DATA WAREHOUSE PROJECT

Air traffic and CO2 emissions

from April to August 2022

Calugaru Maria Diana (1893272)
Mastrandrea Lorenzo (1892793)

What we have done ...

- Data gathering
- ETL process for cleansing and filtering the original data
- Conceptual modeling: DFM schema
- Logical modeling: SNOWFLAKE schema
- ROLAP queries on SQL
- Analysis with Tableau

The premise of our project is that the original file containg the flights data is incomplete (missing flights). Therefore the analysis we carried should not be taken as accurate ...

Data Gathering

5 different operational data sources taken from Kaggle:

airlines → informations about airlines like name IATA code*, ICAO code*, alias, callsign, country and whether the airline is still active.

airports → informations about airports like name IATA code*, ICAO code*, city, country, latitude, longitude, altitude, timezone, DST (daylight saving time).

aircraft → informations about aircrafts like name IATA code*, ICAO code*.

flights → informations about flights like departure and arrival airports, departure and arrival time, duration, flight number, airline, aircraft, price (USD), co2 emission, average co2 emission for that specific route .

fleets → informations about airlines and their current, past and future aircrafts together with the associated cost.

* **IATA** and **ICAO** codes are standard identifiers given by the **International Air Transport Association** and by **the International Civil Aviation Organisation**

ETL processes

Cleansing and integration phases of the original data were made through Python scripts. Below there are the detailed operations perfomed on each of the files.

airlines

- Filtering the airlines keeping only the ones with the *active* set to *Y (YES)*.
- Removed columns *airline_id, alias, callsign, active.*
- The not-defined *Icao_code, iata_code* and *country* replaced by *null* values.
- *Name* was set to lower case (useful for future joins).

The resulting file has the following shape:

Name	*IATA code*	*ICAO code*	*Country*

ETL processes

Cleansing and integration phases of the original data were made through Python scripts.
Below there are the detailed operations perfomed on each of the files.

airports

- *Airport_Id, DST(Daylight saving time), Type, Source* columns removed.
- Missing *iata_code* and *icao_code* (marked with \N) were replaced by *null* values.
- The *Latitude, Longitude* and *Altitude* casted to Float.
- The *Timezone* casted to Int.

The resulting file has the following shape:

Name	*City*	*Country*	*IATA code*	*ICAO code*	*Latitude*	*Longitude*	*Altitude*	*Timezone*	*TZ*

ETL processes

Cleansing and integration phases of the original data were made through Python scripts. Below there are the detailed operations perfomed on each of the files.

aircrafts

- *Index* column removed.
- Missing *iata_code* and *icao_code* (marked with *N*) were replaced by null values.

The resulting file has the following shape:

Name	*IATA code*	*ICAO code*

ETL processes

Cleansing and integration phases of the original data were made through Python scripts.
Below there are the detailed operations perfomed on each of the files.

flights

- Filtering the flights keeping only those with *stops* set to 0.
- Squared brakets removed from *airline name*, then converted to lower case and eventually made coherent with the values present in the *airlines* file.
- *Stops, currency, scan date* columns removed.
- *%* symbol removed from *co2 percentage*.
- *Duration, co2 emissions, avg co2 emission for this route, co2 percentage* casted to int.
- *Price* casted to float.
- *Departure time* and *arrival time* are splitted in *time, day, month, year* columns.

The resulting file has the following shape:

From airport code	*From country*	*Dest Airport Code*	*Dest country*	*Aircraft type*	*Airline number*	*Airline name*	*Flight Number*	*Departure date*	*Departure month*	*Departure year*

Departure time	*Arrival day*	*Arrival month*	*Arrival year*	*Arrival time*	*Duration*	*Price*	*Co2 emissions*	*Avg co2 emissions for this route*	*Co2 percentage*

ETL processes

Cleansing and integration phases of the original data were made through Python scripts.
Below there are the detailed operations perfomed on each of the files.

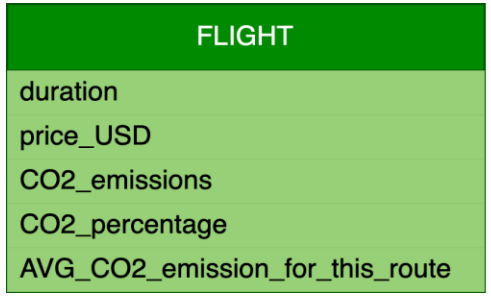
fleets

- *Parent airline* and *airline* were set to lower case.
- *Orders* columns removed.
- *Current, future, historic* and *total* were casted to int.
- *%* symbol removed from *Unit cost* and *Total cost* then multiplied per 1.000.000 (the original cost was expressed in millions of dollars) then casted to float.

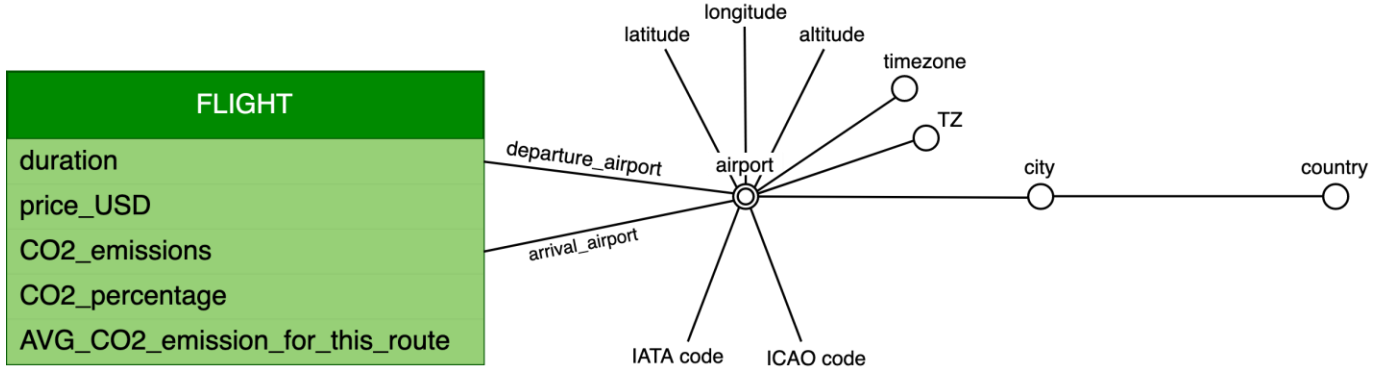
The resulting file has the following shape:

