

# Project Report

## Data Analysis and Integration – 2022/2023

**Group:** 2

**Students:** Francisco Ribeiro (95578) and Diogo Lopes (96732)

---

### Contents

Data Warehouse SQL .....	1
Transformations .....	3
dim_airline .....	3
dim_airplane .....	4
dim_arrival .....	5
dim_departure .....	6
dim_airport .....	7
fact_flight .....	8
Jobs.....	10
load_dw.....	10
load_fact.....	10
Cube Definition .....	11
Saiku Analysis .....	14
airports.sql .....	14
passengers and revenue by airline and month .....	14
airplane type and month of departure by month of arrival .....	15
how much money spent/gained on flights between countries .....	16
airports-large.sql .....	17
passengers and revenue by airline and month .....	17
airplane type and month of departure by month of arrival .....	18
how much money spent/gained on flights between countries .....	19
airports-large-extra.sql.....	20
passengers and revenue by airline and month .....	20
airplane type and month of departure by month of arrival .....	21
how much money spent/gained on flights between countries .....	22
Comparisons.....	23

---

## Data Warehouse SQL

*airports\_dw.sql*

```
DROP DATABASE IF EXISTS airports_dw;
```

```
CREATE DATABASE airports_dw;
```

```
USE airports_dw;
```

```
CREATE TABLE dim_airport (  
    AIRPORT_ID INT,  
    AIRPORT_NAME VARCHAR(255),  
    CITY VARCHAR(255),  
    COUNTRY VARCHAR(255),  
    PRIMARY KEY (AIRPORT_ID)  
);
```

```
CREATE TABLE dim_departure (  
    TIME_ID DATETIME,  
    YEAR_ID INT,  
    MONTH_ID INT,  
    MONTH_NAME VARCHAR(255),  
    DAY_ID INT,  
    PRIMARY KEY (TIME_ID)  
);
```

```
CREATE TABLE dim_arrival (  
    TIME_ID DATETIME,  
    YEAR_ID INT,  
    MONTH_ID INT,  
    MONTH_NAME VARCHAR(255),  
    DAY_ID INT,  
    PRIMARY KEY (TIME_ID)  
);
```

```
CREATE TABLE dim_airplane (  
    AIRPLANE_ID INT,  
    AIRPLANE_TYPE INT,  
    PRIMARY KEY (AIRPLANE_ID)  
);
```

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

```
);  
  
CREATE TABLE fact_flight (  
    FLIGHT_ID INT,  
    TOTALBOOKINGS INT,  
    REVENUE DECIMAL(65,2),  
    AIRLINE_ID INT,  
    AIRPLANE_ID INT,  
    ORIGIN_ID INT,  
    DESTINATION_ID INT,  
    DEPARTURE DATETIME,  
    ARRIVAL 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_ID) REFERENCES dim_airport (AIRPORT_ID),  
    FOREIGN KEY (DESTINATION_ID) REFERENCES dim_airport (AIRPORT_ID),  
    FOREIGN KEY (DEPARTURE) REFERENCES dim_departure (TIME_ID),  
    FOREIGN KEY (ARRIVAL) REFERENCES dim_arrival (TIME_ID)  
)
```

# Transformations

## dim\_airline

**Table input**

Step name: Table input  
Connection: airports

SQL

```
SELECT  
  airline_id  
,  airlinename  
FROM airports.airline
```

Line 5 Column 0

Store column info in step meta data: ☐  
Enable lazy conversion: ☐  
Replace variables in script?: ☐  
Insert data from step:   
Execute for each row?: ☐  
Limit size: 0

**Insert / update**

Step name: Insert / update  
Connection: airports\_dw  
Target schema: airports\_dw  
Target table: dim\_airline  
Commit size: 100

Don't perform any updates: ☐

The key(s) to look up the value(s):

Table field	Comparator	Stream field1	Stream field2
AIRLINE_ID	=	airline_id	

Update fields:

Table field	Stream field	Update
AIRLINE_ID	airline_id	Y
AIRLINE_NAME	airlinename	Y

Examine preview data		
Rows of step: Table input (14 rows)		
	airline_id	airlinename
1	13	Bulgaria Airlines
2	18	Croatia Airlines
3	20	Cyprus Airlines
4	21	Czech Airlines
5	23	Denmark Airlines
6	31	Estonia Airlines
7	35	France Airlines
8	40	Greece Airlines
9	44	Hungary Airlines
10	49	Italy Airlines
11	63	Luxembourg Airlines
12	77	Poland Airlines
13	87	Slovakia Airlines
14	89	Spain Airlines

Examine preview data		
Rows of step: Insert / update (14 rows)		
	airline_id	airlinename
1	13	Bulgaria Airlines
2	18	Croatia Airlines
3	20	Cyprus Airlines
4	21	Czech Airlines
5	23	Denmark Airlines
6	31	Estonia Airlines
7	35	France Airlines
8	40	Greece Airlines
9	44	Hungary Airlines
10	49	Italy Airlines
11	63	Luxembourg Airlines
12	77	Poland Airlines
13	87	Slovakia Airlines
14	89	Spain Airlines

## dim\_airplane

The screenshot shows the Informatica Designer interface with the 'dim\_airplane' dimension table being configured. The 'Table input' step is configured with the following SQL query:

```
SELECT
  airplane_id
, type_id
FROM airports.airplane
```

The 'Insert / update' step is configured with the following settings:

- Step name: Insert / update
- Connection: airports\_dw
- Target schema: airports\_dw
- Target table: dim\_airplane
- Commit size: 100
- Don't perform any updates: ☐
- The key(s) to look up the value(s):

Table field	Comparator	Stream field1	Stream field2
1 AIRPLANE_ID	=	airplane_id	

Update fields:

Table field	Stream field	Update
1 AIRPLANE_ID	airplane_id	Y
2 AIRPLANE_TYPE	type_id	Y

Examine preview data		
Rows of step: Table input (410 rows)		
▲	airplane_id	type_id
1	1	228
2	2	38
3	3	60
4	4	232
5	5	21
6	6	48
7	7	41
8	9	40
9	10	41
10	11	60
11	12	6
12	13	232
13	14	232
14	15	21
15	17	316

Examine preview data		
Rows of step: Insert / update (410 rows)		
▲	airplane_id	type_id
1	1	228
2	2	38
3	3	60
4	4	232
5	5	21
6	6	48
7	7	41
8	9	40
9	10	41
10	11	60
11	12	6
12	13	232
13	14	232
14	15	21
15	17	316

## dim\_arrival

**Table input**

Step name: Table input  
Connection: airports

SQL:

```
SELECT
  arrival
FROM airports.flight
```

**Calculator**

Step name: Calculator

Fields:

New field	Calculation	Field A
1 year_id	Year of date A	arrival
2 month_id	Month of date A	arrival
3 day_id	Day of month of date A	arrival

**Value mapper**

Step name: Value mapper

Fieldname to use: month\_id

Target field name: month\_name

Field values:

Source value	Target value
1	Jan
2	Feb
3	Mar
4	Apr
5	May
6	Jun
7	Jul
8	Aug
9	Sep
10	Oct
11	Nov
12	Dec

**Insert / update**

Step name: Insert / update

Connection: airports\_dw

Target schema: airports\_dw

Target table: dim\_arrival

Commit size: 1000

Don't perform any updates: ☐

The key(s) to look up the value(s):

Table field	Comparator	Stream field1	Stream field2
1 TIME_ID	=	arrival	

Update fields:

Table field	Stream field	Update
1 TIME_ID	arrival	Y
2 YEAR_ID	year_id	Y
3 MONTH_ID	month_id	Y
4 DAY_ID	day_id	Y
5 MONTH_NAME	month_name	Y

