

GARDA CRIME DATA ANALYSIS

DATA WAREHOUSE & BUSINESS INTELLIGENCE (CA-2)



Name: Raghav Mehta
Student ID: x17155151
Email ID: x17155151@student.ncirl.ie
MSc in Data Analytics
National College of Ireland

1. INTRODUCTION

Gardaí or “the Guards” is the name for the police force of the Republic of Ireland, which falls under the Irish Government. In this project, we analyze the past criminal record data of Ireland generated by the Gardaí. We attempt to map this data with some key metrics such as population of different divisions, unemployment rates and depression levels to analyze the relation between crime types and criminal intense areas. The objective behind this data warehouse is to be able to derive business intelligence for the Gardaí and provide insights in order to regulate and reduce the occurrence of crimes in the country. For this analysis, we shall consider the data from 2012 to 2015 as it is complete and available. We shall also capture the sentiments of the Irish citizens through Twitter to understand their reactions to the crime and how safe they feel in their resident provinces.

2. SOURCES OF DATA

To populate such a data warehouse, we have considered the following sources of data:

- a) **Gardaí offences data** – This data is a structured set of data and our primary source available on the www.data.gov.ie website. This data includes the number of offences for each type of crime, in all the Gardaí stations across the country. Each station is mapped to one of the 26 divisions of Republic of Ireland. The data is available from 2010 to 2016*, however we have filtered the data from 2012 to 2015 as the data for 2016 is incomplete. Data is downloaded is downloaded from the website in CSV format.

URL: <https://data.gov.ie/dataset/crimes-at-garda-stations-level-2010-2016>

- b) **Wikipedia data** – Our second set of data is a semi-structured dataset extracted from a Wikipedia page. This data consists of the population and province for each of the divisions of Ireland. We have extracted this dataset using the R and have subsequently cleansed the data in the

same code. Some unwanted rows related to Northern Ireland and some unrequired columns have been removed.

URL: https://en.wikipedia.org/wiki/List_of_Irish_counties_by_population

- c) **Mock data** – To aid our analysis we have mocked data which was unavailable with the Irish CSO and other Irish data websites. We have generated an unemployment rate (in %) and depression level (score of 0-100) for each of the 26 divisions for 4 years. This is required to do comparative analysis on the occurrence of different crime types and whether any relation exists. This data has been downloaded and structured in CSV format.
- d) **Twitter Sentiment data** – Finally we have also extracted tweets to run a sentimental analysis based on certain keywords such as “extortion”, “assault”, “theft”, “kidnapping”, “violence”, “murder”. Each time we have entered the name of the division to generate tweets related to that area, performed a sentimental analysis and saved the score in a different data frame. This data frame consisting of the sentiment score for all 26 divisions have been written out to a CSV from the R code.

3. ARCHITECTURE

There are two popular methodologies of defining the architecture of a data warehouse. The Kimball approach uses bottom-up approach and implements the generation of multi-dimensional tables around a central fact table. This data will be extracted and transformed to be housed in a data mart. This method advocates denormalization of data across the warehouse. We shall build a star schema using SQL JOIN and queries to populate the dimensional model for faster analytical processing through OLAP technologies.

On the other hand, Inmon approach is a top-down approach where the dimensions are created after the population of the data warehouse. It uses normalized data and leads to slower querying and JOIN.

For our project, we have used the Kimball approach, where data from multiple sources are housed in the staging area, undergo ETL process and used to build the data warehouse. Then we finally use OLAP to derive reports and analysis for our end client.