Perent airline	*Airline*	*Aircraft type*	*Current*	*Future*	*Historic*	*Total*	*Unit cost*	*Total cost*	*Average age*

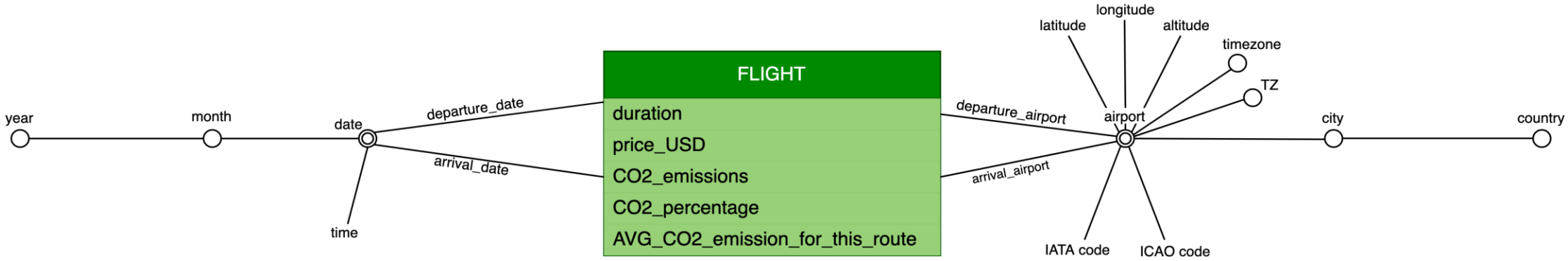
Conceptual modeling: DFM schema



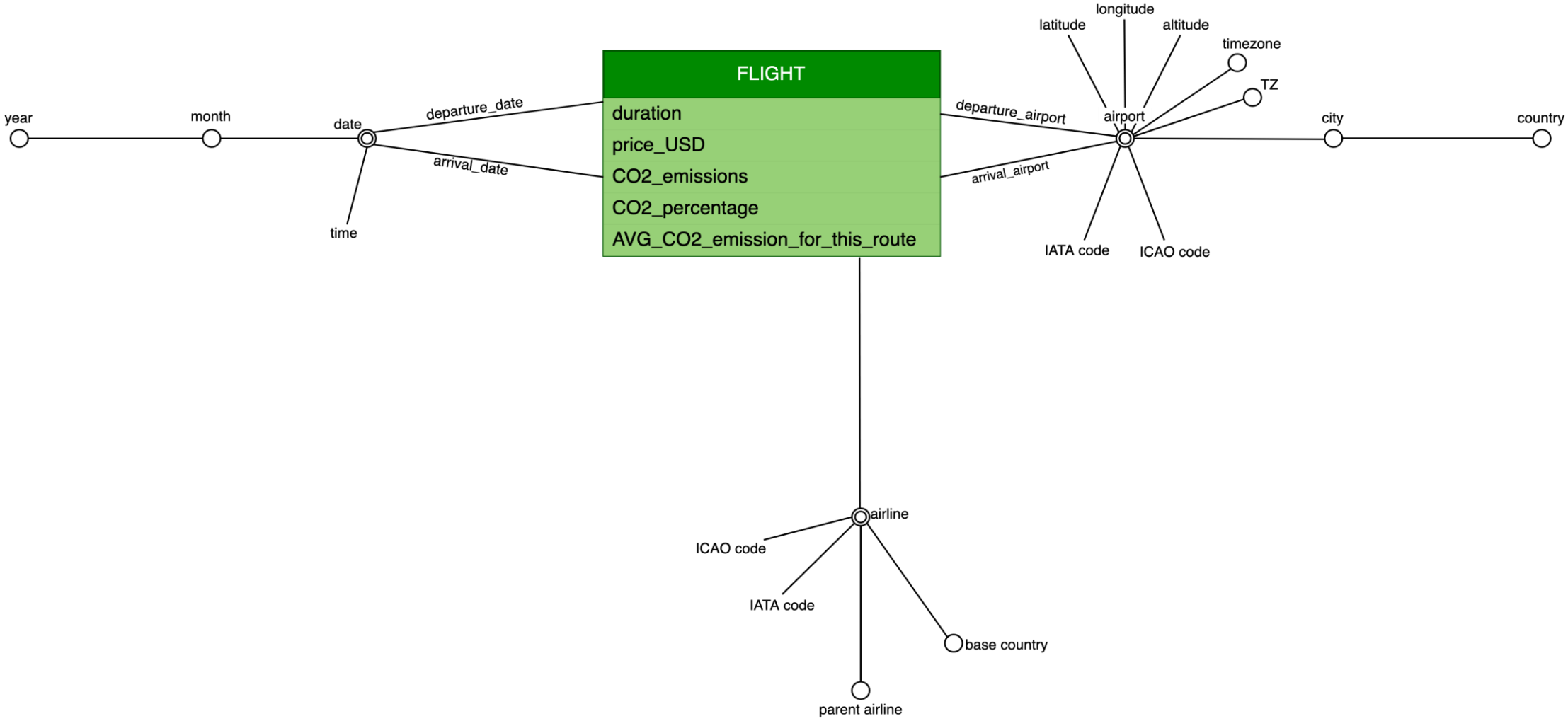
