Centro Universitario de Occidente

Seminario de Sistemas 2

Ing. Pedro Domingo

Primer Semestre 2025

**“Proyecto Farmacia”**

#4 - 201930699 - Erick Daniel Morales Xicará

# **Índice**

[**Índice 1**](#_vgbnxawjdyg)

[**Introducción 3**](#_tiqewpcj6fgs)

[**Diagnóstico Actual 4**](#_ga4rwnoenzug)

[Conclusión de Diagnóstico 5](#_pdeq5rv3c9rl)

[Importante 5](#_hlww562b8p9x)

[**Fase de Descubrimiento 7**](#_kixbkunvbnze)

[Fuentes Externas 8](#_ch4tdjofjzq9)

[Web Access 9](#_k3wk5pjchfs4)

[**Fase de Preparación 10**](#_mp8b57sb1ptz)

[Operational Data Store (ODS) 10](#_rmexc77kgsn)

[**Fase de Planeación 17**](#_7t1mw11iqeco)

[Herramientas de implementación 19](#_wknnax9f59uc)

[Roles de Recurso Humano 21](#_aozjtomlz6ft)

[**Fase de Construcción 23**](#_bde3ww5ojlce)

[Presentación 23](#_wex9x0h1631m)

[Reportes 24](#_qf1is1qdnxkd)

[Data Marts 28](#_x73xv4kyminc)

[Data Mart #1 : Ventas\_DM 28](#_6ewuabagcfhj)

[Data Mart #2 : Inventario\_DM 30](#_4uchiqcu57m1)

[Data Mart #3 :Pedidos\_DM 32](#_wdcnca5i3q7y)

[Data Mart #4 :Compras\_DM 34](#_dxzgorc6eq36)

[Data Mart #5 :Distribuciones\_DM 36](#_yrzawdfz4k7o)

[Data Mart #6 :Produccion\_DM 38](#_cawlx11y946u)

[Data Mart #7 :RRHH\_DM 39](#_rqc2m0a5o7me)

[Data Mart #8 :Inspeccion\_DM 41](#_91nt1edx3828)

[Data Mart #9 :Marketing\_DM 43](#_1g1mljwou1xp)

[Data Mart #10 :Stock\_Sucursal\_DM 45](#_bgd1zs1hsw58)

[Datos por Tema (Cubos y Modelos) 48](#_dltq1om5yolb)

[1. Cubo OLAP - Ventas 48](#_obflwhwxkgb2)

[2. Cubo OLAP - Inventario 50](#_yu92bl19rozz)

[3. Cubo OLAP - Pedidos 52](#_67bd2r8kh8ws)

[4. Cubo OLAP - Compras 53](#_bb6pyijgnhtc)

[5. Cubo OLAP - Distribuciones 55](#_eocheikux8i)

[6. Cubo OLAP - Producción 56](#_p31a62e8szlr)

[7. Cubo OLAP - RRHH 58](#_hkoso5407wbt)

[8. Cubo OLAP - Devoluciones 59](#_mjfhbt38gd5h)

[9. Cubo OLAP - Marketing 61](#_t7x32wm7jjpk)

[10. Cubo OLAP - Stock por Sucursal 63](#_c6j2lu5hm36y)

[11. Cubo OLAP - Rentabilidad por Producto 64](#_61s70bht91x9)

[12. Cubo OLAP - Tiempos de Respuesta Operativa 65](#_y1x0kzh1ql0e)

[13. Cubo OLAP - Demanda Estacional 66](#_a9dp17f57nn3)

[Indicadores 67](#_wk4pitupid5z)

[**Fase de Comunicación 74**](#_5sms1obk3m1j)

[● Dashboard de Análisis de Ventas 75](#_a5qk8i4ksw0)

[● Dashboard de Pedidos 75](#_d7p36pyt4lv4)

[● Dashboard Inventario y Stock Crítico 76](#_16zl4r9i719c)

[● Dashboard de Producción 76](#_cbs6negobgzx)

[● Dashboard de Inspección y Devoluciones 77](#_ec7reo7hcmf)

[● Dashboard de Recursos Humanos 77](#_8cti5knnmp2k)

[● Dashboard de Marketing 78](#_12qcz1kx8s3m)

[● Dashboard de Stock por Sucursal 79](#_8u8kzwb6n1h)

[● Dashboard de Distribución Logística 79](#_1akjyhwrmq1l)

[● Dashboard de Compras y Proveedores 80](#_yb5pi2m6yi90)

[**Fase de Operacionalización 83**](#_sq9txuq33g1)

[1. Mantenimiento y Actualización de la Solución BI 83](#_ylga2o74ggsz)

[2. Roles de Mantenimiento y Gobierno de Datos 83](#_4rw96owvqe3y)

[3. Estrategia de Escalabilidad 84](#_su9oxt70qtus)

[**Arquitectura general del sistema de BI 85**](#_7ewutp2nabg1)

[**Conclusión 86**](#_xkbdoe1dqfe8)

[**Anexos 88**](#_z09wjdhxhdbq)

[**Bibliografía 90**](#_od9fop70jgbk)

# 

# 

# **Introducción**

El proyecto a desarrollar tiene como objetivo diseñar e implementar un sistema de BI (Inteligencia de negocios) para una cadena de farmacias que se denomina “Enfermito te ves más bonito”, donde se aplican los conceptos del ciclo de vida del datos junto con la arquitectura de BI. La solución propuesta permitirá que se pueda consolidar la información de las diversas fuentes de datos, facilitando así las posteriores toma de decisiones estratégicas pero basadas en datos precisos y actualizados junto con una estructura y datos más maduros para tener la certeza durante la toma de dichas decisiones.

Para poder contemplar estos escenarios y el objetivo, se diseñará una arquitectura de BI que integrará los sistemas operacionales existentes y nuevas soluciones que van a permitir la mejora en la eficiencia de los procesos de esta empresa. Se desarrollará un modelo de datos centralizados en un DW (Data Warehouse), y Data Marts específicos que permiten el análisis de los distintos aspectos de esta empresa, como por ejemplo, inventarios, ventas, distribución y producción de farmacéuticos.

Algunos puntos importantes en la solución de BI, va a ser capaz de responder preguntas como:

* Cuales son los productos de menor rotación y cómo se optimiza su manejo en los o el inventario
* Donde es conveniente ubicar nuevas bodegas para reducir costos logísticos
* Como se puede mejorar el sistema de pedidos de las sucursales para evitar retrasos y los desabastecimientos
* Cuales son los medicamentos que deben fabricarse en los laboratorios propios y en qué cantidades deben ser fabricados, según la demanda y la disponibilidad de la materia prima

A través de este sistema de Business Intelligence, la empresa “Enfermito te ves más bonito” podrá optimizar sus procesos operacionales, reducir costos, mejorar la disponibilidad de productos y sobre todo fortalecer su competitividad en el mercado farmacéutico nacional.

# **Diagnóstico Actual**

Visión general

La empresa “Enfermito te ves más bonito” es una cadena de farmacias con cobertura a nivel nacional que con el paso del tiempo se ha podido diversificar su operación a través de importar y fabricar medicamentos propios. A pesar del crecimiento, expansión y potencial, como cualquier empresa enfrenta problemas derivados de sus datos, de no tener una madurez en sus datos y comenzando por la descentralización de datos, ineficiencias en temas de logística, falta de coordinación entres sus sistemas y a su vez también el bajo uso de sistemas para ciertos procesos que podrían mejorar los mismos.

Para poder obtener un diagnóstico, seguidamente de una abstracción de información, encontramos puntos claves que los cuales nos servirán como objetivos / metas para poder ver los cambios una vez hayamos implementado nuestra arquitectura de BI en esta empresa:

* Fragmentación de sistemas y falta de integración de datos
  + Cada sucursal opera de una manera independiente del sistema de ventas, lo que realmente nos impide tener una visión a nivel nacional de estos datos.
  + Dado lo anterior se puede ver que realmente no existe una plataforma centralizada que consolide todos los datos operativos, de inventario, pedidos, compras, entre otros.
  + Esta fragmentación de data, limita realmente a tomar decisiones basadas en los mismos datos y en tiempo real, dado que de acuerdo a cada sucursal se tomarán distintas decisiones pero no serán para beneficio general que es lo que se propone con la solución de BI.
* Problemas de Inventario
  + El inventario tiene un crecimiento de los costos esto derivado del almacenamiento prolongado de los productos que no tienen una alta rotación o venta
  + No se puede hacer un análisis de las gestiones de los niveles de stock
  + Se menciona que la bodega central está saturada y la empresa no tiene esos criterios en sus datos para decidir donde realizar una nueva bodega
* Ineficiencia en los pedidos
  + Los pedidos no tienen criterios o políticas establecidas para hacerlos lo que genera:
    - Duplicidad de pedidos
    - Pedidos Fragmentados
    - Retrasos por acumulacion de pedidos
* Desabastecimiento
  + Las sucursales tienen un desabastecimiento en temporadas clave, derivado de un mal estudio de mercado
  + Esto hace que exista una pérdida de los clientes dada la competencia y por no saber en qué temporada se requieren ciertos medicamentos.
* Subutilización de los Laboratorios
  + Se cuenta con 3 laboratorios en distintas regiones del país pero no están siendo productivos de la misma manera los 3.
  + Todo surge a raíz que no se sabe que medicamentos se necesita, que cantidad por región, cuales hay disponibles y cual es la capacidad instalada de cada laboratorio
* Saturación Bodega Central
  + La bodega central enfrenta un desorden operativo al querer manejar inspecciones, devoluciones y despachos en simultáneo
  + No existe información sobre tiempos de respuesta a reclamos, causas de devoluciones

## **Conclusión de Diagnóstico**

La empresa presenta una fuerte desconexión entre sus sistemas operativos y carece de una plataforma unificada para el análisis estratégico. Esto genera ineficiencias logísticas, altos costos operativos y pérdida de competitividad en el mercado.

Por lo cual se necesita de una solución de BI para que la empresa pueda madurar en sus datos y asi podes reparar todas esas brechas que no tienen contempladas y así poder enseñarles la importancia de manejar bien sus datos, y que los datos nos brindan información útil para la toma de decisiones basadas en los mismos datos y su entendimiento.

## **Importante**

Aquí nos encontramos en un punto importante donde para poder realizar un mejor análisis de BI y poder entregar una solución óptima de BI se necesita que la empresa contenga los siguientes sistemas, los cuales ayudaran a poder mejorar la operatividad y así tener en cuenta un mejor panorama y así poder tomar mejores decisiones que es lo que busca la implementación de BI que se quiere realizar, por lo cual se RECOMIENDA implementar los siguientes sistemas:

1. ***Sistema de Recursos Humanos***

Este sistema, permite entender la disponibilidad y eficiencia del personal en las áreas críticas como lo es en distribución, bodega, laboratorios y permite tener nombre, puesto, área de trabajo, horarios, capacidad técnica o experiencia, relación con otras tareas y otros datos que se detallarán más adelante.

1. ***Sistema de Devoluciones e Inspección de Calidad***

Este sistema registra los procesos de revisión que ocurre cuando los medicamentos llegan a la bodega central, entonces pueden haber devoluciones por errores en el pedido, problemas de calidad, empaques defectuosos o inclumplimientos en las especificaciones de los productos y también este sistema contempla datos como la fecha de ingreso de bodega, el resultado de la inspección, motivo de devolución, proveedor, cantidad de lotes aceptados y cantidad de lotes no aceptados, cantidad por lotes, empleado encargado, estado del reclamo, observaciones entre otros datos que se detallarán más adelante.

1. ***Sistema de Promociones y Marketing***

Este sistema registra los procesos internos de la empresa para implementar campañas de promociones de productos y estrategias de marketing orientadas a mejorar las ventas o el volumen de las mismas, atraer nuevos clientes y poder promocionar los medicamentos propios. Estas actividades son gestionadas a través de un sistema que registra campos como fecha de inicio y fin de la campa;a, productos incluidos de promoción, canal de promoción, algun cupon exclusivo por la campana, ventas asociadas a las mismas por medio de la aplicación del cupón, el motivo de las mismas, y entre otras que se detallarán mejor más adelante.

1. ***Sistema de Control de Stock por sucursal***

Este sistema se pone como un sistema general pero se plantea que cada una de las sucursales de dicha empresa, y más adelante se hará por cada sucursal, el cual podrá ser un excel, hojas de papel o algo el cual se determinará en su momento para un mejor análisis, contemplando que tiene datos como el nombre del producto, cantidad disponible en tiendas, niveles minimos y maximos a tener en stock por producto, fecha de entrada, fecha salida, persona que realizó el movimiento, y entre otros más campos que se detallaran mas adelante dependiendo de las distintas sucursales.

# **Fase de Descubrimiento**

Como se mencionó anteriormente, para aportar la integración de múltiples fuentes de datos que provendrán o provienen de los distintos sistemas operacionales existentes en dicha empresa, así como que otras fuentes pueden aportarnos valor a este análisis, comenzaremos a identificar los mismos:

1. ***Sistema de Ventas por sucursal***

Cada sucursal de la cadena de farmacias cuenta con su propio sistema de ventas desplegado en la nube. Todas estas instancias o sistemas son independientes por lo cual nos dice que actualmente no hay una vista centralizada de estos datos a nivel nacional. Esta fuente contendrá datos de sitios como productos vendidos, cantidades, precios, fecha y hora de la venta, el empleado que lo vendió, forma de pago y muchos otros campos que definiremos más adelante.

1. ***Sistema de Pedidos de sucursales***

Este sistema es el que gestiona todos los pedidos que las sucursales hacen hacia la bodega central, donde actualmente no tienen un criterio estandarizado para realizar los pedidos bajo los criterios que cada uno considere, generando así las duplicaciones o solicitudes incompletas. Este sistema registra los productos solicitados, cantidades, fechas, estado del pedido y otros campos que se detallarán más adelante.

1. ***Sistema de compras***

Este sistema es el encargado de gestionar las adquisiciones de medicamentos con farmaceitura internacionales, el cual permite llevar el control sobre las cotizaciones, ordenes de compra, tiempos de entrega, condiciones de garantía, devoluciones y otros campos que se detallarán más adelante.

1. ***Sistema de Inventario / Bodega Central***

Este sistema es el encargado de registrar todos los movimientos del inventario, ingreso de medicamentos, inspecciones de los mismos, devoluciones, almacenamiento, entrega a las sucursales y más datos. También información sobre la ubicación de cada una de las estanterías en su bodega, fecha de vencimiento del lote y los lotes.