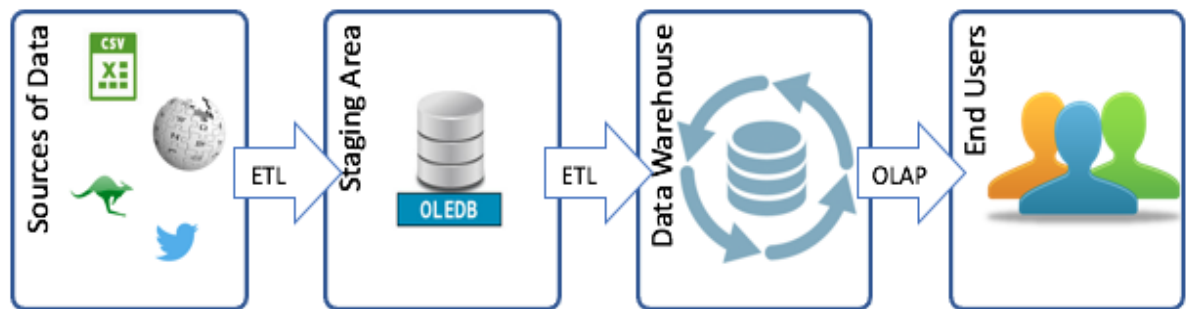


Fig.1. DW Architecture

4. IMPLEMENTATION

- a) We collect our data from the various sources – this includes the CSV from data.gov.ie, semi-structured data from Wikipedia, Twitter sentimental analysis and CSV of mock data.
- b) Using truncate and load method, we put the required data in to the staging area. The offence data of Gardaí forms our principle data for the business objective.
- c) Next, we clean our data. The data from Wikipedia was scrapped and cleansed in a single R script. The dataset of crimes was organized in Excel itself and then cleansed using R.
- d) We load all the data sets as flat files in to staging area to raw data tables.
- e) We generate our Dimension tables using these raw data tables and performing SQL JOIN wherever required
- f) Then we feed the keys from the Dimension tables and the measures from the raw data tables in to our Fact table.

- g) We check our data in SQL Server Management Studio to ensure that the data loaded in to Fact and Dimension tables are as required
- h) Using the Analysis Services of Visual Studio, we deploy our cube
- i) Finally, we draw our business intelligence from the cube data through visualization tools such as Tableau, Excel, Power BI etc.

5. DATA MODEL

We first load our data from the source to the staging area. All our CSV's are imported using Flat File source to the OLE DB Destination. This creates the raw data tables in SQL Server Management Studio.

The four sources of data are being put in the raw tables:

- Gardaí Crime data – structured data which is slightly cleaned in R
- Wikipedia data – semi structured data scrapped and cleansed through R
- Mock Index data – structured mocked data
- Twitter data – Using R to extract tweets on different keywords, averaging the scores and storing in another data frame before writing out to csv

The dimensions and fact tables are generated according to our data and requirement as per the business objective. We have generated 3 dimensional tables and 1 central fact table as follows :-

- **Dim_station:** it is the dimension table consisting of the primary key offence_id, every station, their station_id, the division it falls in and finally the province it belongs to. This is a way of drilling down as one province has multiple divisions and each division has multiple stations.

- **Dim_crimetype:** this dimension table consists of the various types of crime that are recorded at the stations every year and a type_id which acts as the primary key for this table.
- **Dim_year:** since our data is collected annually, our date dimension only consists of the year and the year_id associated with it. The year_id acts as primary key.
- **Fact table:** The fact table consists of the foreign keys for each of the dimension table. Along with that there are four measures: population of each division, unemployment rate, depression score and the number of offences recorded at each station, under each crime type for each of the years (2012 to 2015)

