LTA Service Agreement UNDP México (UNDP-MEX-10029 / CSG-2024-002)

mottum. Analytica
Control y Gestión

Consultoría

"Diseño y programación de soluciones tecnológicas, ciencia de datos, análisis y visualización de datos"

Entrega 1 de 3

Extracción de datos EDCA y generación de tablas

Este entregable contiene:

- Script 1 - Transformación semi-automática (Incluye la descarga y generación de tablas)
- Script 2 - Enfoque teórico del script de Transformación automática.
- Documentación para la puesta en funcionamiento (Read me file).
- Informe de avance

SECRETARÍA EJECUTIVA SNA

5 de Abril 2024

PNUD

ÍNDICE

1. Introducción.

El Sistema Nacional Anticorrupción de México constituye el mecanismo gubernamental destinado a fomentar la prevención, identificación, investigación y castigo tanto de actos corruptos como de irregularidades administrativas en los ámbitos federal, estatal y municipal.

Dentro de sus variadas funciones, la Secretaría Ejecutiva de este sistema (SESNA) se encarga de diseñar estrategias comprensivas para combatir la corrupción, además de desarrollar borradores de metodologías y criterios de evaluación para medir este problema.

En este marco, el Programa de las Naciones Unidas para el Desarrollo (PNUD) en México colabora con SESNA en la creación e implementación de estrategias de mejora.

Este documento representa el informe de progreso del primer entregable, en el cual se ha establecido el entorno de trabajo necesario y se han generado los primeros códigos en Python para el análisis y manejo de datos.

Repositorio de código
https://github.com/MottumData/Anticorruption_ETL_MEX

2. Enfoque técnico.

El objetivo de este proyecto es procesar los datos de contrataciones públicas de México provenientes de Compranet Info en un formato Zip. Para ello el proyecto se enfoca desde un punto de vista técnico con varios scripts de Python que serán documentados y guardados en un repositorio de código abierto de Github. Esto ayudará al mantenimiento y reproducción del código en un futuro.

Los datos alojados en el link de origen se descargan en formato .Zip, que a fecha de Marzo 2023 pesan 3 GB comprimidos y 25 GB en formato json (sin comprimir).

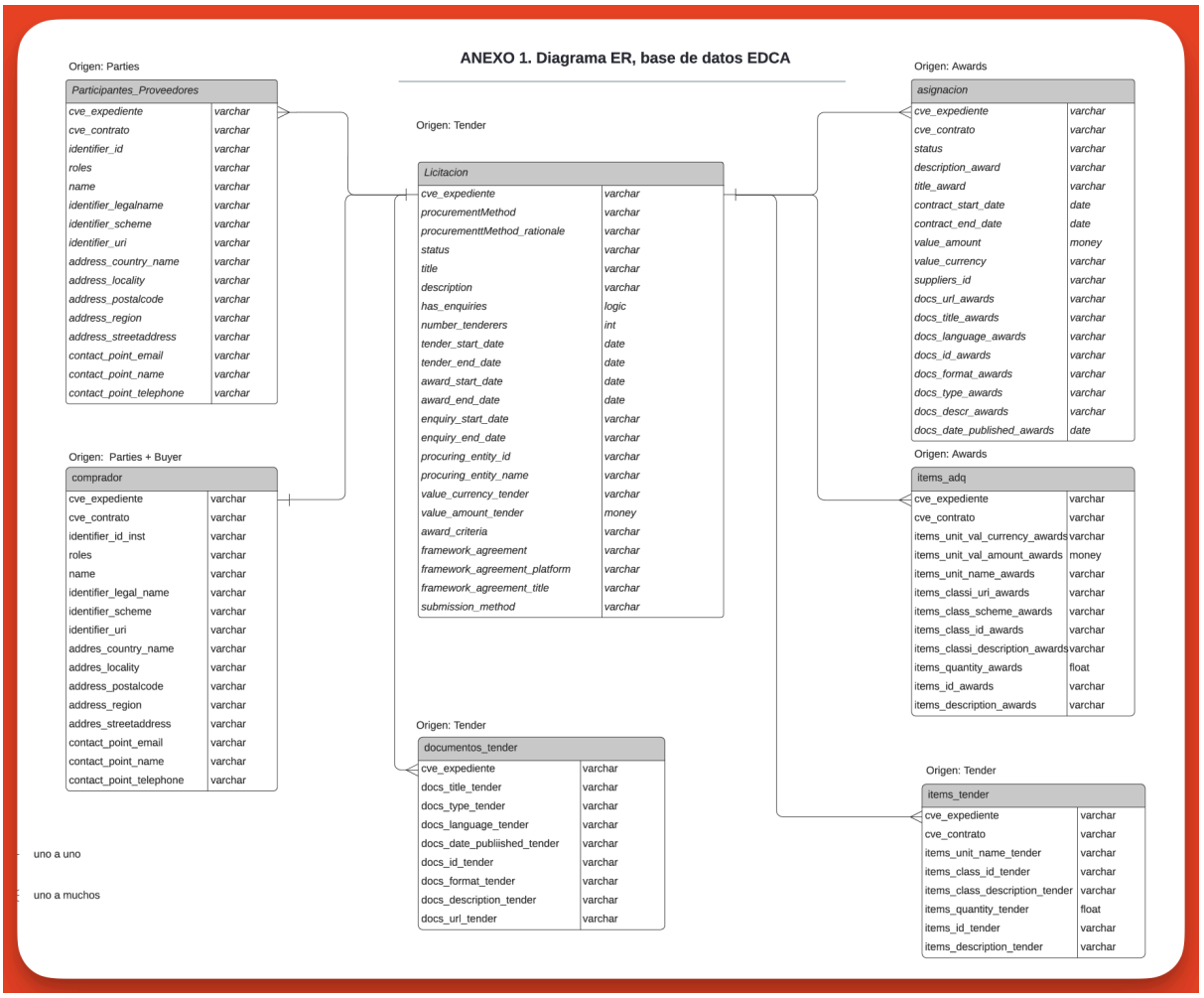
Link de origen de los datos:
https://compranetinfo.hacienda.gob.mx