1. ***Sistem de Distribución y Logística***

Este sistema controla el proceso de envío desde la bodega central hacia las sucursales, datos como disponibilidad de vehículos, marca, a;o, millas por galón, rutas, regiones cantidad de pedido a distribuir, entre otros datos que se detallarán más adelante.

1. ***Sistema de Producción***

Este sistema registra todos los procesos internos de la producción de los tres laboratorios de la empresa, la cual están ubicados en el oriente, centro y occidente del país. Contiene datos como medicamentos fabricados, número de lote, cantidad, fechas, que maquinaria uno, disponibilidad de insumos, empleados, turnos de producción entre otros elementos que se detallarán más adelante.

## **Fuentes Externas**

1. ***Ministerio de Salud Pública (MSPAS)***

Este es una entidad externa al negocio, la cual esta empresa no tiene acceso a estos sistemas por lo cual son datos estadísticos y abiertos pero no se pueden obtener mediante una API que exponga esta empresa lo cual puede servir para mejorar el análisis de mercado para poder enriquecer el análisis de BI. La data es probable que esté en csv o excel.

1. ***Sistema de Información Geográfica***

Este es un sistema que contempla poder realizar una integración de una fuente de datos geoespaciales para poder almacenar coordenadas y ubicaciones físicas de las sucursales, bodegas y los laboratorios de esta empresa. Esta información permite poder ver puntos críticos del negocio por ejemplo en momentos en que se puede detectar bloqueos y eso servirá para poder analizar los envíos o así, y se encontrará datos como el nombre del punto, ubicación, dirección física, departamento, municipio, coordenadas de punto, que instalación o tipo es, que región pertenece, zona de influencia.

1. ***Instituto Nacional de Estadística (INE)***

Esta es una fuente de información de datos abiertos, que es una entidad gubernamental de Guatemala, y el aporte a nuestro análisis son datos sociodemográficos detallada por región y lo que hará que el análisis de ventas por decir un ejemplo, puede verse influenciado por qué tantas personas hay en la región de donde estan las sucursales, para poder ver expansiones de sucursales de la empresa y vamos a encontrar datos como la región, total de población, densidad poblacional, promedio de ingreso mensual, nivel educativo promedio, nivel de desempleo, clasificación económica por departamento y proyección de crecimiento poblacional y otros mas datos que se detallarán más adelante.

1. ***Sistema de información climatológica***

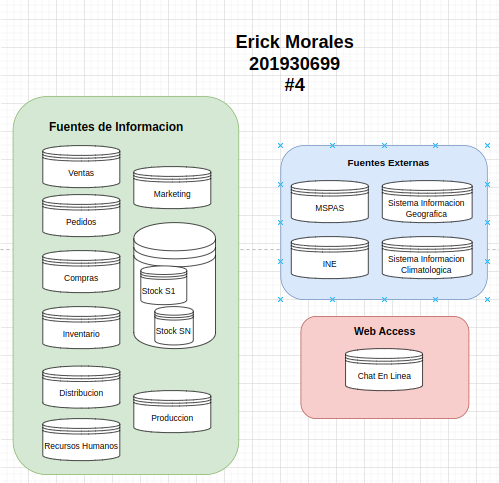
Esta también es una fuente externa que nos dará datos meteorológicos obtenidos de servicios públicos de clima, se enfoca en poder recopilar información sobre condiciones climáticas y así ver cómo influye en la salud de la población y que medicamentos o productos tienden a tener más demanda por lo mismo.

## **Web Access**

1. ***Chat En Línea***

Se cuenta con un Web Access en el cual la empresa cuenta con canales de mensajería directa como WhatsApp mediante los cuales los clientes pueden realizar consultas sobre productos, disponibilidad, horarios de atención, promociones, precio o ubicación de sucursales y lo cual estos datos son importantes para mejorar el análisis y comportamiento de los clientes, donde se encuentran campos como datos del contacto, motivó , producto(s) mencionados, sucursal más cercana, tiempo de contacto, estado del caso y otros elementos que se detallaran mejor más adelante.

Por lo anteriormente mencionado se obtiene el inicio de nuestro diagrama de BI para ir entendiendo como esta la arquitectura de BI de este caso:



Ahora bien, luego que ya definimos las fuentes de información que nos ofrece la empresa “Enfermito te ves más bonito” pasamos a la parte de cómo nosotros debemos de tratar estos datos para así poder tener la centralización de nuestros datos lo cual permitirá crear nuestro DataWarehouse del cual se podrá extraer información que ya se tiene o se puede obtener con nuestros ODS y a su vez otro tipo de información a partir de la centralización de nuestros datos.

El DataWarehouse consiste en esa gran base de datos centralizada donde integraremos, organizaremos y almacenaremos todos los datos de las fuentes de datos y a su vez generamos nuevo DataMart y Cubos (Indicadores) basados en los mismos datos.

# **Fase de Preparación**

## Operational Data Store (ODS)

A continuación se detallan los datos que se encuentran en cada uno de los ods (o lo que se espera encontrar que el sistema operacional)

* ODS - Sistema de Ventas por sucursal
  + Region
    - id\_region
    - nombre\_region
  + Departamento
    - id\_departamento
    - nombre departamento
    - id\_region (FK)
  + Municipio
    - id\_municipio
    - nombre\_municipio
    - id\_departamento (FK)
  + Sucursal
    - id\_sucursal
    - nombre\_sucursal
    - ubicacion
    - id\_municipio (FK)
  + Jornada
    - id\_jornada
    - tipo\_jornada
  + Empleado
    - id\_empleado
    - nombre
    - cargo
    - dpi
    - edad
    - genero
    - fecha\_ingreso
    - estado\_empleado
    - id\_jornada(FK )
    - id\_sucursal (FK)
  + Categoria
    - id\_categoria
    - nombre\_categoria
  + Producto
    - id\_producto
    - nombre
    - id\_categoria(FK)
    - marca
    - presentacion
    - precio\_base
    - es\_propio
    - unidad\_de\_medida
  + Cliente
    - id\_cliente
    - nit
    - nombre
    - tipo\_cliente
    - genero
    - edad
    - numero\_de\_compras
    - frecuencia\_compra
  + Venta
    - id\_venta
    - fecha\_venta
    - hora\_venta
    - id\_sucursal (FK)
    - id\_empleado (FK)
    - id\_cliente (FK)
    - canal\_venta
    - total\_bruto
    - descuento\_total
    - total\_neto
    - forma\_pago
    - estado\_pago
  + DetalleVenta
    - id\_detalle
    - id\_venta (FK)
    - id\_producto (FK)
    - cantidad\_vendida
    - precio\_unitario
    - margen\_unitario
    - subtotal
    - descuento\_producto
    - total\_línea
* ODS - Sistema de Pedidos de Sucursales
  + Pedido
    - id\_pedido
    - id\_sucursal(FK)
    - fecha\_solicitud
    - fecha\_estimada\_entrega
    - urgencia
    - estado
    - observaciones
  + Detalle Pedido
    - id\_detalle\_pedido
    - id\_pedido (FK)
    - id\_producto (FK)
    - cantidad\_solicitada
    - cantidad\_entregada
* ODS - Sistema de Compras
  + Proveedor
    - id\_proveedor
    - nombre
    - pais
    - canal\_contacto
  + Compra
    - id\_compra
    - id\_proveedor (FK)
    - fecha\_cotizacion
    - fecha\_orden
    - fecha\_estimada\_entrega
    - fecha\_real\_entrega
    - estado
  + Detalle Compra
    - id\_detalle\_compra
    - id\_compra (FK)
    - id\_producto
    - cantidad
    - precio\_unitario
    - condiciones
* ODS - Inventario General
  + Ubicacion Bodega
    - id\_ubicacion
    - descripcion
  + Lote
    - id\_lote
    - id\_producto (FK)
    - id\_proveedor (FK)
    - fecha\_ingreso
    - fecha\_vencimiento
    - cantidad\_total
    - cantidad\_disponible
    - id\_ubicacion (FK)
* ODS - Distribucion y Logistica
  + Vehículo
    - id\_vehiculo
    - marca
    - modelo
    - anio
    - capacidad\_kg
    - rendimiento\_kmgl\_carretera
    - rendimiento\_kmgl\_ciudad
  + Piloto
    - id\_piloto
    - nombre
    - licencia
    - telefono
    - correo
    - fecha\_ingreso
    - estado
  + Envío
    - id\_envio
    - id\_pedido (FK)
    - id\_vehiculo (FK)
    - id\_piloto (FK)
    - fecha\_salida
    - fecha\_entrega
    - ruta
    - estado
* ODS - Producción
  + Laboratorio
    - id\_laboratorio
    - nombre
    - id\_region (FK)
    - ubicación
  + Producción
    - id\_produccion
    - id\_producto (FK)
    - id\_laboratorio (FK)
    - fecha\_inicio
    - fecha\_fin
    - cantidad\_producida
    - id\_empleado (FK)
* ODS - Recursos Humanos
  + Asistencia
    - id\_asistencia
    - id\_empleado (FK)
    - fecha
    - turno
    - estado
  + Gestion
    - id\_gestion
    - id\_empleado (FK)
    - fecha
    - tipo\_gestion
    - comentario
* ODS - Devoluciones E Inspección
  + Motivo Dev
    - id\_motivo
    - descripcion
  + Devolución
    - id\_devolucion
    - id\_lote (FK)
    - id\_producto (FK)
    - id\_proveedor (FK)
    - fecha\_inspeccion
    - resultado
    - id\_motivo (FK)
    - cantidad\_aceptada
    - cantidad\_rechazada
    - id\_empleado\_inspector (FK)
    - estado\_reclamo
  + Detalle Inspeccion
    - id\_detalle
    - id\_devolucion (FK)
    - criterio
    - resultado\_criterio
    - observaciones
* ODS - Promociones y Marketing
  + Promoción
    - id\_promocion
    - nombre
    - canal
    - fecha\_inicio
    - fecha\_fin
    - motivo
  + Detalle Promoción
    - id\_detalle\_promocion
    - id\_promocion (FK)
    - id\_producto (FK)
    - descuento
  + Detalle Actividad
    - id\_actividad
    - nombre\_actividad
    - id\_promocion (FK)
    - presupuesto
    - asistencia
    - compras\_asociadas
  + FeedBack
    - id\_feedback
    - nombre
    - edad
    - comentario
    - id\_region (FK)
* ODS - Control de Stock Sucursal
  + Stock Sucursal
    - id\_stock
    - id\_sucursal
    - id\_producto
    - cantidad\_actual
    - minimo
    - maximo
    - fecha\_ultima\_actualizacion
  + Movimiento Stock
    - id\_movimiento
    - id\_sucursal (FK)
    - id\_producto (FK)
    - tipo\_movimiento
    - cantidad
    - fecha\_movimiento
    - motivo

# **Fase de Planeación**

La solución a partir de lo visto en clase consistirá en abordar la problemática actual de la cadena de farmacias “Enfermito te ves más bonito”, entonces vamos a proponer un dise;o e implementación teórica de una solución de Inteligencia de Negocios (BI) que permite integrar, transformar y analizar los datos de la empresa para mejorar sus procesos logísticos, de ventas, producción y toma de las decisiones estratégicas basado en los datos que tiene, por lo cual el primer paso será poder descubrir estos sistemas.

Seguidamente de descubrir estos datos procederemos a hacer el proceso de los mismos a través de una capa ETL (Extracción, Transformación y Carga), donde se realizará la limpieza, estandarización e integración de los mismos para poder tener un Data Warehouse el cual nos permitirá tener centralizados estos datos. Desde este punto donde ya tendremos los datos almacenados, permitirá generar los Data Marts (como por ejemplo de Ventas, Logistica y Produccion) que los cuales servirán para crear los indicadores clave, y así mismo poder facilitar el análisis mediante los cubos y los dashboards interactivos.

Luego de poner “en marcha” nuestra solución propuesta se estima que la misma permitirá observar lo siguiente:

* Poder identificar productos de baja rotación o movimiento y optimizar el inventario
* Poder analizar la distribución geográfica de las sucursales para evaluar la ubicación óptima de posibles nuevas bodegas
* Poder ordenar y sistematizar aquellos pedidos de las sucursales, para poder optimizar , evitar duplicaciones y retrasos de los mismos
* Poder prever situaciones de desabastecimiento basado en patrones históricos y estacionales
* Poder optimizar el uso de los laboratorios, mediante una planificación en la producción en función de las demandas reales del mercado, la capacidad instalada y la disponibilidad de la materia prima
* Poder además de generar reportes transaccionales, permitir generar reportes ejecutivos y visualizaciones para poder respaldar la toma de decisiones estratégicas

Seguidamente la solución seguirá el ciclo de vida del dato, aplicando cada una de sus fases y se documentada con diagramas, modelos relaciones, álgebra relacional y en su defecto scripts SQL para sustentar el proceso de análisis que se realizará.

También en la fase de planeación debemos identificar las herramientas que usaremos en el transcurso de la implementación de nuestra arquitectura de Business Intelligence.

## **Herramientas de implementación**

**PostgreSQL**

Este es un motor de base de datos para DW y ODS que nos permitirá ciertas funcionalidades para diferentes etapas del ciclo del dato (Fase de preparación, fase de construcción, fase de operacionalización) , para:

* Crear tablas normalizadas (ODS)
* Diseñar el esquema estrella (DW)
* Crear Data Mart
* Ejecutar consultas SQL complejas que permitirán realizar los indicadores y reportes
* Almacenar datos históricos

**Apache Nifi**

Esta es una herramienta visual para la integración de datos (por medio de ETL), y esta nos ayudará en la fase de planeación y fase de construcción, para:

* Diseñar los flujos de datos por medio de pipelines
* Automatizar extracción de datos en medio de archivos ya sea Excel, CSV, Json y APIs
* Transformar y enturas datos con lógica de condicional
* Carga los datos procesados directamente en MariaDB (ODS o DW)

**Python + Pandas**

ETL programático / script personalizados para aplicarlo en la fase de preparación y fase de construcción para:

* Análisis exploratorio de datos
* Limpieza de las columnas
* Limpieza de registros
* Cálculo de métricas auxiliares
* Integración con API’s o alguna otra fuente externa
* Simulaciones de datos si se necesita poblar la BD con casos de prueba

**Power BI**

Esta herramienta se utiliza para poder permitir la visualización de dashboards e indicadores los cuales se usarán para la fase de comunicación y fase de operacionalización, para:

* Conectar al DW o a los Data Marts
* Crear paneles por área: Ventas, Inventario, Producción, Marketing y otras áreas que se contarán
* Mostrar KPI’s gráficos, tablas dinámicas
* Filtrar por región, sucursales, productos, y otros campos

**Draw IO**

Esta herramienta se utiliza para permitir poder crear diagramas para la documentación visual y nos servirá para la fase de descubrimiento, fase de planeación y la documentación general.