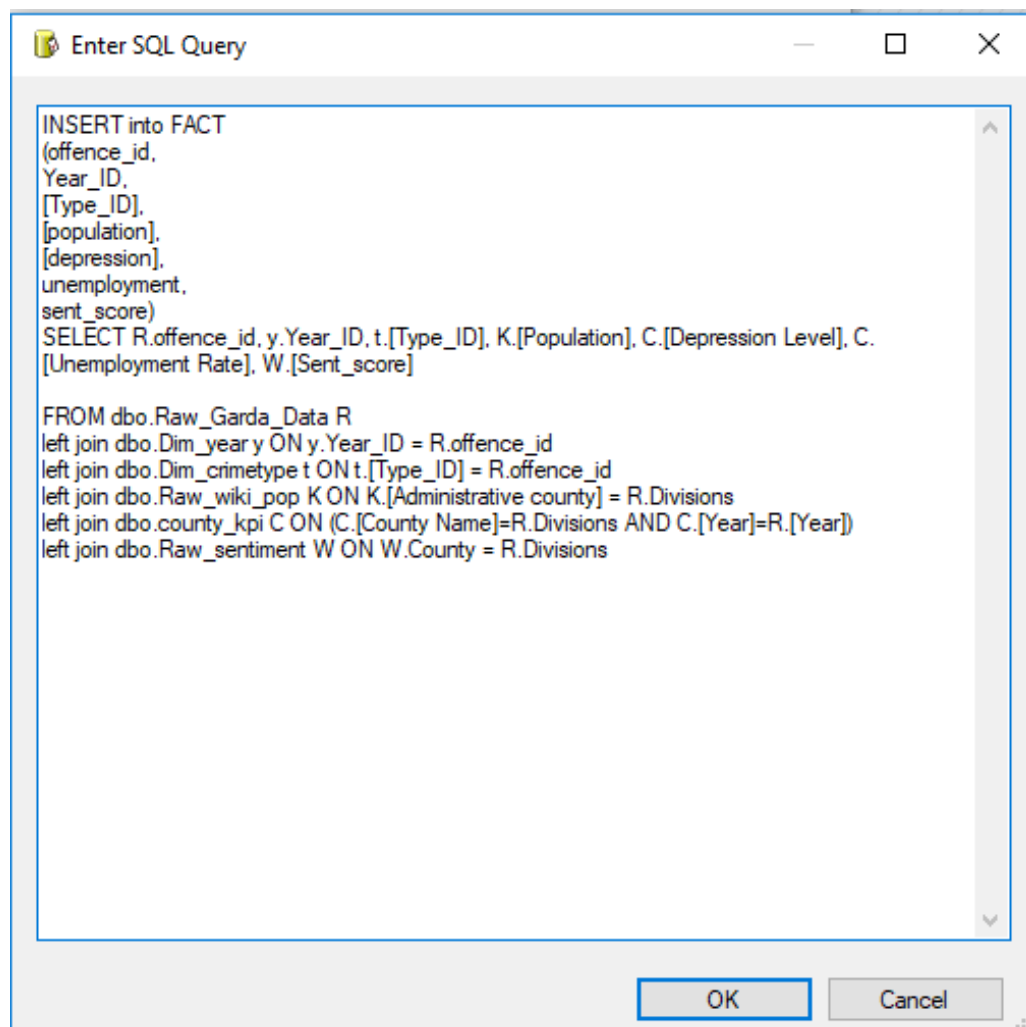


Fig. 2. Fact Table Query

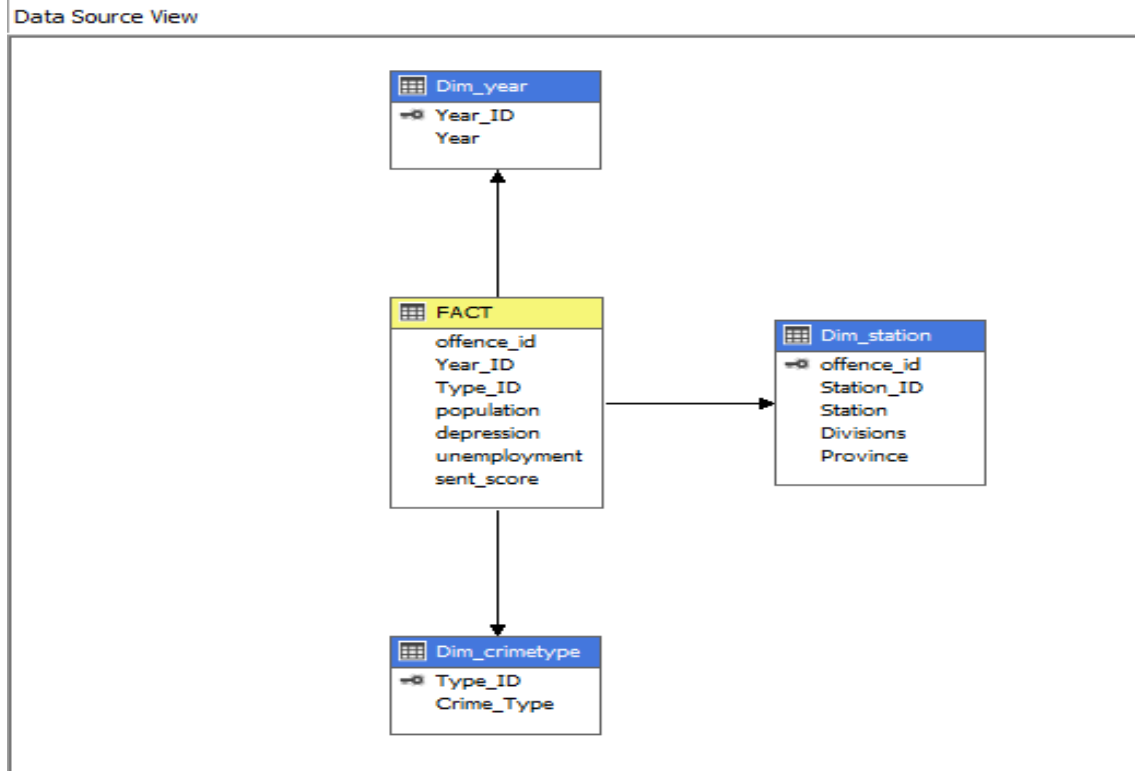


Fig. 3. Dimensional Model

6. ETL STRATEGY

Extraction:

Our ETL process is carried out in Integration Services in Visual Studio. We first truncate our raw data tables to remove all values and load the data on to it. This is done so that if there is a change in the source data, it will reflect in our data warehouse. This ETL process is re-runnable and is executed each time we wish to generate our fact and dimension tables.

