not arrived, this field may include an estimated gate arrival time. | Date/Time |
| **estimated_gate_arrival** | For flights that have not arrived, this field may include an estimated gate arrival time. | Date/Time |
| **utc_actual_gate_arrival** | Actual gate arrival time in UTC. | Date/Time |
| **utc_actual_gate_departure** | Actual gate departure time in UTC. | Date/Time |

| | | |
|---|---|---|
| **utc_actual_runway_arrival** | Actual runway arrival time in UTC. | Date/Time |
| **utc_actual_runway_departure** | Actual runway departure time in UTC. | Date/Time |
| **utc_arrival_date** | The arrival date referenced to UTC. | Date/Time |
| **utc_departure_date** | The departure date referenced to UTC. | Date/Time |
| **utc_scheduled_arrival** | The scheduled arrival time in UTC. | Date/Time |
| **utc_scheduled_departure** | The scheduled departure time in UTC. | Date/Time |
| **utc_scheduled_runway_departure** | The scheduled runway departure time in UTC. | Date/Time |
| **utc_scheduled_runway_arrival** | The scheduled runway arrival time in UTC. | Date/Time |
| **utc_status_active_at** | The time the flight status went active in UTC. | Date/Time |
| **utc_status_cancelled_at** | If a flight is cancelled, this is the time the cancel message was received in UTC. | Date/Time |
| **utc_status_diverted_at** | If a flight is diverted, this is the time the diverted message was received in UTC. | Date/Time |
| **utc_status_redirected_at** | If a flight is redirected, this is the time the redirect message was received in UTC. | Date/Time |
| **utc_status_landed_at** | The time the flight landed in UTC. | Date/Time |
| **arr_delay** | Arrival delay in minutes. This number can be negative for early arrivals. | Integer |
| **dep_delay** | Departure delay in minutes. This number can be negative for early departures. | Integer |
| **distance** | Distance in kilometers | Integer |
| **freighter** | Set to 1, If the flight is a cargo/freighter flight, 0 otherwise. | Integer |
| **published** | Set to 1, If the flight record was created from a published schedule, 0 otherwise. | Integer |
| **cancel** | A flag indicating when a confirmed cancellation was captured. | Flag (1 or 0) |
| **divert** | A flag indicating a diverted flight. Note that diverted flights are not considered "completed" and they may in fact have a delay associated with the departure, arrival, or both. | Flag (1 or 0) |
| **est_dep_delay** | Difference in estimated gate departure and scheduled gate departure in minutes. | Integer |
| **est_arr_delay** | Difference in estimated gate_arrival and scheduled gate_arrival in minutes. | Integer |
| **service_type** | The scheduled service type for a flight.<br>**Value** — **Description**<br>**J** — Scheduled Passenger (Normal Service) | String |

| | |
|---|---|
| **S** | Scheduled Passenger (Shuttle Service) |
| **U** | Scheduled Passenger (Service Vehicle) |
| **F** | Scheduled Cargo/Mail (Loose loaded cargo and/or preloaded devices) |
| **V** | Scheduled Cargo/Mail (Surface Vehicle) |
| **M** | Scheduled Cargo/Mail (Mail Only) |
| **Q** | Scheduled Passenger/Cargo in Cabin |
| **G** | Non-scheduled Passenger (Normal Service) |
| **B** | Non-scheduled Passenger (Shuttle Service) |
| **A** | Non-scheduled Cargo/Mail |
| **C** | Charter (Passenger Only) |
| **O** | Charter (Special Handling-Migrants/Immigrants) |
| **H** | Charter (Cargo and/or Mail) |
| **L** | Charter (Passenger and Cargo and/or Mail) |
| **P** | Non-revenue |
| **T** | Technical Test |
| **K** | Training |
| **D** | General Aviation |
| **E** | Special (FAA/Government) |
| **W** | Military |
| **R** | Additional Flights - Passenger/Cargo |
| **Y** | IATA Special Internal(Y) |
| **Z** | IATA Special Internal(Z) |

**irregular_operations**

The type of irregular operation.

| Value | Description |
|---|---|
| **CANCELLATION** | Flight has been cancelled for any reason. |
| **CONTINUATION_OF** | Identifies a linked flight that this flight is a continuation of. Typically, a continuation occurs when a flight is diverted to an alternate destination airport in order to get passengers to their scheduled destination. |

String

| | | |
|---|---|---|
| | **CONTINUED_BY** | Identifies a linked flight which will be a continuation of this one. Typically, a continuation occurs when a flight is diverted to an alternate destination airport in order to get passengers to their scheduled destination. |
| | **DIVERSION** | Flight is directed to land at a different airport than its scheduled destination. |
| | **FLOWN_OVER** | A fly-over occurs when a plane flying a route comprised of multiple stops skips one or more of the scheduled stops for some reason. The FLOWN_OVER type indicates that the flight in question is not expected to occur because the plane flying the route will not stop at the departure airport specified. |
| | **FLYOVER** | A fly-over occurs when a plane flying a route comprised of multiple stops skips one or more of the scheduled stops for some reason. The FLY_OVER type indicates that the flight in question will not stop at one of the scheduled destinations. |
| | **MISCELLANEOUS** | An irregular operation that does not qualify as one of the other identified types listed here. |
| | **REINSTATEMENT** | A flight was cancelled for a period of time and subsequently reinstated to operational status. |
| | **REPLACED_BY** | Any flight that is cancelled or does not operate for some reason may be replaced by another flight. Provides linking information for the flight that has replaced this one. |
| | **REPLACEMENT_FOR** | Identifies this as a flight that replaced some other flight. Provides linking information for the flight that was replaced by this one. |
| | **RETURN_TO_GATE** | Flight backed away from the gate and started to taxi, does not take off, and returns to gate. |
| | **RETURN_FROM_AIR BORNE** | Flight has taken off and is required to return to its original departure airport. |
| | **SUBSEQUENT_OPER ATION_BY** | Identifies the ID of a new flight operation record that is the continuation of the current flight record. For example, a flight with ID 100000000 was diverted, but later continues on to its destination with the ID of 10001010110. |
| | **SUBSEQUENT_OPER ATION_FOR** | Identifies the ID of a flight operation that is the origin of the current flight record. For example, a flight with ID 10001010110 was created based on the diversion of the flight with ID 100000000. |
| **has_irregular_operation** | A flag indicating if there was an irregular operation registered | Flag (1 or 0) |
| **est_taxi_in** | Estimated Arrival taxi time in minutes (Null if either gate or runway time is not available). | Integer |
| **est_taxi_out** | Estimated Departure taxi time in minutes (Null if either gate or runway time is not available). | Integer |

| | | |
|---|---|---|
| **icao_aircraft_type_actual** | ICAO code for the aircraft for the given flight. Same as “aircrafticao” in “fleet_aircraft” | String |
| **actual_aircraft_type** | IATA code for the aircraft for the given flight. Same as “aircraftiata” in “fleet_aircraft” | String |
| **fleet_aircraft_id** | Primary key used for joining the fleet table | Integer |
| **subsequent_flight_history_id** | Primary key for the flight record created when the flight is diverted. (maps to flight_history_id) | Integer |
| **subsequent_flight_type** | Describes the relationship of a flight history record to its subsequent_flight_history_id, if present.<br>SUBSEQUENT_OPERATION_BY – This is used to indicate that the subsequent operation was created when it was indicated via real-time flight status data after a flight returned to its origin by returning-to-gate or returning-from-air.<br>CONTINUED_BY – This is used to indicate that the subsequent operation was created when it was indicated via real-time flight status data after a flight diverted to an airport other than its origin.<br>REPLACED_BY – This is used to indicate that a cancelled flight was replaced by another operation based on real-time flight status data. Typically, this will be a flight departing from a different airport in the same metro area. | String |
| **upstream_flight_history_id** | Primary key for the flight record that resulted in a subsequent irregular operation event, which resulted in the creation of this flight record. (maps to flight_history_id) | Integer |
| **upstream_flight_type** | Describes the relationship of a flight history record to its upstream_flight_history_id, if present.<br>SUBSEQUENT_OPERATION_FOR – This is used to indicate that this flight history record was created when the upstream flight returned to its origin by returning-to-gate or returning-from-air and this subsequent operation was indicated via real-time flight status data.<br>CONTINUATION_OF – This is used to indicate that this flight history record was created when the upstream flight diverted to an airport other than its origin and this subsequent operation was indicated via real-time flight status data.<br>REPLACEMENT_FOR – This is used to indicate that this flight history record was created when its upstream flight was cancelled, and a subsequent operation was indicated to be departing from a different airport. Typically, this will be a flight departing from a different airport in the same metro area. | String |