* Modelos Entidad-Relación (ER)
* Diagramas del ciclo de vida del dato
* Arquitectura general del sistema BI
* Flujo del proceso ETL
* Representacion de cubos o Data Marts

A su vez, tenemos que realizar la planeación del distribución del tiempo de trabajo por las fases del ciclo de vida del dato y la arquitectura de BI

| **Fase** | **Tiempo Estimado %** | **Justificación Técnica** |
| --- | --- | --- |
| Descubrimiento de fuentes | 20% | Esta fase implica identificar todos los sistemas operacionales y cuales son los faltantes para mejorar el análisis a su vez poder comprender su lógica, procesos, deficiencias y estructura de sus datos. El cual requerirá de un pensamiento sistémico, entrevistas (teóricas) y documentación |
| Modelo y Normalización (ODS) | 25% | Es clave esta fase, para poder garantizar la calidad de los datos del DW. También incluye el diseño de las tablas para cada ODS que se generó y todo lo que conlleva como las relaciones, llaves foráneas, evitar redundancia y aplicar los principios de normalización. |
| Construcción del DW y Data Marts | 30% | Este fase representa la parte más compleja cuando hacemos el modelo, porque se diseña modelo dimensional, también integrar los ODS y definir dimensiones , también diseñar los data marts por cada área |
| Visualización e Indicadore | 15% | Aquí es donde vemos los indicadores de varias dimensiones para poder construir los dashboards interactivos y en esta fase se requiere tanto análisis como criterios para la presentación |
| Documentación y Presentación | 10% | Redacción de todos los informes finales, donde se elabora diagramas (de flujo de ER, Arquitectura) y explicación de toma de decisiones de diseño y preparación para exponer el proyecto |

También para poder realizar un trabajo de manera ordenada y que se tenga el control respecto a las necesidades se recomiendan los siguientes recursos humanos con las funcionalidad que desarrollará cada uno

## **Roles de Recurso Humano**

Aunque el proyecto es desarrollado de forma individual, un sistema BI a la hora de implementar esta solución de BI, se debe involucrar un equipo para poder colaborar distintas fases del ciclo de vida del dato, siento los principales:

**Científico de Datos**

Funciones

* Analizar los datos una vez que ya estén integrados
* Desarrollar los modelos analiticos
* Diseñar los indicadores de negocio
* Participar en las interpretaciones clave

Fases de intervención

* Análisis
* Planeación
* Comunicación

**Ingenierio de Datos**

Funciones

* Diseñar y desarrollar los procesos necesarios de ETL
* Automatizar los flujos de datos desde las distintas fuentes
* Optimizar consultas y la estructura de los datos para mayor eficiencia

Fases de Intervención

* Preparación
* Construcción
* Operacionalización

**Analista de Negocios**

Funciones

* Interactuar con los usuarios clave del negocio como la dirección de la empresa
* Traducir las necesidades en requerimientos de los datos
* Proponer las visualizaciones y la estructura que tendrán los reportes

Fases de Intervención

* Descubrimiento
* Planeación
* Comunicación

**Arquitecto de Datos**

Funciones

* Diseñar la arquitectura general del sistema de BI (tomando en cuenta ODS -> DW -> Data Marts)
* Definir los estándares que debe cumplir el modelado de los datos
* Asegurar que la solución sea escalable y este estructurada de manera óptima

Fases de Intervención

* Descubrimiento
* Preparación
* Construcción

**Administrador**

Funciones

* Encargado de gestionar las plataformas de BI para visualización (Power BI, etc)
* Controlar el acceso a los dashboards e indicadores según los perfiles
* Garantizar disponibilidad y las actualizaciones

Fases de Intervención

* Comunicación
* Operacionalización

Algunas relaciones que se deben tomar en cuenta cuando hablamos de roles y en si son:

* El **Arquitecto de Datos** establece las bases del diseño estructural.
* El **Ingeniero de Datos** implementa y automatiza esos diseños.
* El **Científico de Datos** y el **Analista de Negocio** trabajan con los datos ya procesados para **extraer valor**.
* El **Administrador de BI** entrega ese valor a los usuarios finales mediante dashboards, asegurando seguridad y acceso eficiente.

# **Fase de Construcción**

A continuación se presentan diagramas que representan la arquitectura física de la solución de BI para la empresa “Enfermito te ves más bonito”. En el diagrama se identifican las diferentes fuentes de información internas y externas, junto con el acceso web de los clientes que irán alimentando el Data WareHouse mediante procesos de ETL

Tomando en cuenta como se mencionó antes, vamos a explicar los ODS

## **Presentación**

* ODS - Sistema de Ventas por Sucursal
* ODS - Sistema de Pedidos de Sucursales
* ODS - Sistema de Compras
* ODS - Inventario / Bodega Central
* ODS - Distribución y Logística
* ODS - Producción
* ODS - Recursos Humanos
* ODS - Devoluciones e Inspección
* ODS - Promociones y Marketing
* Control de Stock por Sucursal

## **Reportes**

Presentaremos algunos de los reportes que se podrá obtener dado los diferentes ODS para ver reportajes operacionales.

**ODS - Sistema de Ventas por Sucursal**

* Reporte de ventas diarias por sucursal
* Productos más vendidos por mes
* Ventas por canal (físico, en línea, etc.)
* Comparativo de ventas por género de cliente  
  Ranking de empleados por volumen de venta

**ODS - Sistema de Pedidos de Sucursales**

* Estado actual de pedidos por sucursal
* Productos más pedidos por sucursal
* Pedidos urgentes pendientes de entrega
* Tiempo promedio entre pedido y entrega estimada

**ODS - Sistema de Compras**

* Compras realizadas por proveedor
* Comparativo de precios por producto y proveedor
* Proveedores con más devoluciones
* Órdenes de compra pendientes de entrega

**ODS - Inventario / Bodega Central**

* Lotes próximos a vencerse
* Stock disponible por producto
* Productos con bajo nivel de inventario
* Trazabilidad de lote (ingresó, inspección, salida)

**ODS - Distribución y Logística**

* Estado de envíos en tránsito
* Pedidos entregados por piloto
* Vehículos con más uso o fallas reportadas
* Tiempos promedio de entrega por región

**ODS - Producción**

* Producción por laboratorio y producto
* Turnos más productivos
* Tiempo promedio de fabricación por medicamento
* Capacidad utilizada vs. instalada por laboratorio

**ODS - Recursos Humanos**

* Asistencia por jornada
* Turnos con mayor ausentismo
* Historial de gestiones por empleado
* Empleados con mayor rotación

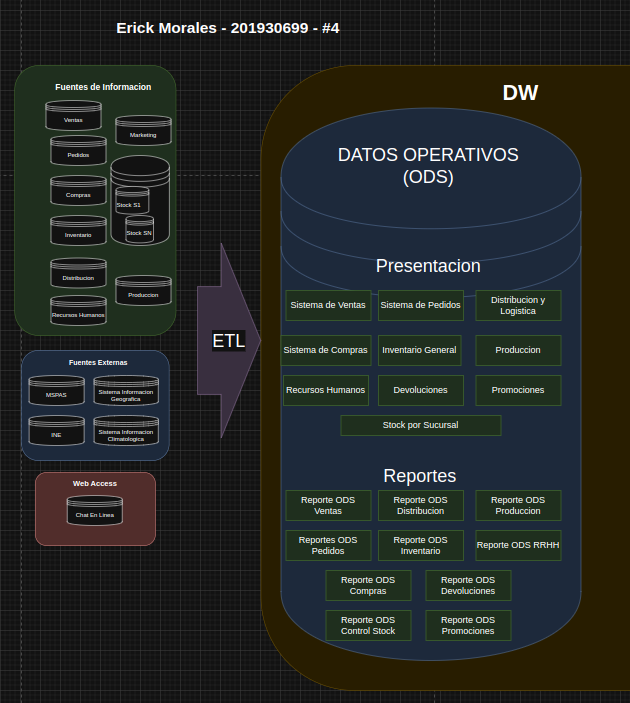
**ODS - Devoluciones e Inspección**

* Tasa de devolución por proveedor
* Motivos más comunes de devolución
* Empleados inspectores con más rechazos  
  Tiempos promedio de respuesta a reclamos

**ODS - Promociones y Marketing**

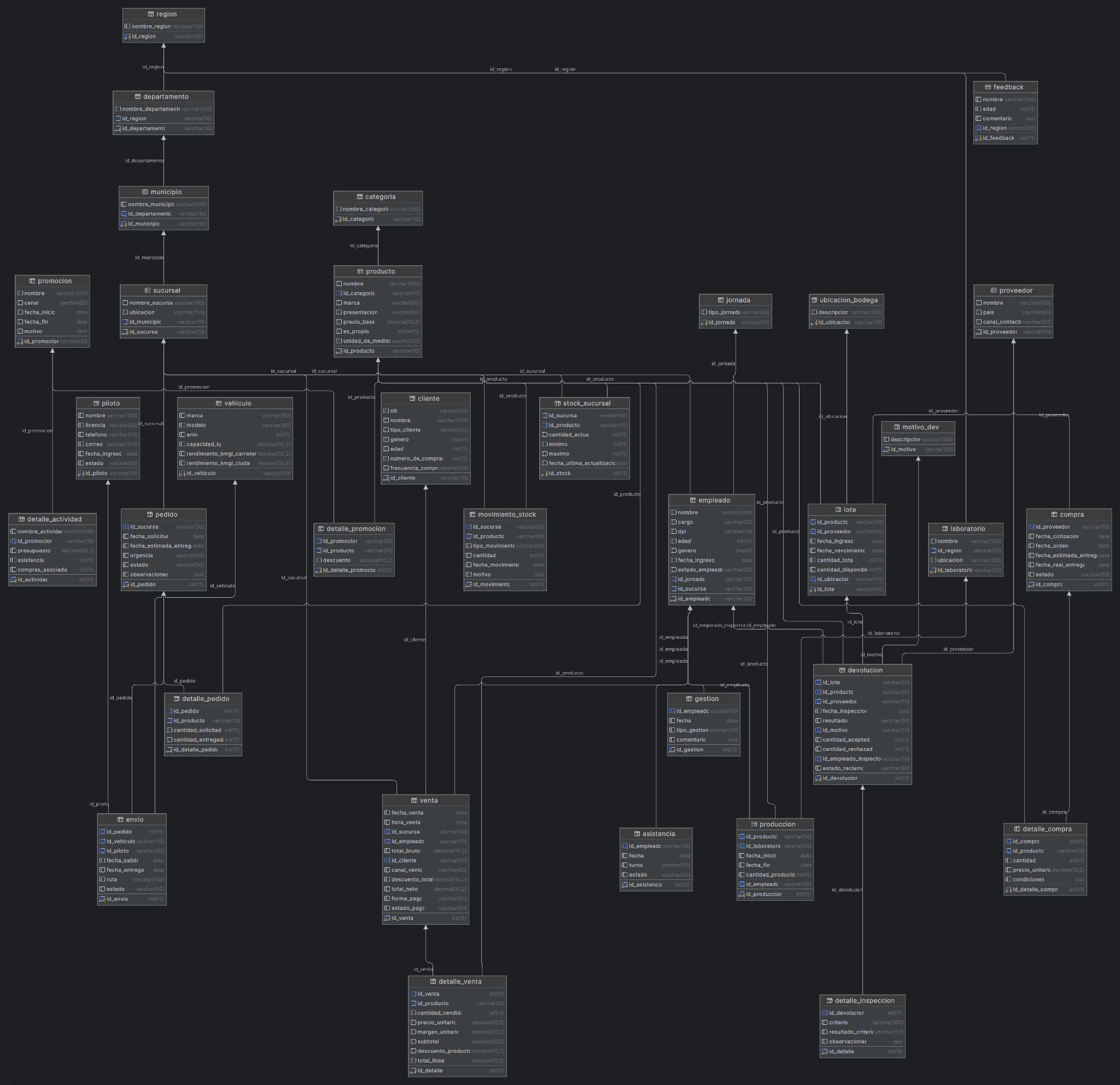
* Promociones activas
* Ventas asociadas a promociones
* Descuento promedio aplicado por producto
* Feedback recibido por promoción

**ODS - Control de Stock por Sucursal**

* Stock actual por producto y sucursal
* Productos con sobrestock o desabastecimiento
* Movimientos de stock recientes  
  Productos más movidos en tienda

Hasta el momento tendríamos lo que se contempla en el diagrama, tomando en cuenta los siguientes documentos que se encuentran en la carpeta siendo esto:

* **script\_farmacia\_bi.sql:** Este es el script DDL de la base de datos
* **ods\_reports\_farmacia.sql:** Este es el script de las consultas de la base de datos. Se realizó dado que aunque no tenemos los datos, si tenemos cómo es que queremos la información y por medio del proceso de ETL vamos a cargar los datos y guardarlos como se debe.
* **carga\_datos.sql:** Este script es para la carga de datos a la base de datos y seguidamente se pasará a su respectiva tabla.

