AIRTRAFFIC

A database for managing aviation networks.

Laura Menotti
Albero Piva
Gianmarco Prando

Database 2 - A.Y. 2021/2022 - January 19th, 2022

Introduction

EUROCONTROL releases flight data for R&D purposes. The data source for flights is Eurocontrol Network Manager flights plans in PRISME Data Warehouse (DWH).

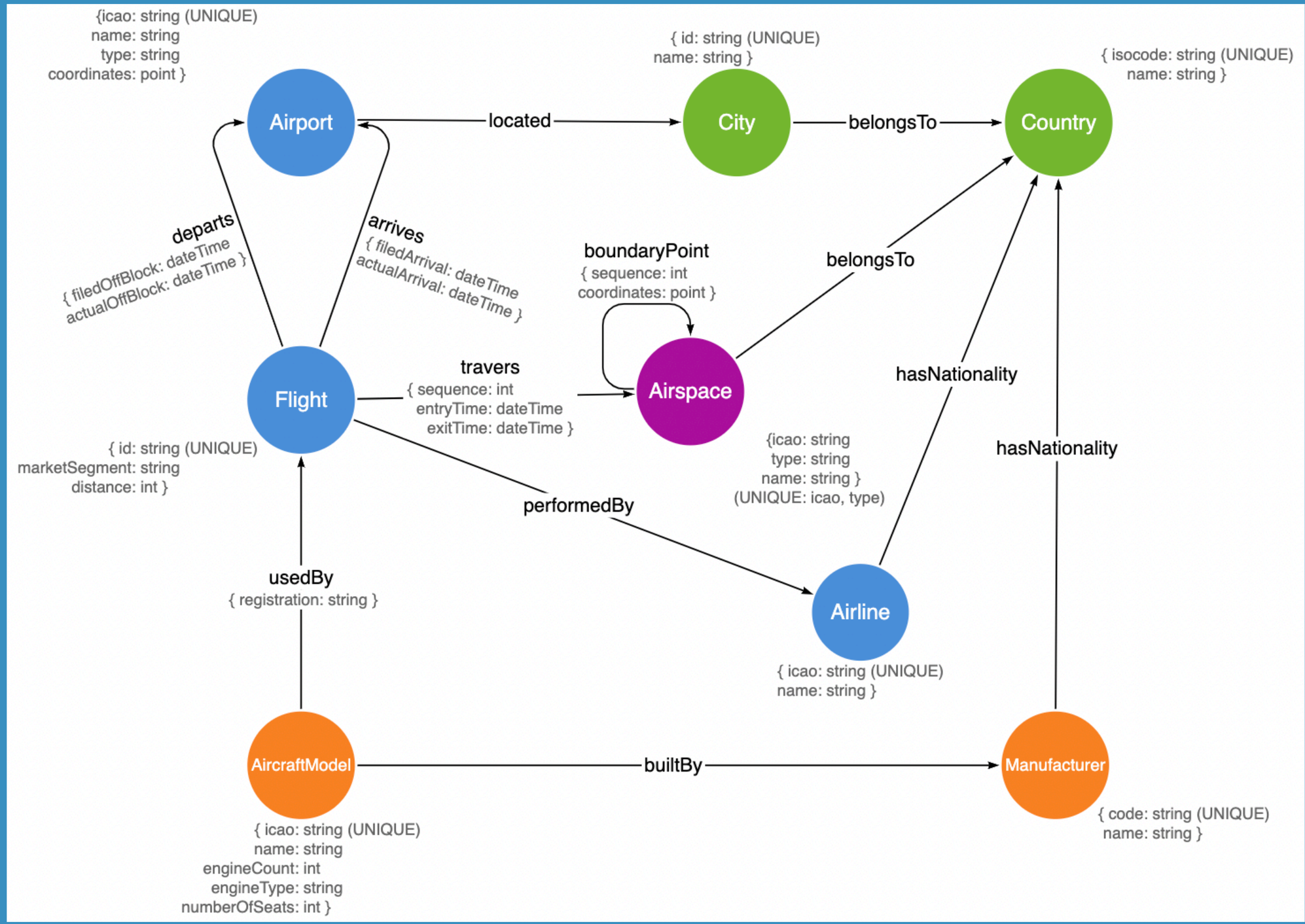
We model flights, airspaces, airports, airline companies and aircrafts in order to analyze the European air traffic.

