

Informe

Clasificación de Trámites de Construcción en La Paz



**Postgrado
en Informática**

Integrantes:

- Ericka Guiserla Cori Avalo
- Paolo Ramos Mendez
- Gaston Nina Sossa

4 Octubre, 2025

INDICE

Introducción.....	1
1. Fuentes de Datos.....	1
1.1 Fuente de datos 1 - API - GEOSERVER.....	1
1.2 Fuente de datos 2 - Scrapy - HTML.....	2
2. Proceso de ingestión, limpieza, clasificación y almacenamiento.....	3
2.1 Proceso de Ingestión.....	3
2.2 Proceso de Limpieza.....	3
2.3 Proceso de clasificación.....	4
2.4 Proceso de Almacenamiento.....	4
3. Diseño de Pipeline.....	5
3.1 Extraction.....	5
3.2 Transformation.....	6
3.3 Load.....	6
4. Limitaciones y mejoras futuras.....	6
4.1 Limitaciones.....	6
4.2 Mejoras futuras.....	7
Conclusion.....	7
BATCH Y/O STREAMING.....	8
DIFICULTADES.....	8
ANEXOS.....	9

Introducción

Este informe documenta el proceso de ETL aplicado al dataset de trámites municipales de La Paz, Bolivia. La información registrada, clave para la gestión urbana y los servicios municipales, requiere calidad e integridad para garantizar transparencia y eficiencia administrativa.

El análisis se enfocó en la detección de registros duplicados, especialmente en el código catastral, que debería ser un identificador único de propiedades. La duplicidad compromete la confiabilidad de los reportes, la integridad referencial y la toma de decisiones en planificación y control urbano.

Este proceso ETL se enmarca en los esfuerzos de modernización de la gestión municipal y busca establecer las bases técnicas para la depuración y normalización de la data, garantizando que la información sobre trámites municipales sea precisa, consistente y confiable para todos los stakeholders involucrados, desde funcionarios municipales hasta ciudadanos que realizan trámites y desarrolladores urbanos que requieren información precisa para la toma de decisiones.

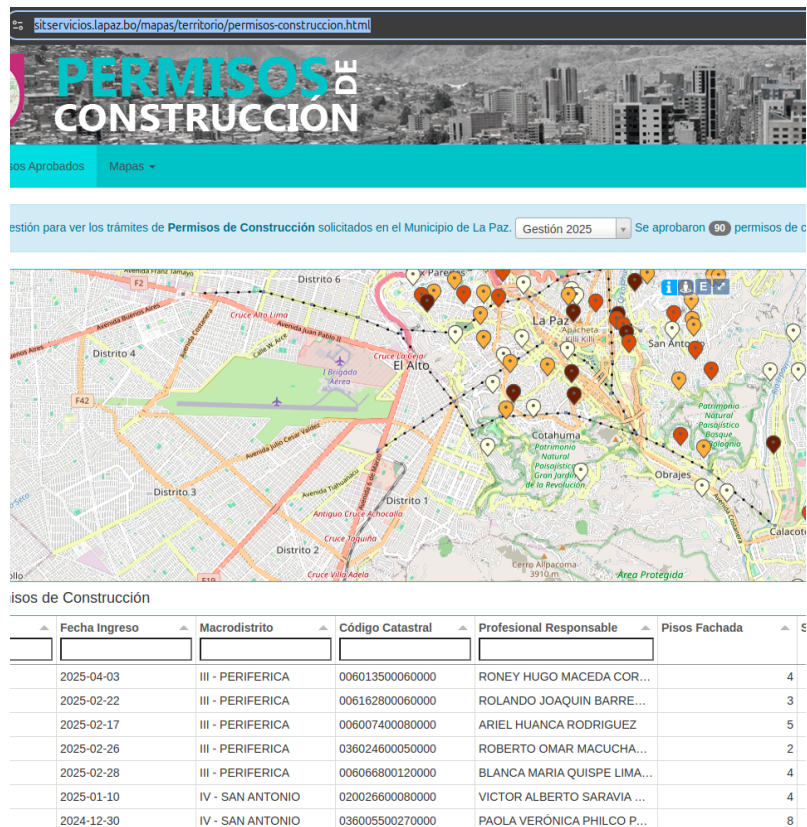
El análisis reveló una significativa presencia de registros duplicados basados en el código catastral. Se identificaron 7,020 registros correspondientes a códigos catastrales repetidos, lo que representa aproximadamente el 50% del total del dataset.