**Examine preview data**

Rows of step: Table input (1210 rows)

arrival
1 2015/06/01 03:19:00.000000000
2 2015/06/01 03:23:00.000000000
3 2015/06/01 04:56:00.000000000
4 2015/06/01 08:16:00.000000000
5 2015/06/01 08:22:00.000000000
6 2015/06/01 08:59:00.000000000
7 2015/06/01 09:11:00.000000000
8 2015/06/01 14:11:00.000000000
9 2015/06/01 16:41:00.000000000
10 2015/06/01 17:39:00.000000000
11 2015/06/01 19:09:00.000000000
12 2015/06/01 19:27:00.000000000
13 2015/06/01 21:18:00.000000000
14 2015/06/01 22:20:00.000000000

**Examine preview data**

Rows of step: Value mapper (1210 rows)

arrival	year_id	month_id	day_id	month_name
1 2015/06/01 03:19:00.000000000	2015	6	1	Jun
2 2015/06/01 03:23:00.000000000	2015	6	1	Jun
3 2015/06/01 04:56:00.000000000	2015	6	1	Jun
4 2015/06/01 08:16:00.000000000	2015	6	1	Jun
5 2015/06/01 08:22:00.000000000	2015	6	1	Jun
6 2015/06/01 08:59:00.000000000	2015	6	1	Jun
7 2015/06/01 09:11:00.000000000	2015	6	1	Jun
8 2015/06/01 14:11:00.000000000	2015	6	1	Jun
9 2015/06/01 16:41:00.000000000	2015	6	1	Jun
10 2015/06/01 17:39:00.000000000	2015	6	1	Jun
11 2015/06/01 19:09:00.000000000	2015	6	1	Jun
12 2015/06/01 19:27:00.000000000	2015	6	1	Jun
13 2015/06/01 21:18:00.000000000	2015	6	1	Jun
14 2015/06/01 22:20:00.000000000	2015	6	1	Jun

**Examine preview data**

Rows of step: Calculator (1210 rows)

arrival	year_id	month_id	day_id
1 2015/06/01 03:19:00.000000000	2015	6	1
2 2015/06/01 03:23:00.000000000	2015	6	1
3 2015/06/01 04:56:00.000000000	2015	6	1
4 2015/06/01 08:16:00.000000000	2015	6	1
5 2015/06/01 08:22:00.000000000	2015	6	1
6 2015/06/01 08:59:00.000000000	2015	6	1
7 2015/06/01 09:11:00.000000000	2015	6	1
8 2015/06/01 14:11:00.000000000	2015	6	1
9 2015/06/01 16:41:00.000000000	2015	6	1
10 2015/06/01 17:39:00.000000000	2015	6	1
11 2015/06/01 19:09:00.000000000	2015	6	1
12 2015/06/01 19:27:00.000000000	2015	6	1
13 2015/06/01 21:18:00.000000000	2015	6	1
14 2015/06/01 22:20:00.000000000	2015	6	1
15 2015/06/01 22:26:00.000000000	2015	6	1

**Examine preview data**

Rows of step: Insert / update (1210 rows)

arrival	year_id	month_id	day_id	month_name
1 2015/06/01 03:19:00.000000000	2015	6	1	Jun
2 2015/06/01 03:23:00.000000000	2015	6	1	Jun
3 2015/06/01 04:56:00.000000000	2015	6	1	Jun
4 2015/06/01 08:16:00.000000000	2015	6	1	Jun
5 2015/06/01 08:22:00.000000000	2015	6	1	Jun
6 2015/06/01 08:59:00.000000000	2015	6	1	Jun
7 2015/06/01 09:11:00.000000000	2015	6	1	Jun
8 2015/06/01 14:11:00.000000000	2015	6	1	Jun
9 2015/06/01 16:41:00.000000000	2015	6	1	Jun
10 2015/06/01 17:39:00.000000000	2015	6	1	Jun
11 2015/06/01 19:09:00.000000000	2015	6	1	Jun
12 2015/06/01 19:27:00.000000000	2015	6	1	Jun
13 2015/06/01 21:18:00.000000000	2015	6	1	Jun
14 2015/06/01 22:20:00.000000000	2015	6	1	Jun

## dim\_departure

**Table input**

Step name: Table input  
Connection: airports

SQL:

```
SELECT
  departure
FROM airports.flight
```

**Calculator**

Step name: Calculator

Fields:

New field	Calculation	Field A
1 year_id	Year of date A	departure
2 month_id	Month of date A	departure
3 day_id	Day of month of date A	departure

**Value mapper**

Step name: Value mapper

Fieldname to u: month\_id

Target field name: month\_name

Field values:

Source value	Target value
1	Jan
2	Feb
3	Mar
4	Apr
5	May
6	Jun
7	Jul
8	Aug
9	Sep
10	Oct
11	Nov
12	Dec

**Insert / update**

Step name: Insert / update

Connection: airports\_dw

Target schema: airports\_dw

Target table: dim\_departure

Commit size: 1000

Don't perform any updates: ☐

The key(s) to look up the value(s):

Table field	Comparator	Stream field1	Stream field2
1 TIME_ID	=	departure	

Update fields:

Table field	Stream field	Update
1 TIME_ID	departure	Y
2 YEAR_ID	year_id	Y
3 MONTH_ID	month_id	Y
4 DAY_ID	day_id	Y
5 MONTH_NAME	month_name	Y

**Examine preview data (Table input)**

departure
1 2015/06/01 01:26:00.000000000
2 2015/06/01 03:04:00.000000000
3 2015/06/01 03:04:00.000000000
4 2015/06/01 07:26:00.000000000
5 2015/06/01 07:35:00.000000000
6 2015/06/01 08:02:00.000000000
7 2015/06/01 08:44:00.000000000
8 2015/06/01 13:35:00.000000000
9 2015/06/01 14:23:00.000000000
10 2015/06/01 15:28:00.000000000
11 2015/06/01 17:31:00.000000000
12 2015/06/01 18:45:00.000000000
13 2015/06/01 19:51:00.000000000
14 2015/06/01 20:23:00.000000000

**Examine preview data (Calculator)**

departure	year_id	month_id	day_id
1 2015/06/01 01:26:00.000000000	2015	6	1
2 2015/06/01 03:04:00.000000000	2015	6	1
3 2015/06/01 03:04:00.000000000	2015	6	1
4 2015/06/01 07:26:00.000000000	2015	6	1
5 2015/06/01 07:35:00.000000000	2015	6	1
6 2015/06/01 08:02:00.000000000	2015	6	1
7 2015/06/01 08:44:00.000000000	2015	6	1
8 2015/06/01 13:35:00.000000000	2015	6	1
9 2015/06/01 14:23:00.000000000	2015	6	1
10 2015/06/01 15:28:00.000000000	2015	6	1
11 2015/06/01 17:31:00.000000000	2015	6	1
12 2015/06/01 18:45:00.000000000	2015	6	1
13 2015/06/01 19:51:00.000000000	2015	6	1
14 2015/06/01 20:23:00.000000000	2015	6	1
15 2015/06/01 21:16:00.000000000	2015	6	1

**Examine preview data (Value mapper)**

departure	year_id	month_id	day_id	month_name
1 2015/06/01 01:26:00.000000000	2015	6	1	Jun
2 2015/06/01 03:04:00.000000000	2015	6	1	Jun
3 2015/06/01 03:04:00.000000000	2015	6	1	Jun
4 2015/06/01 07:26:00.000000000	2015	6	1	Jun
5 2015/06/01 07:35:00.000000000	2015	6	1	Jun
6 2015/06/01 08:02:00.000000000	2015	6	1	Jun
7 2015/06/01 08:44:00.000000000	2015	6	1	Jun
8 2015/06/01 13:35:00.000000000	2015	6	1	Jun
9 2015/06/01 14:23:00.000000000	2015	6	1	Jun
10 2015/06/01 15:28:00.000000000	2015	6	1	Jun
11 2015/06/01 17:31:00.000000000	2015	6	1	Jun
12 2015/06/01 18:45:00.000000000	2015	6	1	Jun
13 2015/06/01 19:51:00.000000000	2015	6	1	Jun
14 2015/06/01 20:23:00.000000000	2015	6	1	Jun

