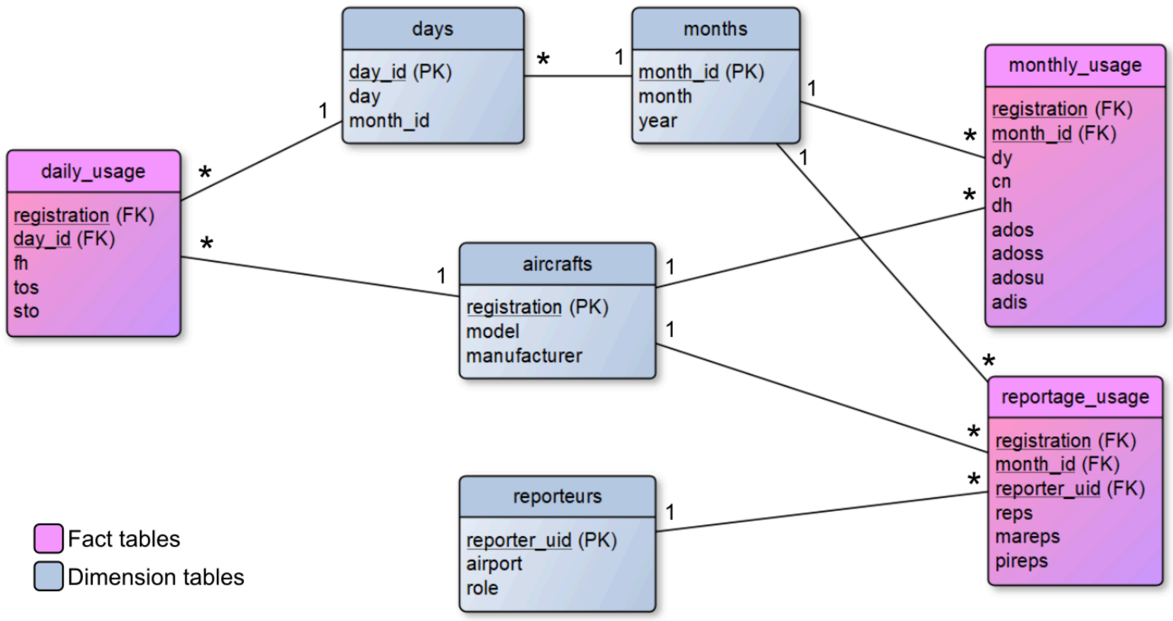
Bases de Dades Avançades (BDA) - Project 1: Data Warehousing (Descriptive analytics)
David Gonzalez and Angel Morales
Document 1/3: multidimensional schema

1. Conceptual design of the multidimensional schema:



2. Assumptions and justifications of the decisions made:

Since the main goal of the data warehouse is to provide the data necessary for the KPIs, we have assumed that the data does not need to be kept in the same structure as in the original database.

