2020

# DATA WAREHOUSING – PROJECT PORTFOLIO

CSC-557 PROJECT PORTFOLIO

YESHWANTH NELLIAKNTI

# Contents

# 1. Overview

We are building a warehouse to be used for the purpose of public health and knowledge. Our data is designed for users looking to explore methods of maintaining & improving living conditions in metropolitan areas, and modeling trends in air quality for a given area. For instance, public health officials using our data warehouse will be able to build applications and systems that can notify of harmful air conditions. Our data is housed in the public domain to be used by all.

# 2. Data Warehouse Requirements

When gathering requirements, it is important to also gather the right "characteristics" of each requirement which can be called as metadata. Each requirement, which is part of the solution to be, requires a unique identifier together with a clear description, the rationale for the requirement, the corresponding owner and the beneficiaries. Unique identifiers for all requirements will enable a better communication between all parties.

## 2.1. Target Application Domain

- Alabama Department of Environment Management.

## 2.2. Potential Users

- Scientist
- State government agency
- Students

## 2.3. Information Requirements

Below are some of the requirements for designing a Data Warehouse.

- Access to historic air quality levels by going back to n years of each test site
- Track the emergence of new, and concentration levels of known toxic substances at each source
- Estimate current air quality levels/conditions for neighboring regions outside the direct test zones
- Predict pollution levels across Alabama for the upcoming year

- Monitor the health status of each air monitoring station
    - Equipment uptime, battery levels, etc.
- Forecast average ozone level for the next *n* years across Alabama

# 3. Data Warehouse Logical Design

## 3.1. Key Decisions

There are few key decisions based on which the data warehouse base can be build.

### 3.1.1. Business Process

Gathering Information on air quality in metropolitan areas and share it with stations.

### 3.1.2. Granularity

Snapshot of Air quality measurement from each station in the Morning, Afternoon and at Nights of a day.

### 3.1.3. Measures

Some of the measures identifies based on requirements are.

- Concentration levels of harmful chemicals (Carbon Monoxide, Nitric Oxide, etc.)
- Atmospheric visibility
- Ozone level
- Current Wind speed
- Air Quality Index

### 3.1.4. Dimensions

Below are the dimensions that are identified and will be used in designing Data Warehouse.

- Test Site
    - Station Name, State, County, Range
- Date
    - Time, Day, Month, Year and Season
- Devices
    - Device Name, Min Range, Max Range, Units, Device Accuracy

- Toxins
    - Name, Type (Primary Standard &Secondary Standard), Harmful Concentration Levels, Description

## 3.2. Types of Measures

**Additive Measures:**

Additive measures can be summed across any of the dimensions associated with the fact table. The additive measures base on above dimensions are like calculation of air quality index based on toxins.

**Semi-additive :**

Measures can be summed across some dimensions, but not all; balance amounts are common semi-additive facts because they are additive across all dimensions except time.

**Non-additive :**

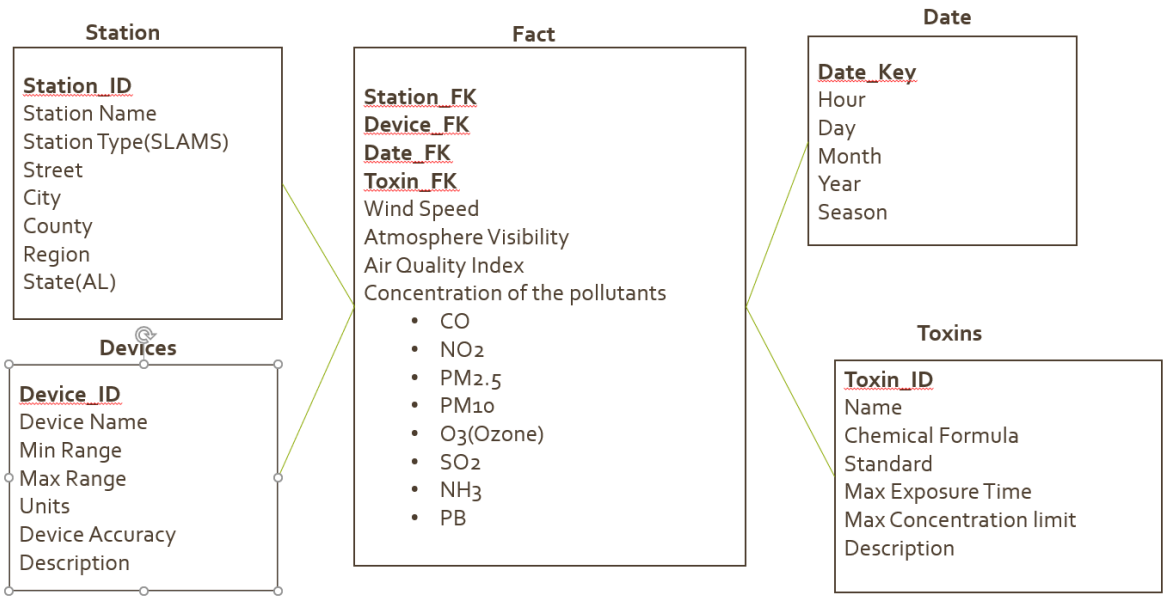
These are those specific class of fact measures which cannot be aggregated across all/any dimension and their hierarchy.