Our database is intended for **people working in the aviation sector**, e.g. Airline companies or Airport Control Departments.

Aviation follows an international identification system for all entities, namely ICAO (*International Civil Aviation Organization*) codes.

What's New? Airports coordinates and Airspace data.

Data available at: https://www.eurocontrol.int/dashboard/rnd-data-archive


{icao: string (UNIQUE)
name: string
type: string
coordinates: point }
Airport
located
{ id: string (UNIQUE)
name: string }
City
belongsTo
{ isocode: string (UNIQUE)
name: string }
Country
departs
{ filedOffBlock: dateTime
actualOffBlock: dateTime }
arrives
{ filedArrival: dateTime
actualArrival: dateTime }
boundaryPoint
{ sequence: int
coordinates: point }
belongsTo
Flight
travers
{ sequence: int
entryTime: dateTime
exitTime: dateTime }
Airspace
hasNationality
{ id: string (UNIQUE)
marketSegment: string
distance: int }
{icao: string
type: string
name: string }
(UNIQUE: icao, type)
hasNationality
performedBy
usedBy
{ registration: string }
Airline
{ icao: string (UNIQUE)
name: string }
AircraftModel
builtBy
Manufacturer
{ icao: string (UNIQUE)
name: string
engineCount: int
engineType: string
numberOfSeats: int }
{ code: string (UNIQUE)
name: string }

What's New : Coordinates

Neo4j supports spatial geometry by means of the ***Point***.

Four Coordinate Reference Systems (CRS) are supported:

- ***Geographic coordinates*** modeling points on the earth (2D or 3D).
- ***Cartesian coordinates*** modeling points in euclidean space (2D or 3D).

We set airports and airspaces coordinates by means of the following instruction:

coordinates=point({latitude: $lat, longitude: $long, crs: 'wgs-84'})

Then we can access the coordinates components as instance properties.
(e.g. coordinates.latitude, coordinates.longitude)

