

14-6-2024

Proyecto integrador

Proceso ETL

Cuatrimestre: 9

Grupo: C

Carrera: IDGS

Presentan:

Bahena Castillo Luis Eduardo

Barrios Tecorral Oscar Miguel

Mata Nieto Iván Samuel

Reynoso Macedo Brayan

Rodriguez Rodriguez Cristian

Rogel Valentin Diego Jared

Contenido

Introducción:	2
Objetivo del Segundo Avance:.....	2
Descripción del Conjunto de Datos:.....	2
Necesidad del Proceso ETL:	3
Proceso ETL:	3
Limpieza de Datos:	5
Ejemplo práctico:	5
Carga:	7
Herramientas y métodos utilizados:.....	7
Procedimiento de carga:	8
Aseguramiento de la integridad y consistencia:.....	8
Diagrama del Proceso ETL:	10
Consideraciones Técnicas y Éticas:	12
Medidas de seguridad:	13
Conclusiones:	14
Próximos pasos:	14
Referencias:.....	15

Introducción:

En el primer avance de este proyecto, exploramos el conjunto de datos "Estaciones de Servicio (Gasolineras) y Precios finales de Gasolina y Diesel" del portal datos.gob.mx. Este conjunto de datos, compuesto por dos archivos XML (Places.xml y Prices.xml), ofrece información valiosa sobre la ubicación y los precios del combustible en las estaciones de servicio de México. Con base en este conjunto de datos, definimos un caso de aplicación que busca ayudar a los consumidores a encontrar la gasolinera más cercana con los mejores precios, utilizando tecnologías como la inteligencia artificial (IA), el aprendizaje automático (ML) y el análisis de Big Data.

Objetivo del Segundo Avance:

El objetivo de este segundo avance es preparar los datos para el análisis de minería de datos. Para lograr esto, llevaremos a cabo un proceso de Extracción, Transformación y Carga (ETL). Este proceso es fundamental para garantizar que los datos estén limpios, estructurados y listos para ser utilizados en el análisis posterior, que nos permitirá descubrir patrones y tendencias relevantes para el caso de aplicación.

Descripción del Conjunto de Datos:

El conjunto de datos seleccionado contiene información detallada sobre las estaciones de servicio en México, incluyendo su nombre, ID, coordenadas geográficas, tipo de combustible (regular, premium, diesel) y precios. El volumen de datos es considerable, ya que abarca miles de estaciones en todo el país. La velocidad de actualización no está especificada, pero se espera que sea periódica. La variedad de datos es buena, ya que incluye tanto información geoespacial como numérica.

Necesidad del Proceso ETL:

El proceso ETL es necesario debido a que los datos en su formato original (XML) no son directamente adecuados para el análisis de minería de datos. Los archivos XML contienen una estructura jerárquica que dificulta la manipulación y el análisis de los datos. Además, es probable que los datos contengan inconsistencias, valores faltantes o duplicados que deben ser corregidos antes de proceder con el análisis. El proceso ETL nos permitirá extraer la información relevante de los archivos XML, transformarla en un formato tabular estructurado y cargarla en un entorno adecuado para el análisis, como una base de datos o un marco de datos.

Proceso ETL:

Extracción:

Origen de los datos: Los datos originales se obtuvieron del portal de datos abiertos del gobierno mexicano, datos.gob.mx, específicamente del conjunto de datos "Estaciones de Servicio (Gasolineras) y Precios finales de Gasolina y Diesel". La URL directa para acceder a estos datos es:

<https://datos.gob.mx/busca/dataset/estaciones-de-servicio-gasolineras-y-precios-finales-de-gasolina-y-diesel>

Formato original y conversión: Los datos se proporcionaron originalmente en formato XML, distribuidos en dos archivos: "Places.xml" (información de ubicación) y "Prices.xml" (información de precios).

Herramientas utilizadas: Para facilitar la manipulación y el análisis, se realizó una conversión manual de los archivos XML a formato JSON. Esta conversión se llevó a cabo debido a la mayor flexibilidad y facilidad de uso que ofrece el formato JSON en comparación con el XML, especialmente en entornos de programación y análisis de datos.