## 3.3. Surrogate Keys

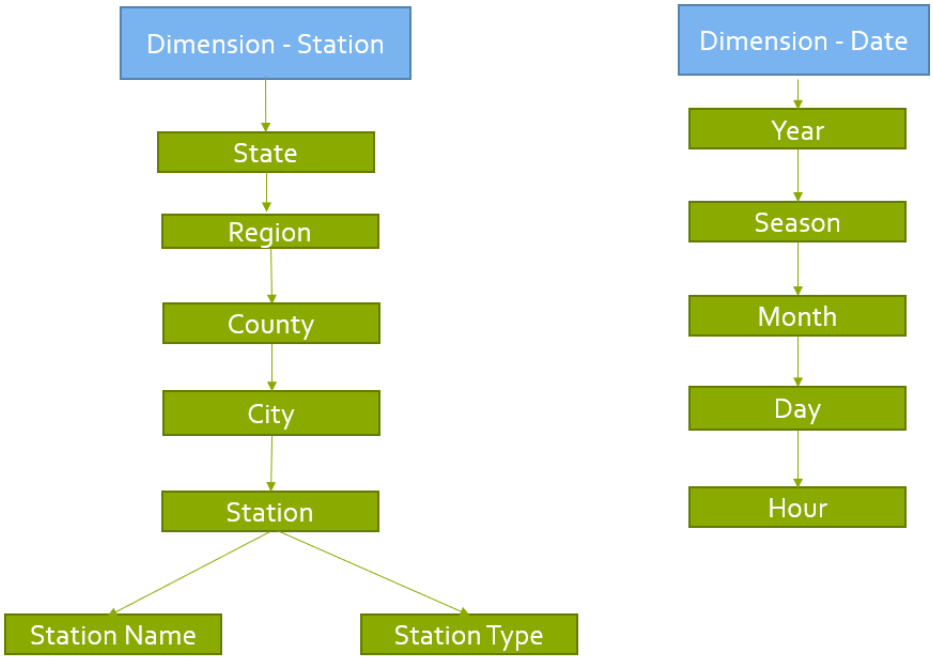
Surrogate Keys are integers that are assigned sequentially in the dimension table which can be used as primary key. We should have defined PK in our tables as per the business requirement and that might be able to uniquely identify any record. As Surrogate Key is just an Integer attached to a record for the purpose of joining different tables in a Star or Snowflake schema-based Data Warehouse. Surrogate Key is much needed when we have very long primary key, or the datatype of the primary key is not suitable for Indexing.

## 3.4. Logical Star Scheme.

A logical star schema is composed of central fact tables, a set of dimension tables, and the joins that relate the dimension tables to the fact table.




## 3.5. Conceptual Hierarchies



# 4. Extraction Transformation and Load (ETL)

## 4.1. ETL Process

Extract Transform and Load (ETL), it is a general procedure of copying data from one or more sources into a destination system which represents the data differently from the source's or in a different context than the sources.

Some of the challenges that can be like user expectations where they expect very refined results from any analysis. Information Driven Analysis, spending enough time on understanding and documenting business needs. Data Structuring and systems optimization, Choosing the right type of warehouse, Balancing resources and Data governance and master data.

## 4.2. Transformations

Below are the few common types of transformations that can be performed on extracted source data.

Merging**:**

Merging is to link the data from multiple data sources.

**Ex**: Data sources from a Flat File and SQL server can be combined by using Tmap processing component in Talend and get the output in a data warehouse.

Splitting:

Splitting a single column into multiple columns.

**Ex**: Consider Address filed which can be split into Street, City, County, Region. This can be done by using tExtractDelimitedField component in Talend.

Deduplication:

Identifying and removing duplicate records.

**Ex**: Consider if there are 3 duplicate records in Customer table. Use tUniqRow component in talend and select the Key attributes column which removes the duplicate records.

Filtering:

Filtering is to select only certain rows and or columns that are required.

**Ex**: By using Tmap or tFilterRow or tFilterCoulmn we can remove the column or row to be not included in data warehouse.

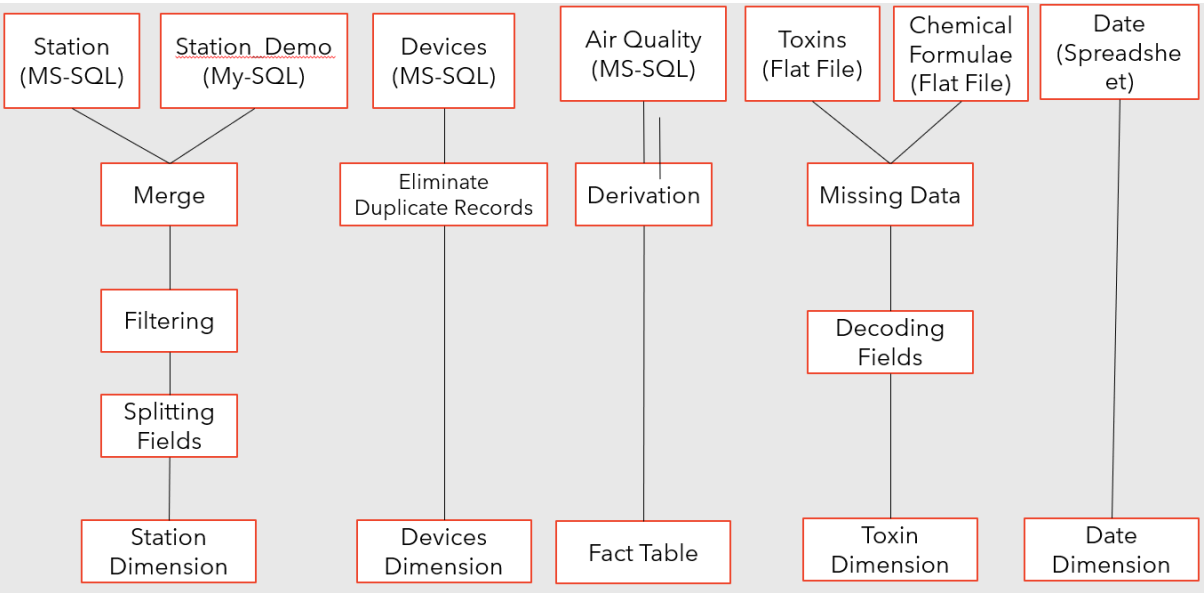
Data Validation:

Simple or complex data validation.

