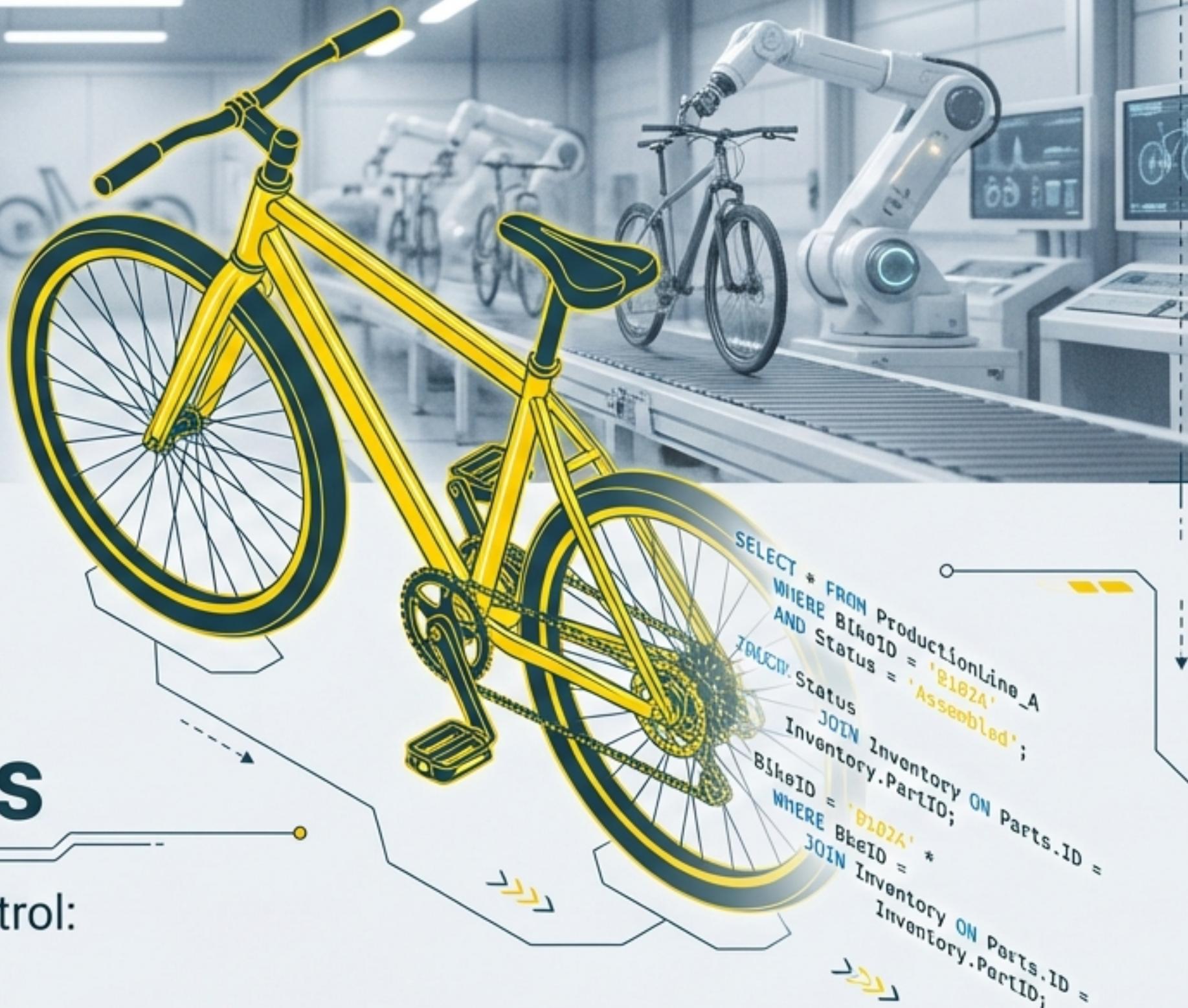




SQL para Análisis Gerencial y Toma de Decisiones

De la línea de ensamblaje al tablero de control:
Optimizando el taller.



Excel: Herramienta Manual



- ⚙️ Límite de 1 millón de filas
- ⚙️ Propenso a error humano
- ⚙️ Lento (VLOOKUP) ⏲

SQL: Automatización Industrial



- ⌚ Millones de registros
- ⌚ Repetible y auditabile
- ⌚ Velocidad instantánea ⚡

Dejen de 'manipular' datos con las manos.
Empiecen a 'interrogarlos' con código.

Conexión de Datos con SQL (JOIN)

Inventario_Bicis		
ID_Bici	Modelo	Costo_Prod
101	MTB-29	300
102	Ruta-Pro	550

Ventas_Semana			
ID_Venta	ID_Bici	Precio_Venta	Fecha
V-99	101	450	2023-10-01
V-100	102	800	2023-10-02

El Eslabón de Conexión



Dejen de 'manipular' datos con las manos.
Empiecen a 'interrogarlos' con código.