What's New : Airspaces

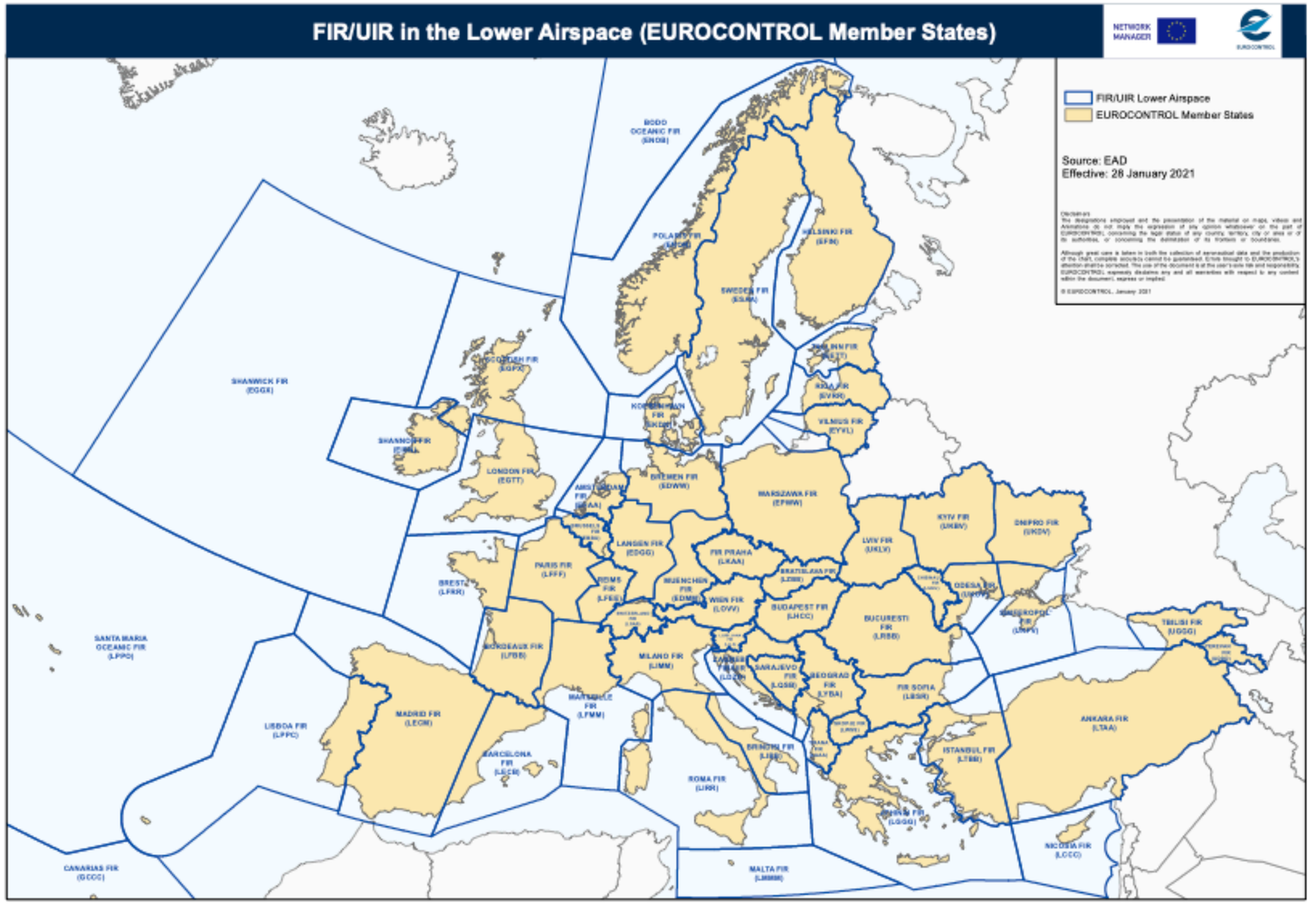
FIR (Flight Information Region)

Specified region of airspace in which a flight information service and an alerting service (ALRS) are provided.

FIRs are the largest regular division of airspace in use in the world today, and have existed at least since 1947.

UIR (Upper Information Region)

Flight Information Region in upper airspace.



European FIR map.

Data available at: https://www.eurocontrol.int/dashboard/rnd-data-archive

Flight_FIRs example

ECTRL ID	Sequence Number	FIR ID	Entry Time	Exit Time
227743256	0	TAXI_OUT	01-03-2019 00:08:00	01-03-2019 00:18:00
227743256	1	UNNTFIR	01-03-2019 00:18:00	01-03-2019 01:18:35
227743256	2	USTRFIR	01-03-2019 01:18:35	01-03-2019 01:49:25
227743256	3	USSSFIR	01-03-2019 01:49:25	01-03-2019 02:13:50
227743256	4	USPPFIR	01-03-2019 02:13:50	01-03-2019 02:38:23
227743256	5	USKKFIR	01-03-2019 02:38:23	01-03-2019 03:02:45
227743256	6	ULWWFIR	01-03-2019 03:02:45	01-03-2019 03:42:56
227743256	7	ULLLFIR	01-03-2019 03:42:56	01-03-2019 04:12:04
227743256	8	EETTFIR	01-03-2019 04:12:04	01-03-2019 04:40:12
227743256	9	EVRRFIR	01-03-2019 04:40:12	01-03-2019 04:49:34
227743256	10	ESAAFIR	01-03-2019 04:49:34	01-03-2019 05:18:31
227743256	11	EPWWFIR	01-03-2019 05:18:31	01-03-2019 05:20:13
227743256	12	EDUUUIR	01-03-2019 05:20:13	01-03-2019 05:37:38
227743256	13	EDVVUIR	01-03-2019 05:37:38	01-03-2019 05:52:40
227743256	14	EDGGFIR	01-03-2019 05:52:40	01-03-2019 06:10:55
227743256	15	EHAAFIR	01-03-2019 06:10:55	01-03-2019 06:16:54
227743256	16	TAXI_IN	01-03-2019 06:16:54	01-03-2019 06:26:54

from file: Flight_FIRs_Actual_20190301_20190331.csv

For each flight we have information about each FIR/UIR it traversed in terms of sequence number, entry time and exit time.

