Snowstorm

# Database Design Document

1. Version 0.5
2. 03/11/2019
3. Team: **Ilia Kassianenko, Pawel Kaluski, Fritz Gyger**

Table of Contents

[1. Overview 3](#_Toc2624387)

[2. Assumptions/Constraints/Risks 5](#_Toc2624388)

[2.1 Assumptions 5](#_Toc2624389)

[2.2 Constraints 6](#_Toc2624390)

[2.3 Risks 6](#_Toc2624391)

[3. Design Decisions 7](#_Toc2624392)

[3.1 Key Factors Influencing Design 7](#_Toc2624393)

[3.2 Functional Design Decisions 7](#_Toc2624394)

[3.3 Database Management System Decisions 8](#_Toc2624395)

[3.4 Security and Privacy Design Decisions 8](#_Toc2624396)

[3.5 Performance and Maintenance Design Decisions 8](#_Toc2624397)

[4. Detailed Database Design 10](#_Toc2624398)

[4.1 Roles and Responsibilities 12](#_Toc2624399)

[4.2 Performance Monitoring and Database Efficiency 12](#_Toc2624400)

[4.2.1 Operational Implications 12](#_Toc2624401)

[4.2.2 Data Transfer Requirements 12](#_Toc2624402)

[4.2.3 Data Formats / Data Dictionary 13](#_Toc2624403)

[Appendix A: Acronyms 20](#_Toc2624404)

[Appendix B: DDL 21](#_Toc2624405)

[Appendix C: Decision chart database system 22](#_Toc2624406)

[Appendix D: Decision tree 23](#_Toc2624407)

No table of figures entries found.

List of Tables

[Table 1 - Acronyms 9](#Table!1|sequence)

## Overview

The purpose of this project is to identify the quantity of snow moved for the City of Montreal to a designated disposal area depot in order to help the city of Montreal as well as the independent snow removers to optimize the snow removal.

Our new application will be complementary to the existing applications:

**Loading**

Snow loading involves picking up snow that was plowed to the side of the road during the loading operations. Snow loading depends on the amount of snow that has fallen, as well as the weather forecast.

*–* ***Planif-Neige*** *est utilisé par les arrondissements pour planifier les parcours de chargement de la neige. Il indique l’avancement des opérations. Planif-Neige alimente les applications telles qu’INFO-Neige MTL et la carte Web par les données ouvertes.*

Since the winter of  2014, Montréal has used specialized software that feeds the INFO-Neige application (available on the [AppleStore](https://itunes.apple.com/ca/app/info-neige-mtl/id935347695?mt=8" \t "_blank) and [GooglePlay](https://play.google.com/store/apps/details?id=com.heritagesoftware.infoneige" \t "_blank)) using [open data](http://donnees.ville.montreal.qc.ca/dataset?q=&tags=D%C3%A9neigement&sort=score+desc%2C+metadata_modified+desc).  The app, which was developed by Sidekick Interactive, helps to accelerate snow loading operations by notifying residents about parking restrictions and places where they can move their car.

**Disposal**

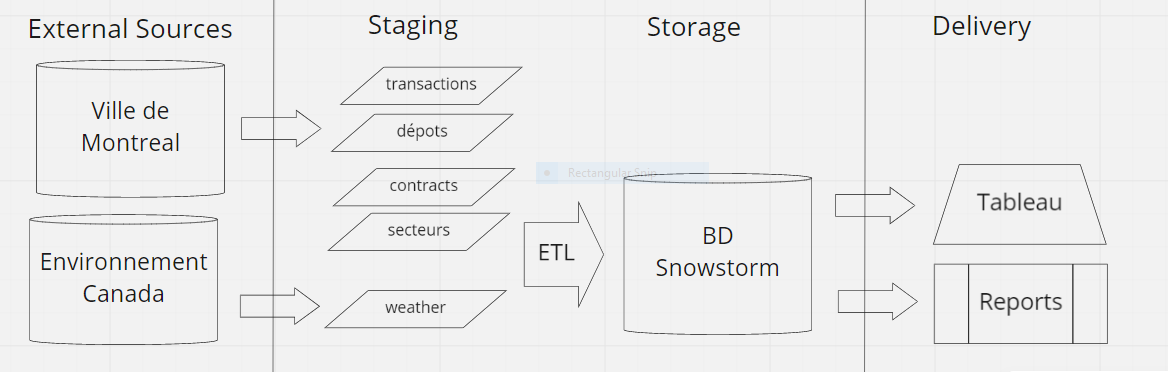
During a snow removal operation, trucks take snow to one of [several disposal sites](http://ville.montreal.qc.ca/snowremoval/elimination-neige#carte-elimination).