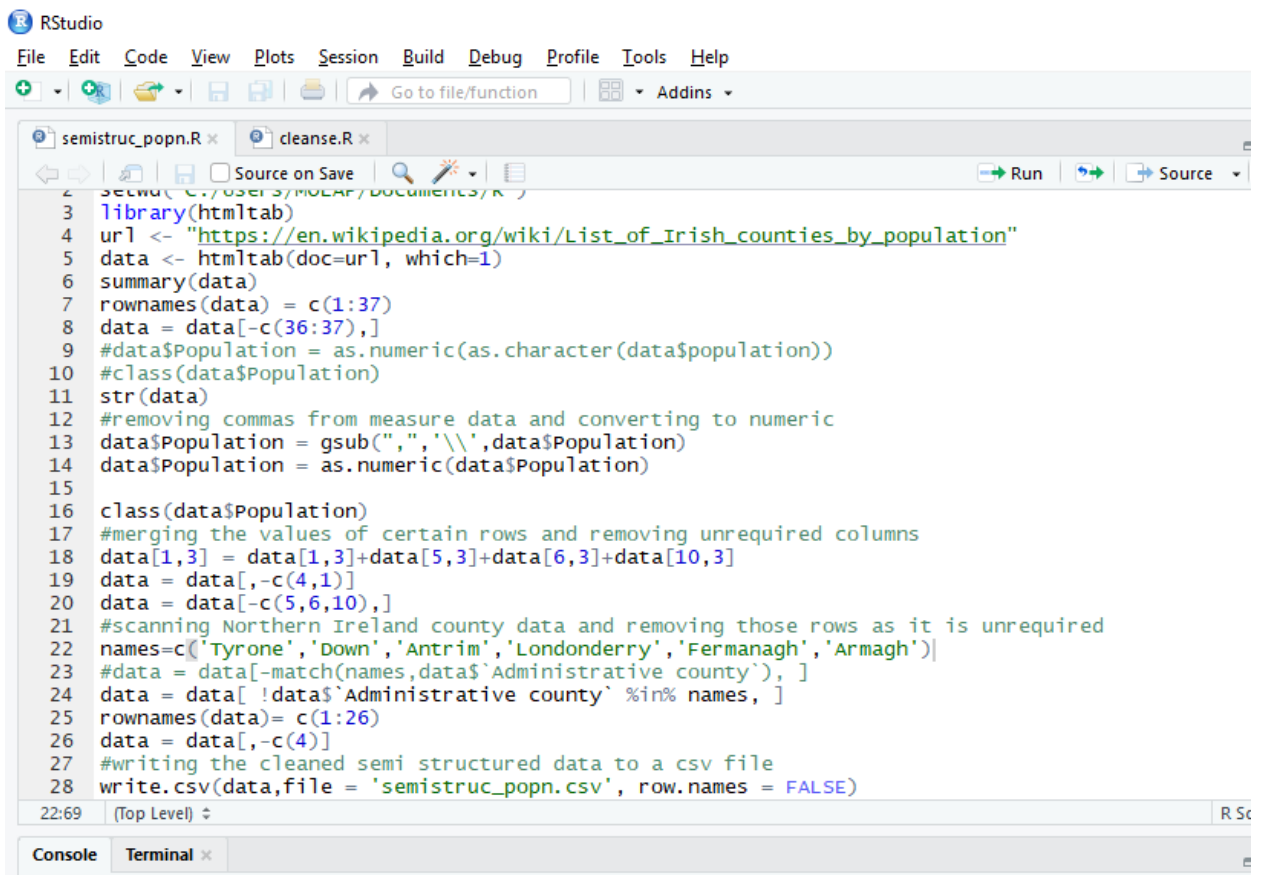


Fig. 4. Loading Data in Staging Tables

As identified earlier, the extraction of semi-structured data from Wikipedia is done using an R script. The data is cleaned within the code to remove unwanted attributes and rows which do not pertain to Republic of Ireland.

We also clean the structured data in by running a simple R script to get rid of certain special characters and unwanted words. This is being written out to a csv file which is then being imported through a flat file source in Visual Studio.

Finally, our Twitter data is also generated using an R script, downloading the tweets, performing sentimental analysis and writing out the data in one code.



```
2 setwd("C:/Users/MOLAP/Documents/R")
3 library(htmllab)
4 url <- "https://en.wikipedia.org/wiki/List_of_Irish_counties_by_population"
5 data <- htmllab(doc=url, which=1)
6 summary(data)
7 rownames(data) = c(1:37)
8 data = data[-c(36:37),]
9 #data$Population = as.numeric(as.character(data$population))
10 #class(data$Population)
11 str(data)
12 #removing commas from measure data and converting to numeric
13 data$Population = gsub(",", "", data$Population)
14 data$Population = as.numeric(data$Population)
15
16 class(data$Population)
17 #merging the values of certain rows and removing unrequired columns
18 data[1,3] = data[1,3]+data[5,3]+data[6,3]+data[10,3]
19 data = data[-c(4,1)]
20 data = data[-c(5,6,10),]
21 #scanning Northern Ireland county data and removing those rows as it is unrequired
22 names=c('Tyrone','Down','Antrim','Londonderry','Fermanagh','Armagh')
23 #data = data[-match(names,data$'Administrative county'), ]
24 data = data[!data$'Administrative county' %in% names, ]