1. Fuentes de Datos

1.1 Fuente de datos 1 - API - GEOSERVER

La plataforma web Permisos de Construcción - La Paz constituye una herramienta de consulta geoespacial que permite acceder a información territorial sobre trámites de construcción en el municipio de La Paz. Esta plataforma se alimenta de un servicio de datos geográficos (GeoServer), el cual expone capas temáticas que pueden ser filtradas dinámicamente según criterios específicos.

Web: <https://sitservicios.lapaz.bo/mapas/territorio/permisos-construccion.html>



1.2 Fuente de datos 2 - Scrapy - HTML

La plataforma SEAT ÚTILES - Permisos de Construcción es una interfaz web que permite la visualización y consulta de datos geospaciales relacionados con permisos de construcción aprobados en el municipio de La Paz. Esta herramienta se integra con un servidor de mapas (GeoServer), desde el cual se extraen datos filtrados para su análisis y visualización.

Web: <https://sitservicios.lapaz.bo/situtiles/pc/?MDQ0MTA0OTAwMDcwMDAwfDYxMTUzfDIwMjU=>

sitservicios.lapaz.bo/sit/utiles/pc/MDQ0MTAD0TAwMDcwMDAwfDYxMTUzDiwMJU=

Gobierno Autónomo Municipal de La Paz • Secretaría Municipal de Planificación

PERMISO DE CONSTRUCCION-LICENCIA AGIL N°61153/2025

Datos Generales

Código Catastral:	044 1049 0007 0000
Informe:	DDAT PSAT 960/2025
Fecha Aprobación:	21/mar./2025
Zona Referencial:	IRPAVI II
Patrón de Asentamiento:	4P - d18

Parámetros de Construcción

Sur 4P - d18			Parámetros Autorizados
ALE	Área de Lote Edificable		516.80 m²
FML	Frente Mínimo de Lote		48.84 m
AMC	Área Máxima a Cubrir	Sótano	0.00 m²
		Semisótano	0.00 m²
		Zócalo	0.00 m²
		Mezzanine	0.00 m²
		Torre	127.83 m²
AME	Área Máxima a Edificar	Zócalo	0.00 m²
		Torre	354.04 m²
AMF	Altura Máxima de Fachada	Sótano	0 plantas
		Semisótano	0 plantas


2. Proceso de ingestión, limpieza, clasificación y almacenamiento

2.1 Proceso de Ingestión.

El proceso de ingestión inicia con el scraping de la página WEB también con consumo de la API, conteniendo 13,907 registros distribuidos en 31 campos. Se implementa una carga controlada utilizando pandas con configuraciones específicas para preservar la integridad de los datos, particularmente en el campo codigo_catastral que se define como string para conservar ceros a la izquierda y formatos catastrales. Durante esta fase se realiza una verificación inicial de integridad que incluye conteo de registros, validación de estructura de columnas, análisis de metadatos básicos y verificación de consistencia en formatos, asegurando que la data cruda mantenga su estructura original antes de proceder a las transformaciones.

2.2 Proceso de Limpieza

La limpieza se focaliza en la identificación y tratamiento de duplicados mediante agrupamiento por codigo_catastral, detectando 7,020 registros repetidos que representan aproximadamente el 50% del dataset. El proceso incluye el análisis de valores nulos across todas las columnas, estandarización de formatos en campos críticos



como fechas y códigos, y validación de consistencia en campos geográficos como coordenadas y distritos. Se implementan técnicas de profiling avanzado para caracterizar la calidad de datos y se establecen reglas de negocio para determinar cuáles duplicados corresponden a trámites legítimos múltiples y cuáles a errores de captura.