## Marketing Airline Table

| Database Field | Description | Data Type |
|---|---|---|
| **airline_code** | The primary key for this table. Typically, this is used as a mechanism for joining to the flight table (typically, flight_history_prod) using the marketing_airline or operating_airline column. | String |
| **name** | The long name of the carrier. This string may include comas (","), therefore, it can cause some problems when exporting data in csv format. Typically, carrier_short_name shouid be used instead of this field. | String |
| **alliance** | The airline alliance of the airline (oneWorld, SkyTeam or Star Alliance). Null if the airline is not a part of an alliance. | String |
| **carrier_category** | Classifies airlines as follows:<br>A (Scheduled Passenger Carrier) | String |

| | | |
|---|---|---|
| | B (Non-scheduled Passenger Carrier)<br>C (Scheduled Cargo Carrier)<br>D (Non-Scheduled Cargo Carrier)<br>I (Scheduled Passenger / Cargo Carrier)<br>J (Non-scheduled Passenger / Cargo Carrier)<br>K (Railway Service) | |
| **carrier_iata_code** | IATA carrier code. | String |
| **carrier_icao_code** | ICAO carrier code. | String |
| **carrier_prefix_number** | Airwaybill (AWB) prefix number used for cargo tracking, ticketing and settlement. | String |
| **carrier_short_name** | The short name of the airline. Generally, this is the field that should be used for displaying an airline name. | String |
| **is_active** | A flag indicating that the airline is currently active and the code in use. | Flag (1 or 0) |
| **classification** | Classifies the airport in the range of 1 to 5, where:<br>1 - top 100 airports by size<br>2 - next 200 airports by size<br>3 - next 400 airports by size<br>4 - all remaining airports that have flights in our flight histories.<br>5 - all airports that don't have flights in our flight histories/permanently closed airports. | String |

## Operating Airline Table

| Database Field | Description | Data Type |
|---|---|---|
| **airline_code** | The primary key for this table. Typically, this is used as a mechanism for joining to the flight table (typically, flight_history_prod) using the marketing_airline or operating_airline column. | String |
| **name** | The long name of the carrier. This string may include comas (","), therefore, it can cause some problems when exporting data in csv format. Typically, carrier_short_name shouid be used instead of this field. | String |
| **alliance** | The airline alliance of the airline (oneWorld, SkyTeam or Star Alliance). Null if the airline is not a part of an alliance. | String |
| **carrier_category** | Classifies airlines as follows:<br>A (Scheduled Passenger Carrier)<br>B (Non-scheduled Passenger Carrier)<br>C (Scheduled Cargo Carrier)<br>D (Non-Scheduled Cargo Carrier)<br>I (Scheduled Passenger / Cargo Carrier) | String |

| | | |
|---|---|---|
| | J (Non-scheduled Passenger / Cargo Carrier)<br>K (Railway Service) | |
| **carrier_iata_code** | IATA carrier code. | String |
| **carrier_icao_code** | ICAO carrier code. | String |
| **carrier_prefix_number** | Airwaybill (AWB) prefix number used for cargo tracking, ticketing and settlement. | String |
| **carrier_short_name** | The short name of the airline. Generally, this is the field that should be used for displaying an airline name. | String |
| **is_active** | A flag indicating that the airline is currently active and the code in use. | Flag (1 or 0) |
| **classification** | Classifies the airport in the range of 1 to 5, where:<br>1 - top 100 airports by size<br>2 - next 200 airports by size<br>3 - next 400 airports by size<br>4 - all remaining airports that have flights in our flight histories.<br>5 - all airports that don't have flights in our flight histories/permanently closed airports. | String |

## Equipment Table

| Database Field | Description | Data Type |
|---|---|---|
| **aircraft_type_code** | The code for the specific configuration of the aircraft. For example, the aircraft_type_code 77W is a Boeing 777-300ER while 773 is a Boeing 777-300. | String |
| **aircraft_type_name** | The name associated with the aircraft_type_code. | String |
| **general_type_code** | The general type code of the aircraft. For example, general type code 777 is associated with 5 configurations that are described by the aircraft_type_code (77L, 77W, 772, 773, 777) | String |
| **is_wide_body** | A flag indicating the aircraft is a wide body aircraft. Set to 't' if true, 'f' if false. | Flag (t or f) |
| **propulsion** | The type of propulsion the aircraft employs. Possible values are:<br>Helicopter<br>Jet-engined aircraft<br>Piston or Turboprop engined aircraft<br>Surface equipment<br>Turboprop-engined aircraft | String |
| **avg_air_speed** | The average airspeed of the aircraft, in miles per hour. | Integer |

## Arrival Airport Table

| Database Field | Description | Data Type |
|---|---|---|
| **airport_code** | The primary key for this table. Typically, this is joined to the flight table (flight_history_prod) using the departure_airport or arrival_airport column. This code is based on either IATA code or ICAO code and sometimes has a trailing "*" that prevents duplicates. Default is to use the IATA code when it exists. | String |
| **airport_faa_code** | The FAA code for the airport, either a 3-letter code or 4-letter code. Sometimes the same as the IATA or ICAO codes but that's not always the case. | String |
| **airport_iata_code** | The 3-letter IATA code for the airport. Note: not all airports have an IATA code. | String |
| **airport_icao_code** | The 4-letter ICAO code for the airport. Note: not all airports have an ICAO code. | String |
| **city** | The city with which the airport is associated. | String |
| **city_code** | The 3-letter city code of the city with which the airport is associated. | String |
| **classification** | Used to classify airports according to size. This is a Cirium generated code that was originally developed to support the dynamic router and it might not have much use in data exports or visualizations. | String |
| **country_code** | The 2-letter ISO country code. | String |
| **country_name** | The full name of the country. | String |
| **district** | The district of the city in which the airport is located. | String |
| **fs_region** | The Region where the airport is located. Regions are defined by Cirium as one of the following:<br>Asia-Pacific<br>Europe<br>Latin America<br>Middle East & Africa<br>North America | String |
| **is_active** | A flag indicating whether the airport is currently operational. | Flag (1 or 0) |
| **is_major_airport** | A flag indicating the airport is considered a major airport. Note: this is an unreliable indicator. | Flag (1 or 0) |
| **latitude** | The latitude of the airport in decimal degrees. In Tableau, this should be converted to a dimension. | Fixed precision number (9.6) |
| **longitude** | The longitude of the airport in decimal degrees. In Tableau, this should be converted to a dimension. | Fixed precision number (9.6) |
| **market_code** | Code associated with the Metro area. For example, AMS market code is associated with the Amsterdam metro area and consists of two cities, Amsterdam and Rotterdam. | String |