Justificación de la conversión a JSON: El formato JSON presenta una estructura más simple y legible que el XML, lo que facilita la extracción de

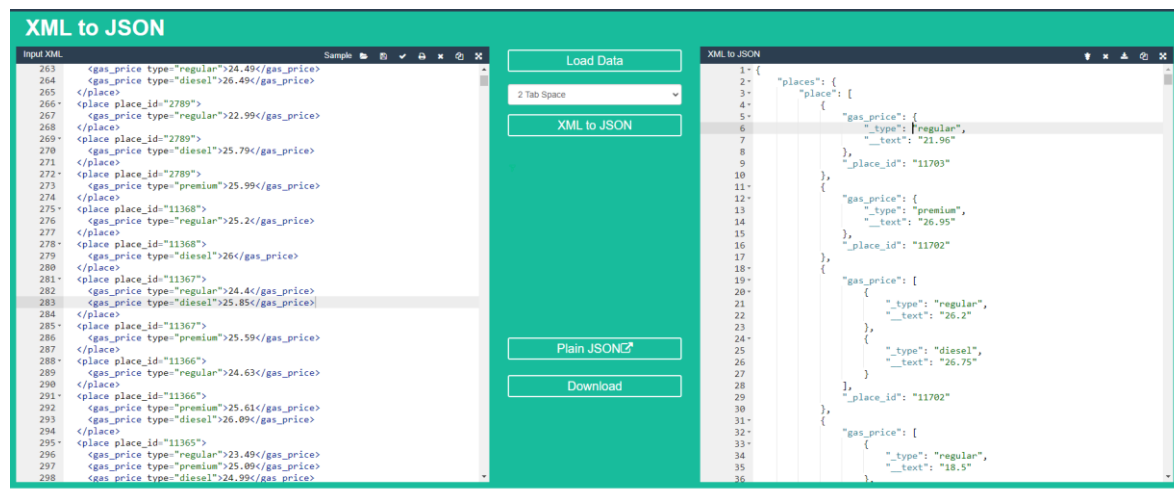
información relevante. Además, JSON es ampliamente compatible con lenguajes de programación y herramientas de análisis de datos, lo que agiliza el proceso de transformación y carga de los datos.

Procedimiento de extracción:

1. **Descarga de datos:** Se descargaron los archivos XML "Places.xml" y "Prices.xml" desde el portal de datos abiertos.
2. **Conversión a JSON:** Se utilizaron herramientas de conversión XML a JSON (ya sea en línea o bibliotecas de software) para transformar los archivos XML en archivos JSON equivalentes.
3. **Almacenamiento:** Los archivos JSON resultantes se almacenaron localmente para su posterior procesamiento en las etapas de transformación y carga del proceso ETL.

Al convertir los datos a JSON, hemos simplificado la estructura de los mismos y los hemos preparado para las siguientes etapas del proceso ETL, donde se limpiarán, transformarán y cargarán en un formato adecuado para el análisis de minería de datos.

Datos Sin limpiar:



The screenshot displays an online XML to JSON converter interface. On the left, the 'Input XML' field contains a snippet of XML data with gas prices for various locations. In the center, there are buttons for 'Load Data', 'XML to JSON', 'Plain JSON', and 'Download'. On the right, the 'XML to JSON' field shows the resulting JSON structure, which organizes the data into a hierarchical format with 'places' as the root and 'place' as an array of objects, each containing 'gas_price' details.

```
Input XML
263 <gas_price type="regular">24.49</gas_price>
264 <gas_price type="diesel">26.49</gas_price>
265 </place>
266 <place place_id="2789">
267 <gas_price type="regular">22.99</gas_price>
268 </place>
269 <place place_id="2789">
270 <gas_price type="diesel">25.79</gas_price>
271 </place>
272 <place place_id="2789">
273 <gas_price type="premium">25.99</gas_price>
274 </place>
275 <place place_id="11368">
276 <gas_price type="regular">25.2</gas_price>
277 </place>
278 <place place_id="11368">
279 <gas_price type="diesel">26</gas_price>
280 </place>
281 <place place_id="11367">
282 <gas_price type="regular">24.4</gas_price>
283 <gas_price type="diesel">25.85</gas_price>
284 </place>
285 <place place_id="11367">
286 <gas_price type="premium">25.59</gas_price>
287 </place>
288 <place place_id="11366">
289 <gas_price type="regular">24.63</gas_price>
290 </place>
291 <place place_id="11366">
292 <gas_price type="premium">25.61</gas_price>
293 <gas_price type="diesel">26.09</gas_price>
294 </place>
295 <place place_id="11365">
296 <gas_price type="regular">23.49</gas_price>
297 <gas_price type="premium">25.09</gas_price>
298 <gas_price type="diesel">24.99</gas_price>