**Ex**: If there is no data/empty in the first three columns of a row then reject the row from processing.

## 4.3. Process of Populating Dimension Tables

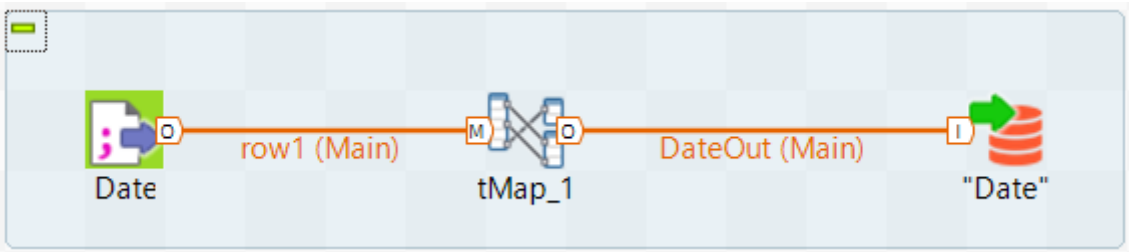
Data can be populated into Dimension table through multiple sources(like flat files, spread sheets, MS-SQL and MySQL). For this Data Warehouse multiple data sources are used and below is the staging plan. While inserting data into dimension table we should make sure they are included with surrogate keys.




Some of the jobs that are used in Talend to populate data into dimension tables are described below.

Data Populating from Spread sheet:

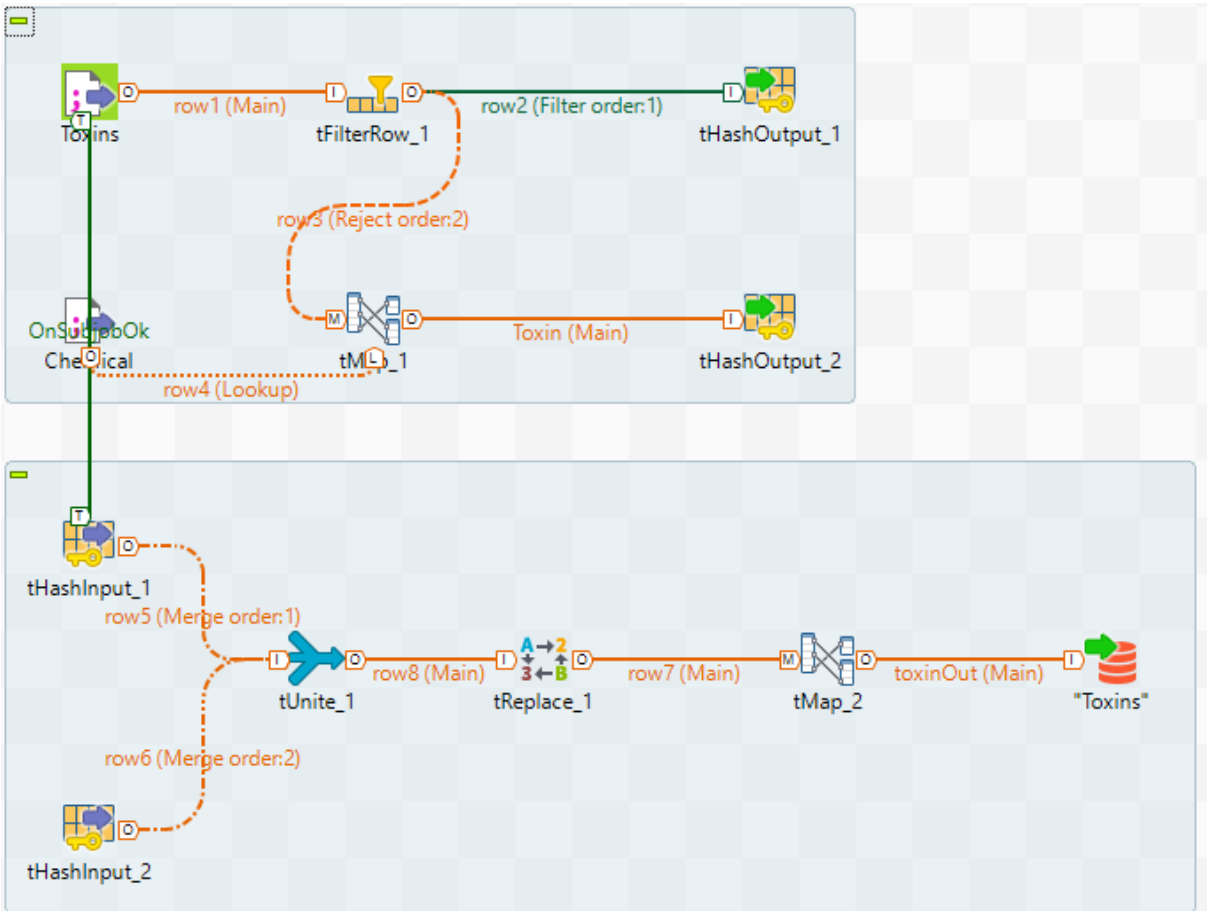
First the data available in a Spread sheet is delimited as a metadata and added to a job. Also connect the Dimension table data base where this data must be incorporated. Use tmap transformation to match the tables and run the job and as output the data content transfers from spread sheet to database dimension table.