The average volume of snow taken to disposal sites per year is 12 million cubic meters – 300,000 truckloads. Meltwater from disposal sites is recovered and treated according to environmental standards.

*–* ***SIT-Neige*** *comprend un système et des équipements télémétriques utilisés pour gérer les transactions liées aux opérations de chargement et d’élimination de la neige. Les GPS fournissent les informations de localisation des souffleuses aux applications Info-Neige MTL et à la carte de déneigement.*

Since the winter of 2014, Montréal has used specialized management software to optimize snow removal operations (loading, transportation and elimination) and monitor billing.

The data is acquired from two main **Data sources:**



We plan to use snow removal information from the city of Montreal and weather information from Environment Canada.

For details of the source data, see chapters [2.1. Assumptions](#_Assumptions) and [4.2.3 Data Formats / Data Dictionary](#__RefHeading___Toc9081_565685251)

Each **depot** has a specified capacity, except for a sewage depot, where the runoff sewage systems capacity is not specified. Also, the quantity of daily transactions per depot is considered to be the quantity of visits to unload snow at a specific depot. The daily frequency for a depot is obtained by dividing the quantity of visits for a depot by the total quantity of visits to all depots on a specific day.

Each daily snow volume unloaded at each depot is divided by the recorded daily average depth of the precipitation. The result from the division is an order of magnitude estimate for the daily surface area that has received precipitation and that has been served by the fleet of vehicles.

Also, by dividing the quantity of visits by the surface area where snow was removed, a service ratio is obtained for a specific depot and day. The difference in time for a transaction between the snow loading and unloading time is considered to be the cycle time. The project herein does not consider the effect on the cycle time following re-assignment of vehicles to depots.

Finally, this analysis will produce for each depot the daily frequency of use and the daily service ratio. Consequently, the analysis will show if constrained capacity depots are used more frequently daily than a sewer depot, and the analysis will show at which depot there is an high quantity of vehicles unloading snow.

The data can also be used to compare the different depots, boroughs and their sectors or the different contracts. For example which contract delivers the best value?

We plan to keep 10 years of history in order to compare different year and are able to see trends.

Since our system is informational (vs operational) we use the **BASE property** for the project:

*As discussed, the Data Property should be Base. Specifically, there is an expectation that the CSV will be consistently available every Monday at 1 AM EST. Also, the city of Montreal provides a disclaimer that previously released records may be corrected retroactively. Service failure is not critical within our environment, since the city of Montreal holds all records. If the portal fails or records become unavailable, the native files are assumed to be available from city of Montreal snow removal management system, in reference to the disclaimer provided. One transaction and one contract file are downloaded once a week, and one weather report is obtained once a week. There are no other consistent inputs, while Deports are assumed to remain unchanged. Data may overwritten when the CSV files are downloaded from the city of Montreal open data portal. ???*

## Assumptions/Constraints/Risks

### Assumptions

Instructions: Describe any assumptions or dependencies regarding the database design for the system. These may concern such issues as: related software or hardware, operating systems, or end-user characteristics.

We will use the following data sources:

T

### Constraints

Instructions: Describe any limitations or constraints that have a significant impact on the database design for the system.

* Weekly extractions as the files are updated weekly by the city, every Monday at 1am EST between November 1st – April 1st.
* Extraction of weather data from Environment Canada at the same time as the one from the city.

### Risks

Instructions: Describe any risks associated with the database design and proposed mitigation strategies.

* Changes of source URL & source format
* Inconsistency of data (*the city of Montreal provides a disclaimer that previously released records may be corrected retroactively).*

## Design Decisions

Instructions: Utilizing the following subsections, describe decisions made that impact the proposed database design. This should include the platform and database management system (DBMS) chosen for the project. Include any other information relevant to the database design decisions (e.g., Data Conversion Plan, Service Level Agreements (SLAs)). The Design Decisions section is written at a higher level than the subsequent Detailed Database Design section, and provides an understanding and rationale for the content in the Detailed Database Design section.