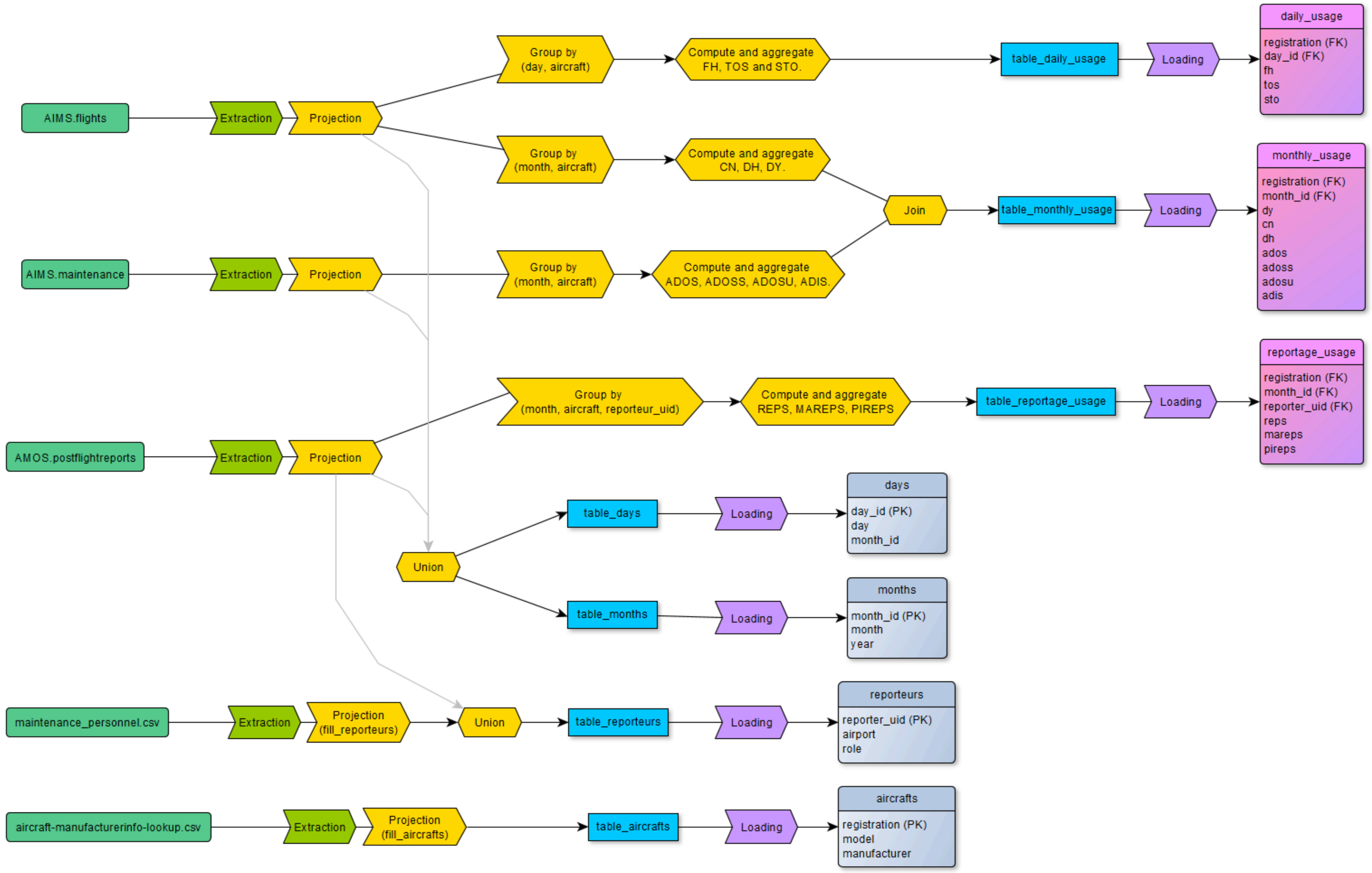
Therefore, our design aims to make the heavy computations on the transform phase, so that the retrieval of the KPIs is faster and more computationally efficient. This ensures that retrieving a KPI is a simple operation that often requires the selection of a single element or the aggregation of a few. In comparison, the same operation in a data-driven database would require making complex selections across tables of flights, maintenance or reports, that would need to traverse and aggregate many elements as well as making more join operations.

Our modeling is especially efficient when there are more flights, maintenances and reports condensed in the same date, since all their information ends up in the same date element. On the other hand, the advantage could not be so noticeable if the events were more sparse.

We structured our fact tables with primary keys linking to separate "days" and "months" dimension tables to match each KPI's desired granularity. This separation ensures that monthly queries don't need to scan and aggregate daily data, optimizing performance and simplicity. Furthermore, we maintain distinct dimension tables for aircrafts and reporteurs using their main identifiers (registration and reporteur_uid) as primary keys. This clear separation ensures efficient data organization while supporting flexible analysis across multiple granularities.

Bases de Dades Avançades (BDA) - Project 1: Data Warehousing (Descriptive analytics), David Gonzalez and Angel Morales

Document 2/3: ETL



Notes: Management of the business rules is done during the computation and aggregation of the metrics. The loading of the dimension tables (days, months, reporteurs and aircrafts) must be done before the loading of the fact tables (daily_usage, monthly_usage and reportage_usage).

Bases de Dades Avançades (BDA) - Project 1: Data Warehousing (Descriptive analytics)
David Gonzalez and Angel Morales
Document 3/3: query comparison

We have tested the three queries proposed. Those were initially written to be executed on the original sources and we have remade the queries to work on our data warehouse.

The main goal was to:

1. Make sure that the whole ETL worked as expected and the resulting data is correct.
2. Check that our solution is more efficient than the original queries on the sources.

In order to do this, we have executed both queries and compared the results retrieved and the time taken to perform the queries. We have also added a debugging trigger to the transform code to decide whether to apply the business rules. This allows a direct comparison between the original queries and ours.

Results:

1. When we don't apply the business rules, the error between the original retrieved values and the values retrieved from our solution is very little. We suspect that the main reason is due to the differences between Python libraries in terms of decimal rounding.

2. Queries are much faster when done on our system. After checking the average times that both implementations take, we see that ours are between 20 and 40 times faster. Here we can see the average elapsed time of each query on 10 executions:

	Aircraft Utilization	Query Reporting	Query Reporting per Role
From original database	0.7492	0.3336	0.3666
From data warehouse	0.0199	0.0157	0.0166
Comparison rate	**x 37.65**	**x 21.25**	**x 22.08**