```

```
XML to JSON
1 {
2   "places": {
3     "place": [
4       {
5         "gas_price": {
6           "_type": "regular",
7           "_text": "21.96",
8         },
9         "_place_id": "11703"
10      },
11      {
12        "gas_price": {
13          "_type": "premium",
14          "_text": "26.95",
15        },
16        "_place_id": "11702"
17      },
18      {
19        "gas_price": [
20          {
21            "_type": "regular",
22            "_text": "26.2",
23          },
24          {
25            "_type": "diesel",
26            "_text": "26.75"
27          }
28        ],
29        "_place_id": "11702"
30      },
31      {
32        "gas_price": [
33          {
34            "_type": "regular",
35            "_text": "18.5"
36          }
37        ]
38      }
39    ]
40   }
41 }
```

Limpieza de Datos:

- **Valores faltantes:** Se identificaron y trataron los valores faltantes en los campos de precios y coordenadas geográficas. Para los precios, se optó por imputar los valores faltantes con la media o la mediana de los precios de la misma estación de servicio para el mismo tipo de combustible. En el caso de coordenadas faltantes, se descartaron las estaciones de servicio que no contaban con información de ubicación completa.
- **Outliers:** Se realizó un análisis exploratorio de los datos para detectar outliers en los precios del combustible. Los outliers se identificaron utilizando métodos como el rango intercuartílico (IQR) y se trataron mediante técnicas como la Winsorización o la transformación logarítmica.
- **Datos erróneos:** Se corrigieron errores tipográficos y de formato en los nombres de las estaciones de servicio y en los tipos de combustible.

Ejemplo práctico:

Una transformación crítica realizada en PDI fue la homogenización de la estructura de datos en el campo "gasprice". En los datos originales, cuando una estación de servicio solo tenía un precio para un tipo de combustible, este no se representaba como un arreglo (array) en el JSON, sino como un objeto simple. Esto generaba inconsistencias en la estructura de los datos y dificultaba su procesamiento posterior.

Para solucionar este problema, se utilizó un paso de "JavaScript" en PDI para transformar los objetos simples en arreglos, incluso si solo contenían un elemento. De esta manera, se aseguró que todas las estaciones de servicio tuvieran una estructura de datos uniforme para el campo "gasprice", independientemente del número de precios disponibles.

```

var gasprice = JSON.parse(gasprice);

if (!Array.isArray(gasprice)) {
    gasprice = [gasprice];
}

gasprice.forEach(function(price) {
    // Procesamiento adicional de los precios (opcional)
});

JSON.stringify(gasprice);

```

Datos Limpios:

```

1  {
2    "places": {
3      "place": [
4        {
5          "gasprice": [
6            {
7              "type": "regular",
8              "text": "21.96"
9            }
10         ],
11         "placeid": "11703"
12       },
13       {
14         "gasprice": [
15           {
16             "type": "premium",
17             "text": "26.95"
18           }
19         ],
20         "placeid": "11702"
21       },
22     ]
23   }

```

Carga:

Destino de los datos: Los datos transformados se cargaron en dos destinos principales:

1. **Archivo JSON único:** Se generó un archivo JSON consolidado que contenía toda la información relevante de las estaciones de servicio, incluyendo su ubicación, precios de combustible y las variables adicionales creadas durante la etapa de transformación. Este archivo sirvió como una copia de seguridad y como fuente de datos para otros análisis que no requirieran una base de datos.
2. **Colección MongoDB:** Los datos también se cargaron en una colección de MongoDB, una base de datos NoSQL orientada a documentos. MongoDB fue seleccionado debido a su flexibilidad para manejar datos semi-estructurados y su escalabilidad para adaptarse al crecimiento del conjunto de datos.