Después del correspondiente proceso de proceso de descarga y limpieza de datos, se requiere transformar los datos para alimentar siete tablas:

1. Participantes_proveedores
2. Comprador
3. Licitación
4. Documents_tender

5. Asignación
6. Items_Adq
7. Items_tender

Dichas tablas se relacionan entre sí según el siguiente diagrama ER:



La fase uno de este proyecto involucra la definición del stack tecnológico con el que se va a trabajar, la descarga inicial de los datos, la creación de las tablas iniciales según el esquema ER y finalmente la documentación inicial del README file y la creación del repositorio en Github.

3. Retos y soluciones.

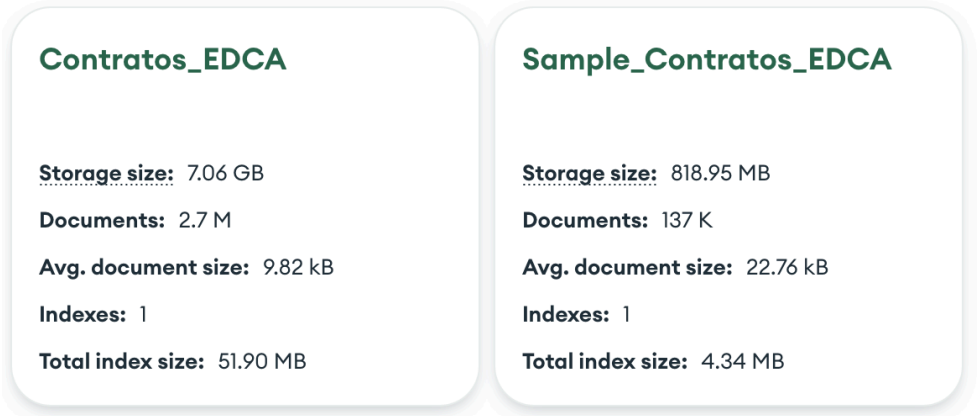
Durante el proceso de la primera parte de este proyecto se han identificado los siguientes retos y soluciones:

1. **Acceso a los datos desde España**
   Debido a restricciones de IP a la web de Compranetinfo.hacienda.mx no se ha podido realizar la descarga desde el lugar de trabajo de los consultores (España). Para solventar

este reto se ha procedido a crear y configurar una VPN que simule una conexión desde México. Esto ha solucionado completamente el reto encontrado.

2. **Base de datos de gran tamaño**
   El procesamiento de la gran cantidad de datos ha dificultado el proceso de lectura y transformación. Para asegurar el correcto funcionamiento se ha propuesto trabajar con una muestra de una base de datos consistente en un 5% del tamaño completo de la base original.

   La muestra de la base de datos ("Sample") tiene un peso de 800 MB frente a los 7 GB de la original. Esto hace que sirva para propósitos de testeo de los scripts.