Conceptual modeling: DFM schema



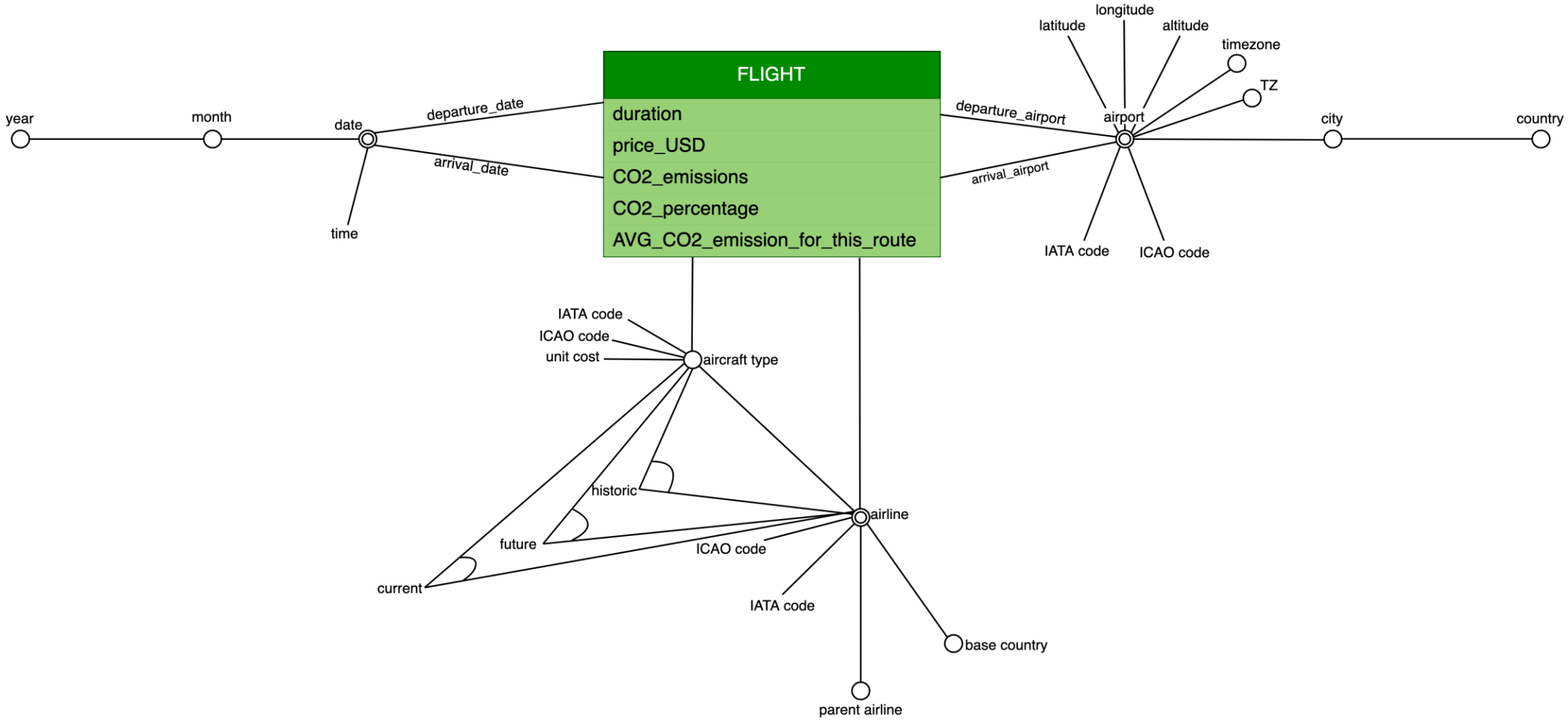
Conceptual modeling: DFM schema



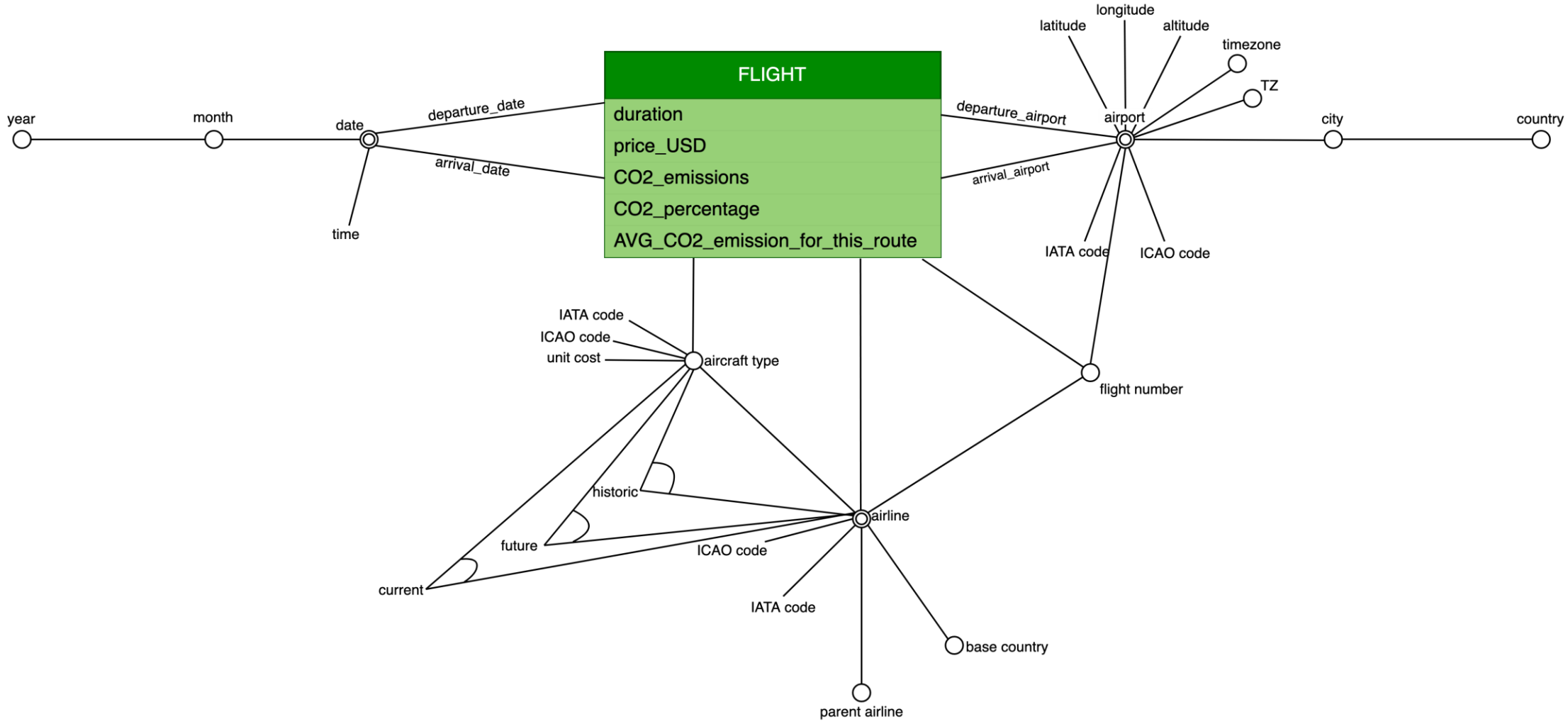
Conceptual modeling: DFM schema



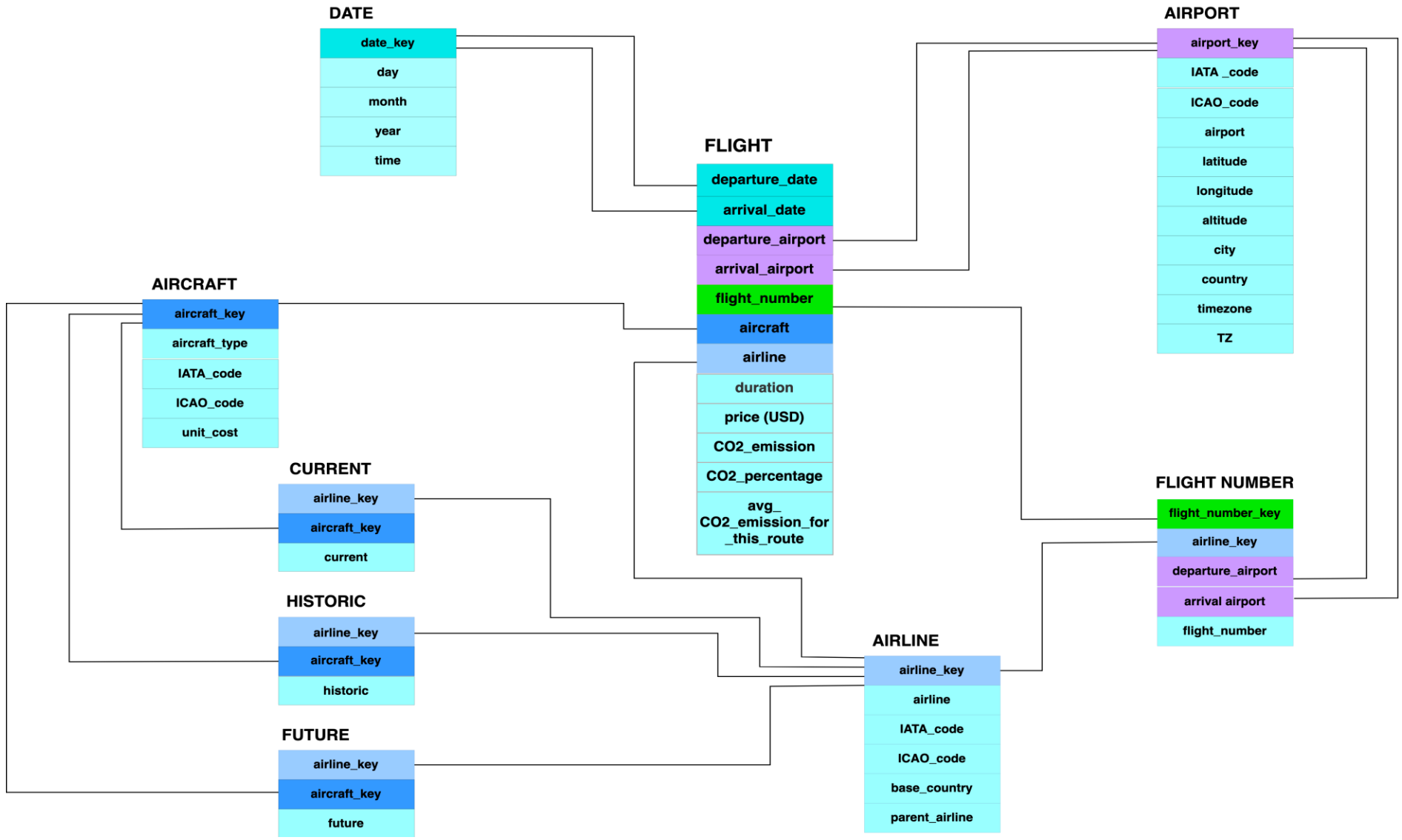
Conceptual modeling: DFM schema