[from file: Flights_20190301_20190331.csv]

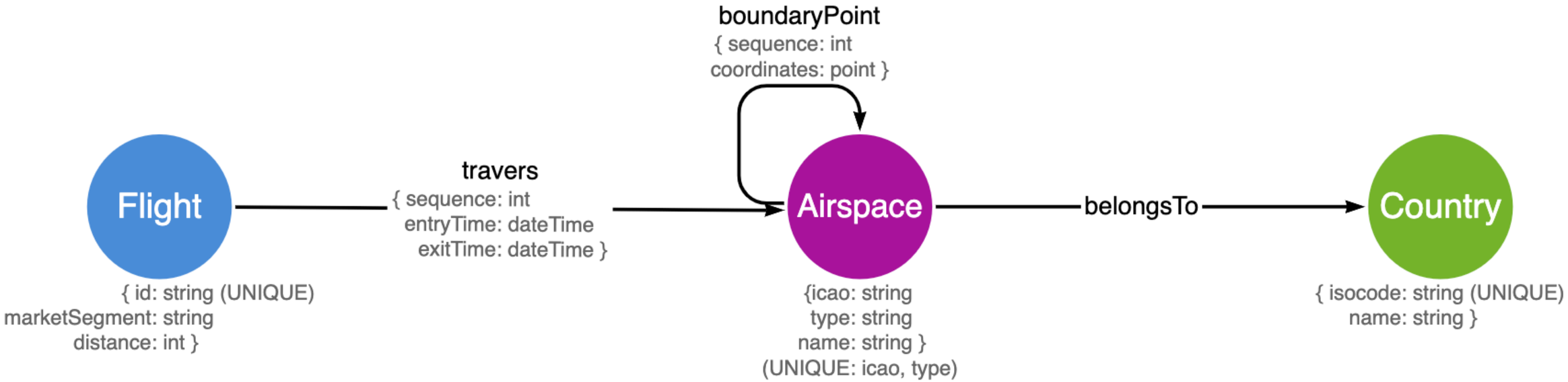
Departure: UNNT (*Tolmachevo Airport, Novosibirsk, Russia*)

Destination: EHBK (*Maastricht Aachen Airport, Maastricht, Netherlands*)

Operator: SAY (*ScotAirways, United Kingdom*)

Data available at: https://www.eurocontrol.int/dashboard/rnd-data-archive

Airspace Modelling



Airspace Boundary Points

Airspace ID	Min Flight Level	Max Flight Level	Sequence Number	Latitude	Longitude
BGGLFIR	0	999	1	89	-60
BGGLFIR	0	999	2	87	-60
BGGLFIR	0	999	3	85	-60
BGGLFIR	0	999	4	82	-60
BGGLFIR	0	999	5	80.18167	-69
BGGLFIR	0	999	6	78	-75
BGGLFIR	0	999	7	77.4	-75.08333
BGGLFIR	0	999	8	76.86	-75.60667
BGGLFIR	0	999	9	76.35	-75.83333
BGGLFIR	0	999	10	76	-76
BGGLFIR	0	999	11	75.82667	-75.45333
BGGLFIR	0	999	12	72.66111	-67.71333
BGGLFIR	0	999	13	71.06722	-64.92639
BGGLFIR	0	999	14	70.25	-63.69083
BGGLFIR	0	999	15	69.92111	-63.22
BGGLFIR	0	999	16	68.95944	-61.94111
BGGLFIR	0	999	17	67.98389	-60.76389
BGGLFIR	0	999	18	65	-57.75
BGGLFIR	0	999	19	63.5	-55.66667
BGGLFIR	0	999	20	58.5	-50

from file: FIR_1904.csv

Airspace ID: Unique identifier of the FIR (or UIR)

Sequence Number: Numeric sequence number of a boundary point of the FIR's shape

Latitude: Latitude in decimal degree

Longitude: Longitude in decimal degree

Data available at: https://www.eurocontrol.int/dashboard/rnd-data-archive

QUERING THE DATABASE

We decide to visualize our data in a more meaningful way than Neo4j Desktop

Homepage

We implemented a homepage where we display all our queries results in geographical maps.