Y un DER de la base de datos la cual se puede visualizar mejor en el archivo **farmacia\_BI\_DER.png**

## **Data Marts**

Siguiendo con la línea de la fase de construcción, los Data Marts son estas tablas nuevas dentro del DataWarehouse (modelo estrella) y las construimos a partir de los datos que ya se integraron en los ODS. Aquí encontraremos las tablas de hechos y las tablas de dimensión, recordando un poco algunos conceptos de que la tabla de hechos son los datos numéricos o métricas que vamos a analizar y las tablas de dimensión son aquellas que nos describen el contexto. Se construirán los siguientes DM’s

* DM\_VENTAS
* DM\_INVENTARIO
* DM\_PEDIDOS
* DM\_COMPRAS
* DM\_DISTRIBUCIONES
* DM\_PRODUCCION
* DM\_RRHH
* DM\_INSPECCION
* DM\_MARKETING
* DM\_STOCK\_SUCURSAL

### **Data Mart #1 : Ventas\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: nos dará el contexto sobre el producto que fue vendido
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_cliente : poder ver quien hizo la compra
   1. id\_cliente
   2. nombre
   3. genero
   4. edad
   5. tipo\_cliente
4. dim\_empleado: ver quien realizó la venta
   1. id\_empleado
   2. nombre
   3. cargo
   4. id\_sucursal
5. dim\_sucursal : ver donde se realizó la venta
   1. id\_sucursal
   2. nombre\_sucursal
   3. ubicacion

**Tabla de Hecho**

1. fact\_ventas : Esta es la que almacena los datos numéricos y las métricas relacionadas a cada venta
   1. id\_fac\_venta
   2. id\_fecha (FK)
   3. id\_producto (FK)
   4. id\_cliente (FK)
   5. id\_empleado (FK)
   6. id\_sucursal (FK)
   7. cantidad\_vendida
   8. total\_bruto
   9. descuento\_total
   10. total\_neto

**Indicadores clave a obtener:**

* Total vendido por día, mes y año.
* Productos más vendidos por sucursal.
* Clientes con mayor frecuencia de compra.
* Empleados con mayor volumen de ventas.
* Comparativo de ventas físicas vs en línea.

**Proceso ETL:**

* Se extraerán los datos desde las tablas venta, detalle\_venta, cliente, producto, empleado y sucursal
* Ya una vez con la data, se realiza la transformación mediante la generación de id fecha, los cálculos totales, y la normalización de los datos
* Se esperaría ya cargar los datos en la tabla de fact\_ventas y hacer un estudio y generación del mismo por medio de python y su librería pandas

**Algebra Relacional**

* VentasConDetalles ← venta ⨝ detalle\_venta
* VentasConProducto ← VentasConDetalles ⨝ producto
* VentasConCliente ← VentasConProducto ⨝ cliente
* VentasConEmpleado ← VentasConCliente ⨝ empleado
* VentasFull ← VentasConEmpleado ⨝ sucursal
* Fact ← π id\_venta, id\_producto, id\_cliente, id\_empleado, id\_sucursal,

cantidad\_vendida,

(cantidad\_vendida \* precio\_unitario) AS total\_bruto,

descuento\_producto,

((cantidad\_vendida \* precio\_unitario) - descuento\_producto) AS total\_neto

(VentasFull)

* FactConFecha ← Fact ⨝σ(Fact.fecha\_venta = dim\_fecha.fecha) dim\_fecha

### **Data Mart #2 : Inventario\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal de los datos
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: este nos dirá que producto es el que se encuentra en inventario
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_proveedor: nos indica quien entregó el producto
   1. id\_proveedor
   2. nombre
   3. pais
   4. canal\_contacto
4. dim\_ubicacion\_bodega : nos indica la ubicación del producto dentro de la bodega
   1. id\_ubicacion
   2. descripcion

**Tabla de Hecho**

1. fact\_inventario: éste tendrá las métricas relacionadas al stock disponible en bodega
   1. id\_fact\_inventario
   2. id\_fecha (FK)
   3. id\_producto (FK)
   4. id\_proveedor (FK)
   5. id\_ubicacion (FK)
   6. cantidad\_total
   7. cantidad\_disponible
   8. dias\_antes\_vencimiento

**Indicadores clave a obtener**

* Productos que tienen un bajo nivel de inventario
* Lotes que están por vencer
* Proveedores con mayor volumen en bodega
* Stock disponible por producto, marca y categoría
* Análisis de ocupación de bodega

**Proceso ETL**

* Extracción de las tablas lote, producto, proveedor y ubicación bodega
* Transformación de por ejemplo el cálculo de días antes del vencimiento esto por cada lote dado la fecha actual, conversión de los formatos y normalizar los datos
* Carga mediante la inserción de los registros transformados y asociar las dimensiones según los id’s

**Algebra Relacional**

* LoteConProducto ← lote ⨝ producto
* LoteConProveedor ← LoteConProducto ⨝ proveedor
* LoteFull ← LoteConProveedor ⨝ ubicacion\_bodega
* Inventario ← π id\_lote, id\_producto, id\_proveedor, id\_ubicacion,

cantidad\_total, cantidad\_disponible,

(fecha\_vencimiento - CURRENT\_DATE) AS dias\_antes\_vencimiento

(LoteFull)

* InventarioConFecha ← Inventario ⨝σ(lote.fecha\_ingreso = dim\_fecha.fecha) dim\_fecha

### **Data Mart #3 :Pedidos\_DM**

**Tabla de Dimensión**

1. dim\_fecha: permite el análisis temporal de los datos
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: este nos dirá que producto es el que se encuentra en inventario
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_sucursal : ver donde se realizó la venta
   1. id\_sucursal
   2. nombre\_sucursal
   3. ubicacion

**Tabla de Hecho**

1. fact\_pedidos: contiene las metricas relaciones a los pedidos hechos por sucursal
   1. id\_fact\_pedido
   2. id\_fecha\_solicitud (FK)
   3. id\_fecha\_estimada\_entrega (FK)
   4. id\_producto (FK)
   5. id\_sucursal (FK)
   6. cantidad\_solicitada
   7. cantidad\_entregda
   8. es\_urgente
   9. estado\_pedido

**Indicadores clave a obtener**

* Estado actual de pedidos por sucursal.
* Productos más pedidos por sucursal y categoría.
* Análisis de cumplimiento de entregas (cantidad entregada vs solicitada).
* Pedidos urgentes pendientes.
* Tiempo promedio entre solicitud y entrega estimada.

**Proceso ETL**

* Extracción desde las tablas pedido y detalle\_pedido, además de producto y sucursal.
* Transformación: Clasificación del campo urgencia a es\_urgente (ej. “Alta” = TRUE).
* Normalización del estado del pedido (ej. “pendiente”, “entregado”, etc.) a valores consistentes.
* Generación de claves para fechas de solicitud y entrega.
* Carga: Inserción de registros en fact\_pedidos y se asociación a las dimensiones mediante claves foráneas.

**Algebra Relacional**

* PedidosFull ← pedido ⨝ detalle\_pedido
* PedidosConProducto ← PedidosFull ⨝ producto
* PedidosConSucursal ← PedidosConProducto ⨝ sucursal
* Fact ← π id\_pedido, id\_producto, id\_sucursal,

cantidad\_solicitada, cantidad\_entregada,

(CASE urgencia WHEN 'Alta' THEN TRUE ELSE FALSE END) AS es\_urgente,

estado AS estado\_pedido

(PedidosConSucursal)

* FactConFecha ← Fact ⨝σ(pedido.fecha\_solicitud = dim\_fecha.fecha) dim\_fecha AS FechaSolicitud ⨝σ(pedido.fecha\_estimada\_entrega = dim\_fecha.fecha) dim\_fecha AS FechaEntrega

### **Data Mart #4 :Compras\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal de los datos
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: este nos dirá que producto es el que se encuentra en inventario
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_proveedor: nos indica quien entregó el producto
   1. id\_proveedor
   2. nombre
   3. pais
   4. canal\_contacto

**Tabla de Hecho**

* fact\_compras: igualmente almacena las métricas asociadas a cada compra realizada por la empresa
  + id\_fact\_compra
  + id\_fecha\_orden (FK)
  + id\_fecha\_entrega (FK)
  + id\_producto (FK)
  + id\_proveedor (FK)
  + cantidad\_comrpada
  + precio\_unitario
  + total\_compra
  + estado\_orden

**Indicadores clave a obtener:**

* Total de compras realizadas por proveedor.
* Comparativo de precios por producto y proveedor.
* Proveedores con mayor volumen de compra.
* Órdenes de compra pendientes vs entregadas.
* Evolución del gasto en compras por mes o año.

**Procesos ETL**:

* Extracción desde las tablas compra, detalle\_compra, producto y proveedor.
* Transformación: Unificación de fechas y generación de claves de fecha seguidamente cálculo del total de la compra y la estandarización del estado de la orden recibida.
* Carga: Los datos transformados son insertados en fact\_compras, enlazados con claves a dim\_fecha, dim\_producto y dim\_proveedor.

**Algebra Relacional**

* ComprasFull ← compra ⨝ detalle\_compra
* ComprasConProducto ← ComprasFull ⨝ producto
* ComprasConProveedor ← ComprasConProducto ⨝ proveedor
* Fact ← π id\_compra, id\_producto, id\_proveedor,

cantidad,

precio\_unitario,

(cantidad \* precio\_unitario) AS total\_compra,

estado AS estado\_orden

(ComprasConProveedor)

* FactConFechas ← Fact ⨝σ(compra.fecha\_orden = dim\_fecha.fecha) dim\_fecha AS FechaOrden ⨝σ(compra.fecha\_real\_entrega = dim\_fecha.fecha) dim\_fecha AS FechaEntrega

### **Data Mart #5 :Distribuciones\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal de los datos
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_piloto: describir el encargado de realizar la entrega
   1. id\_piloto
   2. nombre
   3. licencia
   4. telefono
   5. correo
   6. estado
3. dim\_vehiculo : describir qué vehiculo es usado para la distribución
4. dim\_sucursal : ver donde se realizó la venta
   1. id\_sucursal
   2. nombre\_sucursal
   3. ubicacion

**Tabla de Hecho**

1. fact\_distribucion : almacena métricas relacionadas con los envíos realizados
   1. id\_fact\_distribucion
   2. id\_fecha\_salida (FK)
   3. id\_fecha\_entrada (FK)
   4. id\_piloto (FK)
   5. id\_vehiculo (FK)
   6. id\_sucursal (FK)
   7. cantidad\_productos\_entregados
   8. estado\_envio
   9. tiempo\_entrega\_dias

**Indicadores clave a obtener**

* Envíos en tránsito vs entregados.
* Pedidos entregados por piloto.
* Vehículos más utilizados.
* Tiempos promedio de entrega por región.
* Incidencias logísticas (demoras, fallas en entrega).

**Procesos ETL:**

* Extracción desde envío, piloto, vehículo, pedido, detalle\_pedido, sucursal.
* Transformación: Generación de fechas estándar y cálculo de tiempo\_entrega\_dias y por último agrupación de cantidades por pedido.
* Carga: Inserción en fact\_distribuciones con las dimensiones enlazadas.

**Algebra Relacional**

* EnviosBase ← envio ⨝ pedido ⨝ detalle\_pedido
* EnviosConPiloto ← EnviosBase ⨝ piloto
* EnviosConVehiculo ← EnviosConPiloto ⨝ vehiculo
* EnviosConSucursal ← EnviosConVehiculo ⨝ sucursal
* Fact ← π id\_envio, id\_piloto, id\_vehiculo, id\_sucursal,

SUM(cantidad\_entregada) AS cantidad\_productos\_entregados,

estado AS estado\_envio,

DATEDIFF(fecha\_entrega, fecha\_salida) AS tiempo\_entrega\_dias

(EnviosConSucursal)

* FactConFechas ← Fact

⨝σ(envio.fecha\_salida = dim\_fecha.fecha) dim\_fecha AS FechaSalida

⨝σ(envio.fecha\_entrega = dim\_fecha.fecha) dim\_fecha AS FechaEntrega

### **Data Mart #6 :Produccion\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal de los datos
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: este nos dirá que producto es el que se encuentra en inventario
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
3. dim\_empleado: ver quien realizó la venta
   1. id\_empleado
   2. nombre
   3. cargo
   4. id\_sucursal

**Tabla de Hecho**

1. fact\_produccion: contendrá las métricas de cada evento de producción
   1. id\_fact\_produccion
   2. id\_fecha\_inicio (FK)
   3. id\_fecha\_fin (FK)
   4. id\_producto (FK)
   5. id\_empleado (FK)
   6. id\_laboratorio (FK)
   7. cantidad\_producida
   8. duracion\_dias

**Indicadores clave a obtener:**

* Producción total por producto.
* Comparativo entre laboratorios.
* Tiempo promedio de fabricación por medicamento
* Turnos o empleados más productivos.
* Capacidad instalada vs. utilizada por laboratorio.

**Proceso ETL**

* Extracción desde las tablas producción, producto, empleado, laboratorio.
* Transformación: Normalización de fechas y cálculo de duración en días.
* Carga: Inserción de métricas en fact\_produccion y establecer Relación con dimensiones descriptivas.

**Algebra Relacional**

* ProduccionJoin ← produccion ⨝ producto ⨝ empleado ⨝ laboratorio
* Fact ← π id\_produccion, id\_producto, id\_empleado, id\_laboratorio,

fecha\_inicio, fecha\_fin, cantidad\_producida,

(fecha\_fin - fecha\_inicio) AS duracion\_dias

(ProduccionJoin)

* FactConFechas ← Fact

⨝σ(Fact.fecha\_inicio = dim\_fecha.fecha) dim\_fecha AS FechaInicio

⨝σ(Fact.fecha\_fin = dim\_fecha.fecha) dim\_fecha AS FechaFin

### **Data Mart #7 :RRHH\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_empleado: ver quien realizó la venta
   1. id\_empleado
   2. nombre
   3. cargo
   4. id\_sucursal
3. dim\_jornada: describir la jornada de trabajo del empleado
   1. id\_jornada
   2. tipo\_jornada

**Tabla de Hecho**

1. fact\_rrhh: contiene métricas relacionadas al comportamiento laboral del personal
   1. id\_fact\_rrhh
   2. id\_fecha (FK)
   3. id\_empleado (FK)
   4. id\_jornada (FK)
   5. estado\_asistencia (Asistió, Faltó, Tarde, etc.)
   6. total\_gestiones\_dia
   7. tipo\_gestion\_principal (última o más frecuente del día)

**Indicadores clave a obtener:**

* Asistencia por jornada y fecha.
* Turnos con mayor ausentismo.
* Empleados con mayor volumen de gestiones.
* Historial de asistencias por empleado.
* Empleados con más faltas o rotación.

**Proceso ETL**

* Extracción desde las tablas asistencia, gestion, empleado, jornada.
* Transformación: Normalización de fechas junto con agrupación y conteo de gestiones por día y la categorización de ausencias.
* Carga: Llenado de fact\_rrhh. Relación con dimensiones descriptivas.

