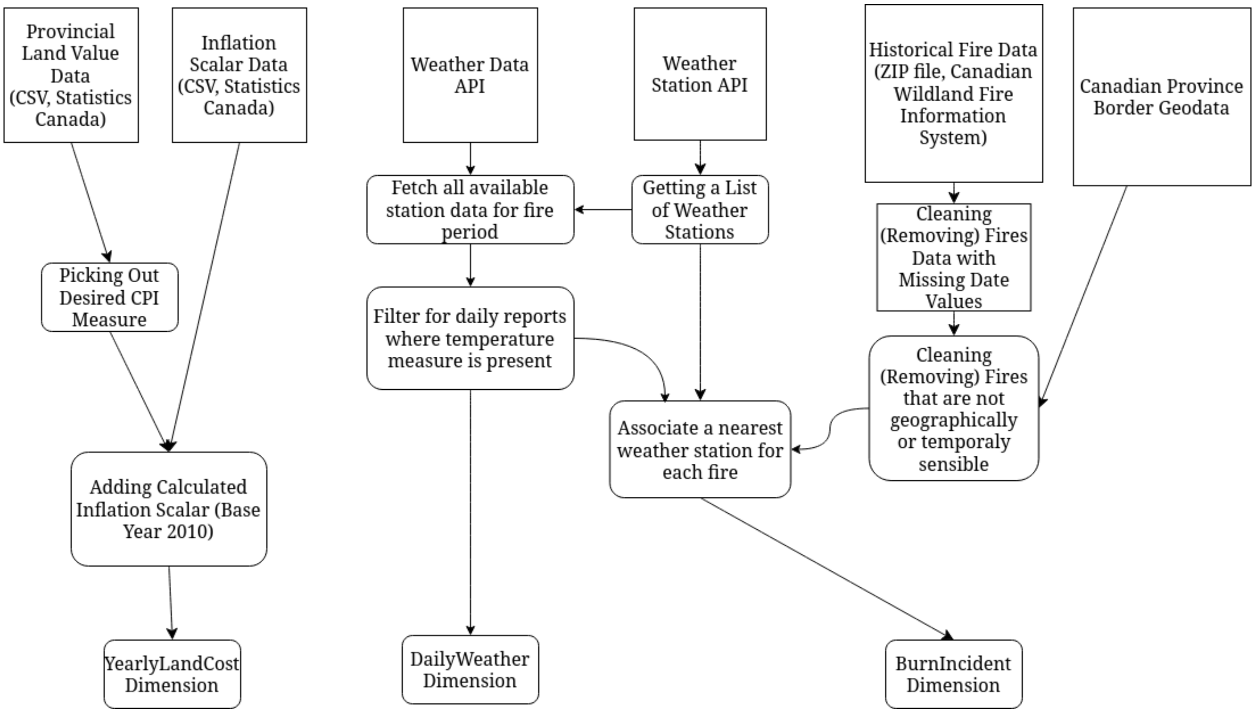
Data Science Project Phase 2: Physical Design and Data Staging

Group Members:

- Jacob Kszan
- Malte Blomqvist
- Alexander De Furia

A: High Level Data Staging Plan Schematic


Provincial Land Value Data (CSV, Statistics Canada)
Inflation Scalar Data (CSV, Statistics Canada)
Weather Data API
Weather Station API
Historical Fire Data (ZIP file, Canadian Wildland Fire Information System)
Canadian Province Border Geodata
Fetch all available station data for fire period
Getting a List of Weather Stations
Cleaning (Removing) Fires Data with Missing Date Values
Picking Out Desired CPI Measure
Filter for daily reports where temperature measure is present
Cleaning (Removing) Fires that are not geographically or temporaly sensible
Associate a nearest weather station for each fire
Adding Calculated Inflation Scalar (Base Year 2010)
YearlyLandCost Dimension
DailyWeather Dimension
BurnIncident Dimension

B: Extra Details

To properly link each wildfire to a corresponding weather station we had to perform a Nearest Neighbour search of the wildfire on the set of all active weather stations. We used the lat/long coordinates of each weather station and fire to perform this search. To reduce the amount of weather reports stored, we used the fire data in the staging of weather data to only query / include weather reports from stations that were nearest to at least one wildfire in our dataset.

Additionally, note that we used an inflation scalar to project nominal costs of a hectare of burned land across years. This is a form of normalization, standardizing the dollar value of land while mitigating / removing the influence of inflation. This will allow us to have a better view of what the relative value lost because of wildfires is across time, without this normalization we couldn't compare this data across years as inflation would cause the value of each dollar (and by extension the actual economic cost of the fires) to be different.