¿Qué modelo de bicicleta generó el mayor margen de ganancia esta semana?

Inventario_Bicis		
ID_Bici	Modelo	Costo_Prod
101	MTB-29	300
102	Ruta-Pro	550

Ventas tiene el dinero (\$).

Inventario tiene el modelo.

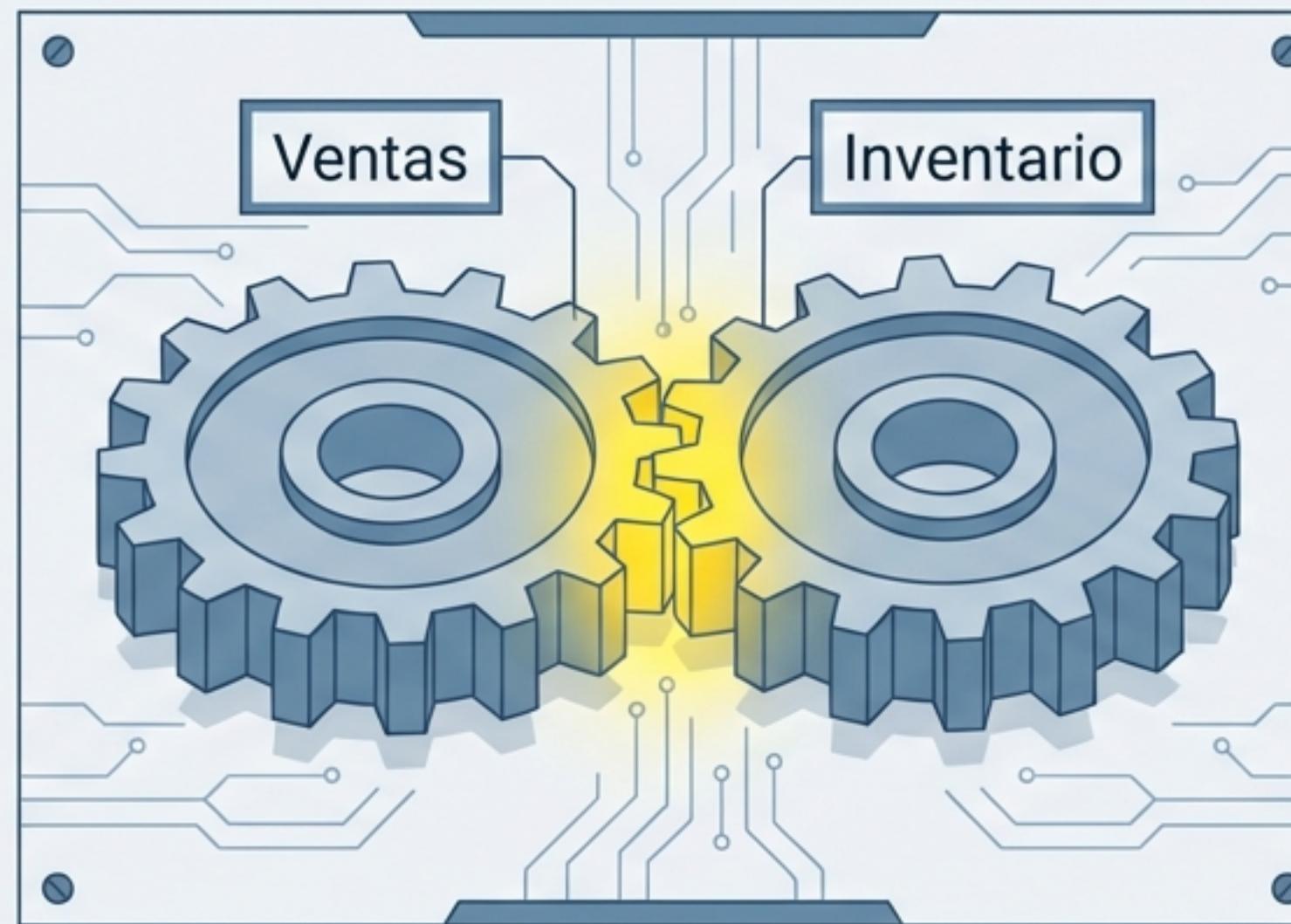
Ventas_Semana			
ID_Venta	ID_Bici	Precio_Venta	Fecha
V-99	101	450	2023-10-01
V-100	102	800	2023-10-02



Están aislados en silos diferentes.

Necesitamos un puente de soldadura.

