ETL PROYECTO 2

ESTEBAN COBO GOMEZ

JOSE DAVID MESA RAMIREZ

CARLOS ANDRES OROZCO

SARA LUCIA ROJAS MEJIA

BREYNER POSSO BAUTISTA

UNIVERSIDAD AUTÓNOMA DE OCCIDENTE

FACULTAD DE INGENIERÍA

INGENIERÍA DE DATOS E INTELIGENCIA ARTIFICIAL

ETL
Proyecto ETL — Agua, Alcantarillado y Aseo (Colombia)

Entrega 2 — Orquestación con Apache Airflow y Modelo Dimensional

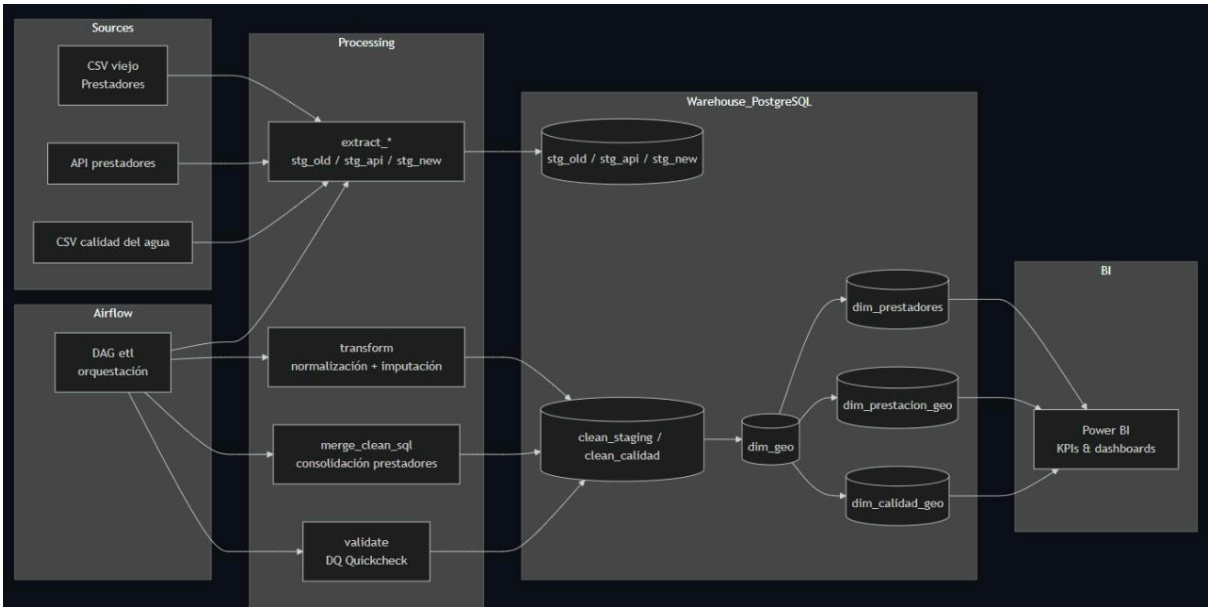
Stack: Python · PostgreSQL · Docker/Compose · Apache Airflow · BI (Power BI)

1) Visión general

El proyecto implementa un pipeline ETL que integra tres fuentes (CSV histórico de prestadores, API de prestadores y CSV de calidad del agua), aplica normalización y controles de calidad, y publica tres dimensiones conformes con clave Departamento–Municipio (y provider_id en prestadores). El diseño sigue modelado dimensional (Kimball), lo que permite comparaciones entre procesos apoyadas en dimensiones compartidas y consultas de tipo *drill-across* cuando en el futuro se agreguen tablas de hechos. (Kimball Group)

La orquestación se realiza con Apache Airflow, que define tareas, dependencias y programación como código (DAGs). (Apache Airflow)

2) Arquitectura



Zonas del flujo:

- **Fuentes**:
  stg_old (CSV histórico de prestadores), stg_api (prestadores desde API), stg_new (CSV de calidad).

- **Procesamiento**:
  extract_* → transform (limpieza y normalización) → merge_clean_sql (consolidación final de prestadores) → validate (calidad de datos).

- **Data Warehouse (DW)**:
  clean_staging (prestadores) · clean_calidad (calidad) · **dimensiones** dim_prestadores, dim_prestacion_geo, dim_calidad_geo.

