# **AID-Project**

Group 39, 89436 - Diogo Moniz Pacheco, 105180 - Roméo Paul-Loup Axel Legoupil

#### 1. Creation of the airports database

Once connected to the MySQL server, we execute the command to create the provided airport database: source airports.sql

In airport database, we create a view called **moreperformance** to use in the fact\_flight transformation to be faster.

Here is the query to **moreperformance**:

```
create or replace view moreperformance(flight_id, passengernumber,
total_receive, airline_id, airplane_id, from, to, departure, arrival) as
select b.flight_id, count(passenger_id) as passengernumber, sum(price) as
total_receive, airline_id, airplane_id, from, to, departure, arrival from
flight as f, booking as b where f.flight_id = b.flight_id group by
b.flight_id order by b.flight_id;
```

#### 2. Creation of the airports data warehouse

We create 4 dimension tables and 1 fact table into the data warehouse airports\_dw:

- dim\_airline : Airline dimension
- dim\_airplane : Airplane dimension
- dim\_airport : Airport dimension for origin and destination
- dim\_time: Time dimension for departure and arrival
- fact\_flight : Fact table for flight

#### Here is the code:

```
DROP DATABASE IF EXISTS airports_dw;

CREATE DATABASE airports_dw;

USE airports_dw;

CREATE TABLE dim_airline (
    AIRLINE_ID INT,
    AIRLINE_NAME VARCHAR(255),
    PRIMARY KEY (AIRLINE_ID)
);
```

```
CREATE TABLE dim airplane (
    AIRPLANE_ID INT,
    AIRPLANE_TYPE VARCHAR(255),
    PRIMARY KEY (AIRPLANE ID)
);
CREATE TABLE dim airport(
    AIRPORT_ID INT,
    AIRPORT_NAME VARCHAR(255),
    CITY VARCHAR(255),
    COUNTRY VARCHAR(255),
    PRIMARY KEY (AIRPORT_ID)
);
CREATE TABLE dim_time(
    TIME_ID DATETIME,
    YEAR ID INT,
    MONTH ID INT,
    MONTH_NAME VARCHAR(255),
    DAY_ID INT,
    PRIMARY KEY (TIME_ID)
);
CREATE TABLE fact_flight(
    FLIGHT_ID INT,
    PASSENGERS_NUMBER INT,
    RECEIVE TOTAL DOUBLE,
    AIRLINE_ID INT,
    AIRPLANE_ID INT,
    ORIGIN AIRPORT ID INT,
    DESTINATION_AIRPORT_ID INT,
    DEPARTURE_TIME_ID DATETIME,
    ARRIVAL_TIME_ID DATETIME,
    PRIMARY KEY (FLIGHT_ID),
    FOREIGN KEY (AIRLINE_ID) REFERENCES dim_airline (AIRLINE_ID),
    FOREIGN KEY (AIRPLANE_ID) REFERENCES dim_airplane (AIRPLANE_ID),
    FOREIGN KEY (ORIGIN_AIRPORT_ID) REFERENCES dim_airport (AIRPORT_ID),
    FOREIGN KEY (DESTINATION_AIRPORT_ID) REFERENCES dim_airport
(AIRPORT_ID),
    FOREIGN KEY (DEPARTURE_TIME_ID) REFERENCES dim_time (TIME_ID),
    FOREIGN KEY (ARRIVAL_TIME_ID) REFERENCES dim_time (TIME_ID)
);
```

Then to create the airports data warehouse we execute the following command: source airports\_dw\_sql