| | | |
|---|---|---|
| **metro** | The name of the metro area (e.g. New York, Los Angeles, London, etc.). Grouping by metro area allows for the analysis of flights that depart or arrive at nearby airports in large metro areas served by multiple airports. | String |
| **name** | The full name of the airport, e.g. "Portland International Airport". | String |
| **postal_code** | The postal code of the airport. | String |
| **region** | The UN region where the airport is located. Regions are one of the following:<br>Africa<br>Antarctica<br>Asia<br>Europe<br>Latin America and the Caribbean<br>North America<br>Oceania | String |
| **state_code** | The state or province code. Null if the city does not associate with a state or province. | String |
| **street_1** | Street address of the airport, line 1. | String |
| **street_2** | Street address of the airport, line 2. | String |
| **sub_region** | Subregions are one of the following (grouped by UN Region):<br>Africa: Eastern Africa; Middle Africa; Northern Africa, Southern Africa, Western Africa<br>Antarctica: Antarctica<br>Asia: Central Asia; Eastern Asia, South-Eastern Asia; Southern Asia; Western Asia<br>Europe: Eastern Europe; Northern Europe; Southern Europe; Western Europe<br>Latin America and the Caribbean: Caribbean; Central America; South America<br>North America: North America<br>Oceania: Australia and New Zealand; Melanesia; Micronesia; Polynesia | String |
| **station_type** | Describes the type of airport:<br>A - Airport<br>B - Bus Station<br>C - Metropolitan Area<br>H - Heliport<br>L - Limousine Station<br>M - Military Base<br>O - Off-Line Point<br>P - Port | String |

| | | |
|---|---|---|
| | R – Railway Station<br>S - Sea Plane Base | |
| **weather_station_code** | The 4-letter weather station code. Often this field *looks* like an ICAO code, but may be a local code, (e.g. FAA code), prepended with an ICAO region prefix. This could also be an ICAO code that was *previously* assigned to the airport. | String |
| **weather_zone** | The 6-character NOAA weather zone (US only) in which the Airport is located. | String |
| **elevation** | The elevation of the airport, in feet. | Integer |
| **Timezone_region_name** | Time zone database name that the airport exists in. | String |

## Departure Airport Table

| Database Field | Description | Data Type |
|---|---|---|
| **airport_code** | The primary key for this table. Typically, this is joined to the flight table (flight_history_prod) using the departure_airport or arrival_airport column. This code is based on either IATA code or ICAO code and sometimes has a trailing "*" that prevents duplicates. Default is to use the IATA code when it exists. | String |
| **airport_faa_code** | The FAA code for the airport, either a 3-letter code or 4-letter code. Sometimes the same as the IATA or ICAO codes but that's not always the case. | String |
| **airport_iata_code** | The 3-letter IATA code for the airport. Note: not all airports have an IATA code. | String |
| **airport_icao_code** | The 4-letter ICAO code for the airport. Note: not all airports have an ICAO code. | String |
| **artcc_code** | Air Route Traffic Control Center codes. These are part of the US air traffic control system. | String |
| **city** | The city with which the airport is associated. | String |
| **city_code** | The 3-letter city code of the city with which the airport is associated. | String |
| **classification** | Used to classify airports according to size. This is a Cirium generated code that was originally developed to support the dynamic router and it might not have much use in data exports or visualizations. | String |
| **country_code** | The 2-letter ISO country code. | String |
| **country_name** | The full name of the country. | String |
| **customs_phone** | A comma separated list of one or more customs telephone numbers. | String |
| **district** | The district of the city in which the airport is located. | String |
| **fs_region** | The Region where the airport is located. Regions are defined by Cirium as one of the following:<br>Asia-Pacific<br>Europe<br>Latin America | String |

| | | |
|---|---|---|
| | Middle East & Africa<br>North America | |
| **is_active** | A flag indicating whether the airport is currently operational. | Flag (1 or 0) |
| **is_major_airport** | A flag indicating the airport is considered a major airport. Note: this is an unreliable indicator. | Flag (1 or 0) |
| **latitude** | The latitude of the airport in decimal degrees. In Tableau, this should be converted to a dimension. | Fixed precision number (9.6) |
| **longitude** | The longitude of the airport in decimal degrees. In Tableau, this should be converted to a dimension. | Fixed precision number (9.6) |
| **market_code** | Code associated with the Metro area. For example, AMS market code is associated with the Amsterdam metro area and consists of two cities, Amsterdam and Rotterdam. | String |
| **metro** | The name of the metro area (e.g. New York, Los Angeles, London, etc.). Grouping by metro area allows for the analysis of flights that depart or arrive at nearby airports in large metro areas served by multiple airports. | String |
| **name** | The full name of the airport, e.g. "Portland International Airport". | String |
| **postal_code** | The postal code of the airport. | String |
| **region** | The UN region where the airport is located. Regions are one of the following:<br>Africa<br>Antarctica<br>Asia<br>Europe<br>Latin America and the Caribbean<br>North America<br>Oceania | String |
| **state_code** | The state or province code. Null if the city does not associate with a state or province. | String |
| **street_1** | Street address of the airport, line 1. | String |
| **street_2** | Street address of the airport, line 2. | String |
| **sub_region** | Subregions are one of the following (grouped by UN Region):<br>Africa: Eastern Africa; Middle Africa; Northern Africa, Southern Africa, Western Africa<br>Antarctica: Antarctica<br>Asia: Central Asia; Eastern Asia, South-Eastern Asia; Southern Asia; Western Asia<br>Europe: Eastern Europe; Northern Europe; Southern Europe; Western Europe<br>Latin America and the Caribbean: Caribbean; Central America; South America<br>North America: North America<br>Oceania: Australia and New Zealand; Melanesia; Micronesia; Polynesia | String |

| | | |
|---|---|---|
| **weather_station_code** | The 4-letter weather station code. | String |
| **weather_zone** | The 6-character NOAA weather zone (US only) in which the Airport is located. | String |
| **elevation** | The elevation of the airport, in feet. | Integer |

## Fleet Aircraft Table

This table contains aircraft configuration data.

| Database Field | Description | Data Type |
|---|---|---|
| **aircraft_history_id** | Primary key for the fleet record, it gets created when the flight record is first created. It captures unique event on a particular aircraft id, hence one aircraft_id registers different aircraft_history_id. | Integer |
| **aircraft_id** | Primary key for an individual aircraft, primarily used internally for joining the tables. Aircraft id remains same for the particular aircraft, even in case of change of events/status. | Integer |
| **aircraft_status** | The aircraft's operational status,<br>Cancelled<br>In Service<br>LOI to Option<br>LOI to Order<br>On option<br>On order<br>Reengineered<br>Retired<br>Storage<br>Type swap<br>Unknown<br>Written off | String |
| **aircraft_configuration_id** | Aircraft can have different configurations during the course of their life. This field identifies the setup of an aircraft. It is used for joining other tables. | Integer |
| **build_year** | Year in which aircraft was built. | Integer |
| **first_flight_date** | Date when aircraft was first in service. | Date |
| **rolled_out_date** | Date when aircraft rolled out of factory door. | Date |
| **aircraft_type_internal_code** | Used internally but can be used for a client who still uses ASO case main, a legacy aircraft type identifier, originally from ASO Case Main | String |