In order to retrieve such results we run a Python script using the library *mplleaflet.*

AirTraffic

EUROCONTROL

Our dashboard allows for a meaningful visualization of some interesting queries by showing the results in a geographic map. All queries are run in Cypher and refers to our database. By simply clicking on the images below you will be redirected to a full-screen visualization of the specific query and you can zoom in to better see the results. For example, the below quereies return: the italian airports by dimension, the european airports, the european airspaces, the worldwide airports traffic and the flight traversing more airspaces.

Screenshot of our homepage.

```python
# Output file name
fileName = "EU_airports_large"

# Query to be executed
query="""
MATCH (a:Airport)
WHERE EXISTS((a:Airport)<-[]-(:Flight)) AND a.type='large_airport'
RETURN a.name AS name, a.coordinates.latitude AS lat, a.coordinates.longitude AS long
"""
# Run the query and save the results into result
result = session.run(query)

# Create two list to store all the latitude and longitude coordinates
latitude = []
longitude = []

# Create the delimiter for our coordinates
nord = 71.185556 #nord point of Europe
sud = 34.85 #sud point of Europe
ovest = -31.275 #ovest point of Europe
est = 68.313889 #est point of Europe

for record in result:
    element = record.data()
    #check if the airport coordinates are inside of the europen delimiter
    if(element["lat"]<=nord and element["lat"]>=sud and element["long"]<=est and element["long"]>=ovest):
        latitude.append(element["lat"])
        longitude.append(element["long"])

#plot the resulting airports
plt.plot(longitude, latitude,"ro")

# load the plot into the map and export the result as html file
mapfile= '../maps/'+fileName+'.html'
map.show(path=mapfile) # to open a browser
```

Example of the computation of a query displaying all European large airports.

Use-Case: Milan Malpensa Airport

In order to show how our database can be used in practice we run a use-case based on Milan Malpensa Airport.

We implemented a simple web application where we display all airports belonging to a given country that we can reach from Milan Malpensa Airport with a direct flight.

Screenshot of the page.

Use-Case: Milan Malpensa Airport

By selecting the desired country from the drop down menu we get a list of such airports together with the cites they are located in.

Note: you can only select countries where there is at least one airport directly reachable from Malpensa.

Select the destination country: United Kingdom LOAD

Airport Name	City Name	
Birmingham International Airport	Birmingham	+
Cardiff International Airport	Cardiff	+
Edinburgh Airport	Edinburgh	+
Liverpool John Lennon Airport	Liverpool	+
London Biggin Hill Airport	London	+
London Gatwick Airport	London	+
London Heathrow Airport	London	+
London Luton Airport	London	+
London Stansted Airport	London	+
Manchester Airport	Manchester	+
East Midlands Airport	Nottingham	+

Results for United Kingdom.

```
//find the countries reachable from Milan Malpensa Airport
String query = """
        MATCH (f:Flight)-[:ARRIVES]->(a:Airport)-[:LOCATED]->(c:City)-[]->(co:Country)
        MATCH (f)-[dep:DEPARTS]->(d:Airport{icao:$icaoFrom})
        WHERE f.marketSegment <> \"All-Cargo\"
        RETURN DISTINCT co.name AS name ,co.isocode AS code ORDER BY name
        """;
```

```
//Given a specific country, find which airport and city are reachable from Milan Malpensa Airport
String query= """
        MATCH (f:Flight)-[arr:ARRIVES]->(a:Airport)-[:LOCATED]->(c:City)-[]->(co:Country{isocode:$iso})
        MATCH (f)-[dep:DEPARTS]->(d:Airport{icao:$icaoFrom})
        WHERE f.marketSegment <> \"All-Cargo\"
        RETURN DISTINCT a.icao AS icao, a.name AS name, c.name AS city
        ORDER BY city,name
        """;
```

Cipher queries used to retrieve such result.