We looked at Google, Amazon, Azure, and Digital Ocean.

Google $8.43 USD/Month:

* 0.6GB of RAM
* Max storage of 3 GB
* Max connections 250

Amazon $11.28 USD/Month ($406 for 3 years):

* 1 server
* 1 core
* 1 GB of RAM
* 1 GB SSD

Azure $984.36USD/Month

* Single Database
* vCore Purchase Model
* 3 year reserved 5 GB

Digital Ocean $25USD/month:

* 1GB RAM
* VM platform with a fully managed PostgreSQL
* 250GB Storage

Storage options: <https://www.digitalocean.com/products/managed-databases/> Pawel

### Key Factors Influencing Design

Instructions: Describe key functional or non-functional requirements that influenced the design. If all such decisions are explicit in the requirements, this section shall so state. Design decisions that respond to requirements designated as critical (e.g., those for performance, availability, security, or privacy) shall be placed in separate subparagraphs.

Because of the nature of our data (structured data), its low volume (MB), frequency (weekly updates) and the use cases (aggregations) we decided to go with a relational database.

See also Appendix C & D.

Warm storage????

### Functional Design Decisions

Instructions: Describe decisions about how the database will behave in **meeting its requirements from a user's point of view** (i.e., functionality of the database from an application perspective), ignoring internal implementation, and any other decisions affecting further design of the database. Include decisions regarding inputs the database will accept **and outputs** (displays, **reports**, messages, responses, etc.) it will need to support, including interfaces with other systems. Describe the general types of processing (sequential versus random for inserts, updates, deletes and queries) required both for data entering the database, and data most frequently accessed. Also include decisions on **how databases/data files will appear to the user**.

Inputs are csv files obtained from different sources, loaded in weekly batch processes on Monday mornings.

For the **reporting** we plan to use Tableau to develop reports and dashboards. We may give external customer access for a small fee (tbd) for self-service.

Currently there are no interfaces with other systems planned, but could be added later.

### Database Management System Decisions

Instructions: Describe design decisions regarding the DBMS intended for the initial implementation. Provide the name of the **DBMS, the reason for selection**, and the **type of flexibility** built into the database for adapting to changing requirements.

Python to develop the ETL process and the data cleansing. See rules in section 4.3.

?

### Security and Privacy Design Decisions

Instructions: Describe design decisions on the levels and types of security and privacy to be offered by the database. General descriptions of classifications of users and their general access rights should be included.

No confidential or private data, all data retrieved is publicly available on the web.

No IP from outside Canada is allowed to use the system.

Basic security system, firewall, etc.

Depending on DBMS?

### Performance and Maintenance Design Decisions

Instructions: Describe how performance and availability requirements will be met. Examples include:

* Describe design decisions **on database** distribution (such as client/server), master database file updates and maintenance, including maintaining consistency, establishing/ reestablishing and maintaining synchronization, enforcing integrity and business rules.
* Describe design decisions to address concurrence issues (e.g., how the data are partitioned or distributed to support multiple applications or competing update functions, if applicable).
* Describe design decisions to support Service Level Agreements (SLAs) for key functions supported by the database.

Eastern time zone home hours during week days: 9 AM – 5PM EST. Application failure is first investigated each week day. Normal site operations during week days between 8 AM – 8 PM EST, which enables nationwide reach; outside site operations hours, the database downloadable content is accessible and support tickets can be created. Embedded analysis tools are disabled outside this period. Although, support ticket resolution will occur the following week day. This permits downtime for operations, for archiving purposes and for automated ETL processes for the source files.

* Describe design decisions on **backup and restoration** including data and process distribution strategies, permissible actions during backup and restoration, and special considerations for new or non-standard technologies such as video and sound. Describe the impact this maintenance will have on availability.

Daniel: In case of a structured database which will be the backup policy in place. = depends on Provider we’ll chose (Pawel?)

* Describe design decisions on data reorganization (i.e., repacking, sorting, table and index maintenance), synchronization, and consistency, including automated disk management and space reclamation considerations, optimizing strategies and considerations, storage and size considerations (e.g., future expansion), and population of the database and capture of legacy data. Describe the impact this maintenance will have on availability.