25 rownames(data)= c(1:26)
26 data = data[-c(4)]
27 #writing the cleaned semi structured data to a csv file
28 write.csv(data,file = 'semistruc_popn.csv', row.names = FALSE)
```

Fig. 5. R script to extract Wikipedia data


```

519 Tweets.text
520
521 analysis = score.sentiment(Tweets.text, pos, neg)# calls sentiment function
522 df24 = data.frame(analysis)
523 avgdf1 = mean(df3$score)
524 df_f23 = data.frame(province,avgdf1)
525 #
526 province = c('#Cork')
527
528 hashtags <- c("#murder", '#thiefs', '#killing', '#homocide', '#assault', '#vandalism', '#mansla
529 needle <- paste(hashtags, collapse = " OR ")
530 hashtags <- province
531 needle <- paste(hashtags,needle, collapse = " AND ")
532
533 tweets <- searchTwitter(needle, n = 20)
534 df <- twListToDF(tweets)
535 #tweets = searchTwitter('#racism',n=50)
536 Tweets.text = lapply(tweets,function(t)t$getText()) # gets text from Tweets
537 Tweets.text
538
539 analysis = score.sentiment(Tweets.text, pos, neg)# calls sentiment function
540 df25 = data.frame(analysis)
541 avgdf1 = mean(df3$score)
542 df_f24 = data.frame(province,avgdf1)
543
544 dffinal = rbind(df_f,df_f1,df_f2,df_f3,df_f4,df_f5,df_f6,df_f7,df_f8,df_f9,df_f10,df_f11,
545 dffinal_1 = gsub('#','',dffinal$province )
546 dffinal_avg = data.frame(dffinal$avgdf1)
547 datafinal = data.frame(provinces = dffinal_1 , average_score = dffinal_avg)
548 #dffinal_1 = gsub("'^(.{1})'", "\\u",dffinal_1)
549 dffinal_1
550 write.csv(file = 'twitter_sentimental.csv',datafinal)
551

```

Fig. 6. Twitter Sentimental Score R Script

Transform & Load:

After the data is loaded in to the raw tables in the staging area, we generate our dimension and fact tables.