Airflow modela el grafo de tareas y dependencias; el **scheduler** aplica políticas de reintentos, SLAs y replanificación. Apache Airflow+2Apache Airflow+2

3) Datasets incluidos

3.1 stg_old — CSV histórico de prestadores

Aporta **NIT**, nombre, **departamento/municipio de prestación**, servicio, estado, clasificación y **contacto** (dirección, teléfono, e-mail).

3.2 stg_api — API de prestadores

Aporta nombre, departamento/municipio, servicio, estado, clasificación (no trae contacto).

3.3 stg_new — CSV de calidad del agua

Aporta dpto/mun, **fecha** de muestreo, **parámetro**, **resultado**, **unidad**, **nombre del punto** y **coordenadas** (latitud/longitud).

Granos lógicos:

- Prestadores: (provider_id, servicio, departamento, municipio).
- Calidad: (departamento, municipio, parámetro, fecha, nombre_punto).

4) Proceso ETL

4.1 Extracción (Extract)

Objetivo: llevar fuentes heterogéneas a *staging* normalizado (tipos de datos seguros, sin lógica de negocio).

- **extract_old**: lee CSV, castea campos, deja en stg_old.
- **extract_api**: consulta API (prestadores), normaliza a esquema stg_api.
- **extract_new**: ingesta del CSV de calidad a stg_new.

Buenas prácticas Airflow: cada extracción es una **tarea**; dependencias explícitas hacia transform. Apache Airflow

4.2 Transformación (Transform)

Meta: construir tablas limpias clean_* aplicando **estandarización**, **reglas de plausibilidad**, **imputación** y **deduplicación**.

4.2.1 Prestadores → clean_staging

- **Normalización de strings**: UPPER, TRIM, **sin tildes** (compatibilidad de dominios).
- **Clave técnica** provider_id:
  COALESCE(nit, md5(UPPER(nombre)|dep|mun|servicio))
  (estable cuando falte NIT; idéntico input ⇒ idéntico id).
- **Servicio**: mapeo a {ACUEDUCTO, ALCANTARILLADO, ASEO}; no mapeables ⇒ DESCONOCIDO (no se pierden filas).
- **Estado**: consolidación {OPERATIVA, SUSPENDIDA, OTRO} con **imputación por moda** (partición por servicio, departamento).
- **Contacto**: direccion/telefono/email se preservan **solo** si vienen de stg_old.
- **Dominio geográfico**: departamento restringido a catálogo oficial del proyecto.
- **Deduplicación**: (provider_id, servicio, departamento, municipio) (ROW_NUMBER/ctid).
- **Salida**:
  clean_staging(provider_id, nombre, departamento, municipio, servicio, estado, clasificacion, direccion, telefono, email).

Razonamiento dimensional: al **conformar** departamento/municipio aseguramos que futuras tablas de hechos puedan **drill-across** con estas mismas llaves. Kimball Group+1

4.2.2 Calidad → clean_calidad

- **Fecha**: parse robusto a DATE (fecha_muestra).
- **Valor**: limpieza de símbolos y casteo a double precision.
- **Coordenadas (Colombia)**: lat ∈ [-5, 15], lon ∈ [-82, -66]; si falta una coordenada, ambas en NULL.
- **Plausibilidad**:

    - **pH**: rango físico **[0, 14]** (valores fuera ⇒ NULL para imputar).
    - **Cloro residual**: **filtro de plausibilidad** 0–5 mg/L; para **operación** se usan umbrales OMS (≥0.5 mg/L tras 30 min a pH<8; ≥0.2 mg/L en entrega), **solo como KPI** de lectura, no para descartar datos. NCBI+2World Health Organization+2

- **Imputación**:
    - unidad: **moda** por parámetro.
    - valor: **mediana** por (parámetro, departamento); *fallback* mediana **global** del parámetro.

- **Unicidad**: (departamento, municipio, parámetro, fecha_muestra, COALESCE(nombre_punto,'')).

- **Salida**:
  clean_calidad(departamento, municipio, parametro, valor, fecha_muestra, unidad, nombre_punto, latitud, longitud).

4.3 Carga (Load) a DW

- **Consolidación de prestadores** (merge_clean_sql): une stg_old + stg_api ya normalizados por transform y refuerza **deduplicación** y **campos de contacto** desde histórico.