Daniel : "Describe design decisions on data reorganization" is if the data will be stored as **SSD** or HDD, RAM, data heat (hot **warm** cold), and if you guys have some policies on moving data from "hot" to "warm" type of storage, is it expected to have a stop on the application to do so?

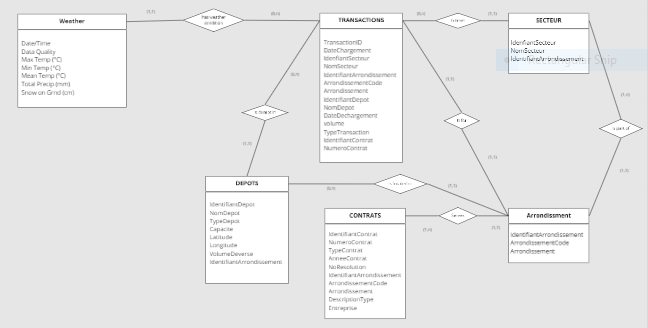
* Describe design decisions to support purging and/or archiving of data to ensure performance and storage objectives are met. Describe the impact this maintenance will have on availability. Describe any needs to recall archived data back into the database.

Since we have low volume and plan to keep 10 years history, no data need to be archived.

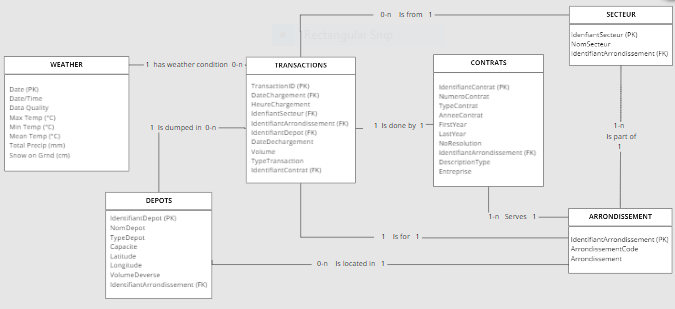
## Detailed Database Design

Instructions: Describe the design of all DBMS structure associated with the system. The headings and sub-headings in this section should be structured according to the information to be presented, and may include discussions about or references to the following:

* Conceptual Data Model (CDM)



* Logical Data Model (LDM) and LDM Entity Relationship Diagram (ERD).



<https://realtimeboard.com/app/board/o9J_kyQw7mY=/?userEmail=fgyger@gmail.com&invite=2aaaaaaadcc970d363df49da3d893532-f09c55633fb9bfe1-e627fdad19a479bc-f83312ccebd6dc98&event=mailInvite&mailUserEmail=fgyger@gmail.com&track=true%22>

* Physical Data Model (PDM) with a description of the DBMS schemas, sub-schemas, records, sets, tables.

See Appendix 3 DDLs

* A comprehensive Data Dictionary showing data stores, data element name, type, length, source, constraints, validation rules, maintenance (create, read, update, delete (CRUD) capability), audit and data masking requirements, expected data volumes, life expectancy of the data, information life-cycle management strategy or at least an archiving strategy, outputs, aliases, and description.

CRUD – text to be added

* Planned implementation factors (e.g., distribution and synchronization) that impact the design. Depending on DB provider - Pawel
* Estimate of the DBMS file size or volume of data per entity.
* Volume :
  + Initial : about 35 MB
  + Weekly growth: about 10MB for transaction & weather data.
* Definition of the update frequency of the database tables, views, files, areas, records, sets, and data pages. Also provide an estimate of the number of transactions, if the database is an online transaction-based system.
* Transactions & Weather :
  + Weekly Mondays after 1am : the city updates the files every Monday at 1am EST
  + Weather Mondays after 1am

The detailed database design information can be included as an appendix, such as DDLs, which would be referenced here.

### Roles and Responsibilities

Instructions: Identify the organizations and personnel responsible for the following database administrative functions: database administrator, system administrator, and security administrator. Describe specific administration skill requirements applicable to the database.

DBA : SQL,

Systems admin :

Security admin : n/a

Data Acquisition & cleansing : ETL

Daniel : At 4.1 is simply listing the departments which will be responsible to manage the system. for example "The database administration will be handled by the Infrastructure department, where as "Query optimization and error handling" will be handled by production support.