**Examine preview data (Insert / update)**

departure	year_id	month_id	day_id
1 2015/06/01 01:26:00.000000000	2015	6	1
2 2015/06/01 03:04:00.000000000	2015	6	1
3 2015/06/01 03:04:00.000000000	2015	6	1
4 2015/06/01 07:26:00.000000000	2015	6	1
5 2015/06/01 07:35:00.000000000	2015	6	1
6 2015/06/01 08:02:00.000000000	2015	6	1
7 2015/06/01 08:44:00.000000000	2015	6	1
8 2015/06/01 13:35:00.000000000	2015	6	1
9 2015/06/01 14:23:00.000000000	2015	6	1
10 2015/06/01 15:28:00.000000000	2015	6	1
11 2015/06/01 17:31:00.000000000	2015	6	1
12 2015/06/01 18:45:00.000000000	2015	6	1
13 2015/06/01 19:51:00.000000000	2015	6	1
14 2015/06/01 20:23:00.000000000	2015	6	1

*dim\_airport*

(note: dim\_origin and dim\_destination were replaced by dim\_airport)

The image shows two configuration windows from a data tool. The 'Table input' window on the left is for a step named 'Table input airport\_geo' connected to 'airports'. It contains an SQL query: 

```
SELECT airport_id, name, city, country FROM airports.airport_geo
```

. The 'Insert / update' window on the right is for a step named 'insert / update' connected to 'airports\_dw'. It shows the target schema 'airports\_dw' and target table 'dim\_airport'. A commit size of 1000 is set. A key is defined with 'AIRPORT\_ID' as the table field and 'airport\_id' as the stream field. Update fields are mapped: 'AIRPORT\_ID' to 'airport\_id', 'AIRPORT\_NAME' to 'name', 'CITY' to 'city', and 'COUNTRY' to 'country'.

The image displays two 'Examine preview data' windows. The left window shows the first 21 rows of the 'Table input airport\_geo' step, which are identical to the first 21 rows of the 'Insert / update' step shown in the right window. Both tables have columns: airport\_id, name, city, and country.

airport_id	name	city	country
1	A CORUNA	A CORUNA	SPAIN
2	AALBORG	AALBORG	DENMARK
3	ABBEYSHRULE	ABBEYSHRULE	IRELAND
4	AGRINION AB	AGRINION	GREECE
5	AIX-LES-BAINS	CHAMBERY	FRANCE
6	AKROTIRI AB	AKROTIRI	CYPRUS
7	ALBENGA	ALBENGA	ITALY
8	ALVERCA AB	ALVERCA	PORTUGAL
9	732 BABIMOST	ZIELONA GORA	POLAND
10	748 BAD DURKHEIM	BAD DURKHEIM	GERMANY
11	815 BALATON	SARMELLEK	HUNGARY
12	1085 BELMONT	ST AFRIQUE	FRANCE
13	1241 BIRR	BIRR	IRELAND
14	1286 BLAUBEUREN	BLAUBEUREN	GERMANY
15	1343 BOHUNOVICE	BOHUNOVICE	CZECH
16	1508 BRASSCHAAT ARMY	BRASSCHAAT	BELGIUM
17	1552 BRILON/ HOCHSAUERLAND	BRILON/ HOCHSAUERLAND	GERMANY
18	1572 BRON	LYON	FRANCE
19	1595 BROUMOV	BROUMOV	CZECH
20	1624 BUCKEBURG ARMY	BUCKEBURG	GERMANY
21	1674 BURGAS	BURGAS	BULGARIA



fact\_flight

load\_dwload\_factdim\_airlinedim\_airplanedim\_arrivaldim\_departuredim\_destinedim\_originfact\_flight

100%

Table inputInsert / update

Table input

Step nameTable input

Connectionairports

Get SQL select statement...

SQL

SELECT  
flight\_id  
from  
to  
departure  
arrival  
airline\_id  
airplane\_id  
count(\*) as totalbookings  
sum(price) as revenue  
FROM airports.flight NATURAL JOIN airports.booking  
GROUP BY flight\_id

Line 7 Column 12

Store column info in step meta data

Enable lazy conversion

Replace variables in script?

Insert data from step

Execute for each row?

Limit size0

HelpOKPreviewCancel

Insert / update

Step nameInsert / update

Connectionairports\_dw

Target schemaairports\_dw

Target tablefact\_flight

Commit size1000

Don't perform any updates:

The key(s) to look up the value(s):

Table field	Comparator	Stream field1	Stream field2
1 FLIGHT_ID	=	flight_id	

Update fields:

Table field	Stream field	Update
1 FLIGHT_ID	flight_id	Y
2 ORIGIN_ID	from	Y
3 DESTINATION_ID	to	Y
4 DEPARTURE	departure	Y
5 ARRIVAL	arrival	Y
6 AIRLINE_ID	airline_id	Y
7 AIRPLANE_ID	airplane_id	Y
8 TOTALBOOKINGS	totalbookings	Y
9 REVENUE	revenue	Y

Get update fields

Edit mapping

HelpOKCancelSQL

Examine preview data									
Rows of step: Table input (1210 rows)									
	flight_id	from	to	departure	arrival	airline_id	airplane_id	totalbookings	revenue
1	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	6	1757.93
2	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	6	1794.15
3	1511	8591	1343	2015/06/01 18:45:00.000000000	2015/06/01 19:27:00.000000000	35	926	18	4166.23
4	1515	3420	748	2015/06/01 08:02:00.000000000	2015/06/01 08:22:00.000000000	35	902	4	651.13
5	1543	3887	10805	2015/06/01 20:23:00.000000000	2015/06/01 22:20:00.000000000	35	902	3	483.37
6	1871	1085	6973	2015/06/01 21:16:00.000000000	2015/06/01 22:26:00.000000000	44	2557	5	1544.67
7	1880	11812	6429	2015/06/01 01:26:00.000000000	2015/06/01 03:19:00.000000000	44	2564	17	3945.94
8	2117	1572	8829	2015/06/01 08:44:00.000000000	2015/06/01 09:11:00.000000000	49	1207	4	1347.69
9	2673	10564	6002	2015/06/01 22:50:00.000000000	2015/06/02 01:24:00.000000000	63	4869	21	4986.82
10	2703	4725	9838	2015/06/01 19:51:00.000000000	2015/06/01 21:18:00.000000000	63	4827	17	4490.16
11	2715	10904	2867	2015/06/01 17:31:00.000000000	2015/06/01 19:09:00.000000000	63	4840	6	1678.74
12	2717	8210	5024	2015/06/01 13:35:00.000000000	2015/06/01 14:11:00.000000000	63	4869	20	4245.1
13	3359	12159	3796	2015/06/01 15:28:00.000000000	2015/06/01 17:39:00.000000000	77	2255	18	4450.93
14	3367	10903	3074	2015/06/01 03:04:00.000000000	2015/06/01 03:23:00.000000000	77	2254	5	1469.83
15	3793	2556	4430	2015/06/01 07:26:00.000000000	2015/06/01 08:59:00.000000000	87	4593	4	783.94