**Algebra Relacional**

* AsistenciasJoin ← asistencia ⨝ empleado ⨝ jornada
* FactAsistencia ← π id\_empleado, id\_jornada, fecha, estado

(AsistenciasJoin)

* GestionesAgg ← γ id\_empleado, fecha, COUNT(id\_gestion) AS total\_gestiones,

MODE(tipo\_gestion) AS tipo\_gestion\_principal

(gestion)

* FactJoin ← FactAsistencia ⨝ GestionesAgg

ON (FactAsistencia.id\_empleado = GestionesAgg.id\_empleado AND

FactAsistencia.fecha = GestionesAgg.fecha)

* FactConFecha ← FactJoin ⨝σ(FactJoin.fecha = dim\_fecha.fecha) dim\_fecha

### **Data Mart #8 :Inspeccion\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal de los datos
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: este nos dirá que producto es el que se encuentra en inventario
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_proveedor: nos indica quien entregó el producto
   1. idproveedor
   2. nombre
   3. pais
   4. canal\_contacto
4. dim\_empleado\_inspector: encargado de la inspeccion
   1. id\_empleado
   2. nombre
   3. cargo
   4. id\_sucursal
   5. estado\_empleado
5. dim\_motivo\_devolucion
   1. id\_motivo
   2. descripcion

**Tabla de Hecho**

1. fact\_inspeccion: contiene datos numéricos relacionados con inspecciones de producto
   1. id\_fact\_inspeccion
   2. id\_fecha (FK)
   3. id\_producto (FK)
   4. id\_proveedor (FK)
   5. id\_empleado (FK)
   6. id\_motivo (FK)
   7. cantidad\_aceptada
   8. cantidad\_rechazada
   9. estado\_reclamo
   10. resultado\_final (Aprobado/Rechazado/Parcial)

**Indicadores clave por obtener**

* Tasa de devolución por proveedor.
* Cantidades rechazadas por producto.
* Empleados con más rechazos asignados.
* Principales motivos de devolución.
* Tiempo promedio entre inspección y resolución.

**Proceso ETL:**

* Extracción desde las tablas devolucion, detalle\_inspeccion, motivo\_dev, producto, proveedor, empleado.
* Transformación: Limpieza de datos duplicados o nulos y generación del identificador id\_fecha junto con consolidación de criterios de inspección.
* Carga: Datos consolidados en la tabla fact\_inspeccion y claves foráneas enlazadas con las dimensiones correspondientes.

**Algebra Relacional**

* InspeccionesJoin ← devolucion ⨝ producto ⨝ proveedor ⨝ empleado ⨝ motivo\_dev
* Fact ← π id\_producto, id\_proveedor, id\_empleado\_inspector, id\_motivo,

fecha\_inspeccion, cantidad\_aceptada, cantidad\_rechazada,

estado\_reclamo, resultado AS resultado\_final

(InspeccionesJoin)

* FactConFecha ← Fact ⨝σ(Fact.fecha\_inspeccion = dim\_fecha.fecha) dim\_fecha

### **Data Mart #9 :Marketing\_DM**

**Tablas de Dimensión**

1. dim\_fecha: permite el análisis temporal
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: nos dará el contexto sobre el producto que fue vendido
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_promocion: informacion general de la campa;a
   1. id\_promocion
   2. nombre
   3. canal
   4. motivo
   5. fecha\_inicio
   6. fecha\_fin
4. dim\_actividad: actividades relacionadas a las campa;as
   1. id\_actividad
   2. nombre\_actividad
   3. presupuesto
   4. asistencia

**Tabla de Hecho**

1. fact\_marketing: métricas asociadas a promociones y campañas
   1. id\_fact\_marketing
   2. id\_fecha (FK)
   3. id\_producto (FK)
   4. id\_promocion (FK)
   5. id\_actividad (FK)
   6. descuento
   7. ventas\_asociadas
   8. feedback\_positivo
   9. feedback\_negativo

**Indicadores clave a obtener**

* Promociones con más ventas asociadas.
* Descuento promedio aplicado por producto.
* Actividades más efectivas según asistencia y compras.
* Canales de promoción más exitosos.
* Análisis de retroalimentación de los clientes por campaña.

**Proceso ETL**

* Extracción desde las tablas promocion, detalle\_promocion, detalle\_actividad, feedback, producto.
* Transformación: Unificación de registros por fechas de campañas y categorización de comentarios en positivos/negativos junto con asociación entre productos y campañas activas.
* Carga: La tabla fact\_marketing consolida métricas de descuentos y resultados y las dimensiones alimentan filtros para análisis.

**Algebra Relacional**

* PromocionesJoin ← detalle\_promocion ⨝ promocion ⨝ producto ⨝ detalle\_actividad
* FeedbackClasificado ← π id\_promocion, COUNT\_IF(comentario LIKE '%bueno%' OR comentario LIKE '%excelente%') AS feedback\_positivo, COUNT\_IF(comentario LIKE '%malo%' OR comentario LIKE '%deficiente%') AS feedback\_negativo (feedback) GROUP BY id\_promocion
* MarketingJoin ← PromocionesJoin ⨝ FeedbackClasificado
* Fact ← π id\_producto, id\_promocion, id\_actividad, fecha\_inicio AS fecha, descuento, compras\_asociadas AS ventas\_asociadas, feedback\_positivo, feedback\_negativo , (MarketingJoin)
* FactConFecha ← Fact ⨝σ(Fact.fecha = dim\_fecha.fecha) dim\_fecha

### **Data Mart #10 :Stock\_Sucursal\_DM**

**Tablas de Dimension:**

1. dim\_fecha: permite el análisis temporal
   1. id\_fecha
   2. fecha
   3. anio
   4. mes
   5. dia
   6. trimestre
2. dim\_producto: nos dará el contexto sobre el producto que fue vendido
   1. id\_producto
   2. nombre
   3. marca
   4. presentacion
   5. unidad\_de\_medida
   6. es\_propio
   7. id\_categoria
3. dim\_sucursal : ver donde se realizó la venta
   1. id\_sucursal
   2. nombre\_sucursal
   3. ubicacion

**Tabla de Hecho**

1. fact\_stock\_sucursal: métricas de inventario en tiendas
2. id\_fact\_stock
3. id\_fecha (FK)
4. id\_producto (FK)
5. id\_sucursal (FK)
6. cantidad\_actual
7. minimo
8. maximo
9. cantidad\_movida
10. tipo\_movimiento (ingreso, salida, ajuste, etc.)
11. motivo (si aplica)

**Indicadores clave por obtener**

* Productos con sobrestock o desabastecimiento.
* Sucursales con más movimientos de stock.
* Análisis de productos más movidos en tienda.
* Control de cumplimiento de niveles mínimos/máximos.
* Comportamiento de inventario por mes, día y tienda.

**Proceso ETL**

* Extracción desde stock\_sucursal, movimiento\_stock, producto, sucursal.
* Transformación: Consolidación de cantidades diarias por tipo de movimiento mediante evaluación de cumplimiento de rangos de stock y enriquecimiento con nombre de producto, sucursal y fechas.
* Carga: Se cargan los datos transformados a fact\_stock\_sucursal y las dimensiones se actualizan con nuevos registros de producto, fecha y sucursal si es necesario.

**Algebra Relacional**

* MovimientosConStock ← movimiento\_stock ⨝ stock\_sucursal
* MovimientosFull ← MovimientosConStock ⨝ producto ⨝ sucursal
* Fact ← π

movimiento\_stock.id\_movimiento AS id\_fact\_stock,

movimiento\_stock.fecha\_movimiento AS fecha,

producto.id\_producto,

sucursal.id\_sucursal,

stock\_sucursal.cantidad\_actual,

stock\_sucursal.minimo,

stock\_sucursal.maximo,

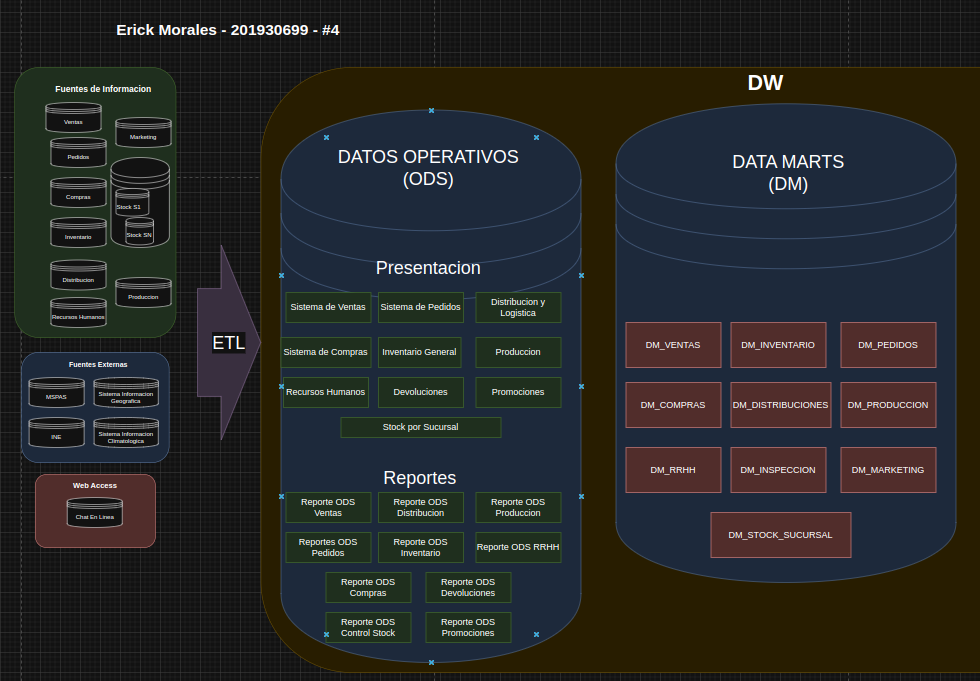
movimiento\_stock.cantidad AS cantidad\_movida,

movimiento\_stock.tipo\_movimiento,

movimiento\_stock.motivo

(MovimientosFull)

* FactConFecha ← Fact ⨝σ(Fact.fecha = dim\_fecha.fecha) dim\_fecha

En este punto llegando a esto:

## **Datos por Tema (Cubos y Modelos)**

### **Cubo OLAP - Ventas**

**Objetivo:** Poder analizar el comportamiento de las ventas desde los diferentes ángulos, así como por tiempo, por productos, clientes, empleados y sucursales

* Dimensiones
  + Fecha : año, mes, dia, trimestre
  + Producto: nombre, marca, categoría.
  + Cliente: tipo (frecuente, nuevo), género, edad.
  + Empleado: quién realizó la venta.
  + Sucursal: en qué ubicación ocurrió.
* Medidas
  + Total vendido (Q)
  + Cantidad vendida
  + Descuento aplicado
  + Total neto
* Dimensiones que permite
  + Detectar productos estrella o en baja rotación.
  + Identificar al mejor vendedor por sucursal.
  + Determinar patrones de compra por género o edad.
  + Comparar sucursales en ventas y rendimiento.

Tablas Involucradas (ODS Y DM)

* venta (ODS)
* detalle\_venta (ODS)
* producto (ODS)
* cliente (ODS)
* empleado (ODS)
* sucursal (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_cliente (DM)
* dim\_empleado (DM)
* dim\_sucursal (DM)
* fact\_ventas (DM)

**Algebra Relacional**

* Unión de Tablas
  + VentasConDetalles ← venta ⨝ detalle\_venta
  + VentasConProducto ← VentasConDetalles ⨝ producto
  + VentasConCliente ← VentasConProducto ⨝ cliente
  + Ventas Con Empleado ← VentasConCliente ⨝ empleado
  + VentasFull ← VentasConEmpleado ⨝ sucursal
* Cálculo de Métricas
  + Fact ← π id\_venta, id\_producto, id\_cliente, id\_empleado, id\_sucursal, cantidad\_vendida, (cantidad\_vendida \* precio\_unitario) AS total\_bruto, descuento\_producto,((cantidad\_vendida \* precio\_unitario) - descuento\_producto) AS total\_neto (VentasFull)

**Juntos con la fecha que se desea:**

* FactConFecha ← Fact ⨝σ(Fact.fecha\_venta = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo será usado desde una herramienta OLAP (Power BI).
* La tabla fact\_ventas será fuente de análisis y podrá ser modelada con las dimensiones ya creadas.
* El cubo se generará por herramienta en forma dinámica o como vista materializada.

### **Cubo OLAP - Inventario**

**Objetivo:** Poder analizar el estado del inventario central de medicamentos para tomar decisiones estratégicas sobre el almacenamiento, vencimiento, ubicación y rotación de productos

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha de ingreso y vencimiento del lote)
  + Producto: nombre, marca, presentación, unidad de medida, es propio, categoría
  + Proveedor: nombre, país
  + Ubicación: descripción del área dentro de la bodega central
* Medidas
  + Cantidad total ingresada
  + Cantidad disponible actual
  + Antiguedad del lote (dado en días)
  + Días restantes a vencimiento
* Dimensiones que permite
  + Productos con bajo stock o sobreinventario.
  + Medicamentos con riesgo de vencimiento.
  + Proveedores con mayor volumen de entrega.
  + Áreas de bodega más saturadas.
  + Eficiencia de rotación de productos por categoría.

Tablas Involucradas (ODS - DM)

* lote (ODS)
* producto (ODS)
* proveedor (ODS)
* proveedor (ODS)
* ubicacion\_bodega (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_proveedor (DM)
* dim\_ubicacion (DM)
* fact\_inventario (DM)

**Algebra Relacional**

* Unión de tablas
  + LotesConProducto ← lote ⨝ producto
  + LotesConProveedor ← LotesConProducto ⨝ proveedor
  + LotesConUbicacion ← LotesConProveedor ⨝ ubicacion\_bodega
* Cálculo de métricas
  + Fact ← π id\_lote, id\_producto, id\_proveedor, id\_ubicacion, cantidad\_total, cantidad\_disponible, (HOY - fecha\_ingreso) AS dias\_antiguedad, (fecha\_vencimiento - HOY) AS dias\_para\_vencer (LotesConUbicacion)
* Unión con Dimensiones temporales
  + FactConFecha ← Fact ⨝σ(Fact.fecha\_ingreso = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* la tabla fac\_inventario se utilizará como la tabla de hechos para el análisis
* el cubo igualmente se modelará dinámicamente desde power BI
* También puede crearse las vistas, para materializar el análisis

### **Cubo OLAP - Pedidos**

**Objetivo:** Poder analizar el comportamiento de los pedidos realizados por las sucursales, evaluando los volúmenes, urgencias, tiempos de entrega estimada y productos más solicitados.