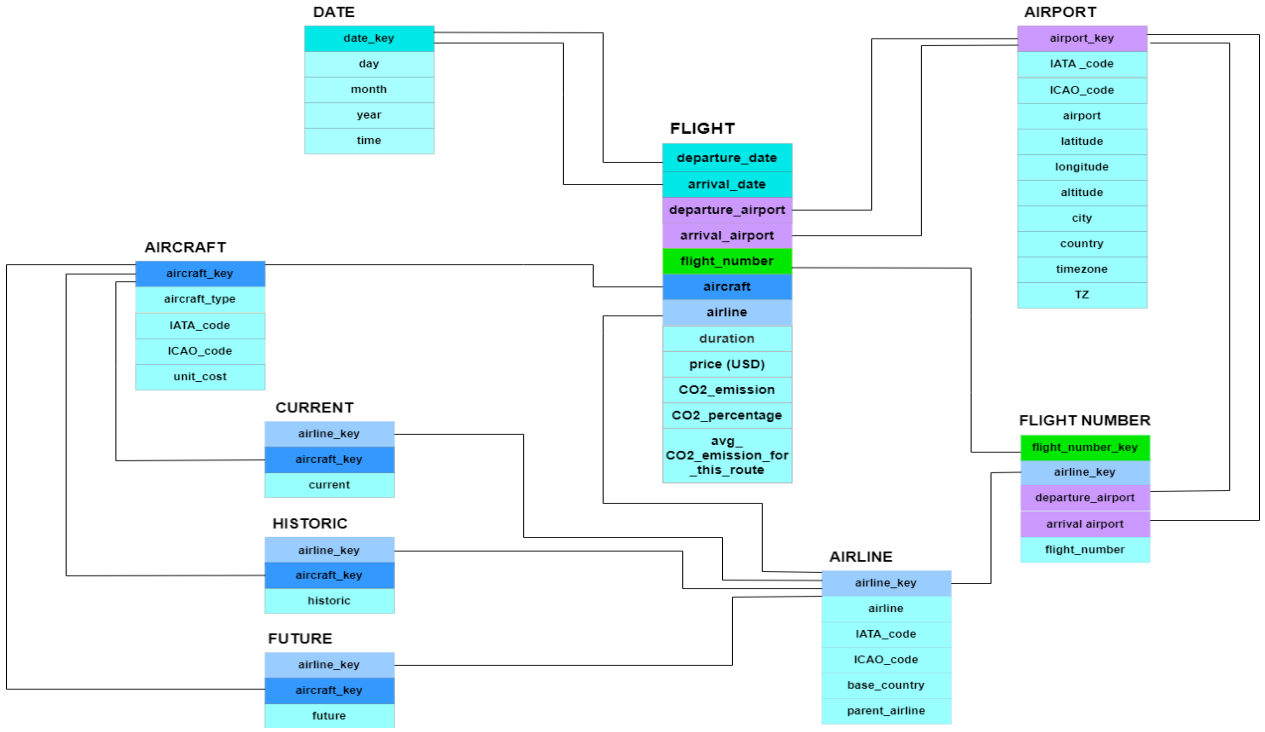
Conceptual modeling: DFM schema



Logical modeling: SNOWFLAKE schema



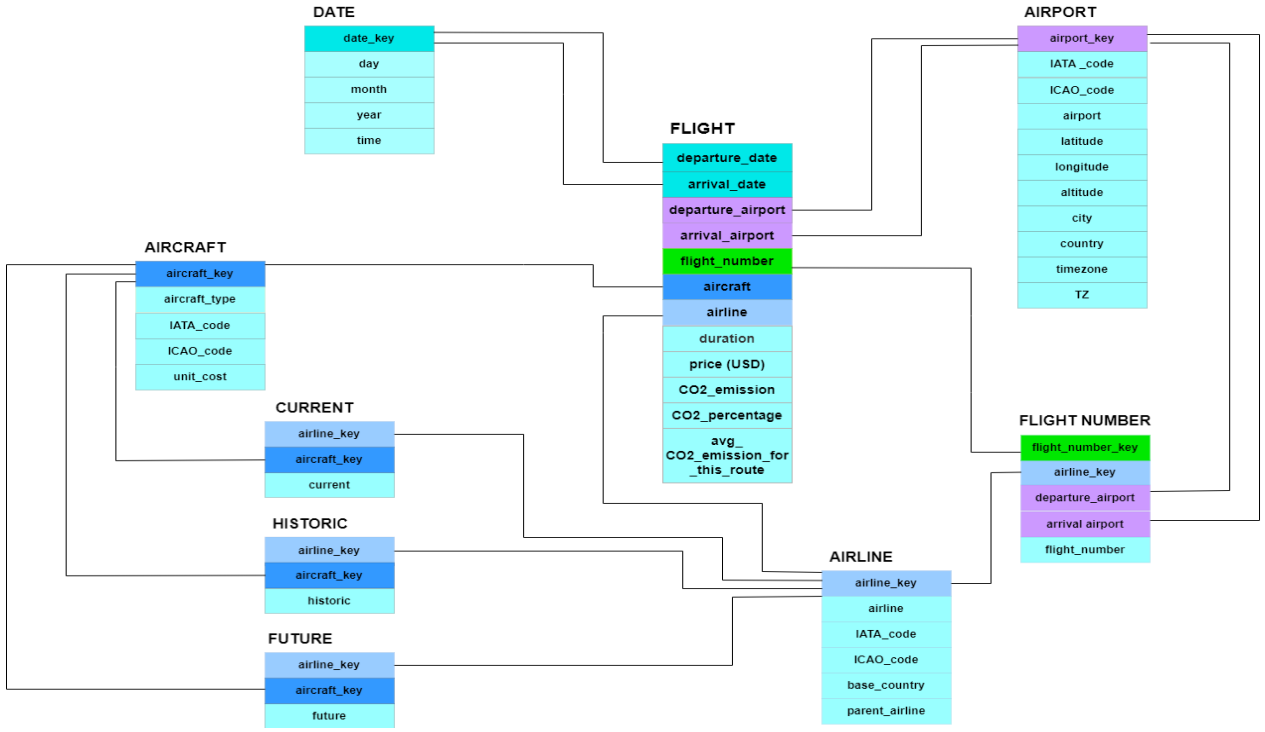
ROLAP queries over the snowflake schema



For each aircraft that perfomed some flights return the average co2 emission per minute:

SELECT ac.aircraft_type, sum(f.co2_emissions)/sum(f.duration) as average_co2_emissions_per_minute
FROM flight f **JOIN** aircraft ac **ON** f.aircraft = ac.aircraft_key
GROUP BY ac.aircraft_type

