Consultas

v_compras_norm:

- Qué hace: normaliza la tabla RAW de compras (unidades → KG, moneda → USD/KG; etiqueta proveedor).
- Join: ninguno (solo BD abastecimiento).
- Analítica: base común para comparar costos entre proveedores y cruzar con otras entidades.

```
CREATE OR REPLACE VIEW alicorp_db_6.v_compras_norm AS
SELECT
 TRY_CAST(purchase_order_id AS BIGINT)          AS purchase_order_id,
 TRY_CAST(po_item AS INTEGER)                   AS po_item,
 TRIM(material_id)                              AS material_id,
 TRIM(material_desc)                            AS material_desc,
 TRIM(category)                                 AS category,
 TRIM(supplier_id)                              AS supplier_id,
 TRIM(supplier_name)                            AS supplier_name,
 TRIM(supplier_country)                         AS supplier_country,
 TRIM(incoterm)                                 AS incoterm,
 UPPER(TRIM(currency))                          AS currency,
 TRY_CAST(price_per_unit AS DOUBLE)             AS price_per_unit,
 TRY_CAST(quantity AS DOUBLE)                   AS quantity,
 UPPER(TRIM(unit))                              AS unit,
 TRIM(plant)                                    AS plant,
 TRIM(order_date)                               AS order_date,
 TRIM(promised_date)                            AS promised_date,
 TRIM(delivery_date)                            AS delivery_date,
 TRY_CAST(projected_usd_per_kg AS DOUBLE)       AS projected_usd_per_kg,
 TRY_CAST(fx_pen_per_usd_at_order AS DOUBLE)    AS fx_pen_per_usd_at_order,
 -- cantidad a KG
 CASE WHEN unit='KG' THEN quantity
    WHEN unit='TM' THEN quantity*1000.0 END     AS qty_kg,
 -- costo a USD/KG
 CASE
  WHEN unit='KG' AND currency='USD' THEN price_per_unit
  WHEN unit='TM' AND currency='USD' THEN price_per_unit/1000.0
  WHEN unit='KG' AND currency<>'USD' THEN price_per_unit/NULLIF(fx_pen_per_usd_at_order,0)
  WHEN unit='TM' AND currency<>'USD' THEN (price_per_unit/1000.0)/NULLIF(fx_pen_per_usd_at_order,0)
 END                                            AS unit_cost_usd_per_kg,

 CASE WHEN LOWER(supplier_country) IN ('perú','peru') THEN 'Local' ELSE 'Internacional' END AS supplier_type
FROM alicorp_db_6.abastecimiento
WHERE purchase_order_id IS NOT NULL;
```

Consultas

v_compras_costomes

- Qué hace: resume costo real promedio USD/KG por material y mes (a partir de v_compras_norm).
- Join: ninguno (aún).
- Analítica: base para comparar vs proyección y para series de tiempo de costos.

```
CREATE OR REPLACE VIEW alicorp_db_6.v_compras_costomes AS
SELECT
 TRIM(material_id) AS material_id,
 date_trunc('month', TRY(CAST(date_parse(TRIM(order_date), '%Y-%m-%d') AS DATE))) AS order_month,
 AVG(unit_cost_usd_per_kg) AS avg_unit_cost_usd_per_kg
FROM alicorp_db_6.v_compras_norm
WHERE order_date IS NOT NULL
GROUP BY 1,2;
```

v_dev_costos_mes

- Qué hace: calcula desviación de costo real vs proyección por mes/material usando la última proyección disponible, mes de compra.
- Join: [v_compras_costomes ↔ v_demanda_agg] por [material_id y [period_month <= order_month] (Para una compra que hice en el mes 3, quiero que la cruces con todos los pronósticos que tenía para ese material en mes 1, mes 2 y mes 3.).
- Analítica: monitoreo de desviaciones, performance de forecast, alertas de sobrecostos/ahorros.

```
CREATE OR REPLACE VIEW alicorp_db_6.v_dev_costos_mes AS
WITH joined AS (
 SELECT
   c.material_id,
   c.order_month,
   c.avg_unit_cost_usd_per_kg,
   d.period_month,
   d.proj_unit_cost_usd_per_kg,
   ROW_NUMBER() OVER (
    PARTITION BY c.material_id, c.order_month
    ORDER BY d.period_month DESC
   ) AS rn
 FROM alicorp_db_6.v_compras_costomes c
 LEFT JOIN alicorp_db_6.v_demanda_agg d
   ON d.material_id = c.material_id
  AND d.period_month <= c.order_month
)
SELECT
 material_id,
 order_month,
 avg_unit_cost_usd_per_kg,
 proj_unit_cost_usd_per_kg,
 100.0 * (avg_unit_cost_usd_per_kg - proj_unit_cost_usd_per_kg)
     / NULLIF(proj_unit_cost_usd_per_kg, 0) AS dev_pct
FROM joined
WHERE rn = 1;
```