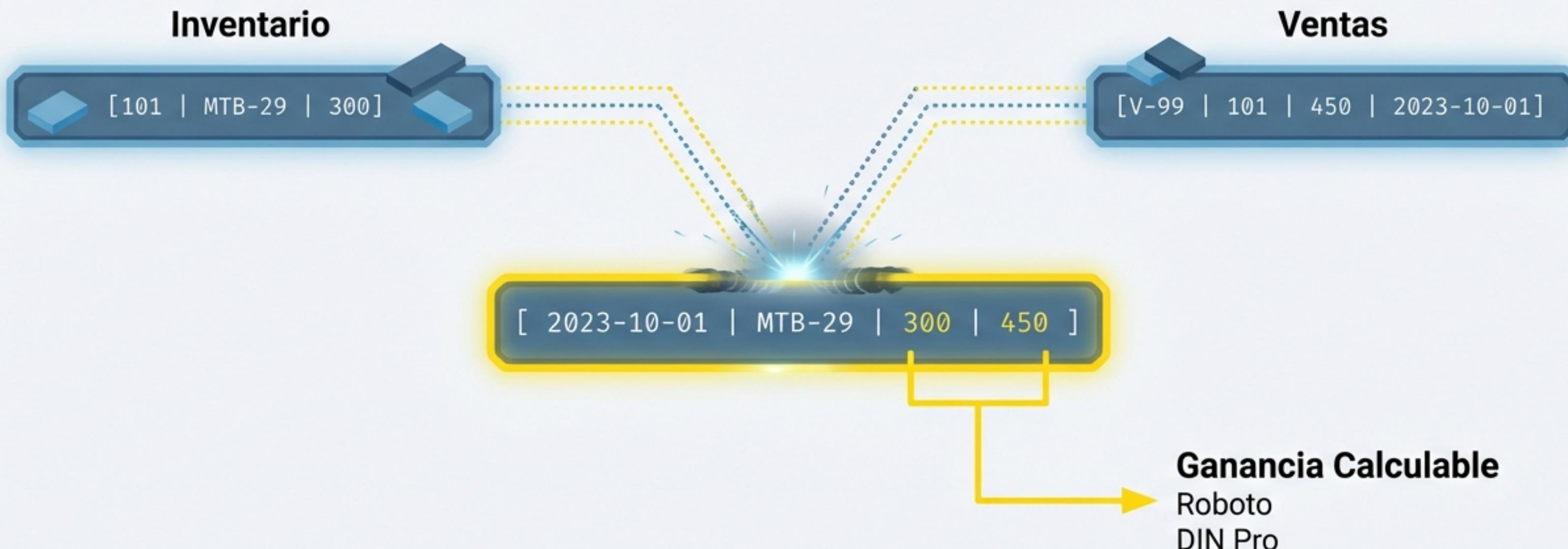
INNER JOIN: La Soldadura de Precisión



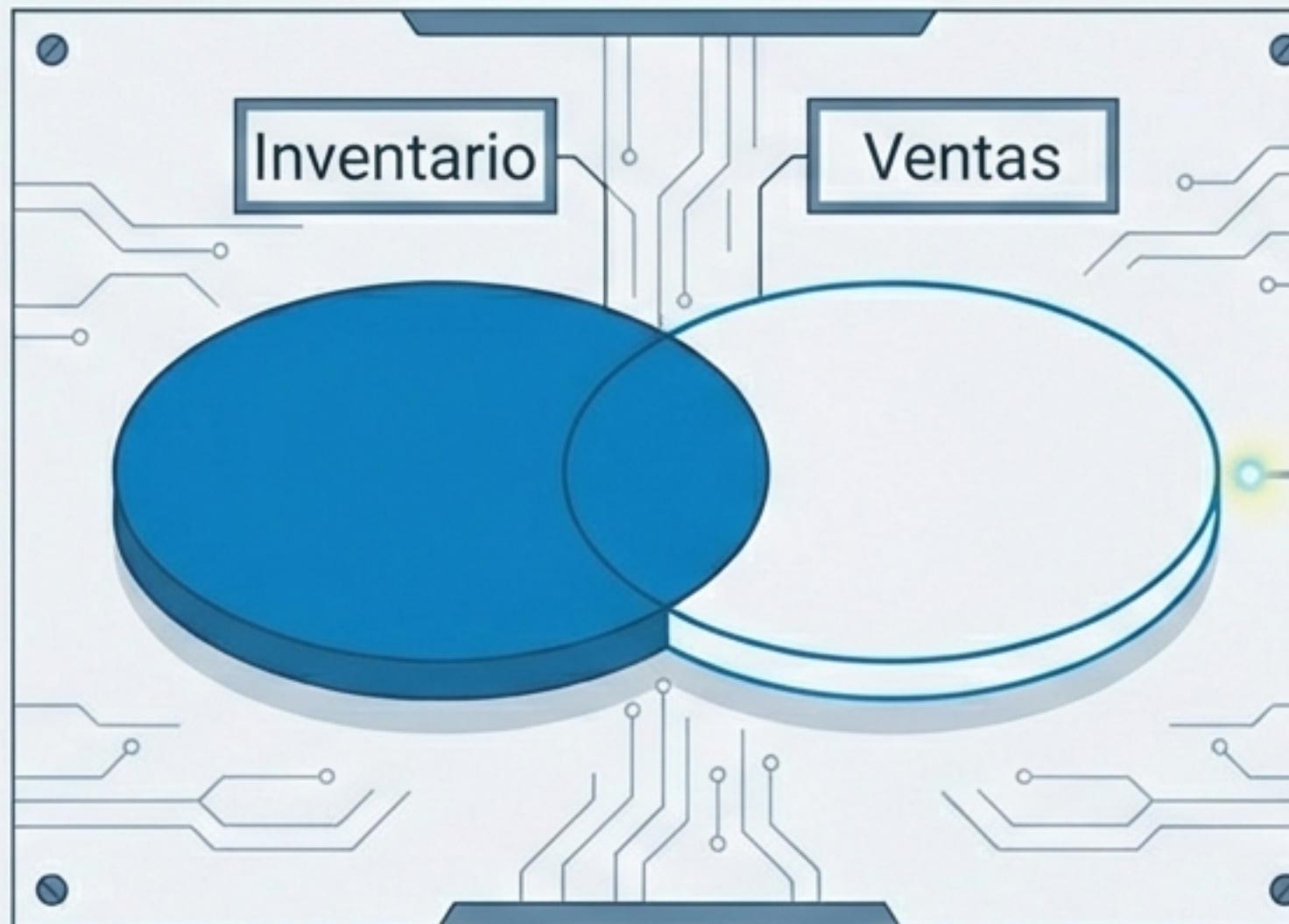
```
1 FROM Ventas  
2 INNER JOIN Inventario  
3 ON Ventas.ID_Bici = Inventario.ID_Bici
```