Tecnologías evaluadas para el procesamiento a gran escala:

La Secretaría Nacional de Anticorrupción (SNA) propone trabajar con tecnología en computación local, es decir, sin utilizar la nube para propósitos de optimización. Esta decisión se toma debido a la futura integración de la nube de Oracle en el SNA.

Para poder disfrutar de la **computación paralelizada** sin necesidad de ir a la nube es necesario disponer de hardware con varios núcleos de procesamiento. Los computadores modernos vienen equipados por lo general con dos núcleos Dual-Core, lo que hace posible simular el cómputo con hasta cuatro procesadores.

El tamaño de **memoria RAM** también es un factor importante en este tipo de procesos ya que es la que guarda en memoria los datos mientras se hace el procesamiento. Esta es la memoria volátil, la que utiliza el ordenador para los procesos de cómputo más comunes.

Los scripts de este entregable han sido computados con el siguiente hardware:

Memoria RAM	Procesador	Sistema Operativo
16 GB	i5 Dual Core	MacOS

Las tecnologías evaluadas y probadas en este proceso fueron:

- Pyspark:
- Lectura desde MongoDB:
- Librería de python Dask:

Tecnología propuesta para continuar el proyecto:
En siguientes etapas de este proyecto, donde se comenzará a analizar los datos y preparar el proceso para que se corra en una máquina del SNA, se propone trabajar con una máquina virtual configurada para un óptimo procesamiento de los datos. Esta estrategia permitirá evitar problemas de configuración y versionado de los distintos códigos y asegurar la sostenibilidad del proceso independientemente de la máquina con la que se ejecute.

Para solucionar los retos del tamaño de la base de datos se propone transformar los datos a un formato parquet que permita la limpieza y procesado de una manera mucho más eficiente.

En resumen, se propone trabajar con el siguiente stack complementando los ya utilizados:

- **Máquina virtual**: Virtualización de un computador optimizado para el procesamiento de estos datos.
- **Python**: Lenguaje de programación principal de este proyecto
- **MongoDB**: Base de datos NoSQL para procesamiento de datos Json.
- **Parquet**: Formato de datos alternativo al csv para el proceso de limpieza y análisis. Luego de este proceso se transformará nuevamente a csv.
- **Pyspark**: Framework de programación (librería) para el procesamiento de datos en paralelo.
- **Librería o sistema de visualización de datos**: (por definir) Para la visualización óptima de los resultados del análisis de propone usar alguna de las siguientes librería y/o tecnologías: Seaborn, Tableau, Looker, Kibana o similar…

3. **Formato de la base de datos**

El formato extensos del json original y la extensas cantidad de llaves en los datos añaden un grado de complejidad a los datos. El excel proveído por el SNA (Relación variables EDCA-MODELO ER.slxs) ha supuesto una ayuda que debe ser tenida en cuenta por cualquier técnico que vaya a ejecutar este proceso en el futuro.

4. Entregables.

4.1. Script de descarga manual.

Por simplicidad para identificar errores en el código, este script se ha escrito en un notebook de python, lo que permite documentar mejor el código.

Nombre del archivo: 1.1 Data Download.ipynb

Archivo encargado de la descarga y descompresión del archivo de un .zip a un json.

4.2. Script de descarga automática.

Por decisión del equipo del SNA este script no se entrega en esta instancia, sin embargo hemos creído conveniente proponer una tecnología que permita descargar el código sin interacción humana.

Existen diversas tecnologías que pueden automatizar la descarga de estos archivos, sin embargo para asegurar que esto funcione correctamente se propone que sí sea un sistema alojado en la nube para evitar la necesidad de la interacción humana.

Un ejemplo de un sistema comúnmente conocido para ejecutar este script podría ser una instancia de Power Automate, tecnología de Microsoft, que compagina adecuadamente con la nube de Azure para la automatización, despliegue de código y alojamiento de archivos. El stack completo para poder correr esto sería:

- Cloud de Azure
- Azure Logic Apps Functions
- Blob Storage

4.3. Script de generación de tablas.

Nombre del archivo: 2_Extraction_MongoDB_V2.ipynb

Este archivo realiza las siguientes acciones a grandes rasgos:

- Lectura al MongoDB
- Comprueba la conexión a la base de datos MongoDB
- Confirma el tamaño y la actualización de datos
- Crea una muestra pequeña de la base de datos
- Crea las tablas definidas en el diagrama ER con la muestra de la BD.