# 3. Transformations developed in PDI

#### **Airline dimension**

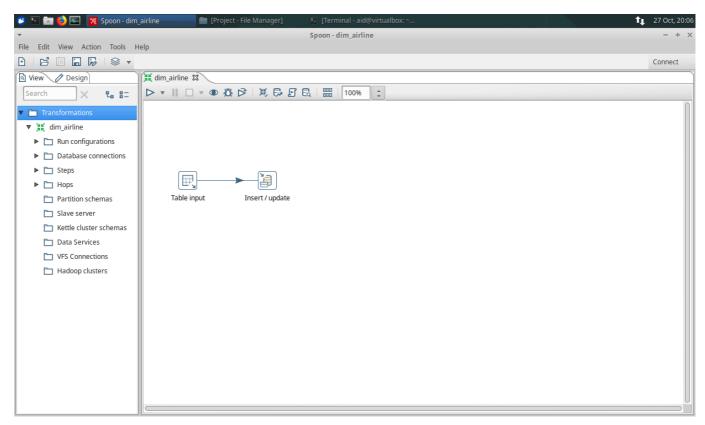


Figure 1 - dim\_airline entire transformation

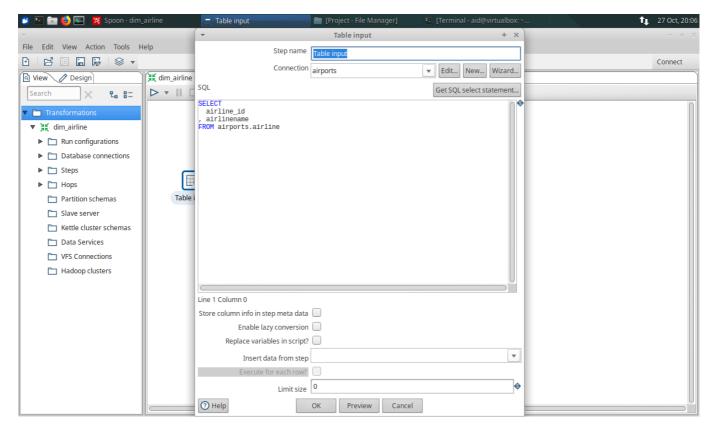


Figure 2 - dim\_airline table input window

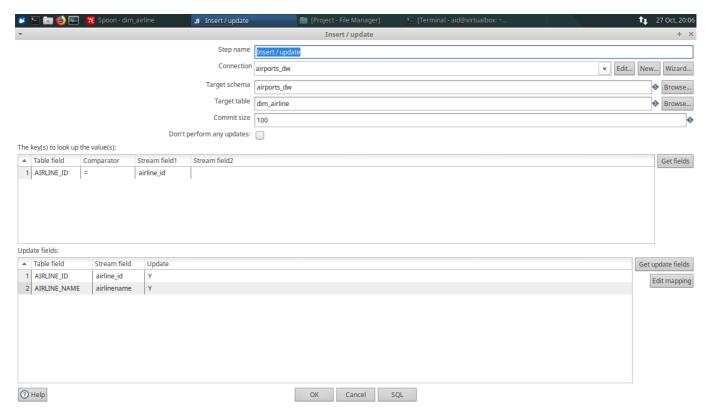


Figure 3 - dim\_airline insert/update window

### **Airplane dimension**

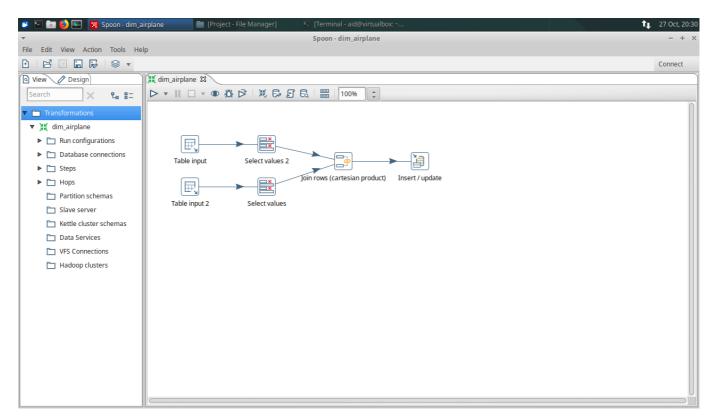


Figure 4 - dim\_airplane entire transformation

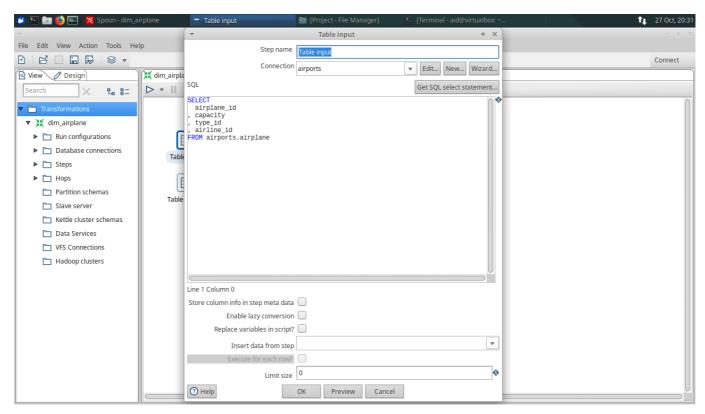