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha\_solicitud).
  + Producto: nombre, categoría, presentación.
  + Sucursal: nombre, región, ubicación.
  + Urgencia: tipo de urgencia (alta, media, baja).
  + Estado: estado del pedido (pendiente, entregado, en tránsito, etc.).
* Medidas
  + Cantidad solicitada
  + Cantidad entregada
  + Diferencia solicitada vs entregada
  + Pedidos urgentes
  + Pedidos pendientes
* Dimensiones que permite
  + Sucursales que más solicitan productos.
  + Productos más pedidos por región.
  + Cumplimiento de entrega (solicitado vs entregado).
  + Análisis temporal del comportamiento de pedidos.
  + Niveles de urgencia más comunes.
  + Identificación de retrasos recurrentes.

Tablas Involucradas (ODS - DM)

* pedido (ODS)
* detalle\_pedido (ODS)
* producto (ODS)
* sucursal (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_sucursal (DM)
* fact\_pedidos (DM)

**Algebra Relacional**

* Unión de Tablas
  + PedidosConDetalles ← pedido ⨝ detalle\_pedido
  + PedidosConProducto ← PedidosConDetalles ⨝ producto
  + PedidosFull ← PedidosConProducto ⨝ sucursal
* Cálculo de Métricas
  + Fact ← π id\_pedido, id\_producto, id\_sucursal, fecha\_solicitud, urgencia, estado, cantidad\_solicitada, cantidad\_entregada, (cantidad\_solicitada - cantidad\_entregada) AS diferencia (FROM PedidosFull)
* Junto con las Fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha\_solicitud = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo podrá ser analizado mediante Power BI
* Se podrá cargar desde fact\_pedidos para poder modelar las dimensiones relacionadas
* Igualmente podrá generarse dinámicamente y materializar las consultas más rápidas

### **Cubo OLAP - Compras**

**Objetivo:**  Poder analizar el comportamiento de las compras realizadas a proveedores internacionales, evaluando los tiempos de entrega, volúmenes por producto, con el cumnplimiento de las condiciones y proveedores más frecuentes

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha\_orden, fecha\_entrega).
  + Producto: nombre, categoría, presentación.
  + Proveedor: nombre, país, canal de contacto.
  + Estado: estado de la orden de compra (pendiente, entregada, parcial, anulada).
* Medidas
  + Cantidad comprada
  + Precio unitario
  + Total de la compra (cantidad \* precio)
  + Diferencia de días entre orden y entrega
  + Órdenes pendientes
  + Órdenes completadas
* Dimensiones que permite
  + Identificar proveedores más usados y confiables.
  + Comparar precios por producto entre proveedores.
  + Medir cumplimiento de tiempos de entrega.
  + Detectar productos más comprados.
  + Evaluar eficiencia en el proceso de compras.

Tablas Involucradas (ODS Y DM)

* compra (ODS)
* detalle\_compra (ODS)
* producto (ODS)
* proveedor (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_proveedor (DM)
* fact\_compras (DM)

**Algebra Relacional**

* Unión de Tablas
  + ComprasConDetalles ← compra ⨝ detalle\_compra
  + ComprasConProducto ← ComprasConDetalles ⨝ producto
  + ComprasConProveedor ← ComprasConProducto ⨝ proveedor
* Cálculo de Métricas
  + Fact ← π id\_compra, id\_producto, id\_proveedor, fecha\_orden, fecha\_entrega, estado, cantidad, precio\_unitario, (cantidad \* precio\_unitario) AS total\_compra, (fecha\_entrega - fecha\_orden) AS dias\_entre\_orden\_entrega (FROM ComprasConProveedor)
* Junto con las fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha\_orden = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo será analizado desde Power BI
* Donde la tabla de fact\_compras es la tabla de hechos
* Igualmente se genera dinámicamente y como vista materializada de datos

### **Cubo OLAP - Distribuciones**

**Objetivo:**  Analizar el comportamiento de las entregas desde la bodega central hacia las sucursales, evaluando la eficiencia logística, tiempos de entrega, desempe;o de los pilotos y vehículos más utilizados

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha\_salida, fecha\_entrega).
  + Piloto: nombre, licencia, estado.
  + Vehículo: marca, modelo, capacidad.
  + Sucursal: nombre, ubicación, región.
  + Estado del envío: entregado, en tránsito, retrasado, cancelado.
* Medidas
  + Cantidad de productos entregados
  + Tiempo de entrega (en días)
  + Cantidad de envíos por piloto
  + Cantidad de envíos por vehículo
  + Envíos pendientes o retrasados
* Dimensiones que permite
  + Identificar envíos en tránsito vs entregados.
  + Medir eficiencia de entrega por región.
  + Detectar vehículos con más entregas o posibles sobreuso.
  + Evaluar desempeño de pilotos.
  + Analizar tiempos promedio de entrega.
  + Localizar incidencias logísticas por zona o vehículo.

Tablas Involucradas (ODS Y DM)

* envio (ODS)
* pedido (ODS)
* detalle\_pedido (ODS)
* vehiculo (ODS)
* piloto (ODS)
* sucursal (ODS)
* dim\_fecha (DM)
* dim\_piloto (DM)
* dim\_vehiculo (DM)
* dim\_sucursal (DM)
* fact\_distribucion (DM)

**Algebra Relacional**

* Unión de tablas
  + EnviosConPedido ← envio ⨝ pedido
  + EnviosConDetalles ← EnviosConPedido ⨝ detalle\_pedido
  + EnviosConSucursal ← EnviosConDetalles ⨝ sucursal
  + EnviosConVehiculo ← EnviosConSucursal ⨝ vehiculo
  + EnviosFull ← EnviosConVehiculo ⨝ piloto
* Cálculo de Métricas
  + Fact ← π id\_envio, id\_vehiculo, id\_piloto, id\_sucursal, fecha\_salida, fecha\_entrega, estado, SUM(cantidad\_entregada) AS cantidad\_entregada\_total,(fecha\_entrega - fecha\_salida) AS tiempo\_entrega\_dias(GROUP BY id\_envio) (FROM EnviosFull)
* Junto con las fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha\_salida = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo será construido desde Power BI
* La tabla de fact\_distribucion será la tabla de hechos
* Se puede segmentar por piloto, vehículo, sucursal o fecha

### **Cubo OLAP - Producción**

**Objetivo:**  Analizar la eficiencia y desempeño de los laboratorios farmacéuticos propios de la empresa, evaluando cantidad de medicamentos producidos, duración de la producción, uso de recursos humanos y capacidad por región

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha\_inicio y fecha\_fin de producción).
  + Producto: nombre, categoría, presentación, si es propio o de marca.
  + Laboratorio: nombre, ubicación, región.
  + Empleado: nombre, cargo, sucursal.
* Medidas
  + Cantidad producida
  + Duración de la producción (días)
  + Producción promedio por laboratorio
  + Medicamentos más fabricados
  + Turnos más productivos
* Dimensiones de permite
  + Comparar la productividad entre laboratorios.
  + Medir duración de fabricación por medicamento.
  + Identificar empleados más activos o eficientes.
  + Detectar qué productos tienen mayor volumen de fabricación.
  + Evaluar uso de capacidad instalada por laboratorio.

Tablas Involucradas (ODS Y DM)

* produccion (ODS)
* producto (ODS)
* laboratorio (ODS)
* empleado (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_empleado (DM)
* fact\_produccion (DM)

**Algebra Relacional**

* Unión de Tablas
  + ProduccionConProducto ← produccion ⨝ producto
  + ProduccionConEmpleado ← ProduccionConProducto ⨝ empleado
  + ProduccionFull ← ProduccionConEmpleado ⨝ laboratorio
* Cálculo de Métricas
  + Fact ← π id\_produccion, id\_producto, id\_empleado, id\_laboratorio, fecha\_inicio, fecha\_fin, cantidad\_producida, (fecha\_fin - fecha\_inicio) AS duracion\_dias(FROM ProduccionFull)
* Unión con las fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha\_inicio = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo se analizará igualmente con Power BI por la tabla fact\_produccion que es la tabla de hechos
* Se modelaran las relaciones con las dimensiones
* Se genera dinámicamente y mediante vistas para materializar para rendimiento

### **Cubo OLAP - RRHH**

**Objetivo:** Analizar la gestión de personal en la empresa mediante la evaluación de asistencia, tipo de jornada, volumen de gestiones diarias y comportamiento por empleado o turno para mejorar la eficiencia operativa.

* Dimensiones
  + Fecha: año, mes, día, trimestre (de asistencia o gestión).
  + Empleado: nombre, cargo, sucursal.
  + Jornada: tipo de jornada (matutina, vespertina, nocturna, etc.).
* Medidas
  + Asistencias (cantidad de días que asistió)
  + Faltas o ausencias
  + Tardanzas
  + Total de gestiones realizadas por día
  + Tipo de gestión más común del día
* Dimensiones que permite
  + Identificar empleados con mayor compromiso o ausentismo.
  + Detectar jornadas con más problemas de asistencia.
  + Medir la productividad diaria por empleado.
  + Analizar tendencias de asistencia según fechas.
  + Evaluar necesidades de rotación o refuerzo en horarios específicos.

**Tablas Involucradas (ODS Y DM)**

* asistencia (ODS)
* gestion (ODS)
* empleado (ODS)
* jornada (ODS)
* dim\_fecha (DM)
* dim\_empleado (DM)
* dim\_jornada (DM)
* fact\_rrhh (DM)

**Algebra Relacional**

* Unión de Tablas
  + Asistencias ← asistencia ⨝ empleado ⨝ jornada
  + Gestiones ← gestion ⨝ empleado
  + FullRRHH ← Asistencias ⨝ Gestiones (por id\_empleado y fecha)
* Cálculo de Métricas
  + Fact ← π id\_empleado, id\_jornada, fecha, estado\_asistencia, COUNT(id\_gestion) AS total\_gestiones\_dia, MAX(tipo\_gestion) AS tipo\_gestion\_principal(GROUP BY id\_empleado, fecha) (FROM FullRRHH)
* Unión con la fecha
  + FactConFecha ← Fact ⨝σ(Fact.fecha = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* Se analizará con Power BI y la tabla de fact\_rrhh que es la tabla de hechos
* Este modelo se basa en las dimensiones de tiempo, empleado y jornada

### **Cubo OLAP - Devoluciones**

**Objetivo:** Analizar el proceso de inspección y devolución de productos en la bodega central, evaluando la calidad, viendo los proveedores con más rechazos, los motivos comunes de devolución y el desempeño de los inspectores.

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha de inspección).
  + Producto: nombre, categoría, presentación.
  + Proveedor: nombre, país, canal de contacto.
  + Inspector (Empleado): nombre, cargo, estado.
  + Motivo de Devolución: descripción del motivo (defecto, error de pedido, etc.).
* Medidas
  + Cantidad aceptada
  + Cantidad rechazada
  + Porcentaje rechazado
  + Número de devoluciones por proveedor
  + Tiempo promedio de resolución (si aplica)
  + Rechazos por motivo
* Dimensiones que permite
  + Identificar los productos más rechazados por inspección.
  + Detectar proveedores con más devoluciones.
  + Conocer los motivos más frecuentes de devolución.
  + Evaluar el desempeño de los inspectores.
  + Estudiar la calidad por marca, proveedor o tipo de producto.

Tablas Involucradas (ODS Y DM)

* devolucion (ODS)
* detalle\_inspeccion (ODS)
* producto (ODS)
* proveedor (ODS)
* motivo\_dev (ODS)
* empleado (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_proveedor (DM)
* dim\_empleado\_inspector (DM)
* dim\_motivo\_devolucion (DM)
* fact\_inspeccion (DM)

**Algebra Relacional**

* Unión de Tablas
  + DevolucionesConProducto ← devolucion ⨝ producto
  + DevolucionesConProveedor ← DevolucionesConProducto ⨝ proveedor
  + DevolucionesConMotivo ← DevolucionesConProveedor ⨝ motivo\_dev
  + DevolucionesConInspector ← DevolucionesConMotivo ⨝ empleado
* Cálculo de Métricas:
  + Fact ← π id\_devolucion, id\_producto, id\_proveedor, id\_empleado\_inspector, id\_motivo, fecha\_inspeccion, cantidad\_aceptada, cantidad\_rechazada, (cantidad\_rechazada / (cantidad\_aceptada + cantidad\_rechazada)) \* 100 AS porcentaje\_rechazo,estado\_reclamo(FROM DevolucionesConInspector)
* Unión con las fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha\_inspeccion = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El análisis mediante Power BI y usando la tabla de hechos de fact\_inspeccion
* Las dimensiones igualmente nos permitirán filtrar los datos
* Se genera dinámicamente y mediante vistas para materializar para rendimiento

### **Cubo OLAP - Marketing**

**Objetivo:** Analizar el impacto de las campañas de marketing y promociones en las ventas, evaluando la efectividad por canal, producto promocionado, feedback recibido y actividades asociadas

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha de la promoción o actividad).
  + Producto: nombre, marca, categoría, si es propio o no.
  + Promoción: nombre, canal, motivo, fechas de inicio y fin.
  + Actividad: nombre de la actividad, presupuesto, asistencia.
  + Feedback (cliente): tipo de comentario (positivo/negativo), edad, región (opcional).
* Medidas
  + Ventas asociadas a promociones
  + Descuento aplicado
  + Cantidad de promociones activas
  + Asistencia a actividades
  + Feedback positivo y negativo por campaña
  + Inversión en marketing (presupuesto por actividad)
* Dimensiones que permite
  + Identificar promociones más efectivas.
  + Medir impacto de campañas por canal y región.
  + Evaluar retorno de inversión por actividad promocional.
  + Analizar la percepción del cliente (feedback).
  + Detectar productos más vendidos durante promociones.