El **INNER JOIN** es nuestro filtro de calidad. Solo nos muestra las bicicletas que se fabricaron **Y se vendieron**. Si se fabricó pero sigue en bodega, no aparece aquí.

Soldadura Completa: Ahora podemos calcular la Ganancia (Precio - Costo) en una sola vista.



LEFT JOIN: Auditoría de Inventario



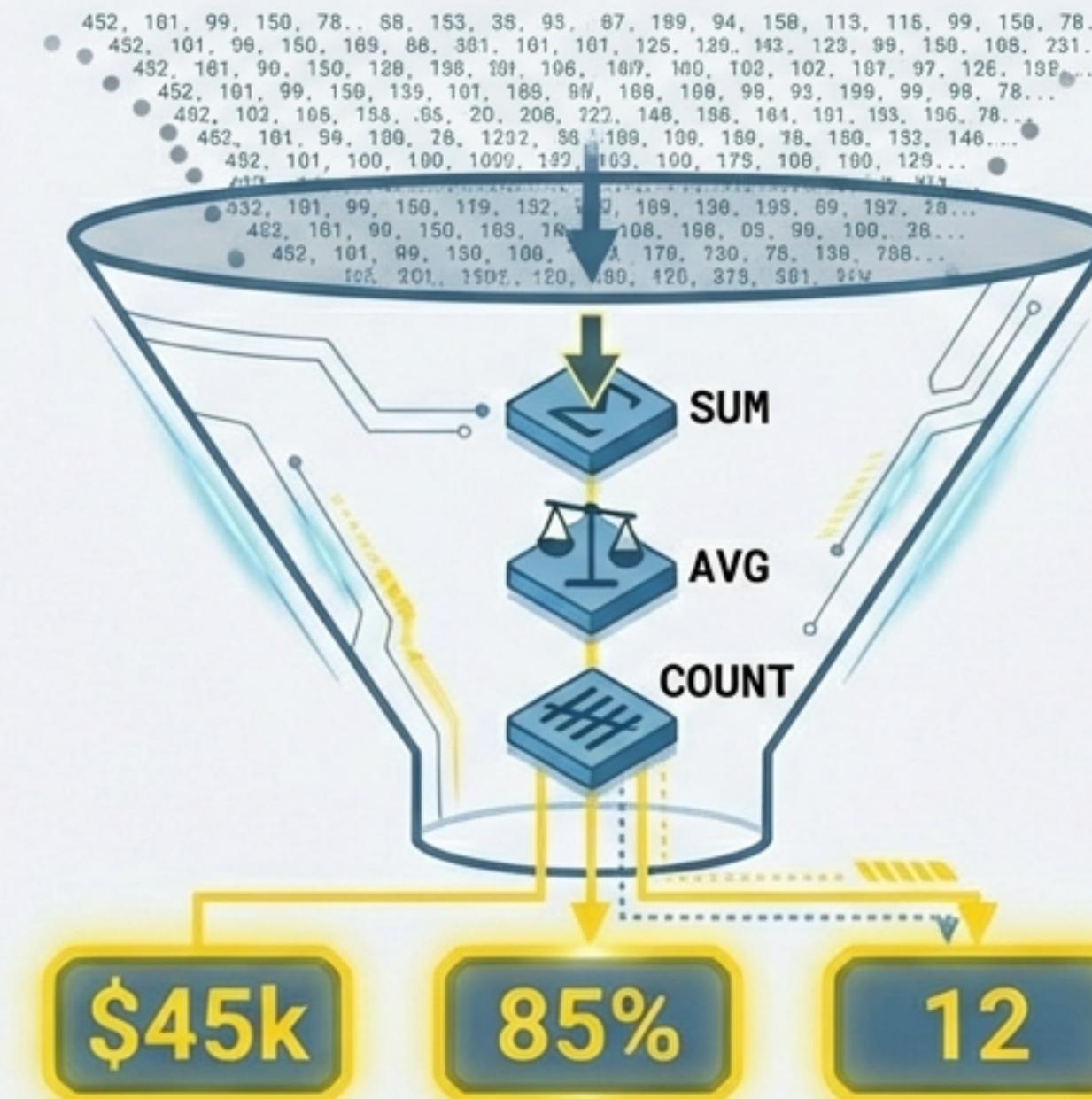
ID_Bici	Modelo	Precio_Venta
101	MTB-29	450
103	Bici Urbana	NULL