Figure 5 - dim\_airplane table input window

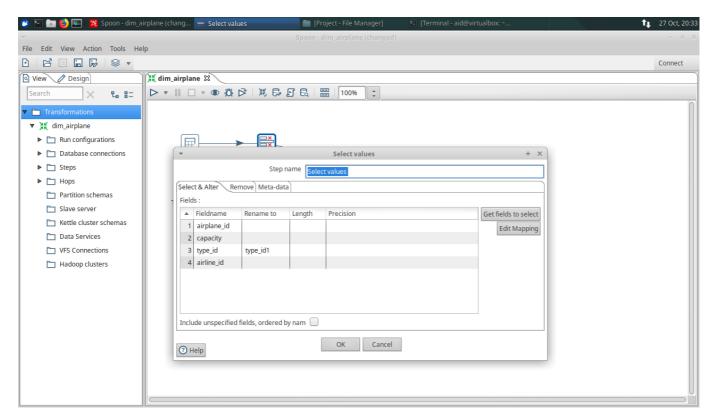


Figure 6 - dim\_airplane select values window

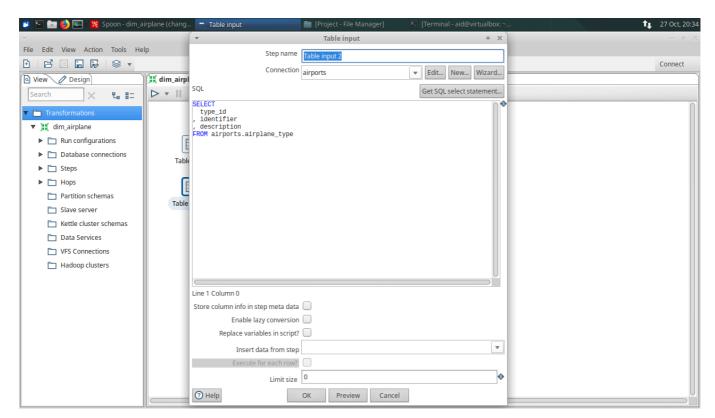


Figure 7 - dim\_airplane table input 2 window

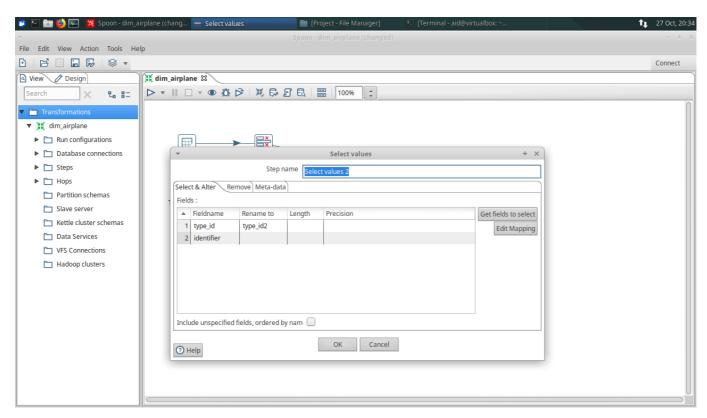


Figure 8 - dim\_airplane select values 2 window

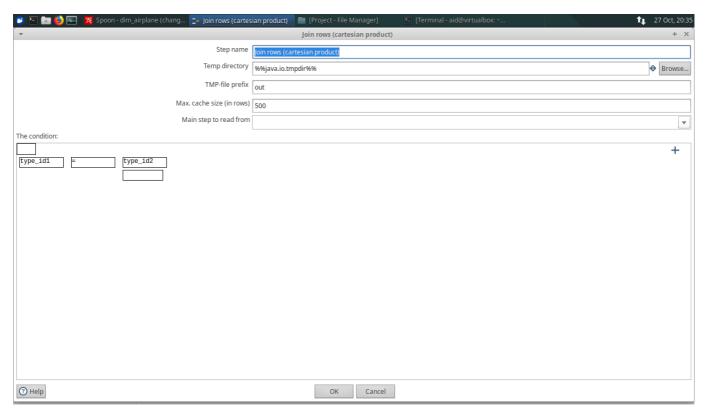


Figure 9 - dim\_airplane join rows window

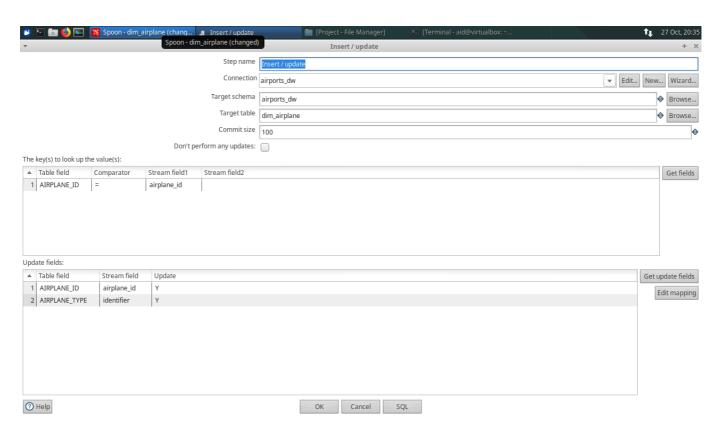


Figure 10 - dim\_airplane insert/update window

