Projet élection

Thibault NG, Camille Du

07 mai 2023

**I - Organisation du groupe :**

Travail commun : conception de la base de données, écriture des requêtes SQL, écriture du rapport

Thibault : transformation et insertion donnée, interface graphique de la carte,

Camille : création de la base de données, interface graphique pour les requêtes sql,

**II - Choix technologiques :**

Langages :

Postgresql : On a préféré une base de données traditionnel comme postgreSQL comme on est plus familiers avec que mySql ou autre base de donnée traditionnel. Aussi, pas d’usage de NoSQL comme MongoDB, car les données que l’on a sont plutôt bien structurées.

Python : L’usage de Python a été choisi ici car c’est le plus pratique pour réaliser des petits projets rapidement et surtout très pratique pour l’aspect du traitement de donnée, on pensera aux librairies très bien conçus comme Pandas

HTML/CSS : Pour l’affichage de la page web, html et css car c’est cela répond à notre besoin, à savoir afficher des pages web(html) et les designer(css).

Logiciels :

Intellij avec plugin Python : nous avons utilisé Intellij pour l’environnement de développement comme c’est l’outil que l’on utilise le plus souvent et que le plugin Python est plutôt bien conçu, on a décidé d’utiliser Intellij plutôt qu’un environnement spécialement adapté à Python.

Librairies :

Traitement de donnée

Pandas : la librairie utilisée pour le traitement des données

Json : la librairie utilisée pour le formatage des données

Geopandas : la librairie utilisée pour les données géographiques qui est comme la librairie json, mais pour les données géographiques

Urlopen : la librairie utilisée pour récupérer des données en ligne

Remarque : parmi les données auxquelles nous avions accès, nous avions uniquement l'état ainsi que le numéro du district. Les données que l'on a trouvé en termes géographique ne contenaient que les comtés. Il a donc fallu trouver des données reliant comtés et numéro de district En combinant les deux et en fusionnant les comtés en un plus gros district nous avons une carte des districts.

Ressource pour les données (bdm-election/ressources)

Une image contenant texte

Description générée automatiquement

Interface graphique

matplotlib, plotly : les librairies utilisées pour tracer le graphique avec une courbe, 2 axes et diverses fonctionnalités comme le zoom

Infrastructure

psycopg2 : la librairie utilisée pour la connexion à la base de donnée et les envois des requêtes SQL et réception de leur résultat

flask : la librairie utilisée pour faire le pont entre html/css et les scripts de traitement, elle est utilisée pour servir de serveur

gevent.pywsgi : la librairie utilisée pour le multi-threading, car flask par lui-même ne le fait pas par défaut

**III - Modèle conceptuel des données** **:**

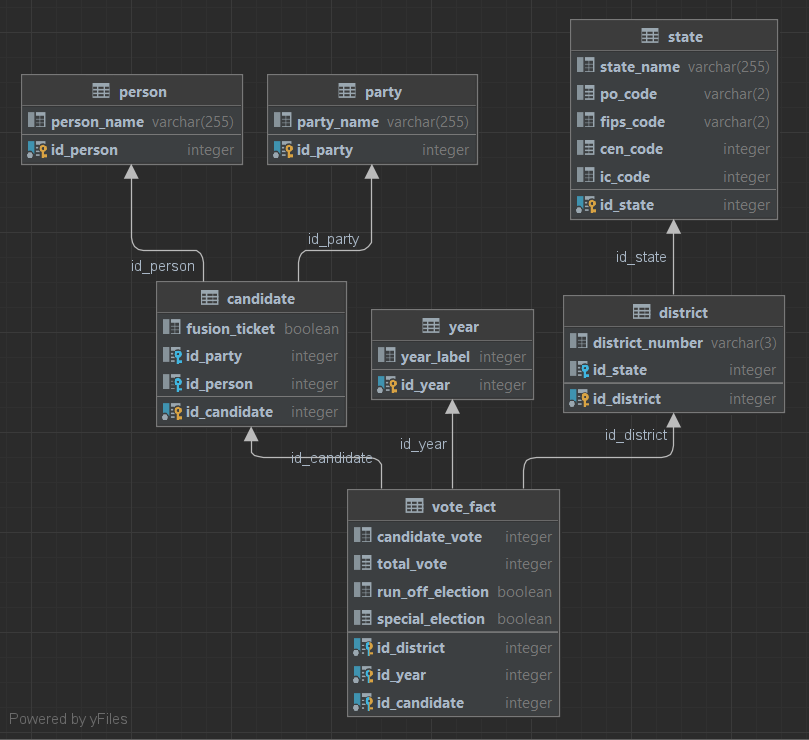


Schéma de la structure interne de l’entrepôt

Structure

Concernant la structure de l’entrepôt, nous avons décidé de prendre un modèle en flocon pour les avantages de la normalisation des dimensions.