| | | |
|---|---|---|
| **aircraft_variant_code** | Legacy field, describes different variant / subtypes of an aircraft | String |
| **aircraft_class** | Legacy taxonomy, now better used as market class and market sector.e.g. Business Jet, Turboprop, Turbine Helicopter | String |
| **aircraft_market_group** | <TUSUP> A more precise categorization of aircraft in each aircraft class, based on their size / role / number of engines and other relevant attributes. Each Market Grouping can belong to only one Aircraft Class. The categorization is on the sub-series level and each sub-series can only belong to one Market Grouping. (part of a 3-level hierarchy i.e. Market Sector, Market Class, and Market Grouping) Example, Civil Turbine Multi – Light, Regional Turboprop – Intermediate, Civil Turbine Multi - Medium | String |
| **aircraft_family** | e.g. Beech 1900, Falcon Family, L-1011 TriStar | String |
| **aircraft_iata** | A unique three-character aircraft designator | String |
| **aircraft_icao** | A unique four-character aircraft designator | String |
| **aircraft_type_name** | Aircraft manufacturers designation for the aircraft | String |
| **aircraft_type_master_series_name** | A high-level grouping of similar aircraft series for analytical purposes, based on common dimensions and to identify aircraft series that share airworthiness properties. A master series contains aircraft series from within one aircraft model, e.g. 747 -100, -200, - 300, -400, -8 and SP. | String |
| **aircraft_type_sub_series_id** | Primary key relating to aircraft type sub series name. | Integer |
| **aircraft_type_sub_series_name** | Lowest designation for the aircraft as used by the aircraft manufacturer, excluding customer variants. | String |
| **minor_variant** | A certain modifier has been added or for the manufacturers marketing | String |
| **aircraft_type_series_name** | An intermediate level between Master Series and Sub Series, including a suffix denoting freight, combi, convertible and range variants. | String |
| **aircraft_market_sector** | A current broad categorization of aircraft into five major categories: Commercial, Business, Helicopter, Utility Transport, and Military fixed wing. | String |
| **aircraft_market_class** | A current broad categorization of the aircraft type in accordance with its design role. | String |
| **aircraft_type_id** | Aircraft manufacturer's designation for the aircraft. | Integer |
| **manufacturer_id** | Unique identification number of aircraft manufacturer | Integer |
| **aircraft_manufacturer** | Name of the aircraft manufacturer responsible for the manufacture of the aircraft. Holds the aircraft type certificate., e.g. Airbus helicopters rather than Euro copter. | String |
| **engine_family_name** | The engine family installed on an aircraft. Engine family is a group of related models sharing airworthiness properties. Part of a 6-tier engine hierarchy: Engine Manufacturer, Engine Family, Engine Type, Engine Master Series, Engine Series, and Engine Sub Series. | String |
| **engine_type_name** | The engine type installed on an aircraft. Engine manufacturer's designation for an engine. Part of a 6-tier engine hierarchy | String |
| **engine_master_series** | <TUSUP> A high-level grouping of similar Engine series for analytical purposes and to identify Engine series that share airworthiness properties. A master series contains Engine series from within one Engine Type. (Part of a six-tier hierarchy: Engine Manufacturer, Engine Family, Engine Type, Engine Master Series, Engine Series, Engine Sub Series.) | String |

| | | |
|---|---|---|
| **engine_series** | <TUSUP> An intermediate level between Master Series and Sub-Series. (Part of a six-tier hierarchy: Engine Manufacturer, Engine Family, Engine Type, Engine Master Series, Engine Series, Engine Sub Series.) | String |
| **number_of_engines** | <TUSUP> Number of Engines. | Integer |
| **engine_type_subseries_name** | The engine sub series installed on an aircraft. Lowest designation for the engine as used by the engine manufacturer. Part of a 6-tier engine hierarchy | String |
| **engine_type_subseries_id** | Unique identifier for the engine sub series installed on an aircraft. | String |
| **engine_manufacturer** | Company responsible for (holds the Engine type certificate) the manufacture of the engines installed on an aircraft | String |
| **apu_type** | <TUSUP> APU Manufacturer's designation for an APU. | String |
| **apu_type_subseries_id** | Identifier for lowest designation for an APU as used by the APU Manufacturer. | Integer |
| **apu_manufactorer** | <TUSUP> Company responsible for the manufacture of the aircraft's APU, ie: company which hold the APU type certificate. | String |
| **certified_mtow_lbs** | <TUSUP>The Maximum Take-Off Weight (in lbs) which is owned for this aircraft and it is entitled to operate at. This may be different from the Operated MTOW | Integer |
| **operating_mtow_lbs** | <TUSUP> The Maximum Take-Off Weight (in lbs) the aircraft is operated at with current operator. This may be different from the Certified MTOW. | Integer |
| **max_landing_weight_lbs** | <TUSUP> Maximum Landing Weight (in lbs) at which the aircraft is certified to land. | String |
| **noise_category** | <TUSUP> Aircraft noise certification identifier e.g. Stage 3 etc. | String |
| **winglets_modifier** | <TUSUP> Indicates if winglets or other wingtip modifiers such as winglet-scimitars or sharklets have been retro-fitted to an aircraft. Does not indicate standard fitment winglets. | Integer |
| **registration_number** | Registration number of the aircraft as per the Certificate of Registration as registered with the governing body of the country where registration resides, e.g. FAA for the USA or CAA for the UK. | String |
| **serial_number** | Aircraft serial number (Constructor No), also known as MSN. | String |
| **hexcode** | Aircraft are assigned a unique ICAO 24-bit address "hex code" when the aircraft is registered, and this address becomes a part of the aircraft's Certificate of Registration. | String |
| **miscode** | Reported error in the Transponder code of an aircraft where the code on the aircraft has not been changed in line with the certified code and registration, generally the previous code is usually still keyed in, Transponder_Miscode indicates what is pinging from the aircraft, the Transponder_Code is what it should be. A Rare occurrence and it is only flagged where it is known to the research team it is not comprehensive. It is but significant when tracking aircraft as incorrect transponders can show one aircraft in 2 different places. | String |
| **event_start_date** | Start of particular historical transaction | Date |
| **event_end_date** | End of particular historical transaction | Date |
| **event_code** | Unique event code e.g. for 'Purchased' event code is I1 | String |
| **current_event** | Event name like 'Purchased', 'Leased In', 'Re-registered' | String |

