

— Arquitectura del Pipeline de Datos (BP010)

1. Introducción

Este documento explica, de forma clara y accesible, cómo funciona el pipeline de datos que procesa la información de los pozos. El objetivo es que cualquier persona del equipo pueda entender:

- Qué hace cada etapa
- Por qué existe
- Cómo se conectan entre sí
- Qué valor aporta al negocio

2. Visión General del Flujo

El pipeline transforma datos crudos del campo en información lista para dashboards.

Código

1. Ingesta Real (Stage)
2. Motor de Calidad de Datos (V5)
3. Reporting Engine (V2)
4. Snapshot Engine (V3)
5. Motor de Colores y Targets (V5)
6. Dataset Final para Dashboards

Cada etapa cumple un rol específico dentro del proceso.

3. Etapa 1 — Ingesta Real (Stage)

“Traemos los datos tal como vienen del campo.”

Qué hace:

- Recibe datos crudos desde archivos, API o telemetría.
- Los guarda sin modificar en tablas de staging.

Tablas involucradas:

- stage.tbl_pozo_produccion
- stage.tbl_pozo_maestra

Por qué existe:

- Para tener un lugar seguro donde almacenar datos sin procesar.
- Para no contaminar las tablas finales con datos incompletos o erróneos.

Ejemplo:

Si un sensor envía 100 lecturas por minuto, aquí se guardan todas tal cual llegan.

4. Etapa 2 — Motor de Calidad de Datos (V5)

“Detectamos valores incorrectos, fuera de rango o sospechosos.”

Qué hace:

- Revisa cada lectura del pozo.
- Compara contra reglas definidas en tablas referenciales.
- Registra errores en stage.tbl_pozo_scada_dq.

Por qué existe:

- Para asegurar que los datos usados en reportes sean confiables.
- Para detectar sensores rotos o valores imposibles.

Ejemplo:

Si un pozo nunca puede tener 5000 PSI y llega ese valor, se marca como error.

5. Etapa 3 — Reporting Engine (V2)

“Construimos el histórico: hora, día y mes.”

Qué hace:

- Calcula agregados históricos:
 - Horarios
 - Diarios
 - Mensuales
- Carga los hechos en tablas de reporting.

Por qué existe:

- Para análisis de tendencias.
- Para reportes de producción.
- Para KPIs históricos.

Ejemplo:

Si un pozo produce 100 barriles por hora, aquí se calcula el total del día y del mes.

6. Etapa 4 — Snapshot Engine (V3)

“Tomamos la última lectura de cada pozo y la dejamos lista para mostrar.”

Qué hace:

- Busca la **última lectura válida** de cada pozo.
- La normaliza.
- La guarda en reporting.dataset_current_values.

Por qué existe:

- Para mostrar en dashboards el estado actual del pozo.
- Para tener un único registro por pozo, siempre actualizado.

Ejemplo:

Si un pozo manda 1000 lecturas por día, aquí solo queda la última.

7. Etapa 5 — Motor de Colores y Targets (V5)

“Convertimos números en semáforos.”

Qué hace:

- Compara los valores actuales contra:
 - límites operativos
 - targets
 - tolerancias
- Calcula:
 - colores (verde, amarillo, rojo)
 - variaciones porcentuales
 - niveles de severidad

Por qué existe:

- Para que un dashboard pueda mostrar si un pozo está:
 - bien
 - en riesgo
 - en condición crítica

Ejemplo:

Si el SPM objetivo es 10 y el pozo está en 14, se pinta amarillo o rojo según la tolerancia.

8. Etapa 6 — Dataset Final para Dashboards

Qué contiene:

- Últimos valores del pozo
- Estado de comunicación
- Semáforos
- Variaciones
- KPIs
- Flags de calidad de datos

Por qué existe:

- Para que cualquier dashboard (Power BI, Grafana, web) pueda consumir datos listos sin cálculos adicionales.

9. Resumen Visual del Flujo Completo

Código

[1] INGESTA REAL

↓

[2] MOTOR DQ (V5)

↓

[3] REPORTING ENGINE (V2)

↓

[4] SNAPSHOT ENGINE (V3)

↓

[5] COLOR & TARGET ENGINE (V5)

↓

[6] DATASET FINAL (para dashboards)