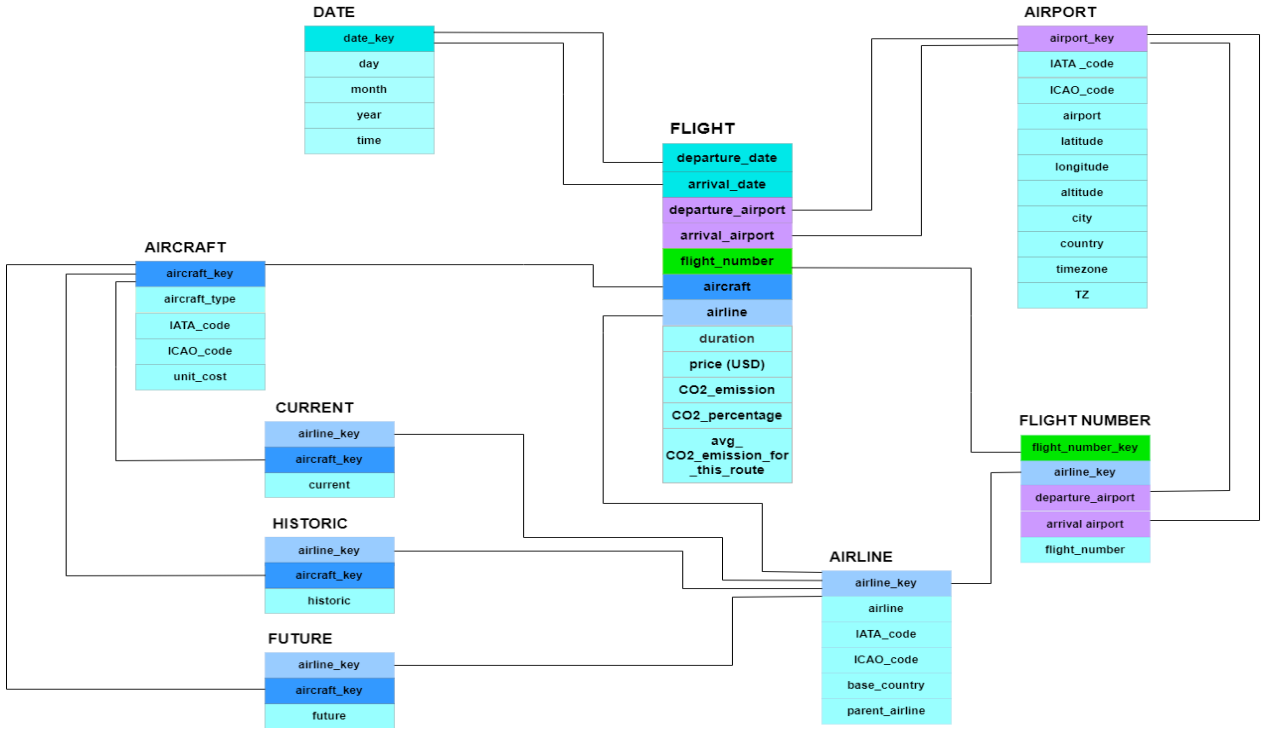
ROLAP queries over the snowflake schema



For each airport and each month return the overall number of flights:

```
SELECT a.airport, d.month, count(*)
FROM flight f JOIN date d ON f.departure_date = d.date_key, airport a
WHERE a.airport_key = f.departure_airport OR a.airport_key = f.arrival_airport
GROUP BY a.airport, d.month
ORDER BY a.airport, d.month
```

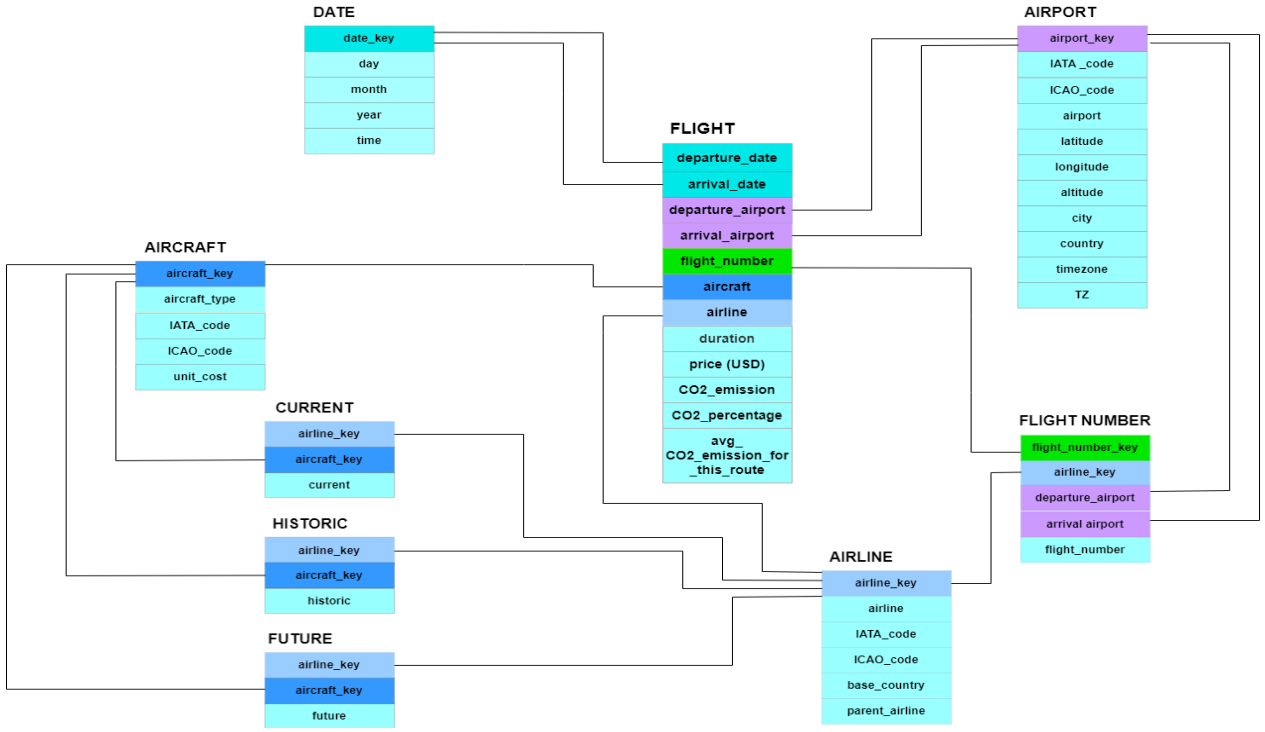
ROLAP queries over the snowflake schema



For each route and month return the average ticket price.

```
SELECT a1.airport, a1.city, a2.airport, a2.city,d.month, avg(f.price)
FROM flight f JOIN airport a1 ON f.departure_airport = a1.airport_key JOIN airport a2
    ON f.arrival_airport = a2.airport_key JOIN date d ON f.departure_date = d.date_key
GROUP BY a1.airport, a1.city, a2.airport, a2.city, d.month
ORDER BY a1.airport, a1.city, a2.airport, a2.city, d.month
```

ROLAP queries over the snowflake schema



For each day return the total ammount of emitted co2 by all the flights.

```
SELECT d.day, d.month, d.year, sum(f.co2_emissions)
FROM flight f JOIN date d ON f.departure_date = d.date_key
GROUP BY d.day, d.month, d.year
ORDER BY d.day, d.month, d.year
```