Herramientas y métodos utilizados:

- **Pentaho Data Integration (PDI):** Se utilizó PDI para orquestar todo el proceso de carga. Se crearon transformaciones en PDI para leer los archivos JSON transformados, realizar un ordenamiento previo por el campo "place_id" y luego unir los datos de ubicación y precios mediante un paso de "Merge Join".
- **MongoDB Output:** Se utilizó el paso "MongoDB Output" de PDI para insertar los datos consolidados en la colección de MongoDB.

Procedimiento de carga:

1. **Ordenamiento:** Se realizó un ordenamiento de los datos de ubicación (places.json) y precios (prices.json) por el campo "place_id" para asegurar una unión eficiente en el siguiente paso.
2. **Merge Join:** Se utilizó un paso de "Merge Join" en PDI para combinar los datos de ubicación y precios en un único flujo de datos, utilizando el campo "place_id" como clave de unión.
3. **Salida a JSON:** Se generó un archivo JSON único con los datos consolidados.
4. **Salida a MongoDB:** Se utilizó el paso "MongoDB Output" de PDI para insertar los datos consolidados en la colección de MongoDB.

Aseguramiento de la integridad y consistencia:

- **Verificación de duplicados:** Se verificó la ausencia de registros duplicados en la colección de MongoDB.
- **Validación de tipos de datos:** Se aseguraron los tipos de datos correctos en MongoDB (por ejemplo, coordenadas geográficas como arrays, precios como números de punto flotante).
- **Pruebas de integridad referencial:** Se verificó que todas las referencias a "place_id" en los datos de precios tuvieran una correspondencia válida en los datos de ubicación.

Al finalizar el proceso de carga, los datos se encontraban disponibles tanto en un archivo JSON como en una colección de MongoDB, listos para ser utilizados en el análisis de minería de datos y en la aplicación final para encontrar la gasolinera más cercana con los mejores precios.

El input de places:

JSON input

Step name: DIM_PLACES

File Content Fields Additional output fields

#	Name	Path	Type	Format	Length	Precision	Currency	Decimal	Group	Trim ty
1	placeid	\$.places.place[*].placeid	String							none
2	y	\$.places.place[*].location.y	String							none
3	x	\$.places.place[*].location.x	String							none
4	cre_id	\$.places.place[*].cre_id	String							none
5	name	\$.places.place[*].name	String							none

Select fields

Help OK Preview rows Cancel

El input de prices:

JSON input

Step name: DIM_PRICES

File Content Fields Additional output fields

#	Name	Path	Type	Format	Length	Precision	Currency	Decimal	Group	Trim ty
1	placeid	\$.places.place[*].placeid	String							none
2	gasprice	\$.places.place[*].gasprice	None							none

Select fields

Help OK Preview rows Cancel

Merge join:

Merge join

Step name: Merge join

First Step: SORT_PLACES

Second Step: SORT_PRICES

Join Type: INNER

Keys for 1st step:

#	Key field
1	placeid

Get key fields

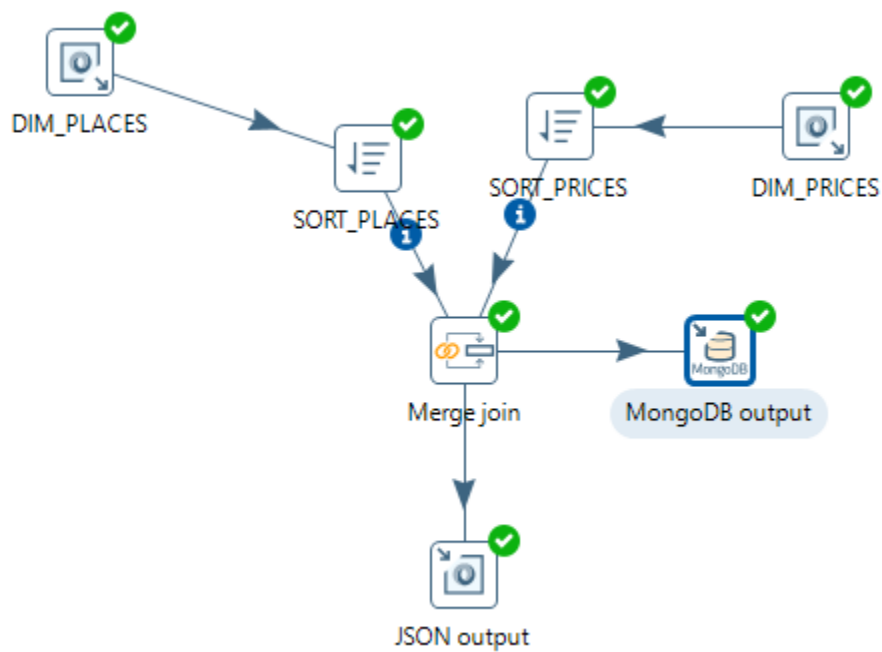
Keys for 2nd step:

#	Key field
1	placeid

Get key fields

Help OK Cancel

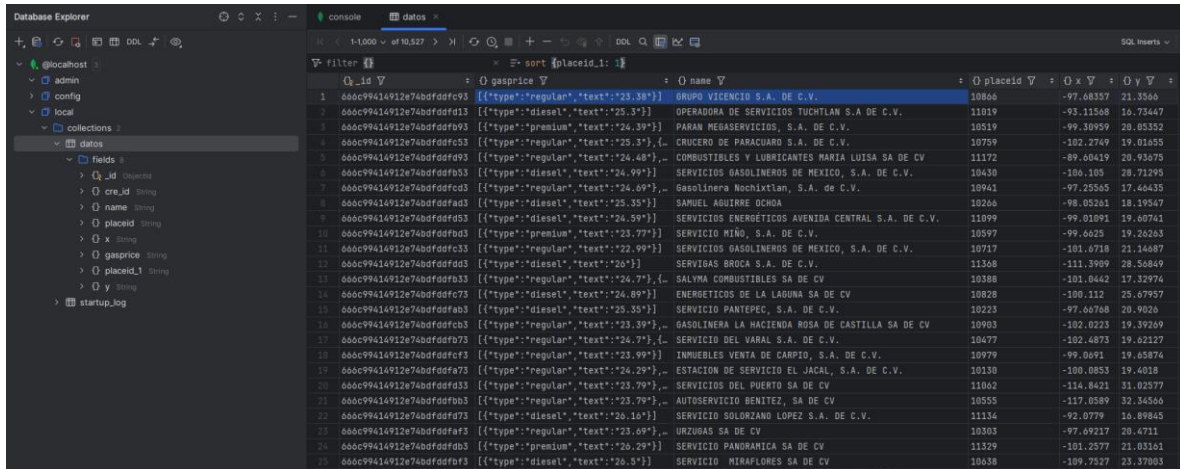
Diagrama del Proceso ETL:



El diagrama presentado ilustra el flujo de trabajo del proceso ETL (Extracción, Transformación y Carga) implementado para el conjunto de datos de estaciones de servicio y precios de combustible. A continuación, se detalla cada componente del diagrama:

1. **DIM_PLACES:** Este paso representa la lectura de los datos de ubicación de las estaciones de servicio desde el archivo JSON "places.json".
2. **SORT_PLACES:** Los datos de ubicación se ordenan ascendentemente por el campo "placeid". Esto asegura una unión eficiente en el siguiente paso.
3. **SORT_PRICES:** De manera similar, los datos de precios del combustible, provenientes del archivo "prices.json", se ordenan también ascendentemente por el campo "placeid".
4. **DIM_PRICES:** Este paso lee los datos de precios ya ordenados.
5. **Merge Join:** Este es el núcleo del proceso de transformación. Aquí, los datos de ubicación y precios se unen en un único flujo de datos utilizando el campo "placeid" como clave. El resultado es un conjunto de datos enriquecido que combina la información de ubicación y precios de cada estación de servicio.
6. **MongoDB output:** Los datos consolidados y enriquecidos se cargan en una base de datos MongoDB. Esto garantiza la persistencia de los datos para su posterior análisis y utilización.
7. **JSON output:** Simultáneamente, los datos consolidados también se escriben en un archivo JSON. Esto proporciona una copia de seguridad de los datos y permite su uso en otros contextos o herramientas de análisis.