Use-Case: Milan Malpensa Airport

By clicking on the plus icon for a specific airport we have a list of all airlines operating from Malpensa to the selected airport and an example with all information about a flight performing the route…

The route **Milan Malpensa-London Heathrow Airport** is covered by these airlines: British Airways, European Air Transport

Flight example in this route:

Flight ID	227846789
Airline Company	British Airways
Departure Airport	Malpensa International Airport
Arrival Airport	London Heathrow Airport
Departure Time	2019-03-04T19:28Z
Arrival Time	2019-03-04T21:37:11Z

Results for Heathrow Airport.

```
//Given a specific destination airport, find which airlines performs flights
//from Milan Malpensa to this airport
String query = """
        MATCH (d:Airport{icao:$icaoFrom})<-[:DEPARTS]-(f:Flight)-[:ARRIVES]->(a:Airport{icao:$icao})
        MATCH (f)-[:PERFORMEDBY]->(air:Airline)
        RETURN DISTINCT air.icao AS icao , air.name AS name
        """;
//Given the code of the flight, find all the important informations
String query = """
        MATCH (d:Airport) <-[dep:DEPARTS]-(f:Flight{id:$id})-[arr:ARRIVES]->(a:Airport)
        OPTIONAL MATCH (f)-[:PERFORMEDBY]->(air:Airline)
        RETURN DISTINCT f.id AS id ,air.name AS airline,a.name AS airArr,d.name AS airDep,
        arr.actualArrival AS timeArr,dep.actualOffBlock AS timeDep
        """;
```

Queries to retrieve such results.

Use-Case: Milan Malpensa Airport

… and all the airspaces traversed by such flight!

Airspaces traversed in this route:

Sequence	ICAO	Type	Country	Entry Time	Exit Time
0	TAXI_OUT	FIR	null	2019-03-04T19:28Z	2019-03-04T19:46Z
1	LIMM	FIR	Italy	2019-03-04T19:46Z	2019-03-04T19:53:02Z
2	LIMM	UIR	Italy	2019-03-04T19:53:02Z	2019-03-04T20:02:08Z
3	LSAS	UIR	Switzerland	2019-03-04T20:02:08Z	2019-03-04T20:12:09Z
4	LFFF	UIR	France	2019-03-04T20:12:09Z	2019-03-04T21:00:36Z
5	EGTT	UIR	United Kingdom	2019-03-04T21:00:36Z	2019-03-04T21:04:01Z
6	EGTT	FIR	United Kingdom	2019-03-04T21:04:01Z	2019-03-04T21:37:11Z
7	TAXI_IN	FIR	null	2019-03-04T21:37:11Z	2019-03-04T21:47:11Z

```
//Given the destination airport, choose one flight from Milan Malpensa to this airport
//and find which airspaces travers during the flight.
String query = """
        MATCH (d:Airport{icao:$icaoFrom})<-[:DEPARTS]-(f:Flight)-[arr:ARRIVES]->(a:Airport{icao:$icao})
        WITH f LIMIT 1
        MATCH (f)-[t:TRAVERS]->(fir:Airspace)
        OPTIONAL MATCH (fir)-[:BELONGSTO]->(cou:Country)
        RETURN f.id AS id, fir.icao AS icao, fir.type AS type, cou.name AS name,
        t.entryTime AS entry,t.exitTime AS exit, t.sequence AS seq ORDER BY t.sequence""";
```

THANK YOU FOR YOUR ATTENTION!

If you are interested keep scrolling for more queries!