As above process the data can be populated to dimension tables by using different transformation like merging different columns, filtering the data, splitting of the fields, eliminate duplicate records, filling out the missing data, decoding of the fields and deriving a new field in to the fact table. Below are the jobs with transformations as showed in Staging diagram.

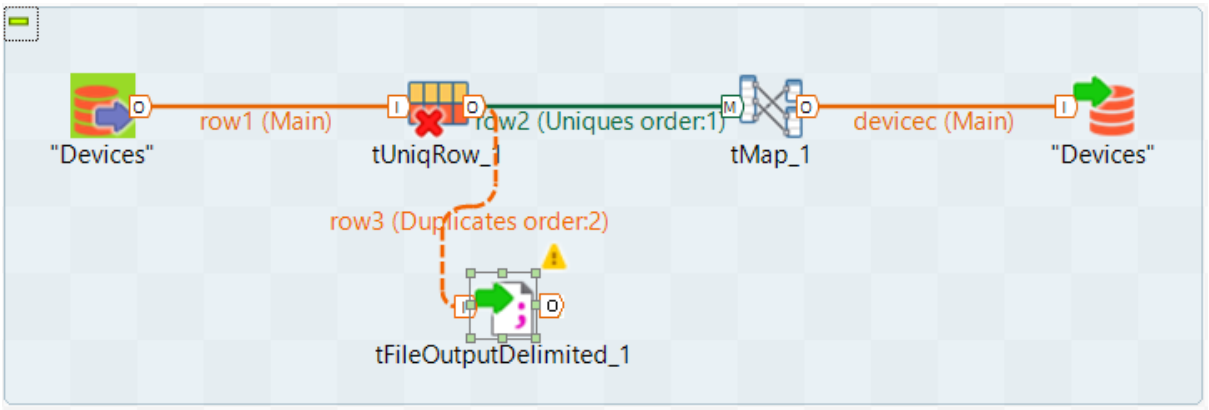
Flat File(Missing Data and Decoding fields):

Here the data available in flat file(Notepad) is mapped with tFilterRow which filter out the missing data and gives the Output to tHashOutput_1. The filter row is connected to tmap which has linked with Chemical flat file that has the missing values. In Tmap the missing values are filled by doing mapping with row1, row4 and inner join as function. The result of missing value records by filling it up is stored in tHashoutput_2. Once this job is success and subjob will be run for decoding of fields by using tReplace. The refined data is transferred to Toxins dimension table.



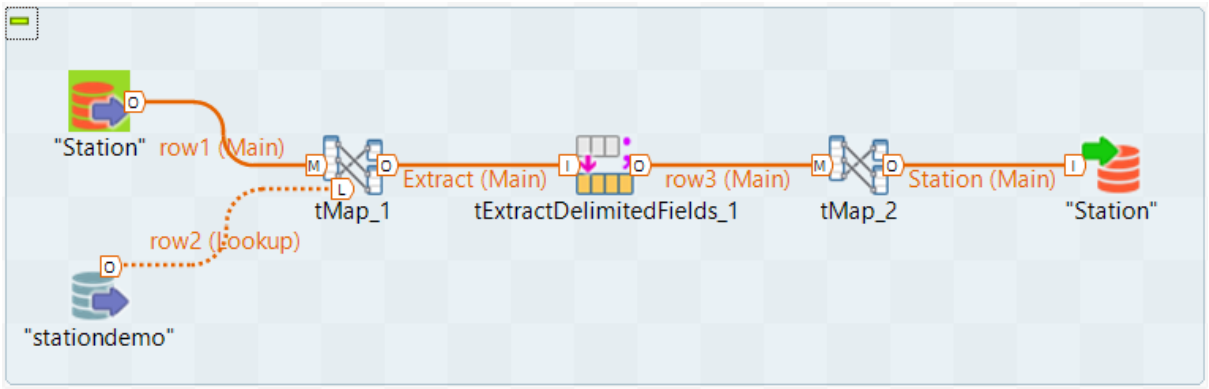
MS-SQL(Eliminate Duplicate Records):

For eliminating duplicate records available in devices, we use tUniqrow transformation. The duplicate records are stored in tFileOutputDelimited_1. By using tMap the data is inserted into Devices table.