Finalmente después de que se ejecutó correctamente el proceso ETL podemos acceder a los datos procesados los cuales vamos a conectar a una api de Python la cual nos hará un estimado de cuáles son las gasolineras más baratas de acuerdo a nuestra longitud y latitud



_id	gasprice	name	placeid	x	y
666c99414912e74bdfdfc93	regular, 25.38	GRUPO VICENCIO S.A. DE C.V.	10866	-97.68357	21.3566
666c99414912e74bdfdfc13	diesel, 25.3	OPERADORA DE SERVICIOS TUCHTLAN S.A. DE C.V.	11019	-93.11568	16.73447
666c99414912e74bdfdfb93	premium, 24.39	PARAM MEGASERVICIOS, S.A. DE C.V.	10519	-99.38959	20.85352
666c99414912e74bdfdfc53	regular, 25.3	CRUCERO DE PARACUARO S.A. DE C.V.	10759	-102.2749	19.01655
666c99414912e74bdfdfc93	regular, 24.48	COMBUSTIBLES Y LUBRICANTES MARIA LUISA SA DE CV	11172	-89.68419	20.93675
666c99414912e74bdfdfb53	diesel, 24.99	SERVICIOS GASOLINEROS DE MEXICO, S.A. DE C.V.	10438	-100.188	28.73195
666c99414912e74bdfdfc03	regular, 24.69	Gasolinera Nechitlan, S.A. de C.V.	10941	-97.25545	17.46435
666c99414912e74bdfdfad3	diesel, 25.35	SAMUEL AGUIRRE OCHOA	10246	-98.02041	18.18947
666c99414912e74bdfdfc53	diesel, 24.59	SERVICIOS ENERGETICOS AVENIDA CENTRAL S.A. DE C.V.	11099	-99.01091	19.40741
666c99414912e74bdfdfb03	premium, 23.77	SERVICIO MIND, S.A. DE C.V.	10597	-99.6425	19.26263
666c99414912e74bdfdfc33	regular, 22.99	SERVICIOS GASOLINEROS DE MEXICO, S.A. DE C.V.	10717	-101.4718	21.14487
666c99414912e74bdfdfc03	regular, 26	SERVISAS BROCA S.A. DE C.V.	11368	-111.3999	28.56849
666c99414912e74bdfdfb33	regular, 24.7	SALMA COMBUSTIBLES SA DE CV	10388	-101.8642	17.32974
666c99414912e74bdfdfc73	diesel, 24.89	ENERGETICOS DE LA LAGUNA SA DE CV	10828	-100.112	25.67957
666c99414912e74bdfdfab3	diesel, 25.35	SERVICIO PANTEPEC, S.A. DE C.V.	10223	-97.66768	28.9026
666c99414912e74bdfdfcb3	regular, 23.39	GASOLINERA LA HACIENDA ROSA DE CASTILLA SA DE CV	10903	-102.0223	19.39269
666c99414912e74bdfdfc73	regular, 24.7	SERVICIO DEL VARAL S.A. DE C.V.	10477	-102.4873	19.02127
666c99414912e74bdfdfc13	regular, 23.99	INMUEBLES VENTA DE CARPIO, S.A. DE C.V.	10979	-99.0691	19.65874
666c99414912e74bdfdfc73	regular, 24.29	ESTACION DE SERVICIO EL JACAL, S.A. DE C.V.	10130	-100.8853	19.4018
666c99414912e74bdfdfc03	regular, 23.79	SERVICIOS DEL PUERTO SA DE CV	11062	-114.8421	31.02577
666c99414912e74bdfdfb33	regular, 23.79	AUTOSERVICIO BENITEZ, SA DE CV	10555	-117.0589	32.34566
666c99414912e74bdfdfc73	diesel, 26.16	SERVICIO SOLORZANO LOPEZ S.A. DE C.V.	11134	-92.0779	16.89845
666c99414912e74bdfdfc13	regular, 23.69	URZUAS SA DE CV	10303	-97.69217	20.4711
666c99414912e74bdfdfc03	premium, 26.29	SERVICIO PANORAMICA SA DE CV	11329	-101.2577	21.03161
666c99414912e74bdfdfc13	diesel, 26.5	SERVICIO MIRAFLORES SA DE CV	10638	-109.7527	23.37063