We start off by truncating our dimension tables, before we load data from the raw tables on to it. After our dimension tables, the fact table is created and the measures as well as primary keys loaded on to it.

The creation of fact and dimension tables are achieved through running EXECUTE SQL TASK and using SQL JOIN queries to merge the data from the raw tables as per our business requirements.

Our dimension and fact tables are ready, and we can check the data in SSMS to ensure data quality.

This process of ETL is illustrated in the below figure.

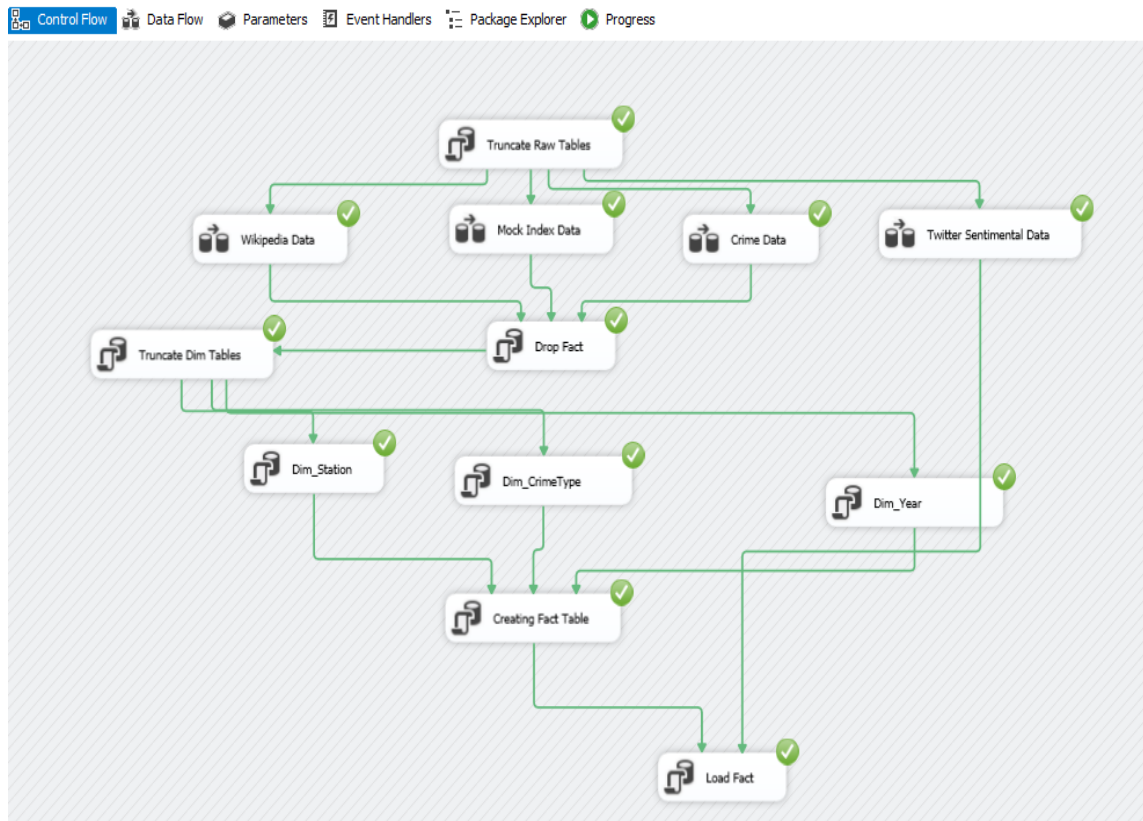


Fig. 7. Complete Data Flow

7. CUBE DEPLOYMENT

After our Fact and Dimension tables are created, we move to the Analysis Services of Visual Studio to deploy our cube. Our dimensional model of star schema is generated over here. Once cube is generated, we visualize our data to form business intelligence analysis. The below figure shows the successfully deployed cube.