ROLAP queries over the snowflake schema

For each route return the less polluting aircraft:

```
CREATE view less_polluting_aircraft AS(
SELECT distinct a1.iata_code as departure, a2.iata_code as arrival, ac.aircraft_type,  f.co2_emissions
FROM flight f JOIN airport a1 ON f.departure_airport = a1.airport_key
              JOIN airport a2 ON f.arrival_airport = a2.airport_key
              JOIN aircraft ac ON f.aircraft = ac.aircraft_key
WHERE f.co2_emissions <= ALL(SELECT f2.co2_emissions
                             FROM flight f2
                             WHERE f.departure_airport = f2.departure_airport
                                   AND f.arrival_airport = f2.arrival_airport))
```

For each airline return the routes it does:

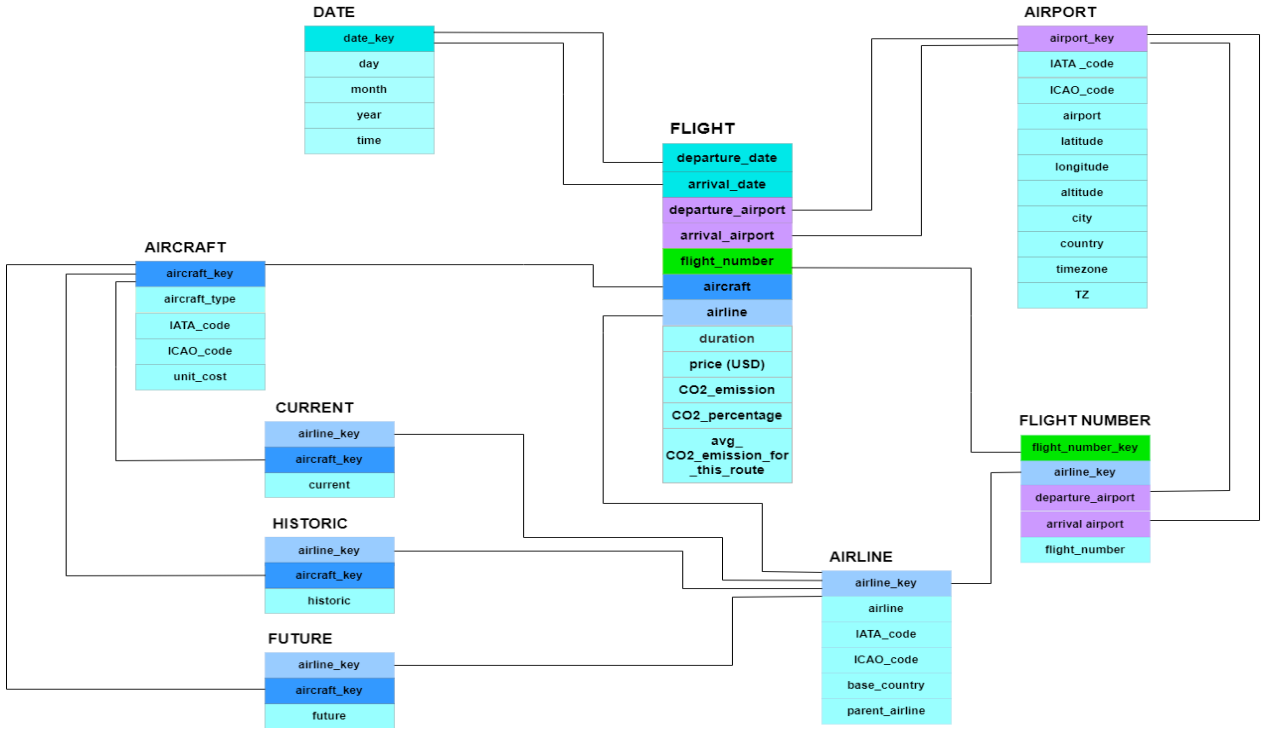
```
CREATE view routes_per_airline AS(
SELECT distinct a1.iata_code AS departure, a2.iata_code AS arrival, al.airline
FROM flight f JOIN airport a1 ON f.departure_airport = a1.airport_key
              JOIN airport a2 ON a2.airport_key = f.arrival_airport
              JOIN airline al ON al.airline_key = f.airline
)
```

ROLAP queries over the snowflake schema

For each route done by an airline see how many of the least polluting aircraft of this route the airline owns or will own and how much it has invesed on these aircrafts:

```
CREATE VIEW airlines_investments AS (
SELECT distinct al.airline, al.base_country, ac.aircraft_type, (c.current + f.future) AS quantity,
        (c.current + f.future)*ac.unit_cost_$ AS investment
FROM routes_per_airline, less_polluting_aircraft, current c, future f, airline al, aircraft ac
WHERE routes_per_airline.departure = less_polluting_aircraft.departure and
        routes_per_airline.arrival = less_polluting_aircraft.arrival and
        al.airline = routes_per_airline.airline and
        al.airline_key = c.airline_key and al.airline_key = f.airline_key and
        ac.aircraft_key = c.aircraft_key and ac.aircraft_key = f.aircraft_key and
        ac.aircraft_type = less_polluting_aircraft.aircraft_type and
        c.current != 0 or f.future != 0
ORDER BY quantity DESC, investment DESC)
```

ROLAP queries over the snowflake schema



Top 20 countries that invest in the least polluting aircrafts for the routes they do:

```
SELECT airlines_investments.base_country , SUM(quantity) AS total_aircrafts, SUM(investment) AS total_investment
FROM airlines_investments
GROUP BY airlines_investments.base_country
ORDER BY total_investment DESC
LIMIT 20
```