C: List of Data Quality Issues and Solutions

1. Fire data
    a. Fires occurring outside of Canada (i.e. In the Ocean, in Alaska)
        - By using geographic boundaries of Canada's different provinces retrieved from separate data source we drop (remove) those fire incidents whose geographic location is outside of Canada.

    b. Incomplete fire entries
        - Many of the reported fire incidents where lacking start and end dates, as we had a plethora of fires to still work with we chose to drop those fires without start and end dates.

    c. Multi-year long fire event outliers
        - These outliers with their immense amount of hectares burnt would throw off the analysis made for fire events able to be quantified on a per day data-grain. As such it was simple to remove the very few occurrences of the outlier 'more than 100 day' fires.

    d. Fire data only specifying hectares burnt on a per-fire basis
        - To solve this issue an estimate of hectares burnt by day was made by splitting up each fire event into one event per day and allowing the assumption that the burn rate could be split equally across the days the fire was active.
2. Weather Data
    a. API not including Average Humidity and Windspeed
        - To solve this problem, we altered our database schema to represent a more realistic measurement, "Max Humidity" and "Max Windspeed"

    b. Different API functions using ClimateID and StationID to identify weather stations
        - Our solution to this problem was to use ClimateID for our API calls and create a mapping from the ClimateID to the corresponding StationID, converting all instances of ClimateID into StationID in our final weather reports

    c. Missing Values for Max Humidity and Windspeed

        - Many weather stations do not report Max Humidity and Max Windspeed (With nearly 70% missing one or both!). We believe that temperature will be the most interesting measure to analyze so we decided not to drop FACTs where either Humidity or Windspeed is missing. Instead, we keep the Null values in the database where they occur.
        - In future analysis of humidity and windspeed, we plan to filter the FACT table / create views on the fact table that look at the cases where humidity and windspeed are instantiated
    d. API occasionally skipping reports due to query size
        - We had this issue when attempting to query all of the station's report data in one request. To solve this issue, we set the limit of the query to 10000 and requested the data in two requests, one spanning from 1987 - 2000 and the other from 2000 - 2023.
    e. Weather stations occasionally not reporting on days
        - We don't include rows in the FACT table where the corresponding weather data does not exist.
        - To minimize the number of these days, we assign closest weather-stations on a per day per fire basis, so if a station lacks data during a day of a fire, we assign it another weather station with data, even though it might be further away.
    f. Weather stations not having reports before their creation / after their termination.
        - While performing Nearest Neighbour to find the closest weather station to a fire, we only consider those stations that were active for the days of the fire.

3. Land Cost and Inflation
    a. The cost of land does not take inflation into account
        - To solve this issue, we normalized the values using yearly Consumer Price Index (CPI) measures calculated by Statistics Canada. These measures are well trusted economic tools and are used to adjust for inflation across different years.

D: Team Planning Spreadsheet

CSI4142 - Project W23
Phase 2- Physical design and data staging
Teamwork - breakdown of duties

Deliverable checklist	Responsible team member(s)	Expected completion date	Actual completion date	Estimated time (hours) to complete	Actual time (hours) to complete	Notes (if any)
Create database Docker image instance / initialization pipeline	Jacob Kszan	13/03/2024	12/03/2024	2	1,5	
Adjusting SQL schema to include lookup tables	Jacob Kszan	22/03/2024	22/03/2024	1	1	
Writing/organizing the surrogate key generation pipeline	Jacob Kszan	22/03/2024	21/03/2024	1	2	
Loading SQL schema into the Docker image	Jacob Kszan	14/03/2024	13/03/2024	1,5	1	
Creating preprocessing script for hotspot (Fire) data	Malte Blomqvist, Alex De Furia	22/03/2024	20/03/2024	6	16	
Fire data restructuring and cleaning	Malte Blomqvist	18/03/2024	22/03/2024	3	4	
Creating preprocessing script for yearly land cost	Malte Blomqvist	22/03/2024	21/03/2024	0,5	0,5	
Calculating Inflation across years for yearly land cost	Malte Blomqvist	22/03/2024	20/03/2024	0,5	0,5	
Creating nearest neighbour script(s) to link fires to their nearest weather stations	Malte Blomqvist, Alex De Furia	22/03/2024	26/03/2024	7	18	This was much more difficult than anticipated due to issues with the weather API
Staging of fact table	Jacob Kszan	22/03/2024	25/03/2024	2,5	2	
Creating Data Staging Scripts	Alex De Furia and Jacob Kszan	22/03/2024	24/03/2024	10	15	
Automating injestion of produced CSV data files into SQL	Alex De Furia	22/03/2024	23/03/2024	2	3	Had to change schema slightly and create additional lookup tables to avoid using operational keys as primary keys
Parsing / Validating Firedata CSVs	Alex De Furia	22/03/2024	21/03/2024	1	1	
Parsing / Validating Land Cost CSVs	Alex De Furia	22/03/2024	21/03/2024	1	1	
Troubleshooting/Bugfixing the SQL schema	Jacob Kszan	22/03/2024	23/03/2024	1	3	
Troubleshooting/Bugfixing the NN algorithm	Malte Blomqvist, Alex De Furia	22/03/2024	26/03/2024	2	6	
Troubleshooting/Bugfixing the staging scripts	Malte Blomqvist, Alex De Furia, Jacob Kszan	22/03/2024	26/03/2024	2	8	Rewrote the staging scripts to move from a direct SQL injection to a CSV-based approach
Discussing SQL Schema Redesign and Solving API Access Issues	Malte Blomqvist, Alex De Furia, Jacob Kszan	22/03/2024	26/03/2024	4	4	