#### **Airport dimension**

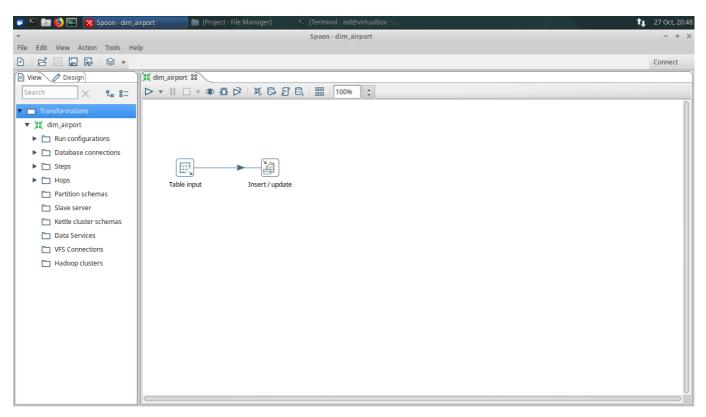


Figure 11 - dim\_airport entire transformation

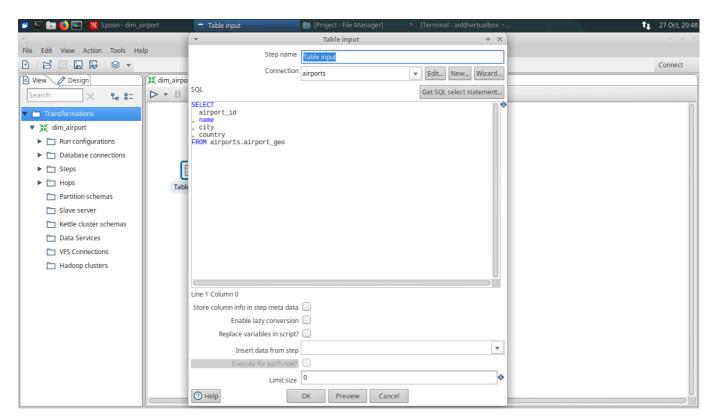


Figure 12 - dim\_airport input table window

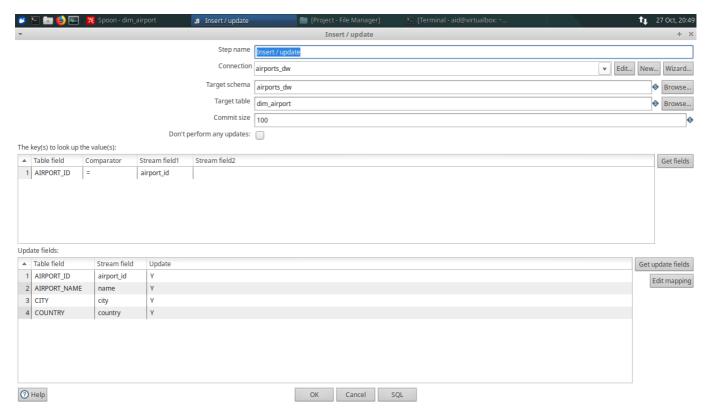


Figure 13 - dim\_airport insert/update window

#### **Time dimension**

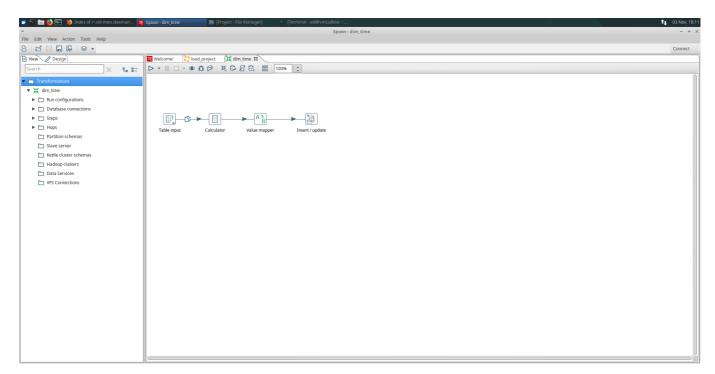


Figure 14 - dim\_time entire transformation

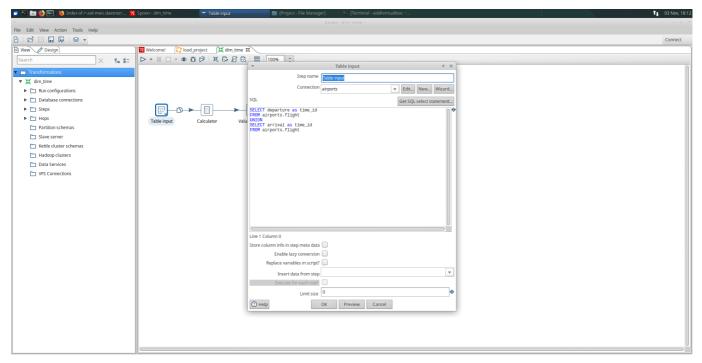


Figure 15 - dim\_time table input window (union between departures and arrivals)