### Performance Monitoring and Database Efficiency

#### Operational Implications

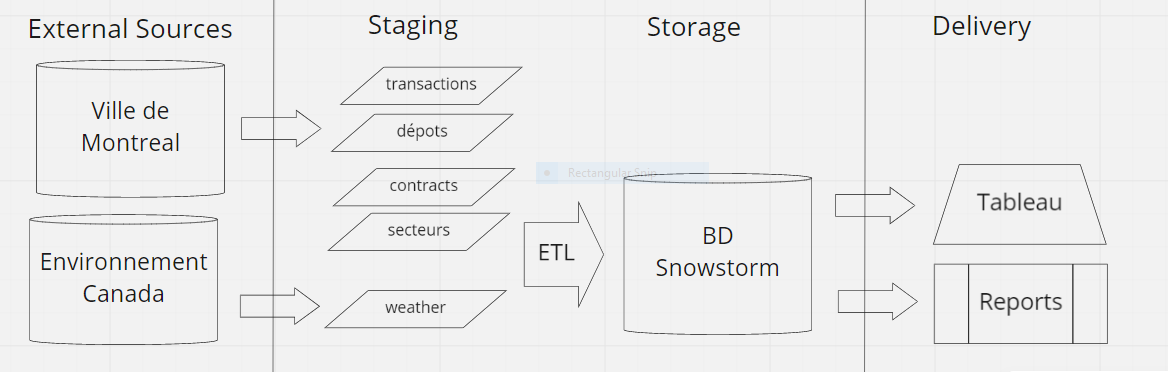
Instructions: Describe operational implications of data transfer, refresh and update scenarios and expected windows.

* Transactions & Weather :
  + Weekly Mondays after 1am : the city updates the files every Monday at 1am EST
  + Weather Mondays after 1am

#### Data Transfer Requirements

Instructions: Describe data transfer requirements to and from the software, including data content, format, sequence, volume/frequency and any conversion issues.

The data is acquired from two main **Data sources:**



* Extract source files in UTF-8 format in order to have French accents (ex. Saint-Léonard instead of Saint-LÃ©onard)
* Volume :
  + Initial : about 35 MB
  + Weekly growth: about 10MB for transaction & weather data.
* Data content and format (See [4.2.3 Data Formats](#_Data_Formats_/))

#### Data Formats / Data Dictionary

Instructions: Describe formats of data for both the sending and receiving systems, including the data item names, codes, or abbreviations that are to be interchanged, as well as any units of measure/conversion issues.

**Source files**

**depots**\_deneigement\_saison\_2018-2019, .csv files

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| IdentifiantDepot | INT |  | N |  |
| NomDepot | VARCHAR | 30 | N |  |
| TypeDepot | VARCHAR | 15 | N | Values : Carierre, Chute\_Egout, Entassement, |
| Capacite | INT |  | Y | 0 = Sewer well |
| Latitude | DECIMAL | 8,6 | N |  |
| Longitude | DECIMAL | 8,6 | N |  |
| VolumeDeverse | DECIMAL | 9,2 | N |  |
| IdentifiantArrondissement | INT |  | N |  |
| ArrondissementCode | CHAR | 3 | N |  |
| Arrondissement | VARCHAR | 30 | N |  |
| MTM8\_X | DECIMAL | 7,1 | N | Unknown Ilia |
| MTM8\_Y | DECIMAL | 8,1 | N | Unknown Ilia |

**secteurs**\_deneigement\_saison\_2018-2019, csv file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| IdenfiantSecteur | INT |  | N |  |
| NomSecteur | VARCHAR | 7 | N |  |
| IdentifiantArrondissement | INT |  | N |  |
| ArrondissementCode | CHAR | 3 | N |  |
| Arrondissement | VARCHAR | 30 | N |  |