| | | |
|---|---|---|
| **event_status** | Event status like ‘In Service’, ‘In storage’, ’Backlog’ | String |
| **operator_id** | Primary Key for an organization/operator. | Integer |
| **operator_classification** | <TUSUP>Describes the type of operator like, Military, Bank, Flying training school, Charter, Corporate/private etc. There are 65 such unique types of operator defined. | String |
| **opcode** | Legacy data field generally refers to ICAO code of operator. In case of lessors or financial companies where ICAO code is not present, it represents the company code of the operator. | String |
| **operator** | Operator of an aircraft. One of eight possible company roles, i.e. Owner, Operator, Manager, Sub Lessor, Operated For, Financed, Trust or Group. | String |
| **operator_iata** | IATA designated code for the Operator. | String |
| **operator_icao** | ICAO designated code for the Operator. | String |
| **operator_type** | Describes the type of operator like, Military, Bank, Flying training school, Charter, Corporate/private etc. There are 65 such unique types of operator defined. | String |
| **operator_country** | Geographic country of the main headquarters of the operator associated with an aircraft | String |
| **operator_region** | Geographic region of the main headquarters of the operator associated with an aircraft | String |
| **operated_for_id** | Primary Key for an organization/operator flying on behalf of the operator. | Integer |
| **operated_for** | An aircraft may be flown by an Operator (in the Operator for the aircraft) on behalf of another organization (in the Operated For role for the aircraft. For example, a private operator may fly an aircraft for a non-commercial organization such as a hospital, at today's date (as opposed to an event date). | String |
| **operated_for_iata** | IATA designated code for the organization/operator flying on behalf of the operator. | String |
| **operated_for_icao** | ICAO designated code for the organization/operator flying on behalf of the operator. | String |
| **primary_usage** | The main use of an aircraft e.g. Passenger, Freight, ASW etc | String |
| **secondary_usage** | The secondary or other use of the aircraft e.g. Passenger, Freight, ASW etc | String |
| **storage_location_name** | The geographic area of the world of the 'storage' location where the aircraft is kept or repaired | String |
| **fleet_number** | Aircraft Fleet Number (as designated by the current operator) | String |
| **last_update_time** | Timestamp when the record was last updated | Timestamp |
| **aircraft_value_subseries_id** | Primary key for aircraft_value_subseries name. | Integer |
| **aircraft_value_subseries_name** | The most granular level an aircraft family can be broken down to for valuation purposes. E.g. A320-200, A320neo, 737-800, 737 Max 8, 777-300ER, 777-200. Even though more granular levels exist (E.g. A320-232, A320-212 etc.), for valuation purposes these are rolled up into Aircraft Value Sub Series. | String |
| **lease_manager_id** | Primary key for the lease manager | Integer |

| | | |
|---|---|---|
| **lease_manager** | Company responsible for the operational management or the leasing arrangements of an aircraft. One of eight possible company roles, i.e. Owner, Operator, Manager, Sub Lessor, Operated For, Financed, Trust or Group. | String |
| **lease_manager_iata** | IATA designated code for the lease manager, mostly same as airline iata code | String |
| **lease_manager_icao** | ICAO designated code for the lease manager, mostly same as airline icao code | String |

## Fleet Seating Table

This table contains the cabin configuration specifications by aircraft.

| Database Field | Description | Data Type |
|---|---|---|
| **aircraft_id** | Primary key for an individual aircraft, primarily used internally for joining the tables. Aircraft id remains same for the particular aircraft, even in case of change of events/status. | Integer |
| **seat_class** | Class of the Aircraft Seat Configuration<br>First Class<br>Business Class<br>Premium Economy<br>VIP<br>Economy Class<br>Economy Sleeper Seat<br>Other/Utility<br>Unknown | String |
| **cabin_name** | This is airline specific cabin classification e.g. 'Delta comfort+', 'United Business First', 'Club ANA Asia' | String |
| **internet_model** | The product name of the in-flight internet hardware installed on the aircraft. e.g. 'eXConnect', '2Ku', 'JetWave MCS-8200' | String |
| **internet_oem** | The company which designs and manufactures in-flight internet hardware. e.g. 'Panasonic Avionics Corporation', 'Gogo', 'Viasat'. | String |
| **number_of_converted_seats** | The number of seats available after conversion from use in one cabin to another. | Integer |
| **number_of_convertible_seats** | The maximum number of seats which can be converted for use from one cabin class to another, typically Business and Economy. | Integer |
| **number_of_seats** | The total number of seats on the aircraft. It is recommended to use over seats in flight_history_prod table. | String |
| **number_of_seats_estimated** | Flag indicating that the datapoint is estimated | Boolean |
| **paid_connectivity** | Flag indicating that payment is required to access the internet connectivity service in that cabin class. | Boolean |
| **phone_model** | The product name of the in-flight mobile connectivity hardware installed. e.g. 'eXphone', 'Mobile OnAir' | String |
| **phone_oem** | The manufacturer of the in-flight mobile connectivity hardware installed, e.g. 'Panasonic', 'SITAOnAir' | String |

| | | |
|---|---|---|
| **ife_screen_size_in** | In Flight Entertainment Screen Size in Inches, measured diagonally. | Integer |
| **ife_model** | In Flight Entertainment Device model product name. e.g. ‘ex3’, ‘TopSeries AVANT’. | String |
| **ife_oem** | In Flight Entertainment hardware manufacturer. e.g. ‘Panasonic’, ‘Thales’, ‘Gogo’, ‘Bluebox’. | String |
| **seat_model** | Seat manufacturer's product name for the seats. | String |
| **seat_oem** | The manufacturer name who has produced the seats. | String |
| **seat_pitch_in** | The distance measured in inches between two rows within that cabin class. | Integer |
| **seat_recline_deg** | Seat Recline in Degrees | Integer |
| **seat_recline_in** | Seat Recline in Inches | Integer |
| **seats_abreast** | The number of seats arranged diagonally across an aircraft cabin. e.g. 1-2-1, 3-3, 3-4-3. | String |
| **seat_support_oem** | The company which maintains support for seats originally produced by a now defunct Seat OEM. | String |
| **seat_width_in** | Seat Width in Inches | Boolean |
| **inseat_power_outlet** | Seat has a power outlet for charging laptop, mobile etc. This can be either a USB port or a 3-pin plug. | Boolean |

## Metar by Flight Table

This table contains the weather information at the departure and arrival airports.

| Database Field | Description | Qualifier | Data Type |
|---|---|---|---|
| **flight_history_id** | Primary key for the flight record created when the flight record is first created. | | Integer |
| **departure_airport_code** | Departure airport FS code. | | String |
| **departure_weather_station_icao** | Pseudo-ICAO code of the weather station issuing the report. Joins to the dim_airports table on weather_station_code. | | String |
| **departure_report_time** | The timestamp of the METAR/SPECI closest to the flight’s actual runway departure time | | Timestamp |
| **departure_raw_report** | Raw METAR/SPECI observation closest to the flight’s actual runway departure time | | String |
| **departure_temperature_celsius** | Outside temperature at departure weather station in degree Celsius/centigrade | | Double |
| **departure_dew_point_celsius** | Outside dew point temperature in degrees Celsius | | Double |
| **departure_wind_direction** | Compass direction from which the wind was blowing at/near departure | 0 – 359 | Integer |
| **departure_wind_direction_min** | Most counterclockwise compass direction from which variable wind was blowing at/near departure | 0 – 359 | Integer |

| | | | |
|---|---|---|---|
| **departure_wind_direction_max** | Most clockwise compass direction from which variable wind was blowing at/near departure | 0 – 359 | Integer |
| **departure_wind_speed** | Wind speed in knots, at departure weather station | 0 - ~60 | Double |
| **departure_wind_variable** | Boolean value indicating whether wind direction is variable. | 0 or 1 | Integer |
| **departure_wind_gusting** | Boolean value indicating whether gust wind is variable. | 0 or 1 | Integer |
| **departure_gust_speed** | Wind gust speed in knots, at departure weather station | 0 -~60 | Double |
| **departure_kmhz_visibility** | Horizontal visibility in kilometers. Generally, the only reason for this field not to be present is if cavok is true, implying unimpeded visibility without a specific numeric measurement, at departure weather station | 0 or 1 | Double |
| **departure_cavok** | Boolean value indicating "Ceiling and Visibility OK" (abbreviated "CAVOK"). | | Integer |
| **departure_drizzle** | It is a type of precipitation called light rain. Based on Intensity METAR code -DZ for Light drizzle. DZ for Moderate drizzle and +DZ for Heavy drizzle.<br>The hazard rating here describes the severity of drizzle at the departure weather station, where 4 is worst. | 0 - 4 | Integer |
| **departure_drizzle_modifier** | Describes the condition of drizzle at departure weather station | PR Partial<br>BC Patches<br>DR Low Drifting<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_rain** | METAR code RA, it is a type of precipitation. The hazard rating here describes the severity of rain at the departure weather station, where 4 is worst. | 0 - 4 | Integer |
| **departure_rain_modifier** | Describes the condition of rain at departure weather station | PR Partial<br>BC Patches<br>DR Low Drifting<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_snow** | METAR code SN, it is a type of precipitation. The hazard rating here describes the severity of snow at the departure weather station, where 4 is worst. | 0 - 4 | Integer |