Grain

La structure interne de l’entrepôt est d’abord composée d’une table de fait « vote\_fact » dont le grain représente les résultats de vote d’un candidat dans un district lors d’une année, sachant qu’un district est situé dans un état et qu’un candidat comprends sa personne et son parti.

Description des tables

Table de fait

Cette table de fait « vote\_fact » est composée des attributs :

- « candidate\_vote » qui signifie le nombre de vote pour ce candidat (pour un district et une année)

-« total\_vote » qui signifie le nombre de vote en tout sur cette élection (pour un district et une année)

- « run\_off\_election » qui signifie qu’il y a eu une élection supplémentaire, ce qui arrive lorsque aucun candidat lors d'une élection primaire ou générale n'a remporté la majorité absolue des votes

-« special\_election » signifie que c’est une élection en dehors du calendrier électoral, ce qui arrive lorsqu’il y a des postes vacants pour diverses raisons comme la démission, la mort ou la destitution d'un élu en exercice.

Table de dimension

Cet entrepôt est ensuite complété par 6 autres tables de dimensions dont :

La table « year » qui correspond aux années des élections avec en attribut :

-« year\_label » qui représente l’année, i.e. 1976, 1978, 1980

La table « district » correspond à une zone géographique délimité et qui compose un état des Etats-Unis, aussi appelés « circonscription électorale » ou « circonscription législative » avec en attribut :

-« district\_number » qui correspond à son numéro de district, i.e. 001, 002, 008

La table « state » qui correspond à une autre zone géographique englobant les ‘districts’, aussi appelés états, elle contient les attributs :

-« state\_name » qui correspond au nom que porte l’état, i.e. Minnesota, Louisiane, Texas

-« po\_code » qui correspond au code postal

-« fips\_code » qui correspond au code fips, utilisé pour identifier de manière unique les États et les comtés aux États-Unis.

-« cen\_code » qui correspond au code cen, utilisé pour identifier de manière unique les entités géographiques dans le contexte du recensement

-« ic\_code » qui correspond au code ic, utilisé pour identifier les zones de communication interurbaine dans un système de numérotation téléphonique

La table « candidate » qui correspond à une personne représentant un parti pour une ou des élections avec en attribut :

-« fusion\_ticket » qui correspond au fait que le candidat représente plusieurs partis ou non

La table « person » qui correspond à l’identité du candidat avec en attribut :

-« person\_name » qui correspond à son nom et prénom

La table « party » qui correspond à un parti politique avec en attribut :

-« party\_name » qui correspond au nom du dit parti

Choix arbitraire sur le fichier csv initial

-Pour la colonne « unofficial », nous l’avons été supprimé et avons décidé de considérer les votes dont les résultats sont non officiels comme tel car ils n’y avaient pas les résultats officiel pour ces élections.

-Pour la colonne « stage », nous avons retiré les lignes correspondant aux valeurs PRI (PRIMARY) car ceux-ci correspondaient aux élections primaires dont le but était d’élire les représentants des différents partis, ce qui n’est pas une donnée qui nous intéresse

Index et agrégat éventuelle :

Index

-btree sur candidate.idCandidat, car un très grand nombre de tuple unique (16 000 environs)

-cluster sur candidat avec parti et personne, car en général on appelle le candidat avec son nom et le nom de son/ses parti/s

-bitmap sur les booleans genre run\_off\_election et special\_election, car ce sont des booleans

Agrégat

-party\_weight\_per\_state\_aggregate : nombre de vote par parti et ceux que ça représente par rapport à tous les votes dans un state en pourcentage

-winner\_party\_per\_state\_per\_year\_aggregate : le candidat gagnant par district par année (vérif run off inclus ou non)

-winner\_party\_globally\_per\_year\_aggregate : pour chaque année le parti ayant eu le plus de district avec un vote majoritaire pour son parti

-district\_weight\_per\_year\_aggregate : le nombre de vote par année et par district tous partis et candidats confondue / le totale de vote tout district tout partis et tout candidats confondus. Cela permet de voir le poids que représente chaque district dans les élections de manière globale