First Project: Data Warehouse design
Mohana Fathollahi, Miona Dimic

Assumptions for Designing Schema[1]

Two schemas are created for the conceptual design.

1) First one consists of the fact Logbooks which has dimensions People, Aircraft_info and MonthYear.
2) Second schema has fact Aircraft metrics with dimensions Aircraft_info and dayMonthYear.

In each dimension corresponding granularity levels are described. As an example within dimension MonthYear lowest granularity level is MonthYearID (e.g. 3/2022) and the highest is Year (e.g. 2022). This separation is meant to allow aggregation or roll-up to a higher level.

Same reasoning has been used with People dimension and level hierarchy ID to Role or ID to Airport. If the ID refers to maintenance personnel we consider the airport. Therefore, we have a conditional relation between ID and Airport that is considered in Indyco and SQL.

In SQL we used P for pilot and M for maintenance and if a user made a mistake by inserting airport for pilot, SQL would return an error.

Aircraft_info dimension consists of lowest level ID that can be aggregated to model and then to manufacturer.

Two conformed hierarchies are noticed - MonthYear and Aircraft_info and those have been used as dimensions for two fact tables. To ensure granularity level - day, we created another dimension dayMonthYear and connected it to the MonthYear dimension. Therefore, we have a snowflake schema that helps to reduce redundancy and easily perform updates.

Assumptions for Materialised views

We considered one MV to answer the first two queries; AIRCRAFT_METRICS_DAILY.

Some KPIs can be calculated from information in this view such as ADOS, DC, DYR, etc. We update this MV every day to have all updates.

For the last two queries we considered two MVs, because we need to relate information from the Logbook to AircraftMetric. Therefore we created one MV over AircraftMetric to extract FH and TO for aircraft per day and year.

Additionally, we defined another MV LOOGBOOK_MONTHLY, to answer last queries and used AIRCRAFT_METRICS_FH to have FH and TO in this MV.

This will help us to reduce the cost of join operation. Moreover, updates are considered every month. To save costs we considered fast refresh instead of complete refresh.Even though the cost of QUERY REWRITE is high, it is useful for answering different queries, and therefore we enable it .

---

[1] In Indyco for dimension we mentioned the name of the dimensions just for clarification, it is not the level of dimension. Additionally, there is a typo in LogBooks that we forgot to write Aircraft_info for its dimension.