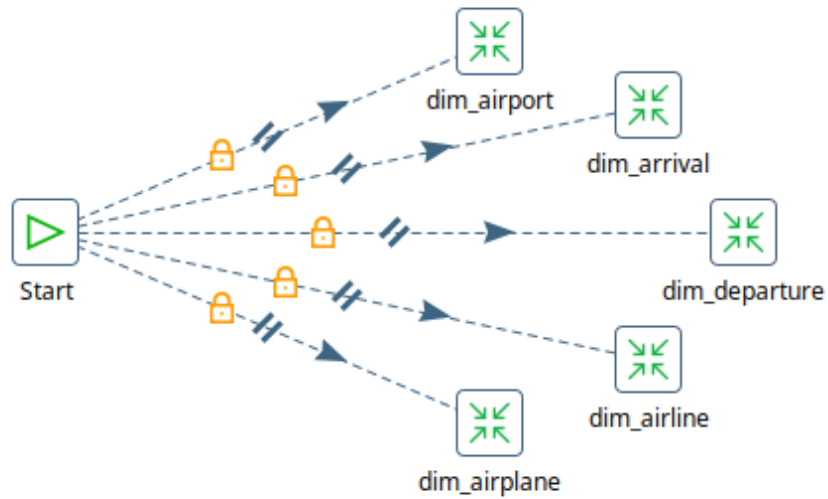
Close

Examine preview data									
Rows of step: Insert / update (1210 rows)									
	flight_id	from	to	departure	arrival	airline_id	airplane_id	totalbookings	revenue
1	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	6	1757.93
2	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	6	1794.15
3	1511	8591	1343	2015/06/01 18:45:00.000000000	2015/06/01 19:27:00.000000000	35	926	18	4166.23
4	1515	3420	748	2015/06/01 08:02:00.000000000	2015/06/01 08:22:00.000000000	35	902	4	651.13
5	1543	3887	10805	2015/06/01 20:23:00.000000000	2015/06/01 22:20:00.000000000	35	902	3	483.37
6	1871	1085	6973	2015/06/01 21:16:00.000000000	2015/06/01 22:26:00.000000000	44	2557	5	1544.67
7	1880	11812	6429	2015/06/01 01:26:00.000000000	2015/06/01 03:19:00.000000000	44	2564	17	3945.94
8	2117	1572	8829	2015/06/01 08:44:00.000000000	2015/06/01 09:11:00.000000000	49	1207	4	1347.69
9	2673	10564	6002	2015/06/01 22:50:00.000000000	2015/06/02 01:24:00.000000000	63	4869	21	4986.82
10	2703	4725	9838	2015/06/01 19:51:00.000000000	2015/06/01 21:18:00.000000000	63	4827	17	4490.16
11	2715	10904	2867	2015/06/01 17:31:00.000000000	2015/06/01 19:09:00.000000000	63	4840	6	1678.74
12	2717	8210	5024	2015/06/01 13:35:00.000000000	2015/06/01 14:11:00.000000000	63	4869	20	4245.1
13	3359	12159	3796	2015/06/01 15:28:00.000000000	2015/06/01 17:39:00.000000000	77	2255	18	4450.93
14	3367	10903	3074	2015/06/01 03:04:00.000000000	2015/06/01 03:23:00.000000000	77	2254	5	1469.83
15	3793	2556	4430	2015/06/01 07:26:00.000000000	2015/06/01 08:59:00.000000000	87	4593	4	783.94

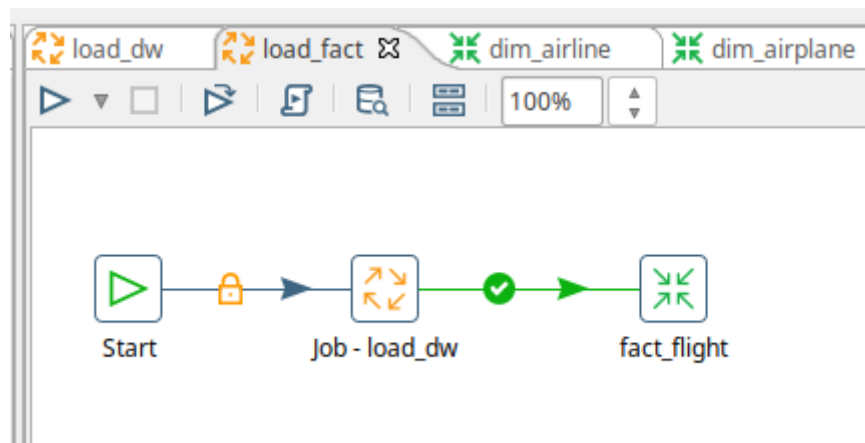
Close

## Jobs

*load\_dw* – loads dimension in parallel



*load\_fact* – loads fact\_flight table after all dimensions loaded



---

## 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" foreignKey="ORIGIN_ID"
highCardinality="false" name="Origin">
      <Hierarchy name="Origin Hierarchy" visible="true" hasAll="true"
allMemberName="AllOrigins" primaryKey="AIRPORT_ID">
        <Table name="dim_airport">
        </Table>
        <Level name="Country" visible="true" column="COUNTRY" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="City" visible="true" column="CITY" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="Airport Name" visible="true" column="AIRPORT_NAME" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true" foreignKey="DESTINATION_ID"
highCardinality="false" name="Destination">
      <Hierarchy name="Destination Hierarchy" visible="true" hasAll="true"
allMemberName="All Destinations" primaryKey="AIRPORT_ID">
        <Table name="dim_airport">
        </Table>
        <Level name="Country" visible="true" column="COUNTRY" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="City" visible="true" column="CITY" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="Airport Name" visible="true" column="AIRPORT_NAME" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="TimeDimension" visible="true" foreignKey="DEPARTURE"
highCardinality="false" name="Departure">
      <Hierarchy name="Departure Hierarchy" visible="true" hasAll="true"
allMemberName="All Departures" primaryKey="TIME_ID">
        <Table name="dim_departure">
        </Table>
        <Level name="Year" visible="true" column="YEAR_ID" type="Integer"
uniqueMembers="false" levelType="TimeYears" hideMemberIf="Never">
        </Level>
```

```

        <Level name="Months" visible="true" column="MONTH_NAME"
ordinalColumn="MONTH_ID" type="String" uniqueMembers="false" levelType="TimeMonths"
hideMemberIf="Never">
        </Level>
        <Level name="Days" visible="true" column="DAY_ID" type="Integer"
uniqueMembers="false" levelType="TimeDays" hideMemberIf="Never">
        </Level>
    </Hierarchy>
</Dimension>
    <Dimension type="TimeDimension" visible="true" foreignKey="ARRIVAL"
highCardinality="false" name="Arrival">
        <Hierarchy name="Arrival Hierarchy" visible="true" hasAll="true"
allMemberName="All Arrivals" primaryKey="TIME_ID">
            <Table name="dim_arrival">
            </Table>
            <Level name="Year" visible="true" column="YEAR_ID" type="Integer"
uniqueMembers="false" levelType="TimeYears" hideMemberIf="Never">
            </Level>
            <Level name="Months" visible="true" column="MONTH_NAME"
ordinalColumn="MONTH_ID" type="String" uniqueMembers="false" levelType="TimeMonths"
hideMemberIf="Never">
            </Level>
            <Level name="Days" visible="true" column="DAY_ID" type="Integer"
uniqueMembers="false" levelType="TimeDays" hideMemberIf="Never">
            </Level>
        </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true" foreignKey="AIRPLANE_ID"
highCardinality="false" name="Airplane">
        <Hierarchy name="Airplane Hierarchy" visible="true" hasAll="true"
allMemberName="All Airplanes" primaryKey="AIRPLANE_ID">
            <Table name="dim_airplane">
            </Table>
            <Level name="Type" visible="true" column="AIRPLANE_TYPE" type="Integer"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
            </Level>
        </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true" foreignKey="AIRLINE_ID"
highCardinality="false" name="Airlines">
        <Hierarchy name="Airline Hierarchy" visible="true" hasAll="true"
allMemberName="All Airlines" primaryKey="AIRLINE_ID">
            <Table name="dim_airline">
            </Table>
            <Level name="Name" visible="true" column="AIRLINE_NAME" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
            </Level>
        </Hierarchy>
    </Dimension>

```

---