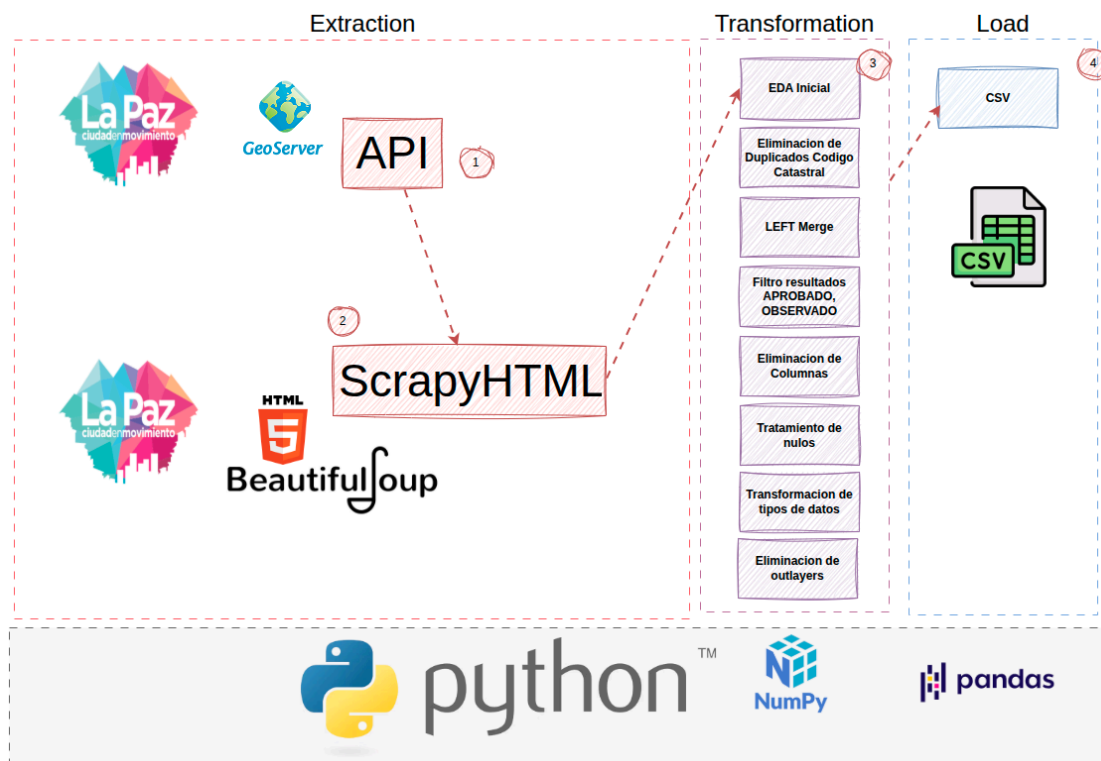
2.3 Proceso de clasificación

Se establece un sistema de clasificación que categoriza los registros según múltiples dimensiones: por tipo de trámite, estado procesal, complejidad del proyecto y distribución territorial. Los duplicados identificados se clasifican en categorías específicas como "múltiples trámites por propiedad", "errores de captura" o "actualizaciones de expedientes". Paralelamente, se enriquece el dataset mediante la integración de metadatos derivados como frecuencia de trámites por zona, tiempos promedio de resolución y patrones de comportamiento por tipo de solicitante, generando insights adicionales para análisis avanzado.

2.4 Proceso de Almacenamiento

El almacenamiento se estructura en capas diferenciadas: datos crudos preservados para auditoría, datos depurados para operaciones diarias, y datos enriquecidos para análisis estratégico. Se implementan controles de calidad post-procesamiento que incluyen validación de integridad referencial, consistencia en relaciones entre tablas y completitud de campos críticos. Finalmente, se establecen mecanismos de disponibilización mediante vistas segmentadas para diferentes usuarios (operativos, analíticos, gerenciales) y se documentan los metadatos del proceso para garantizar la trazabilidad y facilitar futuras actualizaciones del pipeline ETL.

3. Diseño de Pipeline



3.1 Extraction

La extracción es el proceso de obtención de datos desde diversas fuentes hacia nuestro entorno de procesamiento. Implementamos un sistema robusto que maneja múltiples formatos y orígenes, incluyendo archivos locales, APIs REST y bases de datos. Cada fuente requiere configuraciones específicas de conexión y manejo de errores para garantizar la integridad de los datos capturados.

```
# Extracción desde múltiples fuentes
```

```
df_api = pd.read_csv("tramites_lapaz_api_identificador.csv",  
dtype={"codigo_catastral": str})  
datos_externos = pd.read_excel("datos_complementarios.xlsx")
```

Para APIs REST, utilizamos requests con manejo de paginación y límites de tasa, mientras que en bases de datos SQL empleamos conexiones parametrizadas con timeouts. La validación inicial verifica que los datos extraídos cumplan con el schema esperado antes de proceder a la transformación.