| | | | |
|---|---|---|---|
| **departure_snow_modifier** | Describes the condition of snow at departure weather station | DR Low Drifting<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_snow_grains** | Snow grains are characterized as very small (<1 mm), white, opaque grains of ice that are fairly flat or elongated. Unlike snow pellets, snow grains do not bounce or break up on impact. METAR code SG.<br>The rating here describes the severity of snow grains at the departure weather station, where 3 is worst. | 0-3 | Integer |
| **departure_snow_grains_modifier** | Describes the condition of snow grains at departure weather station | BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_ice_crystals** | Diamond dust is a ground-level cloud composed of tiny ice crystal. METAR code IC. The rating here describes the severity of ice crystals at the departure weather station, where 3 is worst. | 0-3 | Integer |
| **departure_ice_crystals_modifier** | Describes the condition of ice crystals at departure weather station | SH Shower(s) | String |
| **departure_ice_pellets** | Also called as freezing rain or sleet, is a type of precipitation where raindrops freeze before hitting the ground and they bounce back. METAR code PL. The rating here describes the severity of ice pellets at the departure weather station, where 3 is worst. | 0-3 | Integer |
| **departure_ice_pellets_modifier** | Describes the condition of ice pellets at departure weather station | SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_hail** | It is a form of solid precipitation; the size of hail is greater than 5mm (0.20in). METAR code GR. The rating here describes the intensity or severity of hail at the departure weather station, where 4 is worst. | 0-4 | Integer |
| **departure_hail_modifier** | Describes the condition of hail at departure weather station | SH Shower(s)<br>TS Thunderstorm | String |
| **departure_small_hail** | Also called as snow pellets, it is a form of solid precipitation, but the size of hail is less than 5mm(0.20in). METAR code GS. The rating here describes the intensity or severity of hail at the departure weather station, where 4 is worst. | 0-4 | Integer |
| **departure_small_hail_modifier** | Describes the condition of small hail at departure weather station | SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |

| | | | |
|---|---|---|---|
| **departure_unknown_precipitation** | This sensor attempts to differentiate between rain and snow and determines the intensity of the precipitation, abbreviated as UP (METAR code). The level of unknown precipitation lies in range 0-3, where 3 is severe. | 0-3 | Integer |
| **departure_unknown_precipitation_modifier** | Describes the condition of unknown precipitation at departure weather station | SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_mist** | Mist is defined as foggy conditions with visibilities greater than 5/8 statute mile. METAR code BR. The rating here describes the level of obscuration caused due to mist at the departure weather station, where 4 is worst. | 0-4 | Integer |
| **departure_mistmodifier** | Describes the condition of mist at departure weather station | PR Partial<br>BC Patches<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_fog** | Fog is defined with visibility 5/8 statute mile or less. METAR code FG. The rating here describes the level of obscuration caused due to fog at the departure weather station, where 4 is worst. | 0-4 | Integer |
| **departure_fog_modifier** | Describes the condition of fog at departure weather station | PR Partial<br>BC Patches<br>DR Low Drifting<br>BL Blowing<br>TS Thunderstorm<br>FZ Freezing | String |
| **departure_smoke** | The foggy effect accompanied with burning smell, generally caused by forest fire or any other large blaze. These burned particles travel 25-100 miles and produce a foggy effect called smoke. METAR code is FU. The rating here describes the level of obscuration caused due to mist at the departure weather station, where 4 is worst. | 0-4 | Integer |
| **departure_smoke_modifier** | Describes the condition of smoke at departure weather station | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |

| | | | |
|---|---|---|---|
| **departure_volcanic_ash** | Volcanic Ash is defined as very small solid particles ejected from a volcano during an eruption which have intermediate axes measuring 2 mm or less. METAR code is VA. The rating here describes the level of obscuration caused due to volcanic ash at the departure weather station, where 4 is worst. | 0-3 | Integer |
| **departure_volcanic_ash_modifier** | Describes the condition of volcanic ash at departure weather station | BL | String |
| **departure_widespread_dust** | The obscuration due to dust spread in larger areas. The rating here describes the level of obscuration caused due to extensive dust at the departure weather station, where 4 is worst. METAR code DU | 0-4 | Integer |
| **departure_widespread_dust_modifier** | Describes the condition of widespread dust at departure weather station. | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **departure_sand** | The obscuration caused due to sand particles in air, generally in desert areas. The rating here describes the level of obscuration caused due to extensive sand at the departure weather station, where 4 is worst. METAR code SA. | 0-3 | Integer |
| **departure_sand_modifier** | Describes the condition of sand at departure weather station. | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **departure_haze** | It is the fog or mist that's caused by small particles in the air. The rating here describes the level of obscuration caused due to haze at the departure weather station, where 4 is probable sandstorm. METAR code HZ. | 0-3 | Integer |
| **departure_haze_modifier** | Describes the condition of sand at departure weather station. | BL Blowing<br>TS Thunderstorm | String |
| **departure_spray** | METAR code PY. The rating here describes the level of obscuration caused due to ocean/lake spray at the departure weather station. | 0 OR 1 | Integer |
| **departure_spray_modifier** | Describes the condition of spray at departure weather station. | BL Blowing | String |
| **departure_dust_sand_whirls** | Well-developed Dust/Sand Whirl, METAR code is PO. | 0-4 | Integer |
| **departure_dust_sand_whirls_modifier** | Describes the condition of spray at departure weather station. | | String |
| **departure_squalls** | A squall is a sudden, sharp increase in wind speed lasting minutes, contrary to a wind gust lasting second. METAR code is SQ. The rating describes the severity of squalls at the departure weather station, 3 being the worst | 0-3 | Integer |

| | | | |
|---|---|---|---|
| **departure_squalls_modifier** | Describes the condition of spray at departure weather station | DR Low Drifting<br>SH Shower(s)<br>TS Thunderstorm | String |
| **departure_funnel_cloud** | Funnel-shaped cloud of condensed water droplets. METAR code is FC. The rating describes the severity of funnel cloud at the departure weather station, 3 being the worst | 0-3 | Integer |
| **departure_funnel_cloud_modifier** | Describes the condition of funnel cloud at departure weather station | MI Shallow<br>BC Patches<br>TS Thunderstorm | String |
| **departure_sandstorm** | METAR code SS. The rating describes the severity of sandstorm at the departure weather station. | 0-3 | Integer |
| **departure_sandstorm_modifier** | Describes the condition of sandstorm at departure weather station | BL Blowing | |
| **departure_duststorm** | METAR code SS. The rating describes the severity of sandstorm at the departure weather station. | 0-3 | Integer |
| **departure_duststorm_modifier** | Describes the condition of duststorm at departure weather station | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **departure_thunder_storms** | Flag identifying thunderstorm at the departure weather station | 0 or 1 | Integer |
| **departure_base_cloud_type** | Fraction of the sky covered by the lowest cloud base at the departure weather station. | SKC – Sky Clear (human-reported)<br>CLR – Clear; No clouds below 12k ft<br>FEW – Few Clouds Covering 0/8ths – 2/8ths of the sky<br>SCT – Scattered Clouds 3/8 – 4/8<br>BKN – Broken Clouds 5/8 – 7/8<br>OVC - Overcast<br>VV – Extremely Poor Vertical Visibility due to fog or precipitation<br>NSC – No Significant Clouds | String |