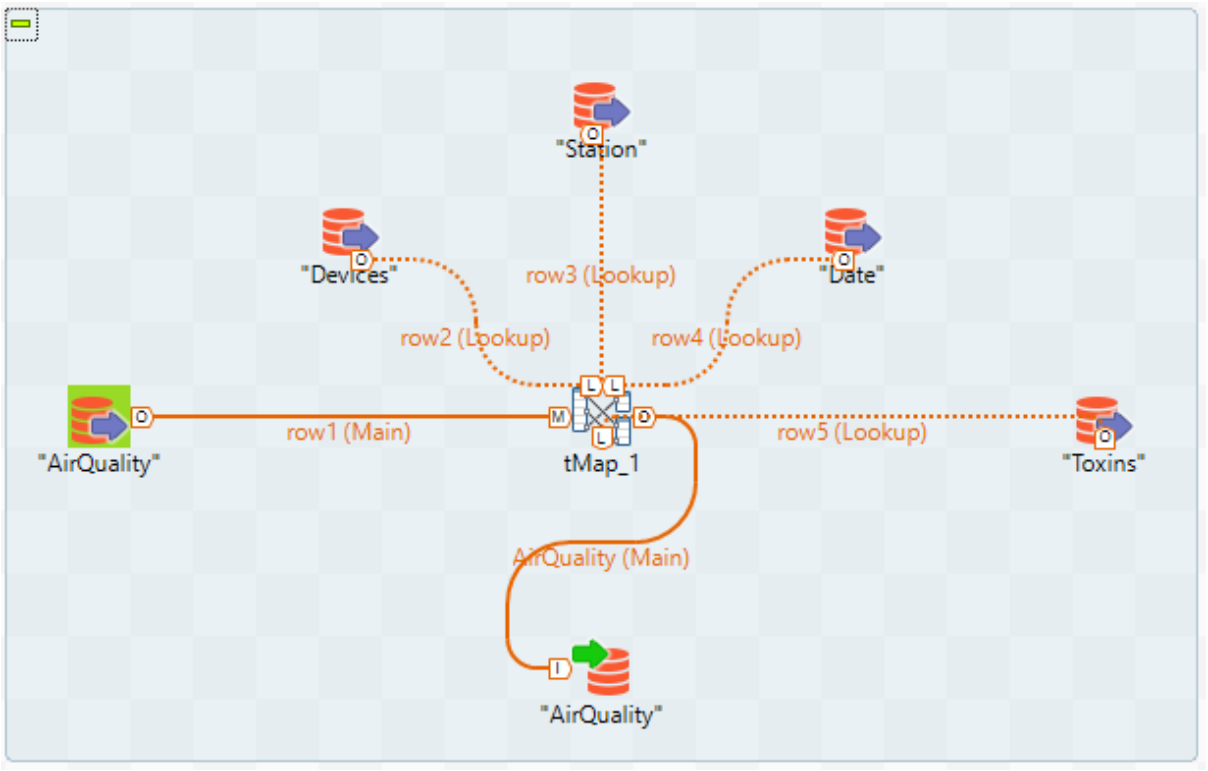

MS-SQL and MY-SQL(Merging, Filtering and Splitting of Fields):

Here the databases are mapped to tMap and which got different data sets and by using the transformations like tExtractDelimitedFields_1 and tMap we merge, filter and splitting of fields and push the job to stations tabnle.



## 4.4. Process of Populating Fact Tables

The source of Fact tables is collected (like Spreadsheet, Flat File, MY SQL, MS-SQL) and data is extracted from it. We create a job for individual and perform transformations. The Foreign keys of all the dimensions table are collected by joining operational keys of the transformed fact data and the operational key of the dimension table. The Foreign key and its respective measures are loaded into Fact Table.




# 5. Physical Data Warehouse Design

## 5.1. Aggregate Fact Tables / generalized cuboids

In these Fact Table / generalized cuboids data is rolled up to increase the query performance. The Aggregates are derived from Fact Tables. They have fewer rows than the Fact table. Then the Aggregate table summaries of most granular data at the higher levels along the dimension hierarchies.

Generally, in Fact table rows represent the numbers with lowest level of dimension hierarchy. By moving the hierarchy to the next level and we obtain the aggregate fact tables. Aggregate tables are advantageous as it reduces the input, output, CPU and RAM usage.

## 5.2. Aggregate fact tables / generalized cuboids to create

- Station: Station Name, Station Type (SLAMS), Address, City, County, Region, State (AL)
- Date: Hour, Day, Month, Year, Season
- Devices: Device Name, Min Range, Max Range, Units, Device Accuracy, Description
- Toxins: Name, Chemical Formula, Standard, Max Exposure Time, Max Concentration limit, Description

Concept Hierarchies of this project are Station and Date.

Station: Street→City→County→Region

Date: Hour→Day→Month→Season→Year

Cuboid-1:[Season, Region, Device Name, Toxin Concentration] where Toxin Name: Nitrogen Dioxide.

Aggregating the data as per cloud-1 gives the information of average Nitrogen Dioxide level concentration in terms of seasons. This helps to analyse the levels of ozone based on the seasonal changes.

Cuboid-2:[County, Toxin Concentration, Device Name, Year] where year = 2019

This cuboid aggregates the data of average Air Quality index recorded in terms of County in the past year 2019.