**Data dictionary:** <http://donnees.ville.montreal.qc.ca/dataset/contrats-transaction-deneigement>

**contrats**\_deneigement\_saison\_2018-2019, csv file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| IdentifiantContrat | INT |  | N |  |
| NumeroContrat | VARCHAR | 7 | N |  |
| TypeContrat | VARCHAR | 12 | Y | Values : Déneigement, Régie, Transport, blank |
| AnneeContrat | CHAR | 9 | N |  |
| NoResolution | VARCHAR | 11 | Y |  |
| IdentifiantArrondissement | INT |  | N |  |
| ArrondissementCode | CHAR | 3 | N |  |
| Arrondissement | VARCHAR | 30 | N |  |
| DescriptionType | VARCHAR | 100 | Y |  |
| Entreprise | VARCHAR | 50 | N |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

**transactions**\_deneigement\_saison\_2018-2019, csv file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| TransactionID | INT |  | N |  |
| DateChargement | TIMESTAMP | 19 | N | YYYY-MM-DD hh:mm:ss |
| IdenfiantSecteur | INT |  | Y |  |
| NomSecteur | VARCHAR | 7 | Y |  |
| IdentifiantArrondissement | INT |  | N |  |
| ArrondissementCode | CHAR | 3 | N |  |
| Arrondissement | VARCHAR | 30 | N |  |
| IdentifiantDepot | INT |  | N |  |
| NomDepot | VARCHAR | 30 | N |  |
| DateDechargement | TIMESTAMP | 19 | N |  |
| Volume | DECIMAL | 4,2 | N |  |
| TypeTransaction | VARCHAR | 7 | N | Values :  AUT, N-AUT, BARCODE |
| IdentifiantContrat | INT |  | Y |  |
| NumeroContrat | VARCHAR | 7 | Y |  |

**weather\_data**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| Date/Time | DATE | 10 | N | YYYY-MM-DD |
| Year | INT | 4 | N |  |
| Month | INT | 2 | N |  |
| Day | INT | 2 | N |  |
| Data Quality |  |  | Y | Unknown type, unknown length – not used by our system |
| Max Temp | DECIMAL | 3,1 | Y |  |
| Max Temp Flag |  |  | Y | Unknown type, unknown length – not used by our system |
| Min Temp (°C) | DECIMAL | 3,1 | Y |  |
| Min Temp Flag |  |  | Y | Unknown type, unknown length – not used by our system |
| Mean Temp (°C) | DECIMAL | 3,1 | Y |  |
| Mean Temp Flag |  |  | Y | Unknown type, unknown length – not used by our system |
| Heat Deg Days (°C) | DECIMAL | 3,1 | Y |  |
| Heat Deg Days Flag |  |  | Y | Unknown type, unknown length – not used by our system |
| Cool Deg Days (°C) | DECIMAL | 3,1 | Y | Always 0 |
| Cool Deg Days Flag |  |  | Y | Always blank |
| Total Rain (mm) |  |  | Y | Unknown type, unknown length – not used by our system |
| Total Rain Flag |  |  | Y | Unknown type, unknown length – not used by our system |
| Total Snow (cm) |  |  | Y | Unknown type, unknown length – not used by our system |
| Total Snow Flag | CHAR | 1 | Y | Value : M |
| Total Precip (mm) | DECIMAL | 3,1 | Y |  |
| Total Precip Flag | CHAR | 1 | Y | Value : M |
| Snow on Grnd (cm) | INT | 2 | Y |  |
| Snow on Grnd Flag |  |  | Y | Unknown type, unknown length – not used by our system |
| Dir of Max Gust (10s deg) | INT | 2 | Y |  |
| Dir of Max Gust Flag | CHAR | 1 | Y | Value : M |
| Spd of Max Gust (km/h) | INT | 2 | Y |  |
| Spd of Max Gust Flag | CHAR | 1 | Y | Value : M |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

**Validation rules:**

* DATES : Need to be valid dates in the right sequence

(ex. 2106-02-07 found in source data)

**Transformation rules:**

* TRANSACTIONS
  + Split DateChargement (format YYYY-MM-DD hh:mm:ss) into   
    DateChargement (YYYY-MM-DD) and HeureChargement (hh:mm:ss) to allow easy join with DATE (YYYY-MM-DD) in WEATHER table.
* CONTRATS
  + Split AnneeContrat (format YYYY-YYYY) in FirstYear & LastYear in order to allow easier filtering.