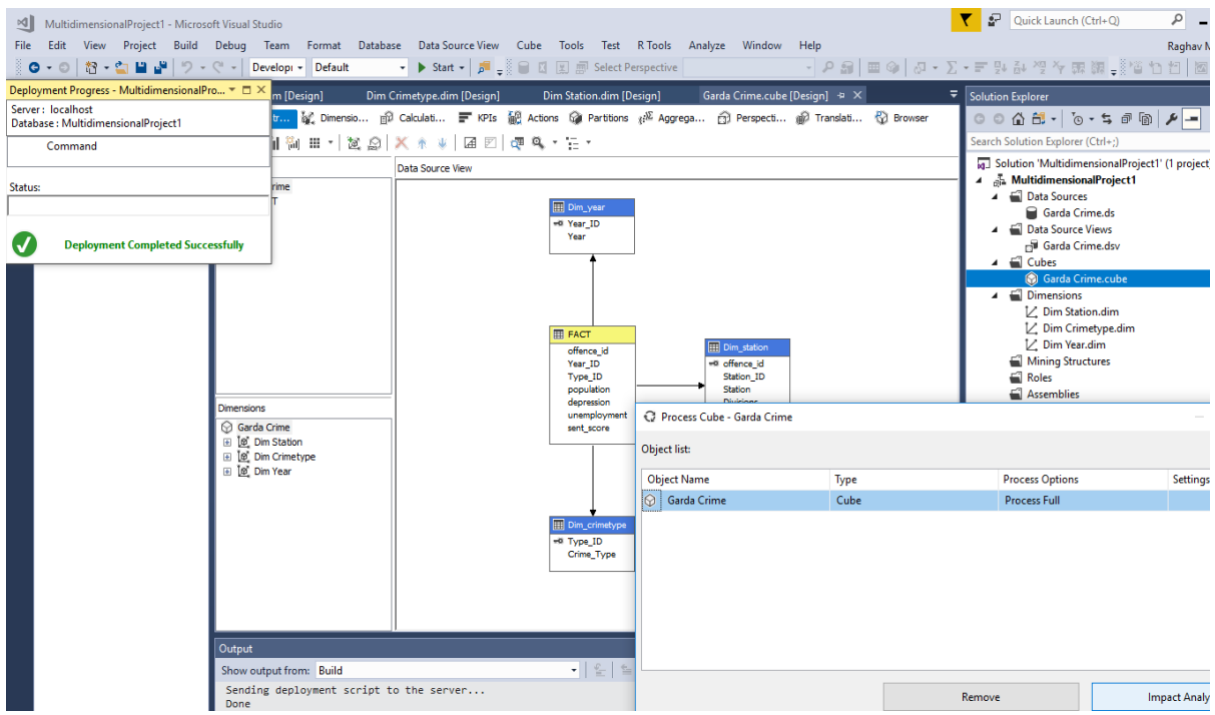


Fig. 8. Successfully Deployed Cube

8. APPLICATIONS OF DATA WAREHOUSE FOR BUSINESS INTELLIGENCE

i) Case Study 1

Gardaí would like to know the major areas of crime, in order to restructure the police stations, deploying more officers from low crime areas to major areas.

Approach : It would be incorrect to simply point out the counties where occurrence of crime is high. Hence we have introduced a calculated field in Tableau, calculating the offences per capita. This gives a clearer picture in assessing the redeploy the police officers, from one county to another. We will ascertain the areas of low and high crimes county wise to answer this query.

2 out of the 4 datasets are being used for this BI query.

Solution : A dashboard has been created with 2 reports. The crimes per capita have been calculated and the top & bottom 4 counties have been shown for each year, with the divisions/counties coloured by province. The crimes per capita for each province are also shown for better understanding at a macro level.

From the graph below, we understand that certain counties of Leinster province are responsible for highest and lowest crime rates in the entire Republic of Ireland. Hence police officers can be deployed intra-province to provide crime stability in the province. Overall an influx of officers is also required in the province 'Leinster' which has the highest crime per capita each year.



Fig. 9. Case Study 1 (Tableau)

ii) Case Study 2

Analyzing certain crimes through certain metrics – unemployment (in %) and depression score (measured on a scale of 0 – 100), division wise.

Approach: From our understanding, not all crimes are inspired by unemployment or depression. Hence we have created a dashboard, again with 2 sheets – one for each. We have applied filters on different relevant crime types for each of the sheet, linking offences such as Attempts/Threats, Drug related offences and Dangerous/Negligent acts with depression and similarly Burglary, Theft, Robbery etc. with unemployment.