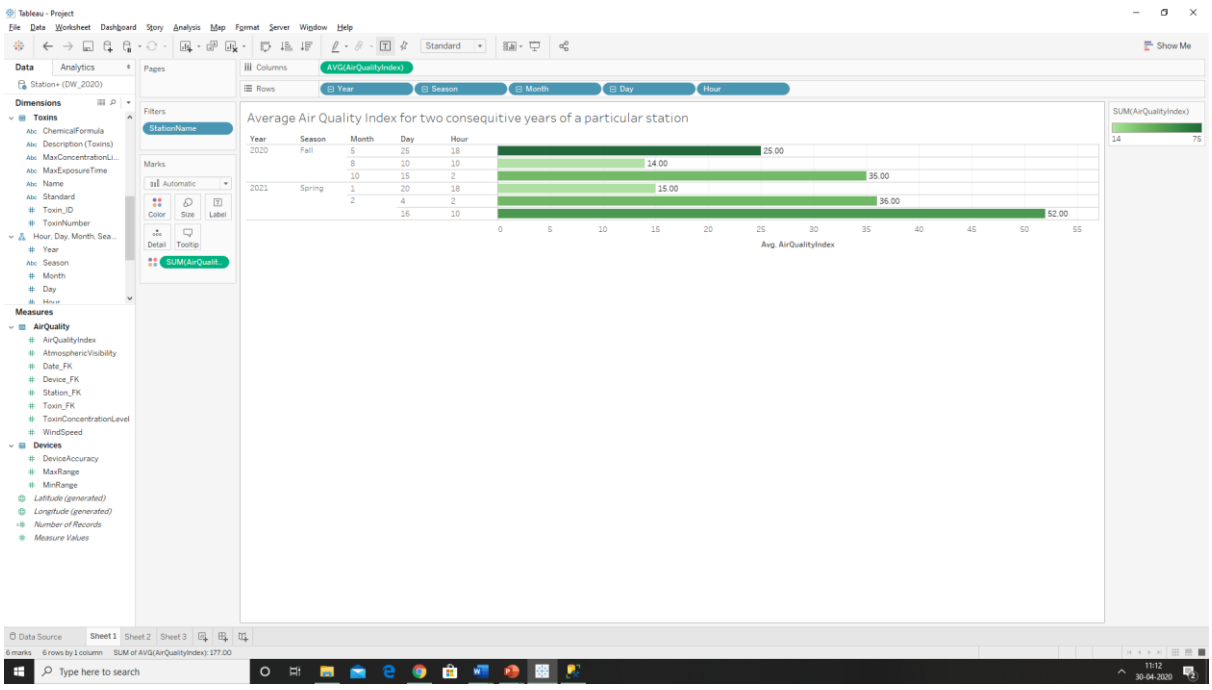
# 6. OLAP and Intelligence Applications

Data Warehouse have deeply rooted applications in every industry which uses structured and unstructured data from disparate sources for forecasting, analytical reporting, and business intelligence, allowing for robust decision-making. Here are some major applications of data warehouses across different industries:

- Banking
- Finance
- Government
- Education
- Healthcare
- Insurance
- Retail and
- Services.