**Target tables**

**DEPOTS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| **IdentifiantDepot** | INT |  | N | Primary Key |
| NomDepot | VARCHAR | 30 | N |  |
| TypeDepot | VARCHAR | 15 | N | Values : Carierre, Chute\_Egout, Entassement, |
| Capacite | INT |  | Y | 0 = Sewer well |
| Latitude | DECIMAL | 8,6 | N |  |
| Longitude | DECIMAL | 8,6 | N |  |
| VolumeDeverse | DECIMAL | 9,2 | N |  |
| IdentifiantArrondissement | INT |  | N | Foreign Key |

**SECTEUR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| **IdenfiantSecteur** | INT |  | N | Primary Key |
| NomSecteur | VARCHAR | 7 | N |  |
| IdentifiantArrondissement | INT |  | N | Foreign Key |

**ARRONDISSEMENT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| IdentifiantArrondissement | INT |  | N | Primary Key |
| ArrondissementCode | CHAR | 3 | N |  |
| Arrondissement | VARCHAR | 30 | N |  |

**CONTRATS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| **IdentifiantContrat** | INT |  | N | Primary Key |
| NumeroContrat | VARCHAR | 7 | N |  |
| TypeContrat | VARCHAR | 12 | Y | Values : Déneigement, Régie, Transport, blank |
| AnneeContrat | CHAR | 9 | N |  |
| FirstYear | CHAR | 4 | N |  |
| LastYear | CHAR | 4 | N |  |
| NoResolution | VARCHAR | 11 | Y |  |
| IdentifiantArrondissement | INT |  | N | Foreign Key |
| DescriptionType | VARCHAR | 100 | Y |  |
| Entreprise | VARCHAR | 50 | N |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

**TRANSACTIONS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| **TransactionID** | INT |  | N | Primary Key |
| DateChargement | DATE | 10 | N | Foreign Key  YYYY-MM-DD |
| TimeChargement | TIME | 8 | N | hh:mm:ss |
| IdenfiantSecteur | INT |  | Y | Foreign Key |
| IdentifiantArrondissement | INT |  | N | Foreign Key |
| IdentifiantDepot | INT |  | N | Foreign Key |
| DateDechargement | TIMESTAMP | 19 | N |  |
| Volume | DECIMAL | 4,2 | N |  |
| TypeTransaction | VARCHAR | 7 | N | Values :  AUT, N-AUT, BARCODE |
| IdentifiantContrat | INT |  | Y | Foreign Key |

**WEATHER**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Length** | **Nullable** | **Comments** |
| **Date** | DATE | 10 | N | Primary Key  YYYY-MM-DD |
| Max Temp | DECIMAL | 3,1 | Y | Celsius |
| Min Temp | DECIMAL | 3,1 | Y | Celsius |
| Mean Temp | DECIMAL | 3,1 | Y | Celsius |
| Total Precip) | DECIMAL | 3,1 | Y | Millimeters |
| Snow on Grnd | INT | 2 | Y | Centimeters |

Appendix A: Acronyms

Instructions: Provide a list of acronyms and associated literal translations used within the document. List the acronyms in alphabetical order using a tabular format as depicted below.

Table 1 - Acronyms

| Acronym | Literal Translation |
| --- | --- |
| csv | Comma Separated Values |
| DDL | Data Definition Language |
| HDD | Hard Drive Disk |
| SSD | Solid State Drive |
| DBMS | Database Management System |
| RDBMS | Relational DBMS |
| ERD | Entity Relationship Diagram |
| PK | Primary Key |
| FK | Foreign Key |
| MB | MegaBytes |
| GB | GigaBytes |
| RAM | Random Access Memory |
| BASE | Basic Availability Soft-State Eventual Consistency |
| ACID | Atomicity Consistency Isolation Durability |
| SQL | Structured Query Language |
| ETL | Extraction Transformation Load |
| SDK | Software Development Kit |
| API | Application Programming Interface |
| ERD | Entity Relationship Model |
| LDM | Logical Data Model |
| CRD | Conceptual Relationship Data Model |

Appendix B: DDL

Appendix C: Decision chart database system

Scofield, Ben (2010-01-14). ["NoSQL - Death to Relational Databases(?)"](http://www.slideshare.net/bscofield/nosql-codemash-2010) Reference