3.2 Transformation

En esta fase aplicamos reglas de negocio y limpieza para convertir los datos en un formato analíticamente útil. El proceso incluye tratamiento de valores nulos, estandarización de formatos, enriquecimiento con datos externos y feature engineering. Cada transformación está documentada y versionada para mantener la trazabilidad.

```
# Limpieza y transformación

df_clean = df_api.drop_duplicates(subset=['codigo_catastral'])

df_clean['fecha_registro'] = pd.to_datetime(df_clean['fecha_registro'])
df_clean['superficie_legal'] = df_clean['superficie_legal'].fillna(0)
```

Las transformaciones se ejecutan en un orden lógico: primero la limpieza básica, luego joins con datos de referencia, después cálculos derivados y finalmente validaciones de calidad. Implementamos pruebas unitarias para cada transformación crítica, asegurando que reglas de negocio como "superficie_construida ≤ superficie_legal" se mantengan consistentes.

3.3 Load

La carga consiste en almacenar los datos transformados en destinos específicos según los requisitos de consumo. Diseñamos estrategias de upsert para actualizar registros existentes e insertar nuevos, optimizando el uso de recursos y manteniendo la consistencia de los datos, en nuestro caso la exportación de la data en un archivo csv.

4. Limitaciones y mejoras futuras

4.1 Limitaciones

El pipeline actual presenta varias limitaciones técnicas y operativas que impactan su escalabilidad y mantenimiento. La principal restricción reside en su arquitectura monolítica, donde todas las transformaciones se ejecutan secuencialmente en un solo proceso, creando cuellos de botella en el procesamiento de grandes volúmenes de

datos. Actualmente manejamos aproximadamente 17,000 registros, pero esta aproximación no sería viable al escalar a cientos de miles o millones de registros.

Limitación actual: procesamiento en memoria

```
df_completo = pd.read_csv("datos_masivos.csv") # Crash con archivos muy grandes
```

Otra limitación significativa es la falta de orquestación y monitoreo automatizado. Las ejecuciones dependen de triggers manuales y no contamos con sistemas de alerta temprana para fallos en las fuentes de datos o degradación en la calidad.


4.2 Mejoras futuras

La integración de metadatos y data lineage mediante herramientas como OpenMetadata permitirá una trazabilidad completa de los datos, desde las fuentes originales hasta los productos finales. Esto facilitará la auditoría, el control de calidad y el diagnóstico de incidencias dentro del flujo de datos. Además, la containerización con Docker y la orquestación con Kubernetes asegurará la portabilidad, escalabilidad y alta disponibilidad del pipeline en diversos entornos de ejecución.

Conclusion

El desarrollo del pipeline ETL en Apache Airflow constituye un avance estratégico en la gestión de datos municipales, al establecer un framework robusto y escalable para la orquestación de procesos. Su implementación permite automatizar completamente el flujo de trabajo, programar ejecuciones periódicas, gestionar dependencias y realizar monitoreo en tiempo real mediante la interfaz web.

Gracias a su arquitectura basada en DAGs (Directed Acyclic Graphs), cada etapa de extracción, transformación y carga se ejecuta de forma coordinada y con capacidad de recuperación ante fallos, superando las limitaciones de los scripts tradicionales. Además, Airflow sienta las bases para ampliar el pipeline con nuevas fuentes, transformaciones y destinos, manteniendo la estabilidad del sistema. Sus capacidades nativas de manejo de



errores, reintentos y logging detallado convierten este proyecto en una plataforma de datos flexible .

BATCH Y/O STREAMING

En un entorno ideal, el procesamiento de datos evoluciona del enfoque batch al streaming. Inicialmente, el procesamiento por lotes resulta más apropiado para el entrenamiento y validación de modelos, ya que maneja volúmenes controlados y requiere una infraestructura sencilla. Este enfoque permite definir transformaciones, validar la calidad de los datos y ajustar reglas de negocio.