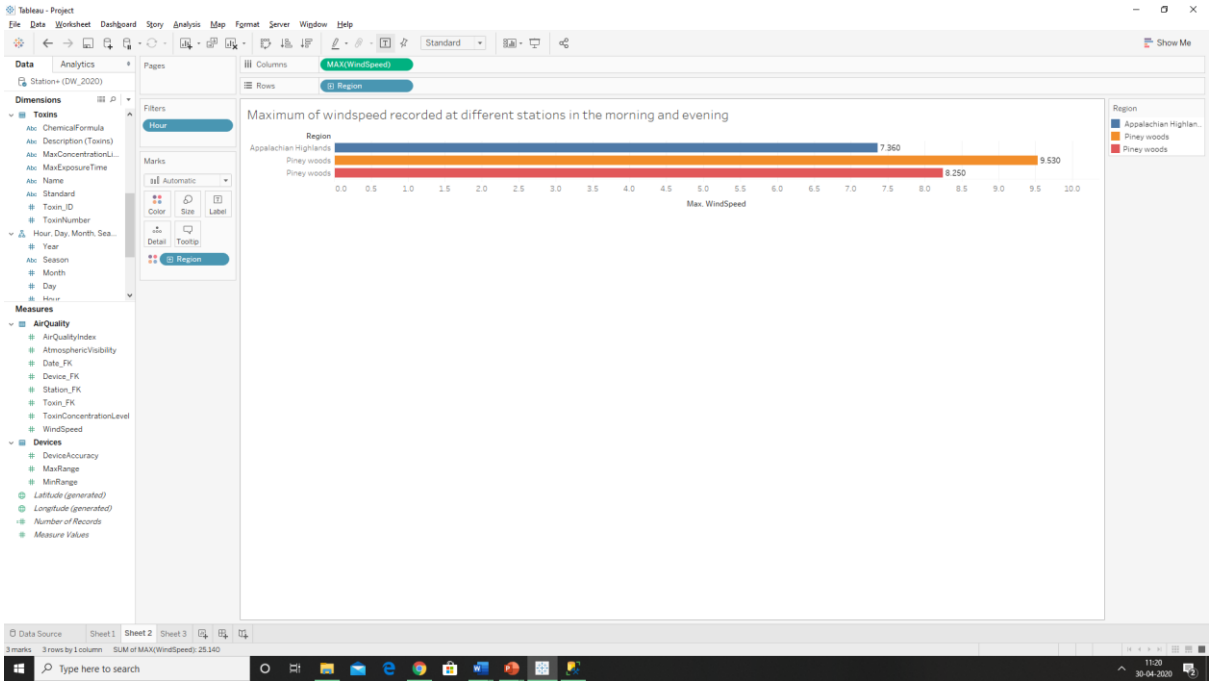
## 6.1. Visual Analytics Reports

**Report 1:** Average Air Quality index for two consecutive years of a design.

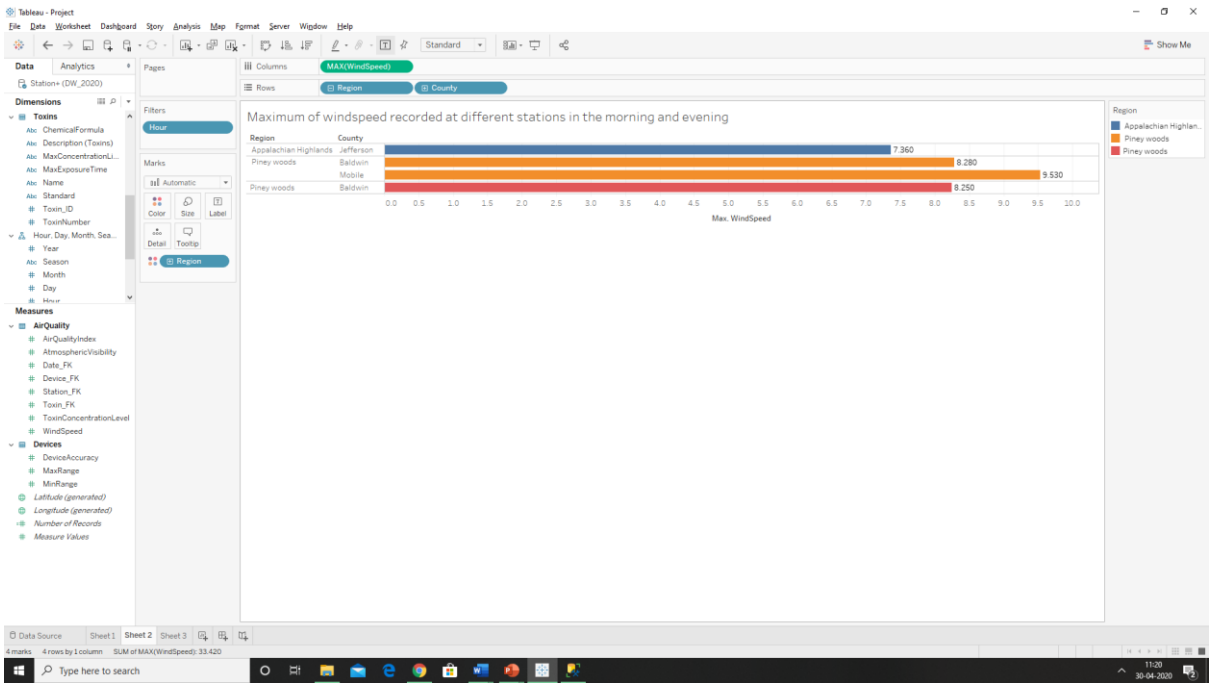



**Report 2:** Maximum of windspeed recorded at different stations in the morning and evening.

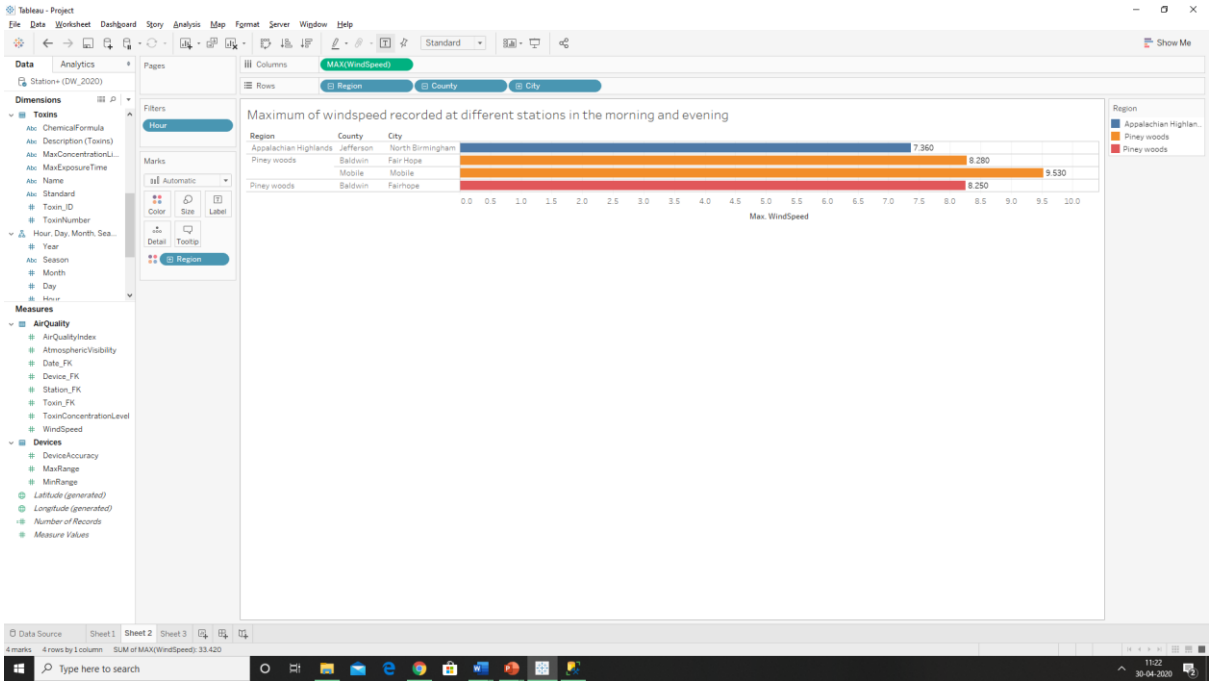
## Maximun Wind speed: Region



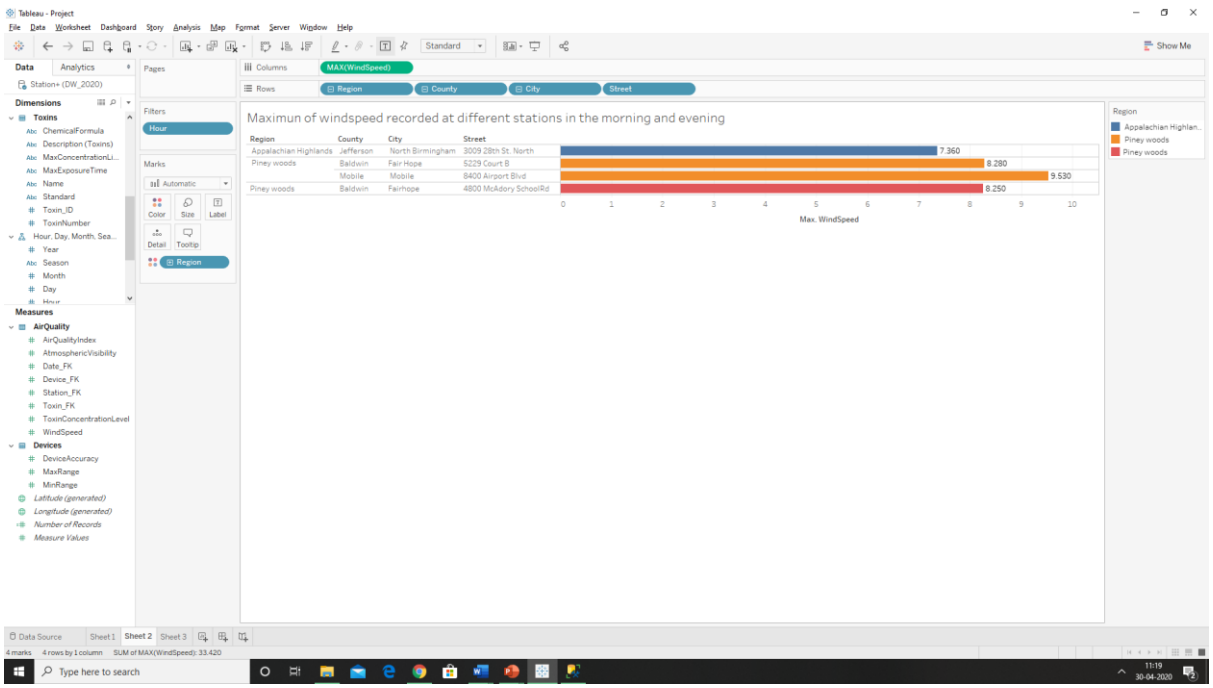