- **Construcción de dimensiones** (SQL):

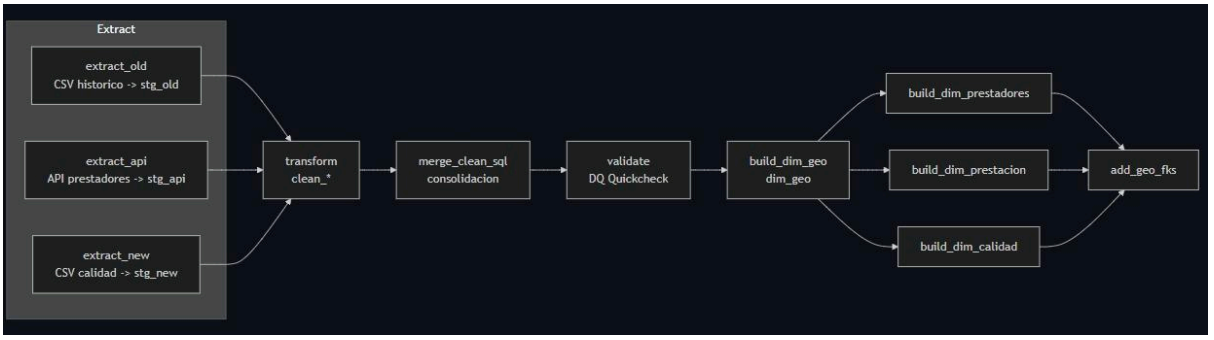
  dim_prestadores (PK (departamento, municipio, provider_id)): atributos de prestador + contacto.

  dim_prestacion_geo (PK (departamento, municipio)): **agregados** por servicio/estado (totales por municipio).

  dim_calidad_geo (PK (departamento, municipio)): agregados de **puntos**, **mediciones**, **parámetros distintos**, **estado_pH/estado_cloro** y **fecha_últ_muestra**.

- **Validación automática** (validate): si fallan **bloqueantes**, el DAG **falla** (early-stop). Apache Airflow

5) DAG de Airflow



extract_old, extract_new, extract_api
→ transform
→ merge_clean_sql
→ validate
→ build_dim_prestadores, build_dim_calidad, build_dim_prestacion

- **Extract**: ingesta a *staging*.
- **Transform**: reglas de limpieza/normalización → clean_*.
- **Merge**: consolidación final de prestadores (dedupe/contacto).
- **Validate**: **DQ Quickcheck** — reglas **bloqueantes** y **avisos** (no bloqueantes).
- **Build dims**: materialización de dimensiones agregadas.

  Declaración de dependencias y semántica de tareas según documentación oficial de Airflow. Apache Airflow+1

Airflow proporciona el marco de DAGs, dependencias y programación como código. (Apache Airflow)

6) Tablas de transformaciones (por variable)

6.1 clean_staging (prestadores)

Variable	Origen	Transformación aplicada	Tipo/Regla	Notas
provider_id	NIT/derivado	`COALESCE(nit, md5(UPPER(nombre)	dep	mun
nombre	texto	UPPER + TRIM + quitar tildes	Normalización	Consistencia nominal
departamento	texto	UPPER/TRIM/sin tildes; validado contra catálogo	Dominio	Rechazo si fuera de catálogo
municipio	texto	UPPER/TRIM/sin tildes	Normalización	—
servicio	texto	Mapeo %ACUED%→ACUEDUCTO; %ALCANT%→ALCANTARILLADO; %ASEO%→ASEO; else DESCONOCIDO	Clasificación controlada	Conservar registros no mapeables
estado	texto	Normalización (OPERATIVA, SUSPENDIDA, OTRO) + imputación por moda (servicio,dpto)	Imputación + dominio	Fallback OTRO
clasificacion	texto	UPPER/TRIM/sin tildes; imputación por moda (servicio)	Imputación	—
direccion	stg_old	Copia directa si existe	Preservación	Sólo histórico
telefono	stg_old	Copia directa; aviso si regex inválida	Validación suave	No bloquea
email	stg_old	Copia directa; aviso si regex inválida	Validación suave	No bloquea
Dedupe	—	(provider_id, servicio, departamento, municipio)	De-duplicación	ROW_NUMBER /ctid