```
<Measure name="Bookings" column="TOTALBOOKINGS" datatype="Integer"
formatString="#,###" aggregator="sum" visible="true">
  </Measure>
  <Measure name="Revenue" column="REVENUE" datatype="Integer" formatString="$
#,###.00" aggregator="sum" visible="true">
    </Measure>
  </Cube>
</Schema>
```

# Saiku Analysis

airports.sql

passengers and revenue by airline and month

Measures

Bookings

Revenue

Columns

Arrival Hierarchy

Months

Rows

Airline Hierarchy

Name

Filter

Months	Jun		Jul		Aug	
Name	Bookings	Revenue	Bookings	Revenue	Bookings	Revenue
Bulgaria Airlines	449	\$ 110,113.24	587	\$ 154,437.99	-	-
Croatia Airlines	436	\$ 109,007.19	489	\$ 122,529.28	-	-
Cyprus Airlines	291	\$ 76,905.68	379	\$ 98,558.57	10	\$ 2,127.63
Czech Airlines	372	\$ 98,762.85	426	\$ 103,871.69	-	-
Denmark Airlines	187	\$ 46,222.46	236	\$ 55,558.29	-	-
Estonia Airlines	405	\$ 107,981.78	530	\$ 133,757.98	-	-
France Airlines	772	\$ 191,586.52	752	\$ 191,868.27	-	-
Greece Airlines	204	\$ 50,663.94	179	\$ 47,767.02	-	-
Hungary Airlines	1,089	\$ 274,893.57	1,143	\$ 286,874.33	-	-
Italy Airlines	165	\$ 42,343.58	210	\$ 56,486.71	-	-
Luxembourg Airlines	1,262	\$ 314,355.07	1,204	\$ 308,794.45	18	\$ 4,498.30
Poland Airlines	552	\$ 138,618.84	598	\$ 146,253.93	-	-
Slovakia Airlines	341	\$ 82,589.22	369	\$ 91,144.70	-	-
Spain Airlines	181	\$ 44,725.48	149	\$ 37,702.61	-	-

14

airplane type and month of departure by month of arrival  
(2 airplane types departed in July and arrived in August)

Measures

Bookings

Columns

Arrival Hierarchy

Months

Rows

Airplane Hierarchy

Type

Departure Hierarchy

Months

Filter

Months		Jun	Jul	Aug
Type	Months	Bookings	Bookings	Bookings
6	Jun	1,354	-	-
	Jul	-	1,597	-
18	Jun	867	-	-
	Jul	-	931	-
21	Jun	885	-	-
	Jul	-	1,148	-
38	Jun	272	-	-
	Jul	-	219	10
40	Jun	129	-	-
	Jul	-	129	-
41	Jun	358	-	-
	Jul	-	343	-
48	Jun	328	-	-
	Jul	-	338	-
60	Jun	566	-	-
	Jul	-	731	18
75	Jun	135	-	-
	Jul	-	184	-
228	Jun	360	-	-
	Jul	-	361	-
232	Jun	129	-	-
	Jul	-	137	-
301	Jun	1,108	-	-
	Jul	-	976	-
316	Jun	215	-	-
	Jul	-	157	-



how much money spent/gained on flights between countries  
(on the 10 days with highest revenue)

Info: 19:32 / 13 x 17 / 0.10s

Measures

Revenue

Columns

Destination Hierarchy

Country

Rows

Origin Hierarchy

Country

Filter

Departure Hierarchy

Days

Country	BELGIUM	CROATIA	CZECH	DENMARK	FINLAND	FRANCE	GERMANY	IRELAND	ITALY	POLAND	SPAIN	SWEDEN
Country	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue
AUSTRIA	-	-	-	-	-	-	-	-	-	\$ 27,681.59	-	-
BELGIUM	-	-	-	-	-	-	\$ 7,676.85	-	-	-	-	-
CZECH	-	-	-	\$ 29,277.79	-	\$ 18,030.66	\$ 31,174.96	\$ 12,139.46	-	-	-	-
DENMARK	-	-	-	-	-	-	-	-	\$ 5,285.41	-	-	-
ESTONIA	-	-	\$ 12,509.64	-	-	-	-	-	-	-	-	-
FINLAND	-	-	-	-	-	-	\$ 45,667.59	-	-	-	-	-
FRANCE	-	-	-	\$ 5,987.47	\$ 9,350.63	\$ 17,511.40	\$ 88,297.39	-	\$ 9,586.81	-	-	\$ 28,313.37
GERMANY	\$ 7,870.68	\$ 18,412.66	\$ 5,029.42	\$ 37,271.34	\$ 35,959.23	\$ 8,873.90	\$ 53,874.95	-	-	-	-	\$ 34,421.65
GREECE	-	-	-	-	-	\$ 4,880.61	-	-	\$ 12,256.94	-	-	-
ITALY	-	-	-	-	-	\$ 9,363.88	\$ 12,908.67	-	-	-	\$ 5,490.45	-
POLAND	-	-	-	-	-	\$ 17,284.92	-	\$ 11,700.39	-	-	-	-
PORTUGAL	-	-	-	-	-	\$ 44,885.51	-	-	-	-	-	-
ROMANIA	-	-	-	-	\$ 12,032.21	-	-	-	-	\$ 6,261.86	-	-
SPAIN	-	-	-	-	-	-	\$ 35,503.45	-	-	-	-	-
SWEDEN	-	-	-	-	-	-	\$ 12,165.32	-	-	\$ 17,427.84	-	-

## airports-large.sql

passengers and revenue by airline and month

**Measures** ▼

Bookings

Revenue

**Columns** ▼

Arrival Hierarchy

Months

**Rows** ▼

Airline Hierarchy

Name

**Filter** ▼

Months	Jun		Jul		Aug	
Name	Bookings	Revenue	Bookings	Revenue	Bookings	Revenue
Afghanistan Airlines	15,352	\$ 3,864,611.80	16,118	\$ 4,062,950.72	318	\$ 81,725.14
Albania Airlines	15,951	\$ 3,991,510.15	17,053	\$ 4,308,360.83	229	\$ 57,998.03
American Samoa Airli	11,252	\$ 2,824,533.49	12,546	\$ 3,134,302.87	102	\$ 23,804.54
Angola Airlines	13,650	\$ 3,425,851.18	14,893	\$ 3,740,791.82	126	\$ 31,979.18
Argentina Airlines	15,661	\$ 3,947,144.14	17,098	\$ 4,309,724.79	274	\$ 67,685.16