El Jefe de Almacén pregunta: ¿Qué bicicletas están juntando polvo?

LEFT JOIN trae todo el inventario. Si no hay venta, la columna de venta aparece como **NULL**.

En manufactura, un **NULL** en ventas significa **capital estancado**.

A la **gerencia** no le interesan 5,000 soldaduras individuales.
Le interesa la **eficiencia total del turno**.



Funciones de Agregación = Reducción de Complejidad.

SUM(): Volumen de Producción

Para presupuestos y control de inventario.

```
SELECT SUM(Costo_Material)  
FROM Produccion_Semana;
```

Total Gasto: \$450,000 USD

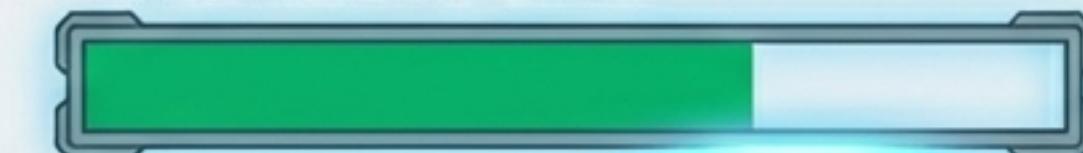


AVG(): Eficiencia del Operario (OEE)

```
SELECT AVG(Tiempo_Ensamble)  
FROM Linea_Montaje;
```



Turno Mañana: 42 min



Turno Noche: 55 min (Alerta)