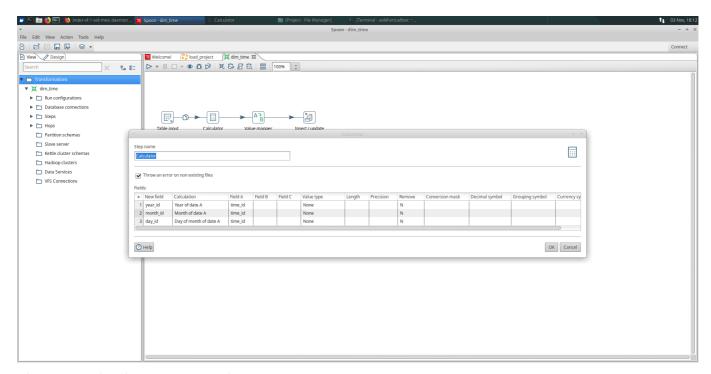


Figure 16 - dim\_time calculator window

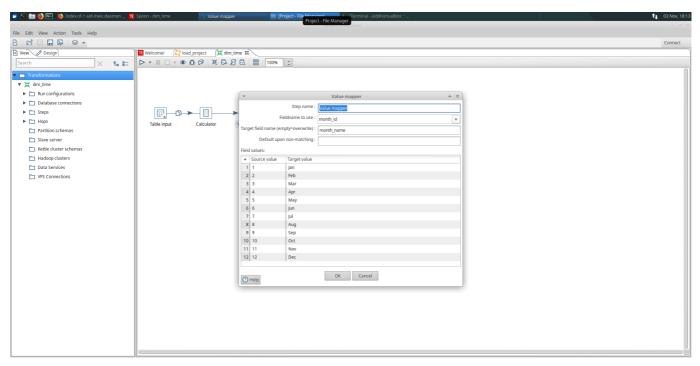


Figure 17 - dim\_time value mapper window

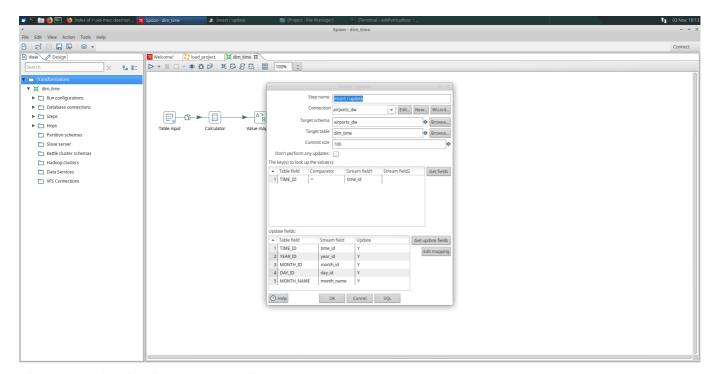


Figure 18 - dim\_time insert/update window

#### Flight fact

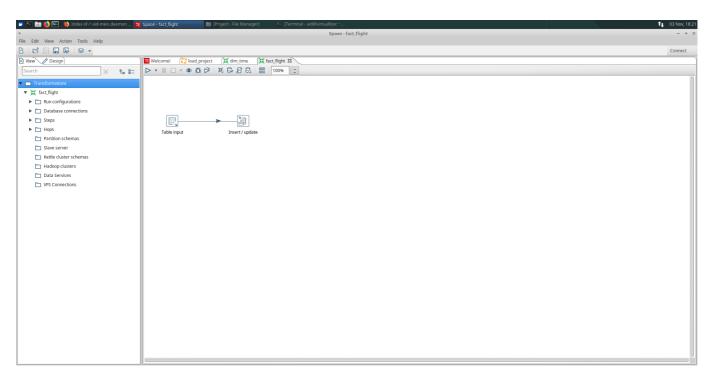


Figure 19 - fact\_flight entire transformation

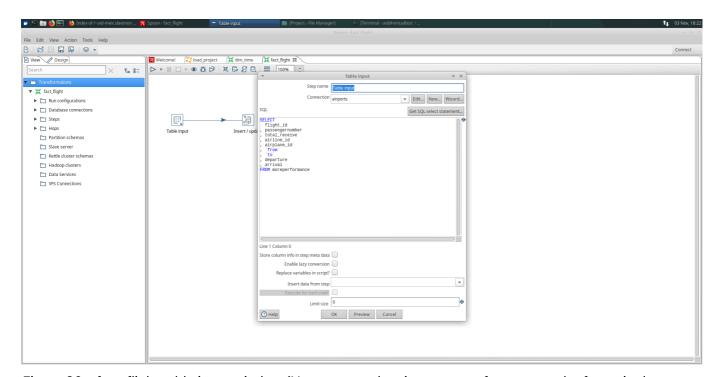


Figure 20 - fact\_flight table input window (Here we use the view moreperformance to be faster in the largest databases)