Australia Airlines	14,726	\$ 3,686,163.92	15,853	\$ 3,987,009.91	271	\$ 65,439.72
Azerbaijan Airlines	11,636	\$ 2,892,437.23	12,449	\$ 3,138,111.32	206	\$ 49,668.41
Bahamas Airlines	16,613	\$ 4,171,537.98	17,918	\$ 4,513,922.77	135	\$ 35,741.87
Belarus Airlines	13,546	\$ 3,412,206.46	14,084	\$ 3,524,337.33	199	\$ 52,043.62
Bhutan Airlines	16,876	\$ 4,217,204.21	18,143	\$ 4,561,337.77	337	\$ 88,020.20
Bolivia Airlines	10,035	\$ 2,514,974.14	10,310	\$ 2,578,467.28	141	\$ 31,541.84
Brazil Airlines	18,264	\$ 4,590,913.75	18,679	\$ 4,698,229.33	268	\$ 65,752.29
Bulgaria Airlines	14,355	\$ 3,615,409.70	15,065	\$ 3,805,913.52	204	\$ 51,009.91
Caicos Is Airlines	13,899	\$ 3,504,475.25	13,677	\$ 3,391,208.05	176	\$ 43,858.96
Central African Rep	15,935	\$ 4,004,911.42	16,829	\$ 4,224,236.25	191	\$ 50,894.75
Chad Airlines	13,582	\$ 3,411,597.28	14,500	\$ 3,635,622.05	345	\$ 83,324.37
Colombia Airlines	13,396	\$ 3,362,594.33	14,495	\$ 3,665,361.21	157	\$ 40,559.92
Croatia Airlines	17,832	\$ 4,461,904.52	18,804	\$ 4,689,909.55	216	\$ 52,837.11
Cuba Airlines	14,201	\$ 3,553,389.12	14,849	\$ 3,716,627.79	89	\$ 24,691.91
Cyprus Airlines	18,091	\$ 4,534,422.14	19,375	\$ 4,875,355.50	306	\$ 78,392.56
Czech Airlines	14,426	\$ 3,613,421.07	15,658	\$ 3,951,107.66	173	\$ 44,100.96
Dakhla And Laayoune	14,152	\$ 3,560,507.07	14,659	\$ 3,676,618.99	278	\$ 68,369.97
Denmark Airlines	12,515	\$ 3,159,702.20	13,695	\$ 3,420,967.75	222	\$ 57,921.59
Djibouti Airlines	14,562	\$ 3,627,921.07	15,373	\$ 3,861,495.67	229	\$ 53,245.86
Dominica Airlines	11,570	\$ 2,900,001.11	11,594	\$ 2,909,360.74	114	\$ 28,687.23
Ecuador Airlines	15,418	\$ 3,886,397.82	16,016	\$ 4,029,561.24	240	\$ 59,310.63

airplane type and month of departure by month of arrival  
(2 airplane types departed in July and arrived in August)

Measures
Bookings
Columns
Arrival Hierarchy
Months
Rows
Airplane Hierarchy
Type
Departure Hierarchy
Months
Filter

	Months	Jun	Jul	Aug
Type	Months	Bookings	Bookings	Bookings
6	Jun	400,251	5,637	-
	Jul	-	403,462	5,273
18	Jun	235,757	3,324	-
	Jul	-	246,377	3,551
21	Jun	183,671	2,589	-
	Jul	-	189,362	2,534
	Aug	-	-	18
38	Jun	101,738	1,418	-
	Jul	-	106,392	1,701
	Aug	-	-	9
40	Jun	40,540	556	-
	Jul	-	41,693	512
41	Jun	58,249	770	-
	Jul	-	60,229	859
48	Jun	70,907	1,137	-
	Jul	-	73,456	968
60	Jun	220,202	3,451	-
	Jul	-	230,522	3,297
	Aug	-	-	18
75	Jun	50,143	700	-
	Jul	-	51,798	615
	Aug	-	-	11
228	Jun	78,211	963	-
	Jul	-	79,969	1,154
232	Jun	27,997	420	-
	Jul	-	28,914	427

how much money spent/gained on flights between countries  
(on the 10 days with highest revenue)

Measures

Revenue

Columns

Destination Hierarchy

Country

Rows

Origin Hierarchy

Country

Filter

Departure Hierarchy

Days

Info: 21.04 / 208 x 205 / 12.01s

Country	AFGHANISTAN	ALGERIA	AMERICAN SAMOA	ANGOLA	ANGUILLA	ANTARCTICA (ARG)	ANTARCTICA (AUS)	ARGENTINA	ARMENIA	ARUBA	AUSTRALIA	AUSTRIA	AZORES	BAHAMAS
Country	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue
AFGHANISTAN	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ALGERIA	-	-	-	-	-	-	-	\$ 40,777.68	-	-	-	-	-	-
AMERICAN SAMOA	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ANGOLA	-	-	-	-	-	-	-	-	-	-	\$ 26,343.92	-	-	-
ANTARCTICA (ARG)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ANTARCTICA (AUS)	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ANTIGUA	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ARGENTINA	-	-	-	-	-	-	-	\$ 34,099.53	-	-	-	-	-	-
AUSTRALIA	-	-	-	-	-	-	-	-	-	-	\$ 184,946.12	\$ 96,013.80	-	-
AUSTRIA	-	-	-	-	-	-	-	-	-	-	\$ 39,074.15	-	-	-
AZERBAIJAN	-	-	-	-	-	-	-	-	-	-	-	-	-	-
AZORES	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BAHAMAS	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BANGLADESH	-	-	-	-	-	-	-	-	-	-	\$ 55,224.25	-	-	-
BELARUS	-	\$ 40,169.99	-	-	-	-	-	-	-	-	-	-	-	-
BELGIUM	-	-	-	-	-	-	-	-	-	-	\$ 34,792.45	-	-	-
BELIZE	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BERMUDA	-	-	-	-	-	\$ 9,619.12	-	-	-	-	-	-	-	-
BHUTAN	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BOLIVIA	\$ 3,503.40	-	-	-	-	-	-	\$ 32,199.49	-	-	-	-	-	-