## Maximun Wind speed: Region→County



Maximun Wind speed: Region→County→City




Maximun Wind speed: Region→County→Street




**Report 3:** Minimum concentration level recorded based on Station name, Chemical Formulae and Units.

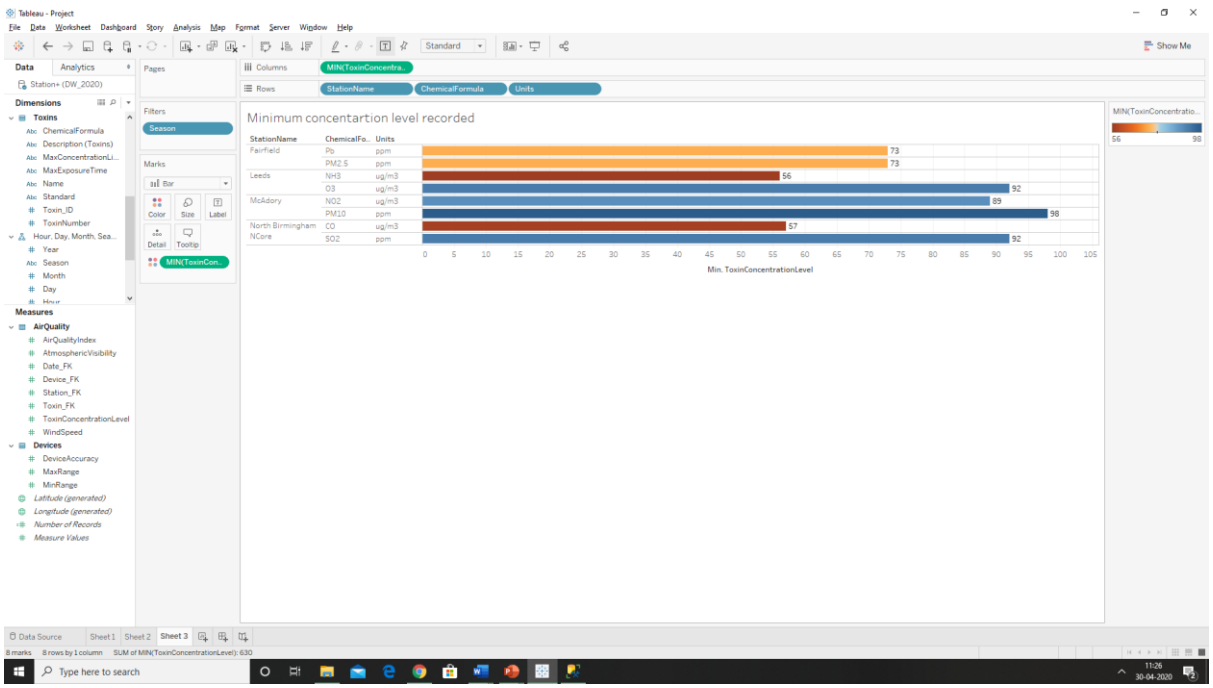


Tableau - Project
Minimum concentartion level recorded
StationName
ChemicalFo..
Units
Fairfield
Leeds
McAdory
North Birmingham
NCore
Min. ToxinConcentrationLevel