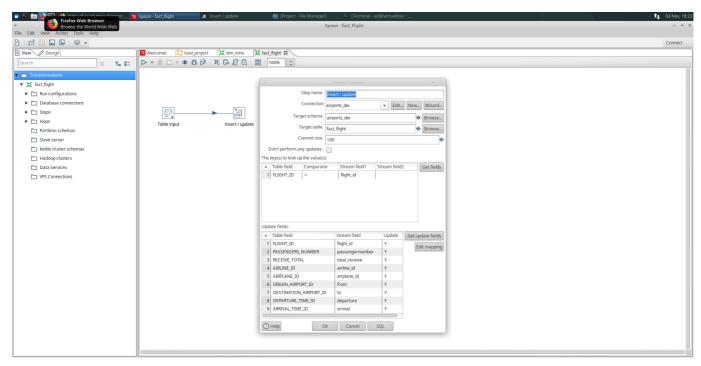


Figure 21 - fact\_flight insert/update window

#### **Final Job**

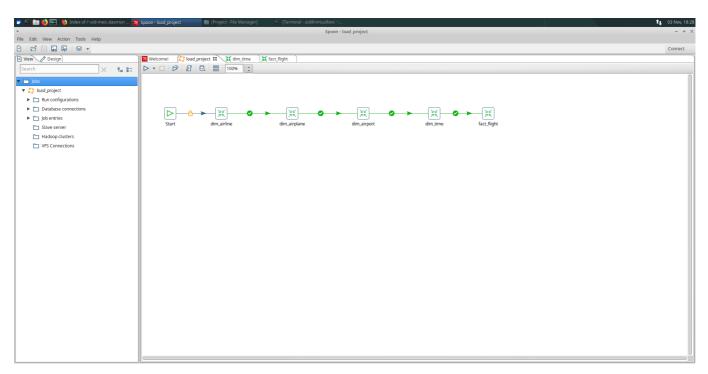


Figure 22 - Complete job

#### 4. XML Code for the cube definition

```
<Schema name="airports_dw">
  <Cube name="Flights" visible="true" cache="true" enabled="true">
    <Table name="fact_flight">
    </Table>
    <Dimension type="StandardDimension" visible="true"</pre>
foreignKey="AIRLINE_ID" highCardinality="false" name="Airline">
      <Hierarchy name="Airline Hierarchy" visible="true" hasAll="true"</pre>
allMemberName="All Arlines" primaryKey="AIRLINE ID">
        <Table name="dim airline">
        </Table>
        <Level name="Name" visible="true" column="AIRLINE NAME"</pre>
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true"</pre>
foreignKey="ORIGIN_AIRPORT_ID" highCardinality="false"
name="0riginAirport">
      <Hierarchy name="Origin Airport Hierarchy" visible="true"</pre>
hasAll="true" allMemberName="All Origin Airports" primaryKey="AIRPORT_ID">
        <Table name="dim_airport">
        </Table>
        <Level name="OriginCountry" visible="true" column="COUNTRY"</pre>
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
        <Level name="OriginCity" visible="true" column="CITY"</pre>
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
    <Level name="OriginName" visible="true" column="AIRPORT_NAME"</pre>
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true"</pre>
foreignKey="DESTINATION_AIRPORT_ID" highCardinality="false"
name="DestinationAirport">
      <Hierarchy name="Destination Airport Hierarchy" visible="true"</pre>
hasAll="true" allMemberName="All Destination Airports"
primaryKey="AIRPORT_ID">
        <Table name="dim_airport">
        </Table>
        <Level name="DestinationCountry" visible="true" column="COUNTRY"</pre>
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
        <Level name="DestinationCity" visible="true" column="CITY"</pre>
```

```
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
    <Level name="DestinationName" visible="true" column="AIRPORT_NAME"</pre>
type="String" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="TimeDimension" visible="true"</pre>
foreignKey="DEPARTURE_TIME_ID" highCardinality="false" name="Time">
      <Hierarchy name="Time Hierarchy" visible="true" hasAll="true"</pre>
allMemberName="All Years" primaryKey="TIME_ID">
        <Table name="dim_time">
        </Table>
        <Level name="Year" visible="true" column="YEAR_ID" type="Integer"</pre>
uniqueMembers="false" levelType="TimeYears" hideMemberIf="Never">
        </Level>
        <Level name="Month" visible="true" column="MONTH NAME"</pre>
ordinalColumn="MONTH ID" type="String" uniqueMembers="false"
levelType="TimeMonths" hideMemberIf="Never">
        </Level>
        <Level name="Day" visible="true" column="DAY ID" type="Integer"</pre>
uniqueMembers="false" levelType="TimeDays" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true"</pre>
foreignKey="AIRPLANE_ID" name="Airplane">
      <Hierarchy name="Airplane Hierarchy" visible="true" hasAll="true"</pre>
allMemberName="All Airplanes" primaryKey="AIRPLANE_ID">
        <Table name="dim_airplane">
        </Table>
        <Level name="Type" visible="true" column="AIRPLANE_TYPE"</pre>
type="String" uniqueMembers="false" levelType="Regular">
        </Level>
      </Hierarchy>
    </Dimension>
    <Measure name="TotalNumberPassengers" column="PASSENGERS_NUMBER"</pre>
datatype="Integer" formatString="#,###" aggregator="sum" visible="true">
    </Measure>
    <Measure name="TotalRevenue" column="RECEIVE_TOTAL" datatype="Numeric"</pre>
formatString="$ #,###.00" aggregator="sum" visible="true">
    </Measure>
  </Cube>
</Schema>
```