Airline Companies

Show the biggest airline companies operating in the Business Aviation segment

```
MATCH(f:Flight{marketSegment:'Business Aviation'})-[:PERFORMEDBY]→(a:Airline)-[:HASNATIONALITY]→(c:Country)
RETURN a.name AS AirlineCompany, c.name AS Country, count(f) AS NumberFlights
ORDER BY NumberFlights DESC
LIMIT 10
```

Airline Companies

Show the biggest airline companies operating in the Business Aviation segment

```
MATCH(f:Flight
RETURN a.name A
ORDER BY Numbe
LIMIT 10
```

```
TY]→(c:Country)
```

AirlineCompany	Country	NumberFlights
"NetJets Europe"	"Portugal"	337
"Vistajet"	"Canada"	151
"Lufttransport"	"Norway"	138
"Air Hamburg"	"Germany"	128
"GlobeAir"	"Austria"	60
"Eisele Flugdienst"	"Germany"	44
"Global Jet Luxembourg"	"Luxembourg"	44
"Gestair"	"Spain"	33
"Air Luxor GB"	"Guinea-Bissau"	31
"Bristol Flying Centre"	"United Kingdom"	31

Aircraft Models

Show the average number of seats of aircraft models for each market segment

```
MATCH (ac:AircraftModel)-[:USEDBY]→(f:Flight)
WHERE ac.numberOfSeats IS NOT NULL
WITH f.marketSegment AS MarketSegment, COLLECT(ac.numberOfSeats) AS listOfSeats
UNWIND(listOfSeats) AS seatsPerMarket
RETURN MarketSegment, avg(seatsPerMarket) AS AvgSeats
```

Aircraft Models

Show the average number of seats of aircraft models for each market segment

MarketSegment	AvgSeats
"Traditional Scheduled"	151.72601449112716
"All-Cargo"	231.76968796433857
"Charter"	152.55852842809378
"Lowcost"	149.297965842127
"Business Aviation"	18.0

Aircraft Models

Show the aircraft model with the highest number of engines and display the information about its manufacturer

```
MATCH (ac:AircraftModel)
WITH max(ac.engineCount) AS maxNumberOfEngines
MATCH (ac:AircraftModel)-[:BUILTBY]→(m:Manufacturer)-[:HASNATIONALITY]→(c:Country)
WHERE ac.engineCount=maxNumberOfEngines
RETURN ac.icao AS AircraftModel, ac.engineCount AS NumberEngines, m.name AS Manufacturer, c.name AS ManNationality
```

Aircraft Models

Show the aircraft model with the highest number of engines and display the information about its manufacturer

```
MATCH (ac:AircraftModel)
WITH max(ac.engineCount) AS maxNumberOfEngines
MATCH (ac:AircraftModel)-[:BUILTBY]→(m:Manufacturer)-[:HASNATIONALITY]→(c:Country)
WHERE ac.engineCount=maxNumberOfEngines
RETURN ac.icao AS AircraftModel, ac.engineCount AS NumberEngines, m.name AS Manufacturer, c.name AS ManNationality
```

AircraftModel	NumberEngines	Manufacturer	ManNationality
"B52"	8	"Boeing Aircraft Company"	"United States"

Aircraft Models

Show the most used aircraft model in all-cargo flights and display the information about its manufacturer

```
MATCH (f:Flight{marketSegment:'All-Cargo'})←[:USEDBY]-(ac:AircraftModel)-
[:BUILTBY]-(m:Manufacturer)-[:HASNATIONALITY]→(c:Country)
RETURN ac.icao AS AircraftModel, m.name AS Manufacturer, c.name AS ManNationality,
count(f) AS OperatedFlights
ORDER BY OperatedFlights DESC
LIMIT 1
```

Aircraft Models

Show the most used aircraft model in all-cargo flights and display the information about its manufacturer

```
MATCH (f:Flight{marketSegment:'All-Cargo'})←[:USEDBY]-(ac:AircraftModel)-
[:BUILTBY]-(m:Manufacturer)-[:HASNATIONALITY]→(c:Country)
RETURN ac.icao AS AircraftModel, m.name AS Manufacturer, c.name AS ManNationality,
count(f) AS OperatedFlights
ORDER BY OperatedFlights DESC
LIMIT 1
```

AircraftModel	Manufacturer	ManNationality	OperatedFlights
"B734"	"Boeing Aircraft Company"	"United States"	364