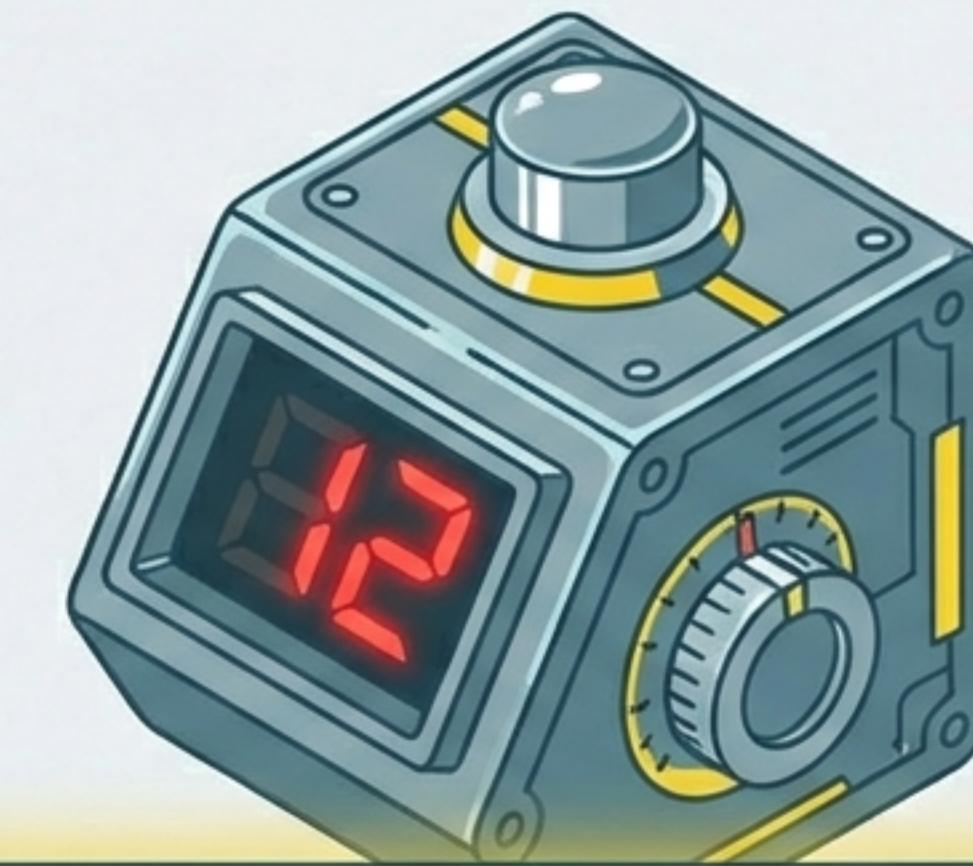


Es vital para medir el **Rendimiento** dentro del cálculo de OEE
(Disponibilidad x Rendimiento x Calidad).

COUNT(): Control de Calidad

Para presupuestos y control de inventario.

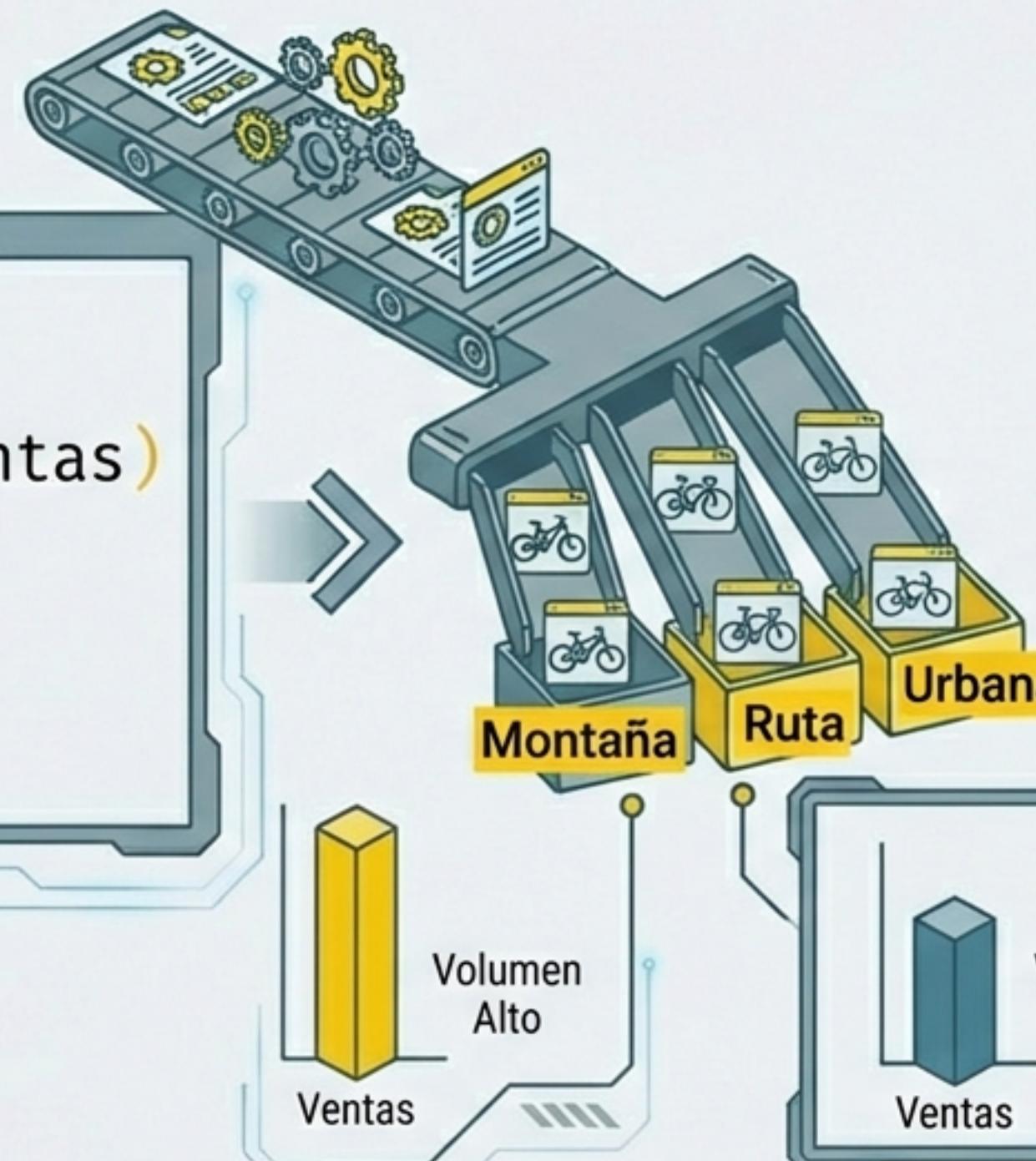
```
SELECT COUNT(*)  
FROM Calidad  
WHERE Estado = 'Falla';
```



No sumamos el valor de la bici,
contamos la ocurrencia del error.
12 fallas en el turno de hoy.