Purgado	—	Eliminar filas con claves nulas/vacías	Calidad (bloqueante)	—

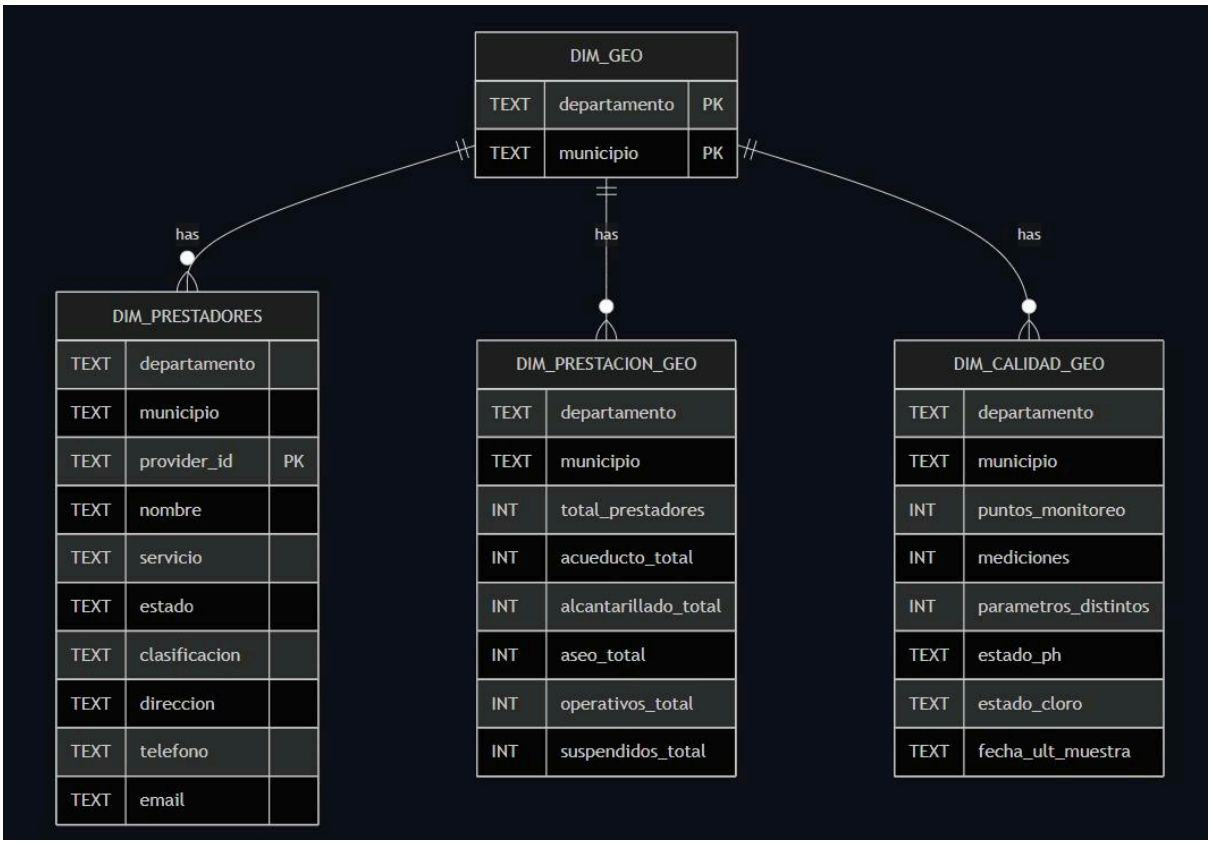
6.2 clean_calidad (calidad del agua)

Variable	Origen	Transformación aplicada	Tipo/Regla	Notas
departamento	texto	UPPER/TRIM/sin tildes; dominio (catálogo)	Dominio	—
municipio	texto	UPPER/TRIM/sin tildes	Normalización	—
fecha_muestra	texto	Parse robusto a DATE	Tipificación	Rango >= 2000-01-01 y <= CURRENT_DATE
parametro	texto	UPPER/TRIM/sin tildes	Normalización	—
valor	texto	Limpiar símbolos → double precision	Tipificación	Plausibilidad: pH [0,14]; Cloro 0–5 mg/L (filtro) support.esdat.net
unidad	texto	TRIM; imputación (moda por parámetro)	Imputación	—
nombre_punto	texto	TRIM	Normalización	Para unicidad por punto
latitud	texto	Casteo seguro; rango [-5, 15]; si longitud nula → poner ambas NULL	Plausibilidad + pareo de nulidad	—
longitud	texto	Casteo seguro; rango [-82, -66]; si latitud nula → poner ambas NULL	Plausibilidad + pareo de nulidad	—
Dedupe	—	(departamento, municipio, parametro, fecha_muestra, COALESCE(nombre_punto,''))	De-duplicación	—

Imputación valor	—	Mediana por (parámetro, departamento); *fallback* mediana global del parámetro	Imputación robusta a outliers	—

Las reglas de cloro **operativas** OMS (≥0.5 mg/L tras 30 min; ≥0.2 mg/L en entrega) se usan para **KPI/semáforos**, no como filtro de eliminación (solo control de plausibilidad 0–5). NCBI+1

7) Dimensiones (modelo estrella)



Se publican tres dimensiones con claves:

1. dim_prestadores — PK: (departamento, municipio, provider_id)
   Atributos: nombre, servicio, estado, clasificacion, direccion, telefono, email.