Analysis with Tableau

Minimum price per route

Airport	Airport (Airport1)	
Addis Ababa Bole International Airport	Brussels Airport	429,00 $
	Cairo International Airport	188,00 $
	Cape Town International ..	467,00 $
	Charles de Gaulle Internat..	476,00 $
	Chhatrapati Shivaji Intern..	510,00 $
	Dubai International Airport	406,00 $
	Eleftherios Venizelos Inte..	405,00 $
	Frankfurt am Main Airport	505,00 $
	Guarulhos - Governador A..	1.033,00 $
	Hamad International Airp..	330,00 $
	Incheon International Air..	911,00 $
	Indira Gandhi Internation..	483,00 $
	Istanbul Airport	531,00 $
	Jomo Kenyatta Internatio..	321,00 $
	Kempegowda Internation..	601,00 $
	Leonardo da VinciFiumicin..	490,00 $
	London Heathrow Airport	421,00 $
	OR Tambo International A..	438,00 $
	Stockholm-Arlanda Airport	473,00 $
	Suvarnabhumi Airport	622,00 $
	Vienna International Airp..	475,00 $
Beijing Capital	Brussels Airport	939,00 $

Number of flights per month for each airline

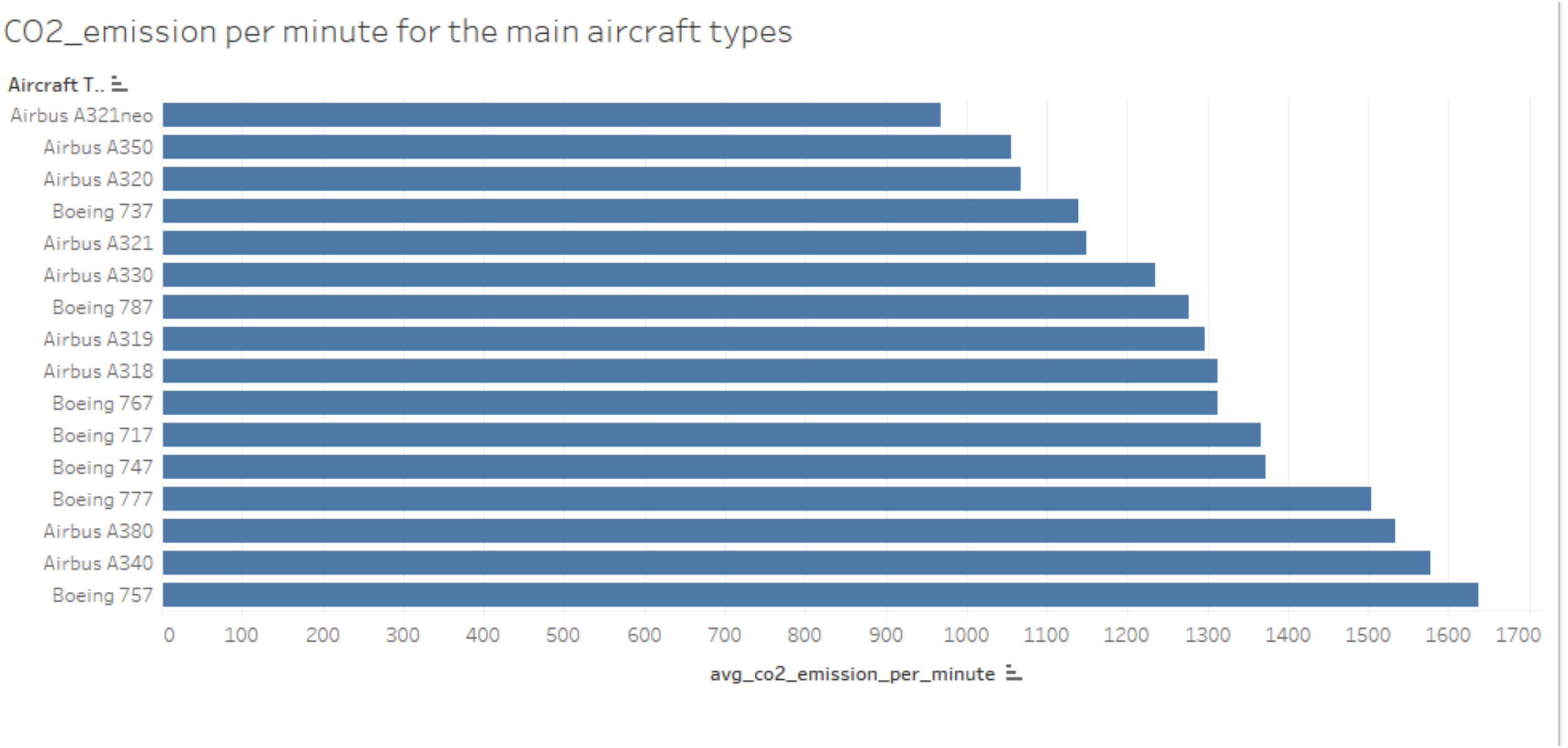
	month			
Airline (Airline)	4	5	7	8
aegean airlines		134	36	32
aer lingus	2	284	63	59
aerolineas argentinas	13	38	15	14
air algerie		10	3	2
air arabia maroc	1	1	1	1
air canada	39	161	47	45
air china		237	81	83
air europa	2	34	8	9
air france	2	502	104	104
air india limited	2	106	22	20
air tahiti nui		4	1	1
air transat	5	21	8	7
air vistara		210	41	40
airasia x		2		
alitalia	1	41	10	12
american airlines	7	101	22	23
asiana airlines	1	3	3	3
austrian airlines	31	244	65	54
avianca - aerovias naciona..	1	153	35	34
azul	8	45	15	11
british airways	7	155	48	38

Analysis with Tableau

For each airline and aircraft type, the number of aircrafts currently owned or to be owned in future

	Aircraft Type										
Airline (Airl..	Airbus A318	Airbus A319	Airbus A320	Airbus A321	Airbus A32..	Airbus A330	Airbus A340	Airbus A350	Airbus A380	Boeing 717	Boeing 737
alitalia		22 0	44 0			14 0					
american airlines		125 0									
asiana airlines						15 0		0 1			
austrian airlines			18 0	6 0							
avianca - aerovias na..											
azul			4 4			5 0					
british airways		44 0	67 0	18 0				0 0			
brussels airlines		21 1	9 0			9 1					
cebu pacific											
china airlines						24 0					
china eastern airl..			165 2	64 3		45 0		0 0			12
china southern ai..		34 0	123 8	88 5		38 1			5 0		16

Analysis with Tableau



Thanks for your attention!

It was a nice journey :)