| | | | |
|---|---|---|---|
| **departure_base_cloud_level** | Cloud layer base (feet above ground level). | 0 - 96500 | Integer |
| **departure_lowest_broken_or_overcast_cloud_base** | Altitude of lowest BKN or OVC cloud base | 0 – 98900 | Integer |
| **departure_cumulonimbus** | Flag indicating that Cumulonimbus clouds (CB) | 0 or 1 | Integer |
| **departure_towering_cumulus** | Flag indicating Towering Cumulus clouds (TCU) | 0 or 1 | Integer |
| **departure_has_runway_visibility** | Flag for runway visibility | 0 or 1 | Integer |
| **departure_lowest_rvr_visibility** | Lowest runway visual range at departure | 0 - 90070 | Integer |
| **departure_greatest_rvr_visibility** | Greatest runway visual range at departure | 0 - 9200 | Integer |
| **departure_instrumentation** | Flight rules/regulations/procedures required to operate given the weather conditions at departure. | VFR – Visual Flight Rules<br>MVFR – Marginal Visual Flight Rules<br>IFR – Instrument Flight Rules<br>LIFR – Low Instrument Flight Rules | String |
| **departure_wxstring** | Raw coded weather conditions. This is a string that starts with an optional one-character intensity flag, where “-“ is “light,” no flag is “moderate,” and “+” is “heavy.” The remainder of the string consists of 2-character weather abbreviations that include descriptors, precipitation types, obscurations, and other weather phenomena. | 322 distinct values like SNDZ<br>-RAHZ<br>RAPL<br>SHGSSN<br>+SHGS<br>-DZFG<br>RAGR<br>RADZ etc. | String |
| **departure_level2cloudtype** | Cloud coverage at the second-highest cloud base from the ground. | SCT<br>FEW<br>OVC<br>BKN<br>SKC<br>VV | String |
| **departure_level2cloudlevel** | Height of the second-highest cloud base in feet above ground level | 100-92900 | Integer |

| | | | |
|---|---|---|---|
| **departure_level3cloudtype** | Cloud coverage at the third-highest cloud base from the ground. | FEW<br>BKN<br>VV<br>OVC<br>SCT | String |
| **departure_level3cloudlevel** | Height of the third-highest cloud base from the ground | 200-95000 | Integer |
| **arrival_airport_code** | Arrival airport FS code. | | String |
| **arrival_weatherstationicao** | Pseudo-ICAO code of the weather station issuing the report. Joins to the dim_airports table on weather_station_code. | | String |
| **arrival_reporttime** | The timestamp of the METAR/SPECI closest to the flight's actual runway arrival time | | Timestamp |
| **arrival_rawreport** | Raw METAR/SPECI observation closest to the flight's actual runway arrival time | | String |
| **arrival_temperaturecelsius** | Outside temperature at arrival weather station in degree Celsius/centigrade | | Double |
| **arrival_dewpointcelsius** | Outside dew point temperature in degrees Celsius | | Double |
| **arrival_winddirection** | Compass direction from which the wind was blowing at/near arrival | 0 – 359 | Integer |
| **arrival_winddirectionmin** | Most counterclockwise compass direction from which variable wind was blowing at/near arrival | 0 – 359 | Integer |
| **arrival_winddirectionmax** | Most clockwise compass direction from which variable wind was blowing at/near arrival | 0 – 359 | Integer |
| **arrival_windspeed** | Wind speed in knots, at arrival weather station | 0 - ~60 | Double |
| **arrival_windvariable** | Boolean value indicating whether wind direction is variable. | 0 or 1 | Integer |
| **arrival_windgusting** | Boolean value indicating whether gust wind is variable. | 0 or 1 | Integer |
| **arrival_gustspeed** | Wind gust speed in knots, at arrival weather station | 0 -~60 | Double |
| **arrival_kmhzvisibility** | Horizontal visibility in kilometers. Generally, the only reason for this field not to be present is if cavok is true, implying unimpeded visibility without a specific numeric measurement, at arrival weather station | 0 or 1 | Double |
| **arrival_cavok** | Boolean value indicating "Ceiling and Visibility OK" (abbreviated "CAVOK"). | | Integer |
| **arrival_drizzle** | It is a type of precipitation called light rain. Based on Intensity METAR code -DZ for Light drizzle. DZ for Moderate drizzle and +DZ for Heavy drizzle.<br>The hazard rating here describes the severity of drizzle at the arrival weather station, where 4 is worst. | 0 - 4 | Integer |
| **arrival_drizzlemodifier** | Describes the condition of drizzle at arrival weather station | PR Partial<br>BC Patches | String |

| | | | |
|---|---|---|---|
| | | DR Low Drifting<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | |
| **arrival_rain** | METAR code RA, it is a type of precipitation. The hazard rating here describes the severity of rain at the arrival weather station, where 4 is worst. | 0 - 4 | Integer |
| **arrival_rainmodifier** | Describes the condition of rain at arrival weather station | PR Partial<br>BC Patches<br>DR Low Drifting<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_snow** | METAR code SN, it is a type of precipitation. The hazard rating here describes the severity of snow at the arrival weather station, where 4 is worst. | 0 - 4 | Integer |
| **arrival_snowmodifier** | Describes the condition of snow at arrival weather station | DR Low Drifting<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_snowgrains** | Snow grains are characterized as very small (<1 mm), white, opaque grains of ice that are fairly flat or elongated. Unlike snow pellets, snow grains do not bounce or break up on impact. METAR code SG.<br>The rating here describes the severity of snow grains at the arrival weather station, where 3 is worst. | 0-3 | Integer |
| **arrival_snowgrainsmodifier** | Describes the condition of snow grains at arrival weather station | BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_icecrystals** | Diamond dust is a ground-level cloud composed of tiny ice crystal. METAR code IC. The rating here describes the severity of ice crystals at the arrival weather station, where 3 is worst. | 0-3 | Integer |
| **arrival_icecrystalsmodifier** | Describes the condition of ice crystals at arrival weather station | SH Shower(s) | String |

| | | | |
|---|---|---|---|
| **arrival_icepellets** | Also called as freezing rain or sleet, is a type of precipitation where raindrops freeze before hitting the ground and they bounce back. METAR code PL. The rating here describes the severity of ice pellets at the arrival weather station, where 3 is worst. | 0-3 | Integer |
| **arrival_icepelletsmodifier** | Describes the condition of ice pellets at arrival weather station | SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_hail** | It is a form of solid precipitation; the size of hail is greater than 5mm (0.20in). METAR code GR. The rating here describes the intensity or severity of hail at the arrival weather station, where 4 is worst. | 0-4 | Integer |
| **arrival_hailmodifier** | Describes the condition of hail at arrival weather station | SH Shower(s)<br>TS Thunderstorm | String |
| **arrival_smallhail** | Also called as snow pellets, it is a form of solid precipitation, but the size of hail is less than 5mm(0.20in). METAR code GS. The rating here describes the intensity or severity of hail at the arrival weather station, where 4 is worst. | 0-4 | Integer |
| **arrival_smallhailmodifier** | Describes the condition of small hail at arrival weather station | SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_unknownprecipitation** | This sensor attempts to differentiate between rain and snow and determines the intensity of the precipitation, abbreviated as UP (METAR code). The level of unknown precipitation lies in range 0-3, where 3 is severe. | 0-3 | Integer |
| **arrival_unknownprecipitationmodifier** | Describes the condition of unknown precipitation at arrival weather station | SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_mist** | Mist is defined as foggy conditions with visibilities greater than 5/8 statute mile. METAR code BR. The rating here describes the level of obscuration caused due to mist at the arrival weather station, where 4 is worst. | 0-4 | Integer |
| **arrival_mistmodifier** | Describes the condition of mist at arrival weather station | PR Partial<br>BC Patches<br>BL Blowing<br>SH Shower(s)<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_fog** | Fog is defined with visibility 5/8 statute mile or less. METAR code FG. The rating here describes the level of obscuration caused due to fog at the arrival weather station, where 4 is worst. | 0-4 | Integer |