BOSNIA AND HERZEGOVINA	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BOTSWANA	-	-	-	-	-	-	-	-	-	-	\$ 16,846.47	\$ 23,076.19	-	-
BRAZIL	\$ 67,377.16	\$ 7,835.24	-	\$ 35,234.85	-	-	-	\$ 94,040.89	-	-	-	\$ 43,673.73	\$ 27,334.74	-
BRITISH VIRGIN IS	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BULGARIA	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BURKINA FASO	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BURUNDI	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CAICOS IS	-	\$ 9,643.50	-	-	-	-	-	-	-	-	-	-	-	-

## airports-large-extra.sql

passengers and revenue by airline and month

Measures											
Bookings											
Revenue											
Columns											
Arrival Hierarchy											
Months											
Rows											
Airline Hierarchy											
Name											
Filter											

Months		Jun		Jul		Aug		Sep	
Name		Bookings	Revenue	Bookings	Revenue	Bookings	Revenue	Bookings	Revenue
Afghanistan Airlines		153,531	\$ 38,484,790.79	161,293	\$ 40,472,054.67	162,391	\$ 40,865,554.24	8,124	\$ 2,039,445.54
Albania Airlines		159,262	\$ 39,897,663.61	170,623	\$ 42,827,184.84	169,506	\$ 42,678,075.18	9,508	\$ 2,408,284.91
American Samoa Airli		112,472	\$ 28,308,790.69	125,591	\$ 31,504,222.81	117,037	\$ 29,368,785.09	4,395	\$ 1,090,446.04
Angola Airlines		136,425	\$ 34,347,453.81	148,966	\$ 37,385,955.74	139,364	\$ 35,013,009.79	6,126	\$ 1,524,536.54
Argentina Airlines		156,706	\$ 39,433,100.48	171,009	\$ 42,933,031.41	158,448	\$ 39,820,229.90	7,249	\$ 1,792,335.79
Australia Airlines		147,264	\$ 36,895,382.74	158,391	\$ 39,717,352.17	154,653	\$ 38,775,426.65	7,486	\$ 1,881,800.70
Azerbaijan Airlines		116,404	\$ 29,158,202.24	124,498	\$ 31,339,897.35	124,711	\$ 31,326,210.56	6,458	\$ 1,613,125.74
Bahamas Airlines		166,189	\$ 41,667,568.68	179,311	\$ 45,011,524.23	170,440	\$ 42,797,620.62	8,441	\$ 2,124,317.51
Belarus Airlines		135,549	\$ 34,147,753.07	140,954	\$ 35,328,487.03	142,070	\$ 35,654,752.12	6,232	\$ 1,563,137.63
Bhutan Airlines		168,839	\$ 42,317,956.17	181,527	\$ 45,582,515.26	175,595	\$ 44,040,959.30	10,125	\$ 2,546,693.14
Bolivia Airlines		100,221	\$ 25,079,321.56	103,110	\$ 25,822,166.50	103,653	\$ 26,018,204.52	5,146	\$ 1,296,724.35
Brazil Airlines		182,562	\$ 45,811,242.30	186,589	\$ 46,786,121.53	186,863	\$ 46,921,737.45	9,054	\$ 2,254,099.02
Bulgaria Airlines		143,482	\$ 36,061,920.02	150,467	\$ 37,736,376.11	145,333	\$ 36,439,479.41	6,509	\$ 1,654,183.65
Caicos Is Airlines		139,123	\$ 34,931,721.84	136,801	\$ 34,289,397.05	136,614	\$ 34,262,735.45	6,519	\$ 1,629,706.29
Central African Rep		159,303	\$ 40,021,712.99	168,365	\$ 42,125,765.54	171,976	\$ 43,124,600.90	9,089	\$ 2,271,431.95
Chad Airlines		135,782	\$ 34,072,100.08	144,879	\$ 36,371,560.39	145,014	\$ 36,362,031.39	6,079	\$ 1,536,691.94
Colombia Airlines		133,934	\$ 33,595,180.27	144,968	\$ 36,335,348.56	144,710	\$ 36,238,243.43	7,400	\$ 1,878,148.08
Croatia Airlines		178,186	\$ 44,646,913.61	188,029	\$ 47,132,982.06	187,251	\$ 46,927,131.22	9,023	\$ 2,296,912.56
Cuba Airlines		142,058	\$ 35,617,149.64	148,646	\$ 37,354,343.65	148,108	\$ 37,196,850.77	6,600	\$ 1,651,337.72
Cyprus Airlines		180,906	\$ 45,455,559.53	193,990	\$ 48,758,424.97	199,919	\$ 50,205,078.87	8,339	\$ 2,089,902.55
Czech Airlines		144,527	\$ 36,249,272.89	156,292	\$ 39,263,017.41	150,664	\$ 37,883,903.19	7,735	\$ 1,936,701.94
Dakhla And Laayoune		141,785	\$ 35,557,711.65	146,538	\$ 36,769,620.16	152,244	\$ 38,203,020.00	9,035	\$ 2,266,459.92
Denmark Airlines		125,033	\$ 31,357,361.07	136,844	\$ 34,271,543.51	130,576	\$ 32,805,019.70	5,613	\$ 1,412,421.64
Djibouti Airlines		145,687	\$ 36,506,386.56	153,697	\$ 38,629,889.57	153,866	\$ 38,656,025.07	5,937	\$ 1,491,655.31
Dominica Airlines		115,913	\$ 29,190,379.84	115,808	\$ 29,054,151.34	120,969	\$ 30,433,939.52	5,332	\$ 1,332,572.25
Ecuador Airlines		154,307	\$ 38,672,659.24	160,267	\$ 40,123,529.57	165,385	\$ 41,497,231.40	8,789	\$ 2,194,314.47