2. dim_prestacion_geo — PK: (departamento, municipio)
   Atributos agregados: total_prestadores, acueducto_total, alcantarillado_total, aseo_total, operativos_total, suspendidos_total.

3. dim_calidad_geo — PK: (departamento, municipio)
   Atributos agregados: puntos_monitoreo, mediciones, parametros_distintos, estado_ph, estado_cloro, fecha_ult_muestra.

Estas dimensiones son conformes (comparten departamento y municipio con dominios normalizados), lo que facilita análisis consistentes entre procesos e implementa la base para *drill-across* cuando existan hechos adicionales. (Kimball Group)

8) Validación — "Dimensión de Calidad de Datos" (reglas, métricas, umbrales)

Basado en dimensiones ampliamente usadas (**DAMA**: completitud, exactitud, consistencia, validez, unicidad, oportunidad), definimos **reglas bloqueantes** y **avisos** con métricas asociadas. sbctc.edu+2Collibra+2

7.1 Tabla de reglas y métricas

Dimensión (DAMA)	Regla / Métrica	Tabla/Ámbito	Umbral	Severidad
Completitud	Claves no nulas (provider_id, nombre, departamento, municipio, servicio)	clean_staging	0 violaciones	**Bloqueante**
Completitud	Claves no nulas (departamento, municipio, fecha_muestra, parametro)	clean_calidad	0 violaciones	**Bloqueante**
Unicidad	Duplicados en (provider_id, servicio, dpto, muni)	clean_staging	0	**Bloqueante**
Unicidad	Duplicados en (dpto, muni, parametro, fecha, punto)	clean_calidad	0	**Bloqueante**
Validez	departamento dentro del catálogo	ambos	100% válidos	**Bloqueante**
Validez	fecha_muestra $\in$ [2000-01-01, hoy]	clean_calidad	100% válidas	**Bloqueante**

Validez	lat/lon en rangos COL; **pareo** de nulidad	clean_calidad	100% válidas/pareadas	**Bloqueante**
Exactitud / Plausibilidad	pH ∈ [0,14] (fuera ⇒ NULL + imputación)	clean_calidad	0 fuera de rango	**Bloqueante**
Exactitud / Plausibilidad	Cloro 0–5 mg/L (fuera ⇒ NULL + imputación)	clean_calidad	0 fuera de rango	**Bloqueante**
Consistencia	% DESCONOCIDO en servicio/dep/mun	clean_staging	≤ 5% (configurable)	**Aviso**
Validez (formato)	Regex e-mail / teléfono	clean_staging	Aviso (no bloquea)	**Aviso**
Oportunidad	Municipios sin muestras en período reciente	clean_calidad	Umbral reportado (p. ej. 90 días)	**Aviso**

> La separación entre **bloqueantes** y **avisos** permite cortar el DAG cuando se compromete integridad, manteniendo alertas informativas para limpieza futura. sbctc.edu

9) KPIs y visualizaciones sugeridas

- Prestación: # de prestadores por municipio/servicio; % operativos vs. suspendidos.
- Calidad: semáforos por municipio para pH y cloro; # de puntos y mediciones; fecha de última muestra.
- Cobertura: municipios sin mediciones recientes.

> Los tableros deben consumir las dimensiones; las claves geográficas compartidas garantizan comparabilidad entre dominios mediante dimensiones conformes. (Kimball Group)

Referencias

- Kimball Group — Conformed Dimensions & Drill-Across. (Kimball Group)
- Apache Airflow — DAGs y conceptos básicos (docs oficiales). (Apache Airflow)

- DAMA — Dimensiones de calidad de datos. (sbctc.edu)

- GitHub — Diagramas Mermaid en Markdown. (GitHub Docs)
- Link Github(https://github.com/EstebanC111s/ETL_Second_Delivery)
- Link diapositivas
- (https://prezi.com/view/HrJpRvnppcXUwNmW4a3S/?referral_token=gyxxkHlnB3FN)