Consideraciones Técnicas y Éticas:

Limitaciones técnicas:

- **Dependencia de datos externos:** El proceso ETL depende de la disponibilidad y actualización constante de los datos en el portal datos.gob.mx. Cualquier interrupción o cambio en el formato de los datos podría afectar el funcionamiento del proceso.
- **Escalabilidad:** Aunque el volumen actual de datos es manejable, el proceso podría requerir optimizaciones si el número de estaciones de servicio o la frecuencia de actualización de precios aumentan significativamente.
- **Automatización limitada:** La conversión inicial de XML a JSON se realizó manualmente. Automatizar este paso podría mejorar la eficiencia del proceso en futuras actualizaciones.

Consideraciones éticas:

- **Privacidad de los datos:** Los datos utilizados no contienen información personal identificable, pero es importante garantizar que cualquier dato agregado o derivado durante el proceso ETL no pueda ser utilizado para identificar o rastrear a individuos.
- **Uso responsable de los datos:** Los datos deben utilizarse únicamente para el propósito declarado en el caso de aplicación (ayudar a los consumidores a encontrar las mejores gasolineras) y no para fines comerciales o discriminatorios.
- **Transparencia:** Es fundamental ser transparente sobre el origen de los datos, las transformaciones realizadas y cómo se utilizarán los resultados del análisis.

Medidas de seguridad:

- **Almacenamiento seguro:** Los datos se almacenan en un entorno seguro, tanto en archivos locales como en la base de datos MongoDB, con acceso restringido y medidas de protección contra pérdida o robo de datos.
- **Cifrado:** Se utilizan técnicas de cifrado para proteger los datos durante la transmisión y el almacenamiento.
- **Anonimización:** Se evita el almacenamiento de información personal identificable y se utilizan técnicas de anonimización en caso de ser necesario generar datos agregados.

Conclusiones:

El proceso ETL descrito ha permitido transformar los datos de estaciones de servicio y precios de combustible en un formato adecuado para el análisis de minería de datos. Se han abordado desafíos técnicos como la estructura de datos inconsistente y se han tomado medidas para garantizar la calidad, integridad y seguridad de los datos.

Se espera que este proceso mejore significativamente la calidad y usabilidad de los datos, permitiendo descubrir patrones, tendencias y relaciones relevantes para el caso de aplicación. El análisis de minería de datos podrá identificar las gasolineras con los mejores precios, las zonas con mayor variación de precios y otros insights que ayuden a los consumidores a tomar decisiones informadas.

Próximos pasos:

- **Análisis de minería de datos:** Aplicar técnicas de minería de datos para descubrir patrones y tendencias en los datos transformados.
- **Desarrollo de la aplicación:** Utilizar los resultados del análisis para desarrollar una aplicación que ayude a los consumidores a encontrar la gasolinera más cercana con los mejores precios.
- **Monitoreo y actualización:** Establecer un proceso de monitoreo y actualización periódica de los datos para mantener la relevancia y precisión de la aplicación.

Referencias:

- **Datos.gob.mx:** Portal de datos abiertos del gobierno mexicano:
<https://datos.gob.mx/>
- **Pentaho Data Integration (PDI):** Herramienta de ETL utilizada para la transformación y carga de datos: <https://pentaho.com/pentaho-community-edition/>
- **MongoDB:** Base de datos NoSQL utilizada para almacenar los datos transformados: <https://www.mongodb.com/>