Tablas Involucradas (ODS Y DM)

* promocion (ODS)
* detalle\_promocion (ODS)
* detalle\_actividad (ODS)
* feedback (ODS)
* producto (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_promocion (DM)
* dim\_actividad (DM)
* fact\_marketing (DM)

**Algebra Relacional**

* Unión de Tablas
  + PromocionesConProducto ← detalle\_promocion ⨝ producto
  + PromocionesFull ← PromocionesConProducto ⨝ promocion ⨝ detalle\_actividad ⨝ feedback
* Cálculo de Métricas
  + Fact ← π id\_promocion, id\_producto, id\_actividad, id\_fecha, descuento, ventas\_asociadas, COUNT(CASE WHEN comentario LIKE '%bueno%' THEN 1 END) AS feedback\_positivo,COUNT(CASE WHEN comentario LIKE '%malo%' THEN 1 END) AS feedback\_negativo,presupuesto, asistencia(GROUP BY id\_promocion, id\_producto, id\_actividad, id\_fecha)(FROM PromocionesFull)
* Unión con las fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo igualmente se analizará desde Power BI por medio de la tabla de hechos fact\_marketing
* La dimensión igualmente se permite filtrar por producto, promoción, canal, actividad y fecha
* Se genera dinámicamente y mediante vistas para materializar para rendimiento

### **Cubo OLAP - Stock por Sucursal**

**Objetivo:** Analizar los niveles de inventario en cada sucursal, permitiendo detectar estos desabastecimientos, sobrestock, comportamiento de productos en cada tienda y cumpliendo los niveles mínimos y máximos establecidos

* Dimensiones
  + Fecha: año, mes, día, trimestre (fecha del movimiento o actualización).
  + Producto: nombre, marca, presentación, unidad de medida, categoría, si es propio o no.
  + Sucursal: nombre, ubicación, región.
  + Tipo de Movimiento: ingreso, salida, ajuste, devolución.
  + Motivo (opcional): razón del movimiento (rotación, daño, vencimiento, etc.).
* Medidas
  + Cantidad actual en stock
  + Cantidad movida (por tipo de movimiento)
  + Diferencia contra mínimo/máximo establecido
  + Número de movimientos por producto o sucursal
  + Variación de inventario en el tiempo
* Dimensiones que permite
  + Detectar productos con sobrestock o desabastecimiento.
  + Ver comportamiento de inventario por tienda, fecha o producto.
  + Medir la frecuencia y causas de movimientos.
  + Comparar cumplimiento de niveles mínimos/máximos entre sucursales.
  + Tomar decisiones para reabastecimiento o traslado de stock.

Tablas Involucradas (ODS Y DM)

* stock\_sucursal (ODS)
* movimiento\_stock (ODS)
* producto (ODS)
* sucursal (ODS)
* dim\_fecha (DM)
* dim\_producto (DM)
* dim\_sucursal (DM)
* fact\_stock\_sucursal (DM)

**Algebra Relacional**

* Unión de Tablas
  + Movimientos ← movimiento\_stock ⨝ producto ⨝ sucursal
  + Stock ← stock\_sucursal ⨝ producto ⨝ sucursal
  + StockFull ← Movimientos ⨝ Stock (por id\_producto y id\_sucursal)
* Cálculo de Métricas
  + Fact ← π id\_producto, id\_sucursal, fecha\_movimiento AS fecha, cantidad\_actual, minimo, maximo, cantidad AS cantidad\_movida, tipo\_movimiento, motivo,(cantidad\_actual - minimo) AS diferencia\_minimo,(cantidad\_actual - maximo) AS diferencia\_maximo(FROM StockFull)
* Unión con las fechas
  + FactConFecha ← Fact ⨝σ(Fact.fecha = dim\_fecha.fecha) dim\_fecha

**Formato de Almacenamiento**

* El cubo se analizará igualmente con Power BI y se conectada a la tabla fact\_stock\_sucursal que es la tabla de hechos
* Igualmente se podrá filtrar las dimensiones por producto, sucursal, fecha y tipo de movimiento para poder segmentar los análisis
* Se genera dinámicamente y mediante vistas para materializar para rendimiento

### **Cubo OLAP - Rentabilidad por Producto**

**Objetivo**: Evaluar qué productos son más rentables considerando costos de compra, producción y descuentos en venta.

* **Fuentes**: DM\_VENTAS, DM\_COMPRAS, DM\_PRODUCCION
* **Medidas**: margen por unidad, ganancia total, costo total, precio medio de venta
* **Dimensiones**: producto, fecha, proveedor, laboratorio, sucursal
* **Valo**r: ayuda a eliminar productos no rentables del catálogo.

Tablas Involucradas

* fact\_ventas
* fact\_compras
* fact\_produccion
* dim\_producto
* dim\_proveedor
* dim\_laboratorio
* dim\_fecha
* dim\_sucursal

**Algebra Relacional**

* Obtener Precios y Ventas
  + Ventas ← fact\_ventas ⨝ dim\_producto ⨝ dim\_sucursal ⨝ dim\_fecha
* Obtener Costos de Compra
  + Compras ← fact\_compras ⨝ dim\_producto ⨝ dim\_proveedor
* Obtener costos de producción
  + Produccion ← fact\_produccion ⨝ dim\_producto ⨝ dim\_laboratorio
* Unión de tablas
  + Rentabilidad ← Ventas ⨝ Compras ⨝ Producción (por id\_producto y fechas aproximadas)
* Calcular las Métricas
* Fact Rentabilidad ← π id\_producto, id\_sucursal, id\_fecha, SUM(total\_neto) AS ingresos,AVG(precio\_unitario\_compra) AS costo\_compra,AVG(costo\_produccion\_unitario) AS costo\_produccion,(AVG(precio\_unitario\_venta) - (costo\_compra + costo\_produccion)) AS margen\_unitario,ingresos - (cantidad\_vendida \* (costo\_compra + costo\_produccion)) AS ganancia\_total(FROM Rentabilidad)

### **Cubo OLAP - Tiempos de Respuesta Operativa**

**Objetivo**: Evaluar qué productos son más rentables considerando costos de compra, producción y descuentos en venta.

* **Fuentes:** DM\_VENTAS, DM\_COMPRAS, DM\_PRODUCCION
* **Medidas:** margen por unidad, ganancia total, costo total, precio medio de venta
* **Dimensiones:** producto, fecha, proveedor, laboratorio, sucursal
* **Valor:** ayuda a eliminar productos no rentables del catálogo.

Tabla Involucradas

* fact\_pedidos
* fact\_distribucion
* dim\_sucursal
* dim\_fecha
* dim\_producto

**Algebra Relacional**

* Unir pedidos con distribuciones
  + PedidosDistribuciones ← fact\_pedidos ⨝ fact\_distribucion (por id\_sucursal y id\_producto)
* Calcular Tiempos
  + Tiempos ← π id\_producto, id\_sucursal, id\_fecha\_solicitud, id\_fecha\_entrega,
  + (id\_fecha\_entrega - id\_fecha\_solicitud) AS dias\_espera,
  + es\_urgente, estado\_pedido, estado\_envio
  + (FROM PedidosDistribuciones)
* Agregar Contexto geográfico
  + TiemposConDim ← Tiempos ⨝ dim\_producto ⨝ dim\_sucursal ⨝ dim\_fecha

### **Cubo OLAP - Demanda Estacional**

**Objetivo:** Cruzar clima o temporadas con ventas para detectar patrones estacionales.

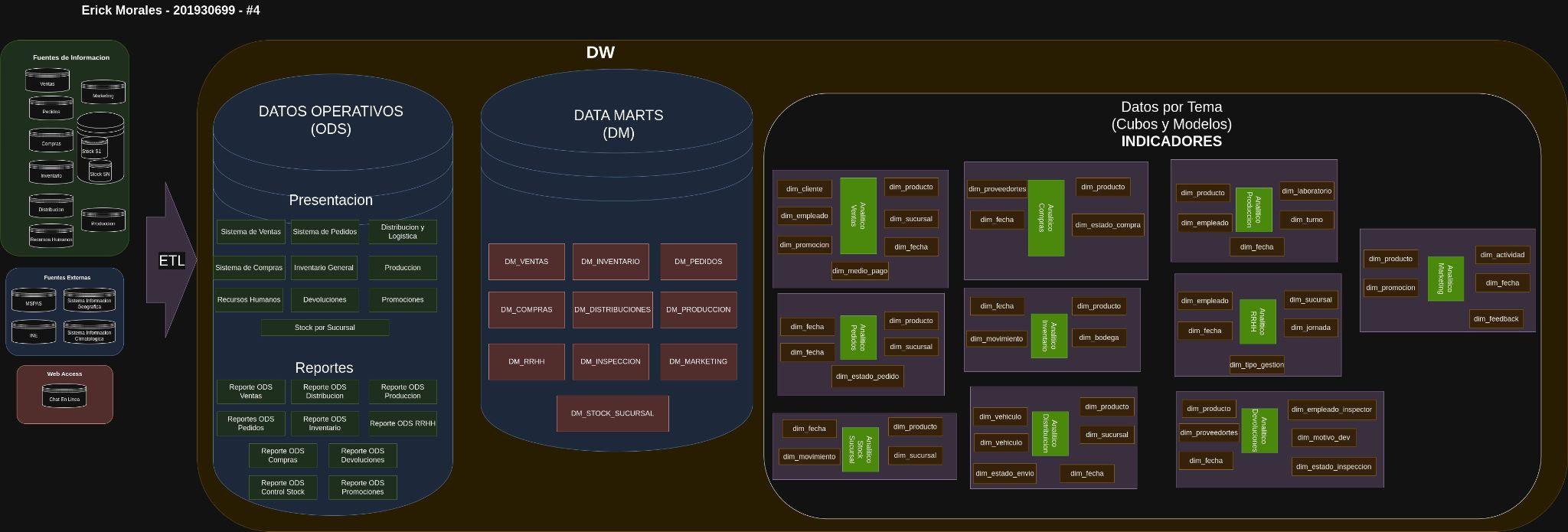
* **Fuentes:** DM\_VENTAS + sistema climatológico
* **Medidas:** ventas por temporada, correlación clima vs demanda
* **Dimensiones:** fecha, producto, región, temperatura, humedad
* **Valor:** anticipa stock por temporada (ej. gripe en invierno, alergias en verano).

Tablas Involucradas

* fact\_ventas
* dim\_producto
* dim\_sucursal
* dim\_fecha
* clima (fuente externa; por fecha y región)

**Algebra Relacional**

* Unir ventas con los productos, sucursal y fecha
  + Ventas ← fact\_ventas ⨝ dim\_producto ⨝ dim\_sucursal ⨝ dim\_fecha
* Unir con los datos climatologicos
  + VentasClima ← Ventas ⨝ climaON (dim\_fecha.fecha = clima.fecha AND dim\_sucursal.region = clima.region)
* Calcular Métricas
  + Estacionalidad ← π id\_producto, id\_sucursal, id\_fecha, temperatura, humedad, SUM(cantidad\_vendida) AS ventas\_totales(GROUP BY id\_producto, id\_fecha, temperatura, humedad) (FROM VentasClima)

Y con esta fase de construcción llegaríamos a esto:

La imagen se puede ver mas amplia en **construccion\_bi.jpg**

### **Indicadores**

Cubo OLAP - Ventas

* Indicadores
  + Total de Ventas
    - Algebra Relacional
      * TotalVentas ← γ SUM(cantidad \* precio\_unitario) (fact\_ventas)
    - SQL
      * SELECT SUM(cantidad \* precio\_unitario) AS total\_ventas FROM fact\_ventas;
  + Ticket Promedio
    - Algebra Relacional
      * Total ← γ SUM(total\_venta) AS total, COUNT(\*) AS num\_ventas (fact\_ventas) TicketPromedio ← π (total / num\_ventas → ticket\_promedio) (Total)
    - SQL
      * SELECT SUM(total\_venta) / COUNT(\*) AS ticket\_promedio FROM fact\_ventas;
  + Top 5 productos más vendidos
    - Algebra Relacional
      * VentasConProducto ← fact\_ventas ⨝ dim\_producto
      * VentasPorProducto ← γ nombre\_producto, SUM(cantidad) AS total\_vendido (VentasConProducto)
      * TopProductos ← τ↓5(total\_vendido) (VentasPorProducto)
    - SQL
      * SELECT dp.nombre\_producto, SUM(fv.cantidad) AS total\_vendido FROM fact\_ventas fv JOIN dim\_producto dp ON fv.id\_producto = dp.id\_producto GROUP BY dp.nombre\_producto ORDER BY total\_vendido DESC LIMIT 5;

Cubo OLAP : Compras

* Indicadores
  + Total de Compras
    - Algebra Relacional
      * TotalCompras ← γ SUM(cantidad \* precio\_unitario) (fact\_compras)
    - SQL
      * SELECT SUM(cantidad \* precio\_unitario) AS total compras FROM fact\_compras;
  + Proveedor más Frecuente
    - Algebra Relacional
      * ComprasProveedor ← fact\_compras ⨝ dim\_proveedor
      * ComprasAgrupadas ← γ nombre\_proveedor, COUNT(\*) AS veces\_comprado (ComprasProveedor)
      * ProveedorTop ← τ↓1(veces\_comprado) (ComprasAgrupadas)
    - SQL
      * SELECT dp.nombre\_proveedor, COUNT(\*) AS veces\_comprado FROM fact\_compras fc JOIN dim\_proveedor dp ON fc.id\_proveedor = dp.idproveedor GROUP BY dp.nombre\_proveedor ORDER BY veces\_comprado DESC LIMIT 1;
  + Tiempo promedio entre orden y entrega
    - Algebra Relacional
      * Tiempos ← π (fecha\_entrega - fecha\_orden) AS dias (fact\_compras)
      * PromedioTiempo ← γ AVG(dias) (Tiempos)
    - SQL
      * SELECT AVG(fecha\_entrega - fecha\_orden) AS dias\_promedio\_entrega FROM fact\_compras;

Cubo OLAP - Pedidos

* Indicadores
  + Porcentaje de cumplimiento de entrega
    - Algebra Relacional
      * Totales ← γ SUM(cantidad\_entregada) AS entregado, SUM(cantidad\_solicitada) AS solicitado (fact\_pedidos)
      * Cumplimiento ← π ((entregado \* 100.0) / solicitado → porcentaje\_cumplimiento) (Totales)
    - SQL
      * SELECT SUM(cantidad\_entregada) \* 100.0 / SUM(cantidad\_solicitada) AS cumplimiento\_pedidos FROM fact\_pedidos;
  + Pedidos Urgentes
    - Algebra Relacional
      * Urgentes ← σ(urgencia = 'Alta') (fact\_pedidos)
      * CantidadUrgentes ← γ COUNT(\*) AS pedidos\_urgentes (Urgentes)
    - SQL
      * SELECT COUNT(\*) AS pedidos\_urgentes FROM fact\_pedidos WHERE urgencia = 'Alta';
  + Tiempo Promedio de entrega
    - Algebra Relacional
      * Tiempos ← π (fecha\_entrega\_estimada - fecha\_solicitud) AS dias\_entrega (fact\_pedidos)
      * PromedioTiempo ← γ AVG(dias\_entrega) AS promedio\_dias (Tiempos)
    - SQL
      * SELECT AVG(fecha\_entrega\_estimada - fecha\_solicitud) AS promedio\_dias FROM fact\_pedidos;