The treemap's size is representative of the number of offences per capita in that division and the intensity of colour is the metric we are analyzing. We have also filtered the top crime counties using the inbuilt Tableau feature under '*Edit Filter -> Top -> By Field*' and a separate filter for year to easily navigate through previous years' data.

3 out of the 4 datasets are being used for this BI query.

Solution: The following example for year '2013' shows that in case of depression, a majority of the top 10 crime affected areas, have a high average depression score (presence of many dark blue squares). Whereas there is not much of a direct relation in the case of unemployment.



Fig. 10. Case Study 2 (Tableau)

iii) Case Study 3

Twitter Analysis of public's sentiments and reaction towards crime in their county.

Approach : A twitter sentimental analysis code was run to score the public's reactions. For each of the 26 divisions, tweets were extracted, cleansed and scored. In the end the average of those tweets were computed to get the mean sentiment score for that division. Now these scores are being compared to the offences per capita in those divisions for the latest year of crime data we have.

3 out of the 4 datasets are being used for this BI query.

Solution : We see no apparent link between people's reaction and the sentiment score derived for that county. The people in high crime areas are not taking to social media to express their concern and raising their voice against this rampant problem.

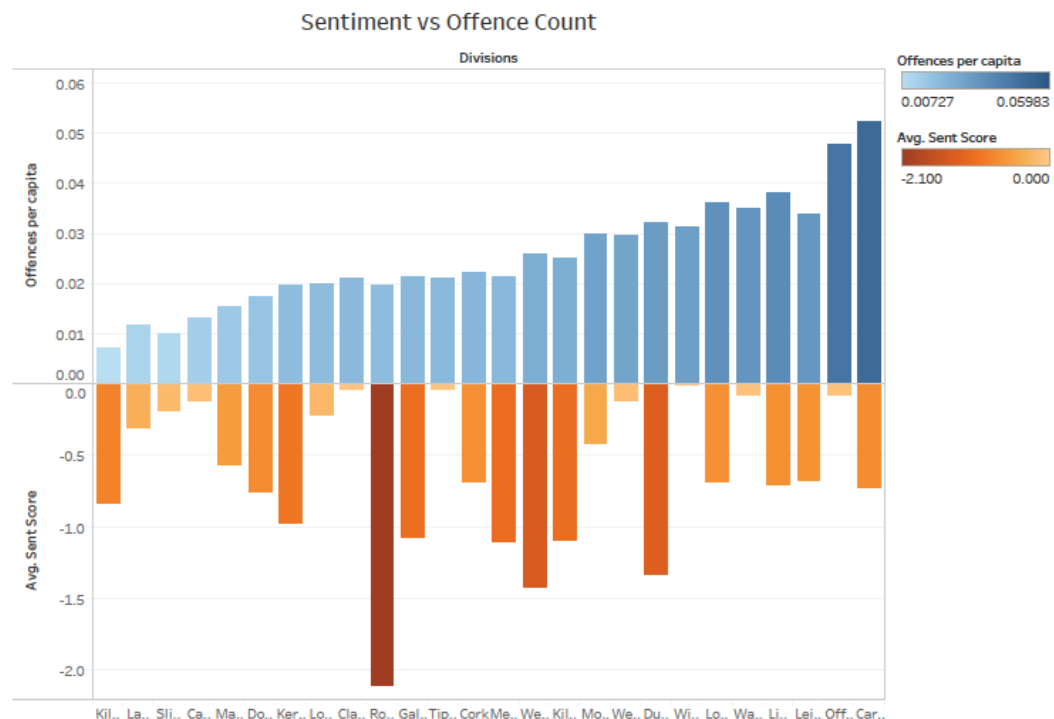


Fig. 11. Case Study 3 (Tableau)

9. REFERENCES

- [1] Kimball, R (1996). *Data Warehouse Toolkit – Practical Techniques for Building Dimensional Data Warehouse*, New York: Wiley & Sons.
- [2] Lamia Yassad and Aissa Labiod, “*Comparative Study of Data Warehouses Modeling Approaches: Inmon, Kimball and Data Vault*”, International Conference on System Reliability and Science, 2016
- [3] Matloff, N., 2011. *The art of R programming: A tour of statistical software design*, No Starch Press
- [4] https://www.w3schools.com/sql/sql_join_left.asp