# 5. Database queries

Query1: passengers and revenue by airline and month (There is not a query because we don't use MDX mode)

Query2 code: Top5 airlines with better revenue

```
SELECT Measures.Members ON COLUMNS,
TOPCOUNT(Airline.[Name].Members, 5, Measures.TotalRevenue) ON ROWS
FROM Flights
```

Query3 code: Average price per passenger from other countries to germany

```
WITH MEMBER Measures.AveragePricePerTicket AS (Measures.TotalRevenue / Measures.TotalNumberPassengers)
SELECT Measures.AveragePricePerTicket ON COLUMNS,
ORDER(OriginAirport.OriginCountry.Members, Measures.AveragePricePerTicket,
DESC) ON ROWS
FROM Flights WHERE (DestinationAirport.GERMANY)
```

## 5.1. Small database (airports.sql) queries

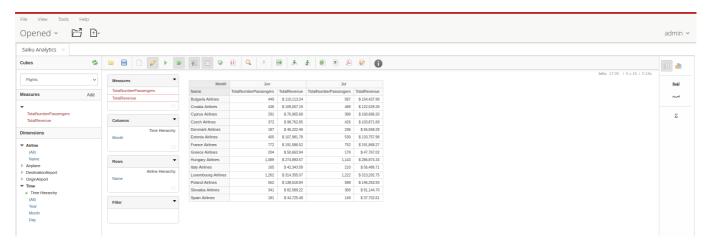


Figure 23 - Query1: passengers and revenue by airline and month (small database - airports.sql)



Figure 24 - Query2: Top5 airlines with better revenue (small database - airports.sql)

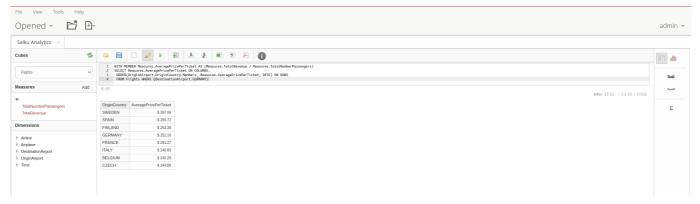


Figure 25 - Query3: Average price per passenger from other countries to germany (small database - airports.sql)

# 5.2. Larger database (airports-larger.sql) queries

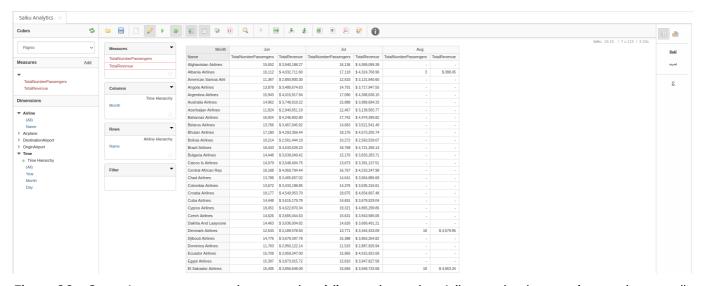


Figure 26 - Query1: passengers and revenue by airline and month pt1 (larger database - airports-larger.sql)

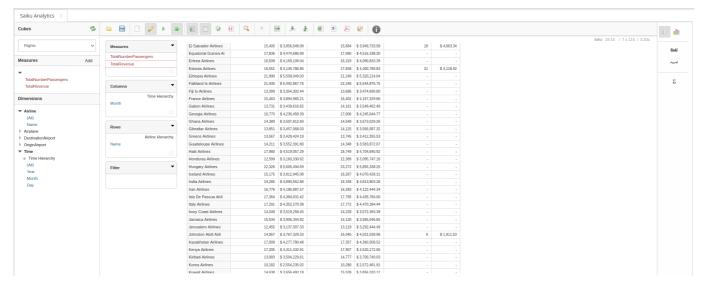


Figure 27 - Query1: passengers and revenue by airline and month pt2 (larger database - airports-larger.sql)