Consultas

v_compras_riesgo

- Qué hace: integra riesgo país a las compras normalizadas (por país del proveedor).
- Join: [v_compras_norm ↔ riesgo_externo] por [supplier_country = country_name]
- Analítica: ranking costo–riesgo, segmentación de proveedores y alertas de vulnerabilidad.

```
CREATE OR REPLACE VIEW alicorp_db_6.v_compras_riesgo AS
SELECT
  c.supplier_id,
  c.supplier_name,
  c.supplier_country,
  c.supplier_type,
  r.country_name,
  r.risk_score,
  r.risk_band,
  AVG(c.unit_cost_usd_per_kg) AS avg_unit_cost_usd_per_kg,
  COUNT(*) AS total_orders
FROM alicorp_db_6.v_compras_norm c
LEFT JOIN alicorp_db_6.riesgo_externo r
  ON LOWER(TRIM(c.supplier_country)) = LOWER(TRIM(r.country_name))
GROUP BY
  c.supplier_id, c.supplier_name, c.supplier_country, c.supplier_type,
  r.country_name, r.risk_score, r.risk_band;
```

Consultas

v_costos_vs_proy_riesgo

- Qué hace: combina costo real vs proyección (como en v_dev_costos_mes) y agrega riesgo país en el mismo resultado.
- Joins:
    - [v_compras_norm ↔ v_demanda_agg] por [material_id y mes].
    - [v_compras_norm ↔ riesgo_externo] por [país del proveedor].
- Analítica: tablero ejecutivo costo, riesgo, desviación para priorizar proveedores y acciones.

```
CREATE OR REPLACE VIEW alicorp_db_6.v_costos_vs_proy_riesgo AS
WITH compras_mes AS (
 SELECT
  TRIM(material_id) AS material_id,
  date_trunc('month', TRY(CAST(date_parse(TRIM(order_date), '%Y-%m-%d') AS DATE))) AS order_month,
  supplier_name,
  supplier_country,
  AVG(unit_cost_usd_per_kg) AS avg_unit_cost_usd_per_kg
 FROM alicorp_db_6.v_compras_norm
 WHERE order_date IS NOT NULL
 GROUP BY 1,2,3,4
),
joined AS (
 SELECT
  c.material_id,
  c.order_month,
  c.supplier_name,
  c.supplier_country,
  c.avg_unit_cost_usd_per_kg,
  d.period_month,
  d.proj_unit_cost_usd_per_kg,
  ROW_NUMBER() OVER (
   PARTITION BY c.material_id, c.order_month, c.supplier_name, c.supplier_country
   ORDER BY d.period_month DESC
  ) AS rn
 FROM compras_mes c
 LEFT JOIN alicorp_db_6.v_demanda_agg d
  ON d.material_id = c.material_id
 AND d.period_month <= c.order_month
)
SELECT
 j.material_id,
 j.order_month,
 j.supplier_name,
 j.supplier_country,
 r.risk_score,
 r.risk_band,
 j.avg_unit_cost_usd_per_kg,
 j.proj_unit_cost_usd_per_kg,
 100.0 * (j.avg_unit_cost_usd_per_kg - j.proj_unit_cost_usd_per_kg)
     / NULLIF(j.proj_unit_cost_usd_per_kg, 0) AS dev_pct
FROM joined j
LEFT JOIN alicorp_db_6.riesgo_externo r
 ON LOWER(TRIM(j.supplier_country)) = LOWER(TRIM(r.country_name))
WHERE j.rn = 1;
```

Consultas

v_leadtime_riesgo

- Qué hace: calcula lead time plan vs real y el atraso promedio por proveedor, integrando riesgo país.
- Join: [v_compras_norm ↔ riesgo_externo] por [país].
- Analítica: cumplimiento de entregas vs riesgo, base para alertas y SLAs.

```
CREATE OR REPLACE VIEW alicorp_db_6.v_leadtime_riesgo AS
SELECT
  c.material_id,
  c.supplier_name,
  c.supplier_country,
  r.risk_band,
  AVG(
    date_diff('day',
      TRY(CAST(date_parse(c.order_date,   '%Y-%m-%d') AS DATE)),
      TRY(CAST(date_parse(c.promised_date, '%Y-%m-%d') AS DATE))
    )
  ) AS lead_plan_d,
  AVG(
    date_diff('day',
      TRY(CAST(date_parse(c.order_date,   '%Y-%m-%d') AS DATE)),
      TRY(CAST(date_parse(c.delivery_date, '%Y-%m-%d') AS DATE))
    )
  ) AS lead_real_d,
  AVG(
    date_diff('day',
      TRY(CAST(date_parse(c.promised_date, '%Y-%m-%d') AS DATE)),
      TRY(CAST(date_parse(c.delivery_date, '%Y-%m-%d') AS DATE))
    )
  ) AS atraso_vs_promesa_d
FROM alicorp_db_6.v_compras_norm c
LEFT JOIN alicorp_db_6.riesgo_externo r
  ON LOWER(TRIM(c.supplier_country)) = LOWER(TRIM(r.country_name))
GROUP BY
  c.material_id, c.supplier_name, c.supplier_country, r.risk_band;
```