GROUP BY: Segmentación Inteligente

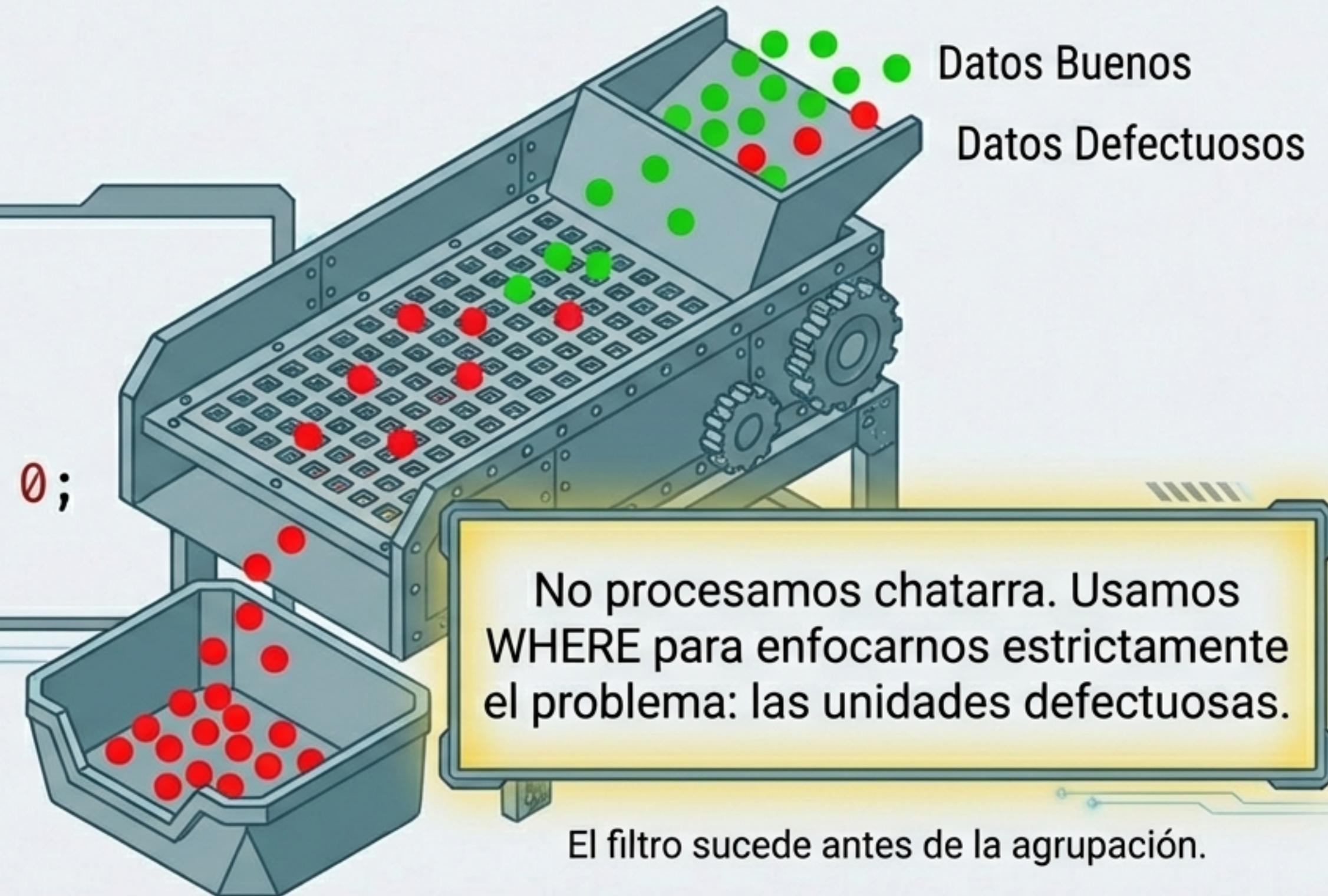
```
SELECT Modelo, SUM(Ventas)  
FROM Ventas_Diarias  
GROUP BY Modelo;
```