Table 2 – Database options

| **Data model** | **Performance** | **Scalability** | **Flexibility** | **Complexity** | **Functionality** |
| --- | --- | --- | --- | --- | --- |
| Key–value store | high | high | high | none | variable (none) |
| Column-oriented store | high | high | moderate | low | minimal |
| Document-oriented store | high | variable (high) | high | low | variable (low) |
| Graph database | variable | variable | high | high | [graph theory](https://en.wikipedia.org/wiki/Graph_theory) |
| Relational database | variable | variable | low | moderate | [relational algebra](https://en.wikipedia.org/wiki/Relational_algebra) |

**Key-value stores** are capable of providing much higher performances than RDBMS

**>>** In our case data organization and management is more important than the performance.

**Column family** databases are designed for large volumes of data, read and write performance, and high availability

**>>** We don’t have large volumes and don’t need great performance nor high availability.

**Document databases** have ability to store varying attributes along with large amounts of data

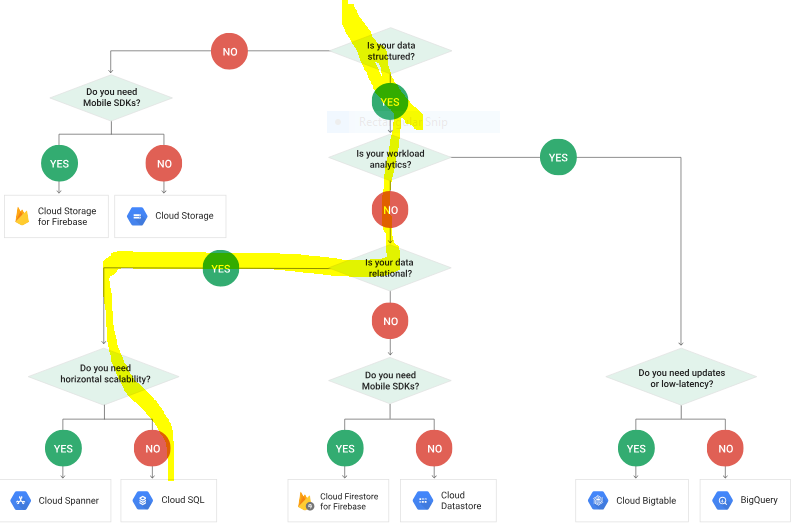
**>>** We don’t have large volumes and the data is static (format doesn’t change much).

**Graph database** is designed to treat the relationships between data as equally important to the data itself. It is intended to hold data without constricting it to a pre-defined model.

**>>** The relationships between the data is static and the pre-defined model does not change.

Appendix D: Decision tree

<https://cloud.google.com/storage-options/> Reference



Appendix E: Highlight of milestone meetings

March 7th: Review of Daniel’s feedback on Slack concerning the report. Update of milestones. Review of Digital Ocean and Microsoft Azure’s pricing. Data flow diagram updated on Real Time board. Discussion concerning the difference between the object-relational database and the entity relational database.

March 5th: Debrief following comments from Daniel in class concerning chosen path for Acid and Relation database with SQL. Review of undefined sections. No response from city of Montreal for measurement questions. Data flow diagram updated on Real Time board, and decision on the type of database (columnar, object vs relational).

February 28th: meeting to review Amazon and Google database options (storage, and Acid/Base characteristics), follow-up on next steps No response from city of Montreal for measurement questions. ERD diagram added to the report. Submission of our questions and report to Daniel on Slack.

February 24th: Discuss open data resources available, similar projects, and streets of Montreal surface area. No response received from city of Montreal, following another attempt made using the data request form. Review of locations in Montreal without depots. Start and end dates of season for snow removal contracts are not confirmed by the city of Montreal. No response from city of Montreal for measurement questions.

February 22nd: No response from city of Montreal for measurement questions concerning the volume measure and the start of the season.

February 19th: Review of video communicated by the city of Montreal concerning the snow collection.

February 13th: Discussion concerning license for city of Montreal data and realism in the application of the proposed solution. Review of Système Planif-Neige API document. Discussion concerning the hierarchy of the data and the data dictionary. Data dictionary including data type updated in project document. Use of Tableau to visualize the boroughs were depots and snow collection transactions are being recorded.

February 11th: meeting kick-off and review of Real Time board contents. Sample of database downloaded, and conducted data profiling, in order to access which data to keep. Initial preparation of the data dictionary.