Egypt Airlines		152,584	\$ 38,312,427.95	157,085	\$ 39,469,676.85	155,740	\$ 39,032,374.87	7,361	\$ 1,844,706.61
El Salvador Airlines		152,013	\$ 38,186,942.96	157,311	\$ 39,541,539.00	162,757	\$ 40,862,191.03	6,817	\$ 1,713,911.04
Equatorial Guinea Ai		175,440	\$ 44,094,510.06	178,659	\$ 44,785,869.08	185,993	\$ 46,608,700.52	9,920	\$ 2,472,782.74
Eritrea Airlines		162,910	\$ 40,821,913.60	163,044	\$ 41,013,770.62	166,612	\$ 41,826,120.76	8,730	\$ 2,170,624.26

airplane type and month of departure by month of arrival  
(2 airplane types departed in July and arrived in August)

**Measures** ▼

Bookings

**Columns** ▼

Arrival Hierarchy  
 Months

**Rows** ▼

Airplane Hierarchy  
 Type

Departure Hierarchy  
 Months

**Filter** ▼

	Months	Jun	Jul	Aug	Sep
Type	Months	Bookings	Bookings	Bookings	Bookings
6	Jun	4,002,690	56,319	-	-
	Jul	-	4,035,047	52,740	-
	Aug	-	-	4,103,815	59,872
	Sep	-	-	-	149,922
18	Jun	2,358,157	33,193	-	-
	Jul	-	2,463,421	35,468	-
	Aug	-	-	2,393,324	38,065
	Sep	-	-	-	81,851
21	Jun	1,837,798	25,875	-	-
	Jul	-	1,893,661	25,333	-
	Aug	-	-	1,917,024	27,036
	Sep	-	-	-	59,000
38	Jun	1,017,201	14,128	-	-
	Jul	-	1,064,500	16,928	-
	Aug	-	-	1,065,543	11,980
	Sep	-	-	-	36,378
40	Jun	405,278	5,555	-	-
	Jul	-	417,154	5,138	-
	Aug	-	-	420,771	6,087
	Sep	-	-	-	14,106
41	Jun	582,638	7,710	-	-
	Jul	-	602,124	8,583	-
	Aug	-	-	601,477	7,684
	Sep	-	-	-	19,412
48	Jun	709,642	11,393	-	-
	Jul	-	734,217	9,694	-
	Aug	-	-	728,225	9,743
	Sep	-	-	-	26,203
60	Jun	2,201,963	34,508	-	-
	Jul	-	2,304,932	32,936	-

how much money spent/gained on flights between countries  
(on the 10 days with highest revenue)

Measures

Revenue

Columns

Destination Hierarchy

Country

Rows

Origin Hierarchy

Country

Filter

Departure Hierarchy

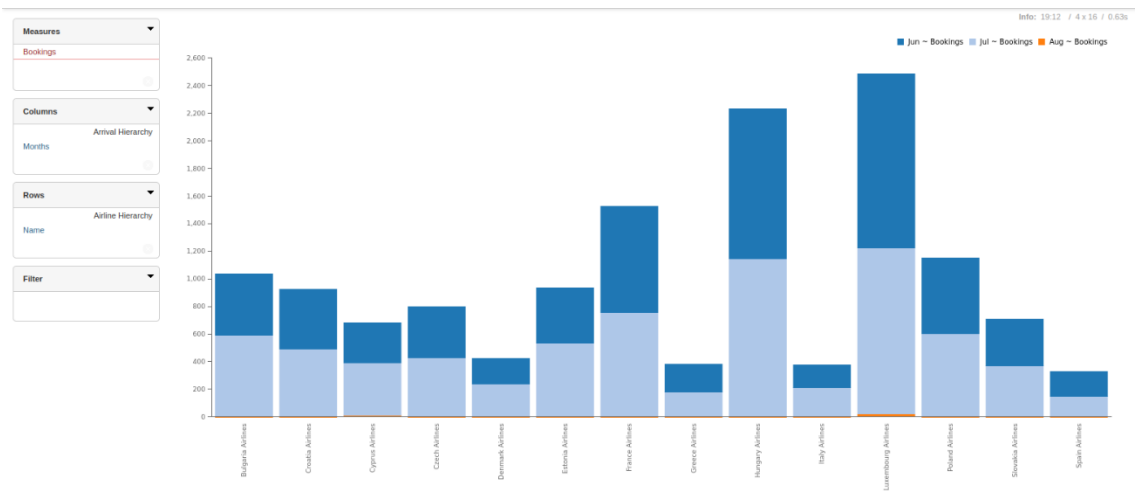
Days

Info: 00:44 / 208 x 205 / 33.06s

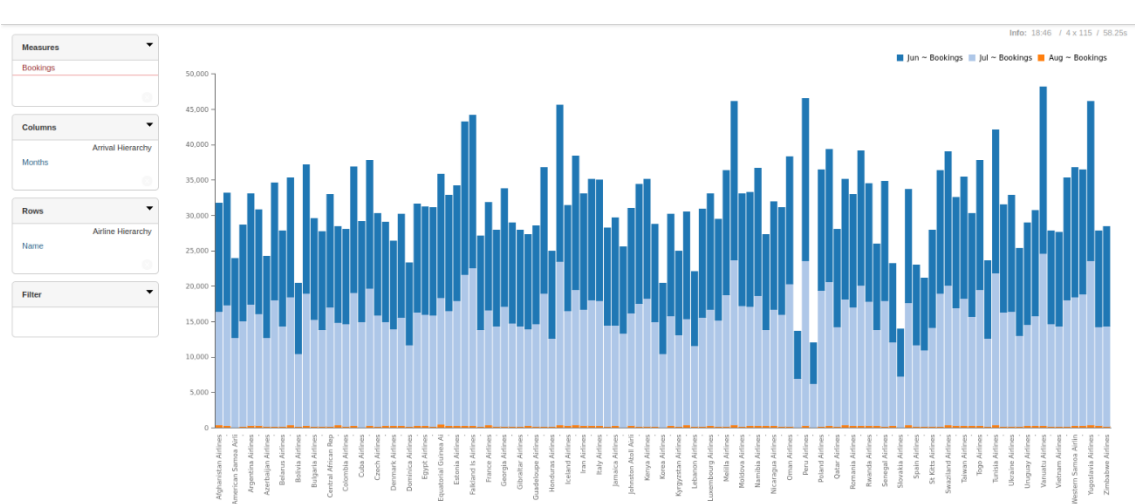
Country	AFGHANISTAN	ALGERIA	AMERICAN SAMOA	ANGOLA	ANGUILLA	ANTARCTICA (ARG)	ANTARCTICA (AUS)	ARGENTINA	ARMENIA	ARUBA	AUSTRALIA	AUSTRIA	AZORES	B
Country	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	Revenue	
AFGHANISTAN	-	-	-	-	-	-	-	-	-	-	-	-	-	
ALGERIA	-	-	-	-	-	-	-	\$ 386,102.32	-	-	-	-	-	
AMERICAN SAMOA	-	-	-	-	-	-	-	-	-	-	-	-	-	
ANGOLA	-	-	-	-	-	-	-	-	-	-	\$ 261,284.63	-	-	
ANTARCTICA (ARG)	-	-	-	-	-	-	-	-	-	-	-	-	-	
ANTARCTICA (AUS)	-	-	-	-	-	-	-	-	-	-	-	-	-	
ANTIGUA	-	-	-	-	-	-	-	-	-	-	-	-	-	
ARGENTINA	-	-	-	-	-	-	-	\$ 331,643.03	-	-	-	-	-	
AUSTRALIA	-	-	-	-	-	-	-	-	-	-	\$ 1,796,940.11	\$ 1,004,368.41	-	
AUSTRIA	-	-	-	-	-	-	-	-	-	-	\$ 465,168.73	-	-	
AZERBAIJAN	-	-	-	-	-	-	-	-	-	-	-	-	-	
AZORES	-	-	-	-	-	-	-	-	-	-	-	-	-	
BAHAMAS	-	-	-	-	-	-	-	-	-	-	-	-	-	
BANGLADESH	-	-	-	-	-	-	-	-	-	-	\$ 537,843.54	-	-	
BELARUS	-	\$ 356,253.74	-	-	-	-	-	-	-	-	-	-	-	
BELGIUM	-	-	-	-	-	-	-	-	-	-	\$ 332,507.47	-	-	
BELIZE	-	-	-	-	-	-	-	-	-	-	-	-	-	
BERMUDA	-	-	-	-	-	\$ 79,107.90	-	-	-	-	-	-	-	
BHUTAN	-	-	-	-	-	-	-	-	-	-	-	-	-	
BOLIVIA	\$ 15,098.57	-	-	-	-	-	-	\$ 261,156.05	-	-	-	-	-	
BOSNIA AND HERZEGOVINA	-	-	-	-	-	-	-	-	-	-	-	-	-	
BOTSWANA	-	-	-	-	-	-	-	-	-	-	\$ 246,207.10	\$ 239,058.34	-	
BRAZIL	\$ 706,268.94	\$ 69,465.56	-	\$ 361,301.95	-	-	-	\$ 961,243.61	-	-	-	\$ 375,928.95	\$ 314,347.42	
BRITISH VIRGIN IS	-	-	-	-	-	-	-	-	-	-	-	-	-	
BULGARIA	-	-	-	-	-	-	-	-	-	-	-	-	-	
BURKINA FASO	-	-	-	-	-	-	-	-	-	-	-	-	-	
BURUNDI	-	-	-	-	-	-	-	-	-	-	-	-	-	
CAICOS IS	-	\$ 47,779.15	-	-	-	-	-	-	-	-	-	-	-	

# Comparisons

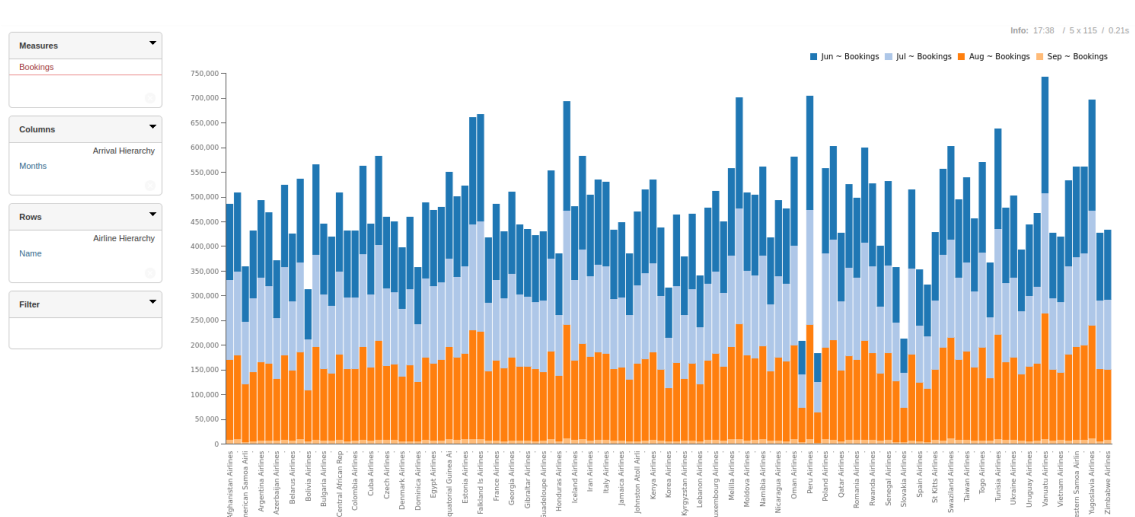
## airports.sql



## airports-large.sql



## airports-large-extra.sql





---

We decided to compare the 3 datasets using the first query from exercise 5, just using the bookings measure.

From the graphs, we can see that scale increases 20-fold between the first and second graphs, and 15-fold between the second and third graphs.

From the first to the second graphs, we see more airlines. From the second to the third graphs, there is around the same number of airlines, but a new month is added, September.

Thus, we can conclude that we are dealing with increasingly denser datasets.