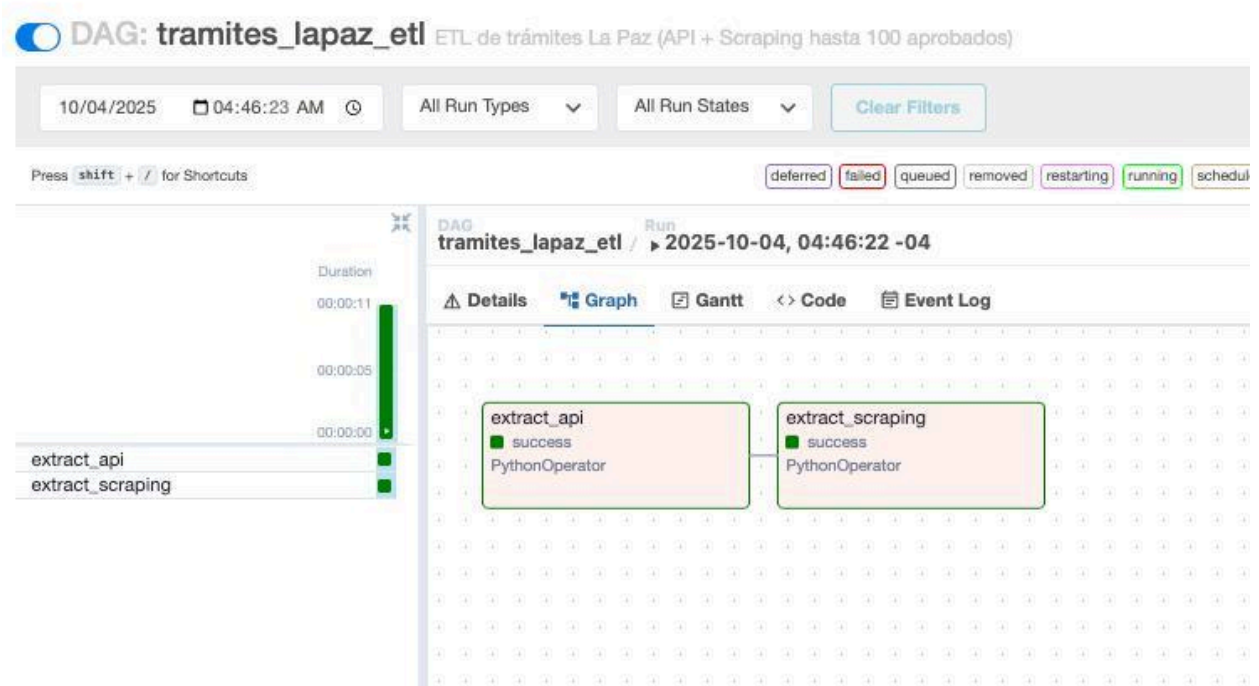
Una vez alcanzada la estabilidad, la transición hacia el procesamiento en tiempo real permite analizar y clasificar datos conforme se generan, combinando precisión analítica e inmediatez operativa, y habilitando respuestas en tiempo real para la toma de decisiones.

DIFICULTADES

La calidad de los datos constituye uno de los principales desafíos en la gestión de información, reflejándose en valores nulos, inconsistentes o erróneos que afectan la confiabilidad del análisis. En el dataset de trámites catastrales, se identificaron columnas con hasta 87.5% de valores faltantes (id_ins_documento, nombre_archivo) y una alta duplicidad de códigos catastrales (25.2% repetidos, algunos hasta 22 veces), lo que distorsiona los resultados y genera redundancia.

Otros problemas comunes incluyen la falta de integridad referencial, que produce registros huérfanos, y la ausencia de estandarización de formatos, especialmente en fechas, códigos y categorías. Asimismo, los problemas de esquema y la escasa documentación dificultan la comprensión y el procesamiento de los datos. Finalmente, la heterogeneidad de tipos, los valores atípicos y los duplicados semánticos requieren técnicas avanzadas de limpieza, normalización y detección para garantizar la consistencia y confiabilidad del dataset.