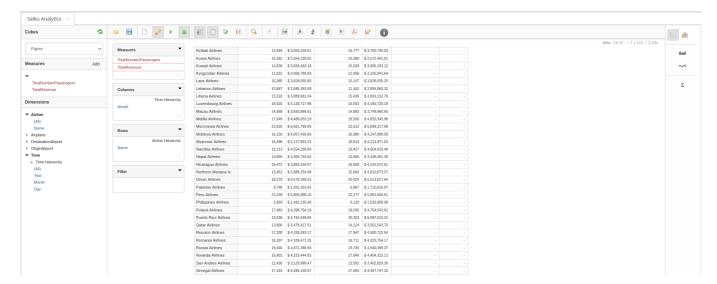


Figure 28 - Query1: passengers and revenue by airline and month pt3 (larger database - airports-larger.sql)

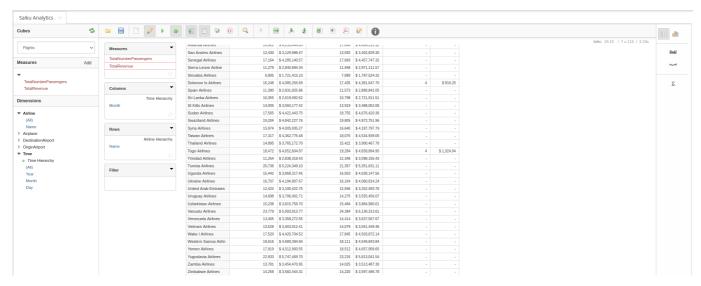


Figure 29 - Query1: passengers and revenue by airline and month pt4 (larger database - airports-larger.sql)

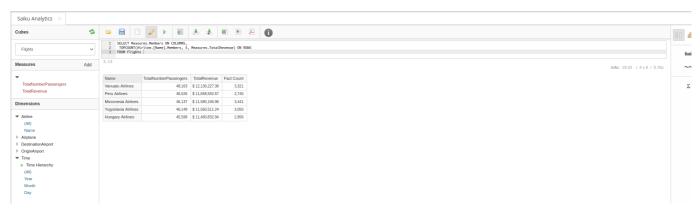


Figure 30 - Query2: Top5 airlines with better revenue (larger database - airports-larger.sql)

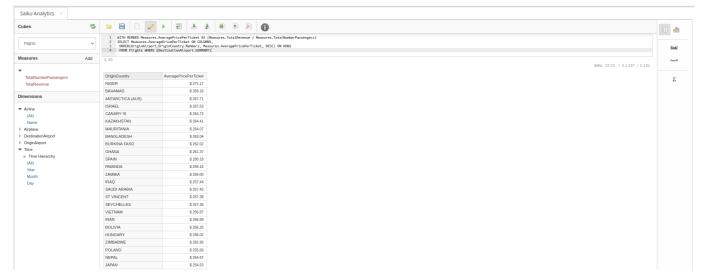


Figure 31 - Query3: Average price per passenger from other countries to germany pt1 (larger database - airports-larger.sql)



Figure 32 - Query3: Average price per passenger from other countries to germany pt2 (larger database - airports-larger.sql)

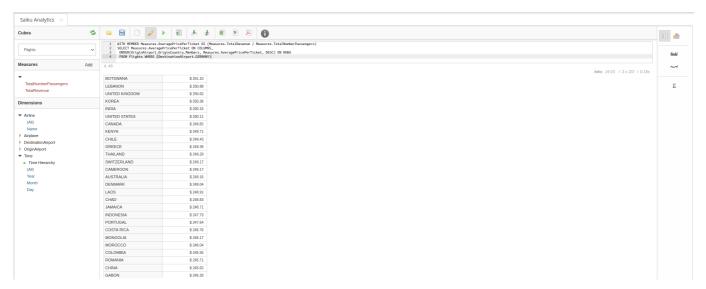


Figure 33 - Query3: Average price per passenger from other countries to germany pt3 (larger database - airports-larger.sql)

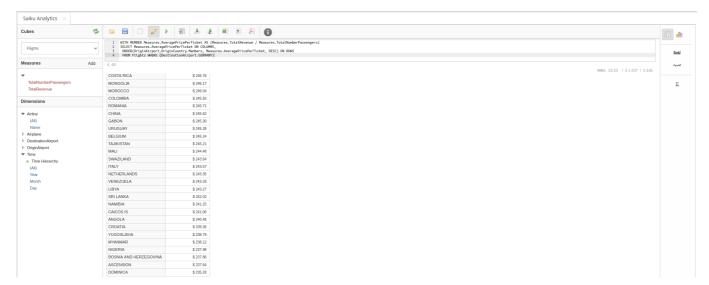


Figure 34 - Query3: Average price per passenger from other countries to germany pt4 (larger database - airports-larger.sql)