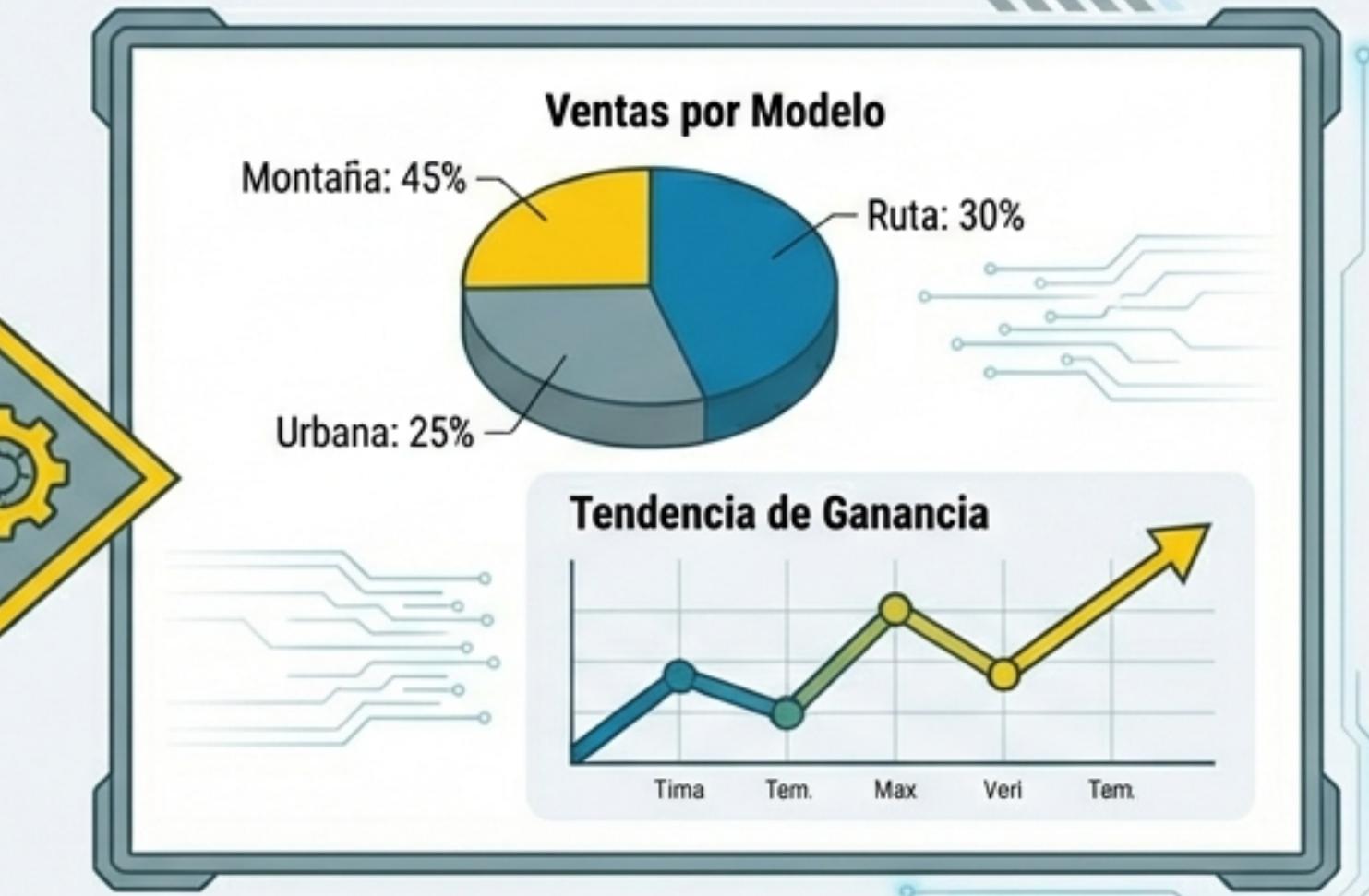
Un promedio global esconde la verdad.
Necesitamos ver la eficiencia por modelo.

WHERE: El Filtro de Materia Prima

```
SELECT ...  
FROM Produccion  
WHERE Defectos > 0;
```

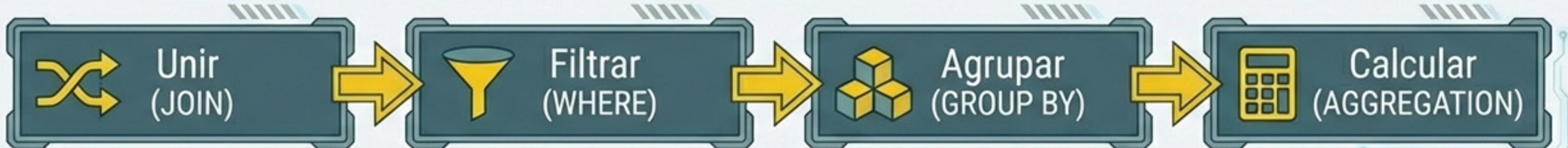


Del Query al Dashboard



Su código SQL es **el motor** que alimenta las decisiones de la gerencia. Sin el Query, el Dashboard está **vacío**.

Resumen: SQL = Rentabilidad



“El analista que domina SQL, domina la planta.
“Datos crudos + SQL = Inteligencia de Negocios.”