ANEXOS



DAG: tramites_lapaz_transform

10/04/2025

04:47:06 AM

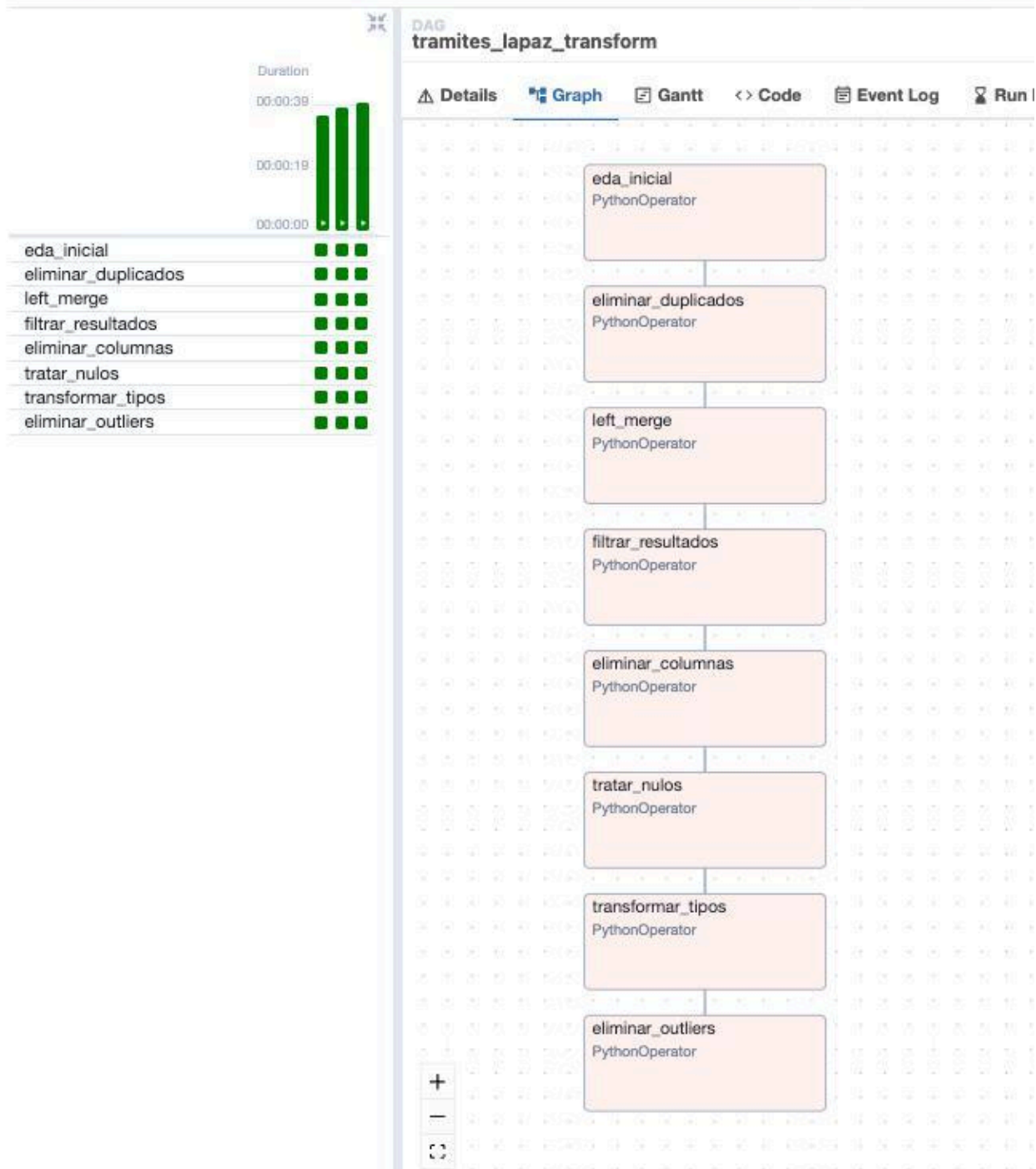
All Run Types

All Run States

Clear Filters

Press **shift** + **/** for Shortcuts

deferred failed queued removed restart



DAG: tramites_lapaz_report

10/04/2025

04:47:43 AM

All Run Types

All Run States

Clear Filters

Press **shift** + **/** for Shortcuts

deferred

failed

queued

removed

restart



DAG tramites_lapaz_report

Details

Graph

Gantt

Code

Event

load_data

PythonOperator

apply_expectations

PythonOperator

generate_report

PythonOperator