| | | | |
|---|---|---|---|
| **arrival_fogmodifier** | Describes the condition of fog at arrival weather station | PR Partial<br>BC Patches<br>DR Low Drifting<br>BL Blowing<br>TS Thunderstorm<br>FZ Freezing | String |
| **arrival_smoke** | The foggy effect accompanied with burning smell, generally caused by forest fire or any other large blaze. These burned particles travel 25-100 miles and produce a foggy effect called smoke. METAR code is FU. The rating here describes the level of obscuration caused due to mist at the arrival weather station, where 4 is worst. | 0-4 | Integer |
| **arrival_smokemodifier** | Describes the condition of smoke at arrival weather station | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **arrival_volcanicash** | Volcanic Ash is defined as very small solid particles ejected from a volcano during an eruption which have intermediate axes measuring 2 mm or less. METAR code is VA. The rating here describes the level of obscuration caused due to volcanic ash at the arrival weather station, where 4 is worst. | 0-3 | Integer |
| **arrival_volcanicashmodifier** | Describes the condition of volcanic ash at arrival weather station | BL | String |
| **arrival_widespreaddust** | The obscuration due to dust spread in larger areas. The rating here describes the level of obscuration caused due to extensive dust at the arrival weather station, where 4 is worst. METAR code DU | 0-4 | Integer |
| **arrival_widespreaddustmodifier** | Describes the condition of widespread dust at arrival weather station. | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **arrival_sand** | The obscuration caused due to sand particles in air, generally in desert areas. The rating here describes the level of obscuration caused due to extensive sand at the arrival weather station, where 4 is worst. METAR code SA. | 0-3 | Integer |
| **arrival_sandmodifier** | Describes the condition of sand at arrival weather station. | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **arrival_haze** | It is the fog or mist that's caused by small particles in the air. The rating here describes the level of obscuration caused due to haze at the arrival weather station, where 4 is probable sandstorm. METAR code HZ. | 0-3 | Integer |
| **arrival_hazemodifier** | Describes the condition of sand at arrival weather station. | BL Blowing<br>TS Thunderstorm | String |

| | | | |
|---|---|---|---|
| **arrival_spray** | METAR code PY. The rating here describes the level of obscuration caused due to ocean/lake spray at the arrival weather station. | 0 OR 1 | Integer |
| **arrival_spraymodifier** | Describes the condition of spray at arrival weather station. | BL Blowing | String |
| **arrival_dustsandwhirls** | Well-developed Dust/Sand Whirl, METAR code is PO. | 0-4 | Integer |
| **arrival_dustsandwhirlsmodifier** | Describes the condition of spray at arrival weather station. | | String |
| **arrival_squalls** | A squall is a sudden, sharp increase in wind speed lasting minutes, contrary to a wind gust lasting seconds. METAR code is SQ. The rating describes the severity of squalls at the arrival weather station, 3 being the worst | 0-3 | Integer |
| **arrival_squallsmodifier** | Describes the condition of spray at arrival weather station | DR Low Drifting<br>SH Shower(s)<br>TS Thunderstorm | String |
| **arrival_funnelcloud** | Funnel-shaped cloud of condensed water droplets. METAR code is FC. The rating describes the severity of funnel cloud at the arrival weather station, 3 being the worst | 0-3 | Integer |
| **arrival_funnelcloudmodifier** | Describes the condition of funnel cloud at arrival weather station | MI Shallow<br>BC Patches<br>TS Thunderstorm | String |
| **arrival_sandstorm** | METAR code SS. The rating describes the severity of sandstorm at the arrival weather station. | 0-3 | Integer |
| **arrival_sandstormmodifier** | Describes the condition of sandstorm at arrival weather station | BL Blowing | String |
| **arrival_duststorm** | METAR code SS. The rating describes the severity of sandstorm at the arrival weather station. | 0-3 | Integer |
| **arrival_duststormmodifier** | Describes the condition of duststorm at arrival weather station | DR Low Drifting<br>BL Blowing<br>TS Thunderstorm | String |
| **arrival_thunderstorms** | Flag identifying thunderstorm at the arrival weather station | 0 or 1 | Integer |
| **arrival_basecloudtype** | Fraction of the sky covered by the lowest cloud base at the arrival weather station. | SKC – Sky Clear (human-reported)<br>CLR – Clear; No clouds below 12k ft<br>FEW – Few Clouds Covering 0/8ths – 2/8ths of the sky | String |

| | | | |
|---|---|---|---|
| | | SCT – Scattered Clouds 3/8 – 4/8<br>BKN – Broken Clouds 5/8 – 7/8<br>OVC - Overcast<br>VV – Extremely Poor Vertical Visibility due to fog or precipitation<br>NSC – No Significant Clouds | |
| **arrival_basecloudlevel** | Cloud layer base (feet above ground level). | 0 - 96500 | Integer |
| **arrival_lowestbrokenorovercastcloudbase** | Altitude of lowest BKN or OVC cloud base | 0 – 98900 | Integer |
| **arrival_cumulonimbus** | Flag indicating that Cumulonimbus clouds (CB) | 0 or 1 | Integer |
| **arrival_toweringcumulus** | Flag indicating Towering Cumulus clouds (TCU) | 0 or 1 | Integer |
| **arrival_hasrunwayvisibility** | Flag for runway visibility | 0 or 1 | Integer |
| **arrival_lowestrvrvisibility** | Lowest runway visual range at arrival | 0 - 90070 | Integer |
| **arrival_greatestrvrvisibility** | Greatest runway visual range at arrival | 0 - 9200 | Integer |
| **arrival_instrumentation** | Flight rules/regulations/procedures required to operate given the weather conditions at arrival. | VFR – Visual Flight Rules<br>MVFR – Marginal Visual Flight Rules<br>IFR – Instrument Flight Rules<br>LIFR – Low Instrument Flight Rules | |
| **arrival_wxstring** | Raw coded weather conditions. This is a string that starts with an optional one-character intensity flag, where “-“ is “light,” no flag is “moderate,” and “+” is “heavy.” The remainder of the string consists of 2-character weather abbreviations that include descriptors, precipitation types, obscurations, and other weather phenomena. | 308 distinct values like SNDZ<br>-RAHZ<br>RAPL<br>SHGSSN<br>+SHGS<br>-DZFG<br>RAGR | |

| | | |
|---|---|---|
| | | RADZ etc. |
| **arrival_level2cloudtype** | Cloud coverage at the second-highest cloud base from the ground. | BKN<br>SKC<br>OVC<br>FEW<br>SCT<br>VV |
| **arrival_level2cloudlevel** | Height of the second-highest cloud base in feet above ground level | 100- 93700 |
| **arrival_level3cloudtype** | Cloud coverage at the third-highest cloud base from the ground. | BKN<br>FEW<br>OVC<br>SCT<br>VV |
| **arrival_level3cloudlevel** | Height of the third-highest cloud base from the ground | 200-90000 |