Cubo OLAP - Distribuciones

* Indicadores
  + Tiempo promedio de entrega
    - Algebra Relacional
      * Entregados ← σ(estado = 'Entregado') (fact\_distribucion)
      * Tiempos ← π (fecha\_entrega - fecha\_salida) AS dias\_entrega (Entregados)
      * PromedioTiempo ← γ AVG(dias\_entrega) AS promedio\_dias (Tiempos)
    - SQL
      * SELECT AVG(fecha\_entrega - fecha\_salida) AS dias\_entrega FROM fact\_distribucion WHERE estado = 'Entregado';
  + Entregas por vehículo
    - Algebra Relacional
      * DistribucionVehiculo ← fact\_distribucion ⨝ dim\_vehiculo
      * EnviosPorVehiculo ← γ placa, COUNT(\*) AS total\_envios (DistribucionVehiculo)
      * RankingVehiculos ← τ↓(total\_envios) (EnviosPorVehiculo)
    - SQL
      * SELECT dv.placa, COUNT(\*) AS total\_envios FROM fact\_distribucion fd JOIN dim\_vehiculo dv ON fd.id\_vehiculo = dv.id\_vehiculo GROUP BY dv.placa ORDER BY total\_envios DESC;

Cubo OLAP - Stock por Sucursal

* Indicadores
  + Sucursales con productos en mínimo
    - Algebra Relacional
      * StockSucursalProducto ← fact\_stock\_sucursal ⨝ dim\_sucursal ⨝ dim\_producto
      * StockMinimo ← σ(cantidad\_actual < cantidad\_minima) (StockSucursalProducto)
      * Resultado ← π nombre\_sucursal, nombre\_producto, cantidad\_actual, cantidad\_minima (StockMinimo)
    - SQL
      * SELECT ds.nombre\_sucursal, dp.nombre\_producto, ss.cantidad\_actual, fss.cantidad\_minima FROM fact\_stock\_sucursal fss JOIN dim\_sucursal ds ON fss.id\_sucursal = ds.id\_sucursal JOIN dim\_producto dp ON fss.id\_producto = dp.id\_producto WHERE fss.cantidad\_actual < fss.cantidad\_minima;
  + Stock Actual por Sucursal
    - Algebra Relacional
      * StockConSucursal ← fact\_stock\_sucursal ⨝ dim\_sucursal
      * StockAgrupado ← γ nombre\_sucursal, SUM(cantidad\_actual) AS stock\_total (StockConSucursal)
    - SQL
      * SELECT ds.nombre\_sucursal, SUM(fss.cantidad\_actual) AS stock\_total FROM fact\_stock\_sucursal fss JOIN dim\_sucursal ds ON fss.id\_sucursal = ds.id\_sucursal GROUP BY ds.nombre\_sucursal;

Cubo OLAP: Producción

* Indicadores
  + Cantidad total producida por laboratorio
    - Algebra Relacional
      * ProduccionLaboratorio ← fact\_produccion ⨝ dim\_laboratorio
      * TotalProducido ← γ nombre\_laboratorio, SUM(cantidad\_producida) AS total\_producido (ProduccionLaboratorio)
    - SQL
      * SELECT dl.nombre\_laboratorio, SUM(fp.cantidad\_producida) AS total\_producido FROM fact\_produccion fp JOIN dim\_laboratorio dl ON fp.id\_laboratorio = dl.id\_laboratorio GROUP BY dl.nombre\_laboratorio;
  + Tiempo promedio de Producción
    - Algebra Relacional
      * Duraciones ← π (fecha\_fin - fecha\_inicio) AS duracion (fact\_produccion)
      * PromedioDuracion ← γ AVG(duracion) AS duracion\_promedio (Duraciones)
    - SQL
      * SELECT AVG(fecha\_fin - fecha\_inicio) AS duracion\_promedio FROM fact\_produccion;

Cubo OLAP - RRHH

* Indicadores
  + Tasa de asistencia general empleado
    - Algebra Relacional
      * Asistencias ← σ(estado = 'Asistencia') (fact\_rrhh)
      * TotalAsistencias ← γ COUNT(\*) AS asistencias (Asistencias)

TotalRegistros ← γ COUNT(\*) AS total (fact\_rrhh)

* TasaAsistencia ← π ((asistencias \* 100.0) / total → tasa\_asistencia) (TotalAsistencias × TotalRegistros)
  + - SQL
      * SELECT COUNT(\*) FILTER (WHERE estado = 'Asistencia') \* 100.0 / COUNT(\*) AS tasa\_asistencia FROM fact\_rrhh;
  + Faltas por empleado
    - Algebra Relacional
      * Faltas ← σ(estado = 'Falta') (fact\_rrhh)
      * FaltasEmpleado ← Faltas ⨝ dim\_empleado
      * FaltasAgrupadas ← γ nombre\_empleado, COUNT(\*) AS total\_faltas (FaltasEmpleado)
    - SQL
      * SELECT de.nombre\_empleado, COUNT(\*) AS total\_faltas FROM fact\_rrhh fr JOIN dim\_empleado de ON fr.id\_empleado = de.id\_empleado WHERE fr.estado = 'Falta' GROUP BY de.nombre\_empleado;

Cubo OLAP - Inspección

* Indicadores
  + Porcentaje de devoluciones
    - Algebra Relacional
      * Totales ← γ SUM(cantidad\_rechazada) AS rechazadas, SUM(cantidad\_aceptada) AS aceptadas (fact\_inspeccion)
      * PorcentajeRechazo ← π ((rechazadas \* 100.0) / (rechazadas + aceptadas) → porcentaje\_rechazo) (Totales)
    - SQL
      * SELECT SUM(cantidad\_rechazada) \* 100.0 / SUM(cantidad\_rechazada + cantidad\_aceptada) AS porcentaje\_rechazo FROM fact\_inspeccion;
  + Motivo más frecuente de devolucion
    - Algebra Relacional
      * Motivos ← fact\_inspeccion ⨝ dim\_motivo\_devolucion
      * MotivosAgrupados ← γ descripcion\_motivo, COUNT(\*) AS frecuencia (Motivos)
      * MotivoTop ← τ↓1(frecuencia) (MotivosAgrupados)
    - SQL
      * SELECT dm.descripcion\_motivo, COUNT(\*) AS frecuencia
      * FROM fact\_inspeccion fi JOIN dim\_motivo\_devolucion dm ON fi.id\_motivo = dm.id\_motivo GROUP BY dm.descripcion\_motivo ORDER BY frecuencia DESC LIMIT 1;

Cubo OLAP - Marketing

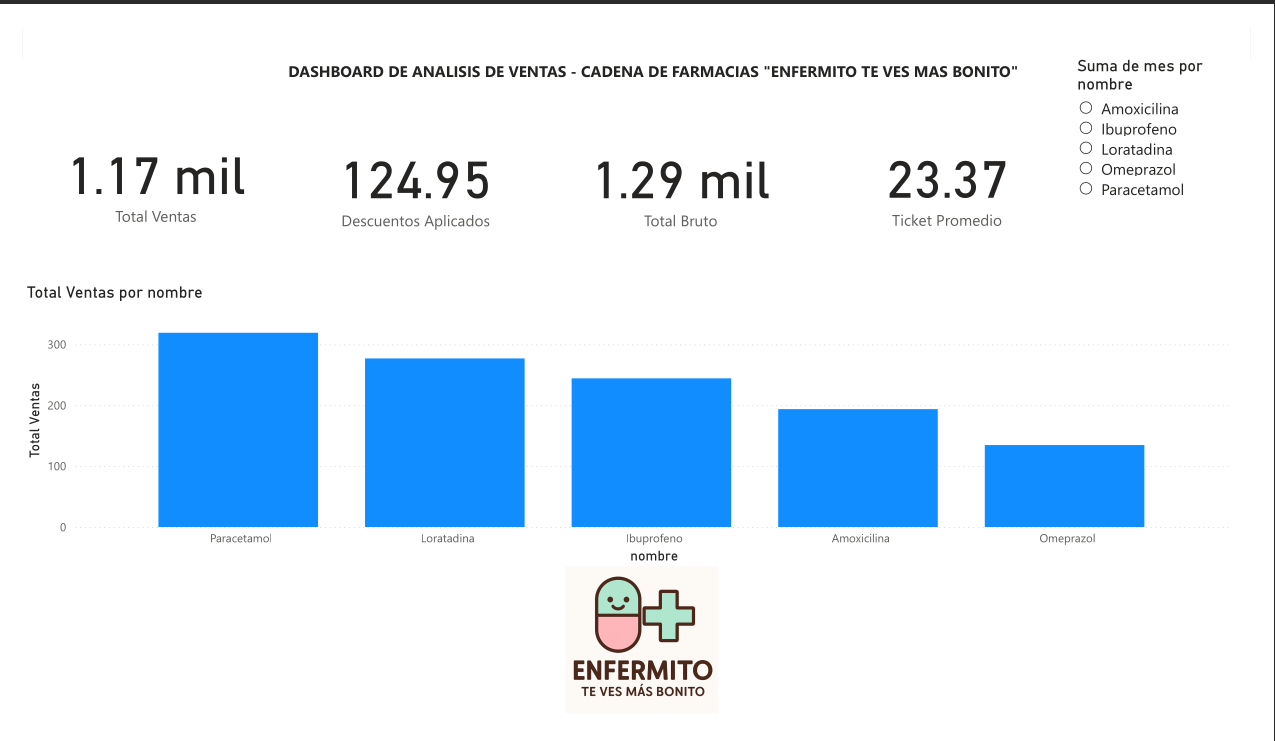
* Indicadores
  + Ventas durante campañas
    - Algebra Relacional
      * VentasConPromo ← σ(id\_promocion IS NOT NULL) (fact\_ventas)
      * TotalVentasPromo ← γ SUM(total\_venta) AS ventas\_en\_promocion (VentasConPromo)
    - SQL
      * SELECT SUM(total\_venta) AS ventas\_en\_promocion FROM fact\_ventas WHERE id\_promocion IS NOT NULL;
  + Promoción con mayor impacto
    - Algebra Relacional
      * VentasPromociones ← fact\_ventas ⨝ dim\_promocion
      * ImpactoPorPromo ← γ nombre\_promocion, SUM(total\_venta) AS total (VentasPromociones)
      * PromoTop ← τ↓1(total) (ImpactoPorPromo)
    - SQL
      * SELECT dp.nombre\_promocion, SUM(fv.total\_venta) AS total FROM fact\_ventas fv JOIN dim\_promocion dp ON fv.id\_promocion = dp.id\_promocion GROUP BY dp.nombre\_promocion ORDER BY total DESC LIMIT 1;

# **Fase de Comunicación**

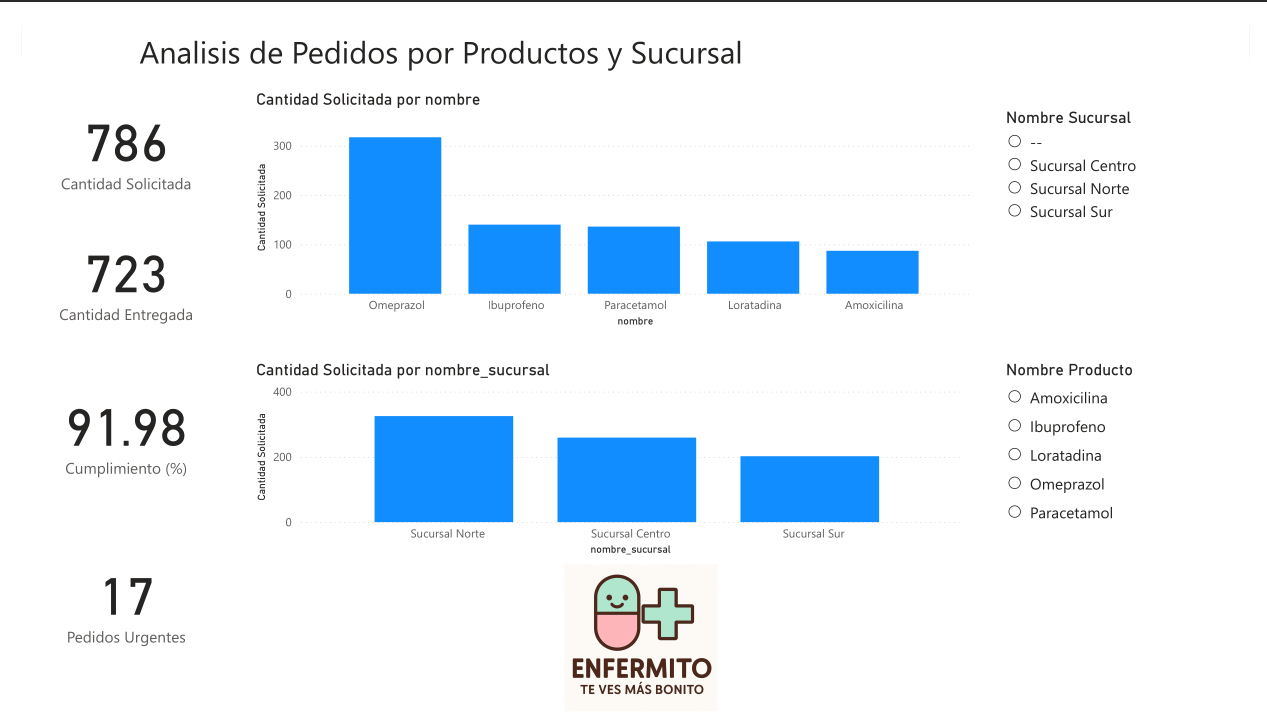
Como sabemos en esta el objetivo es poder pasar esa traducción de los datos ya procesados en información que se ha comprensible y útil para los diferentes niveles de la organización. A través de dashboards temáticos, donde se busca que los responsable de las tomas de decisiones puedan acceder fácilmente a los indicadores clave, detectar anomalías, comparar resultados y planificar estrategias basadas en datos que ya hemos generado, limpiado y alistado mediante una madurez de los datos alta, junto con una buena arquitectura de BI .

Luego de hacer el proceso de carga mediante a BI con datos de prueba, se pudieron generar los siguientes reportes:

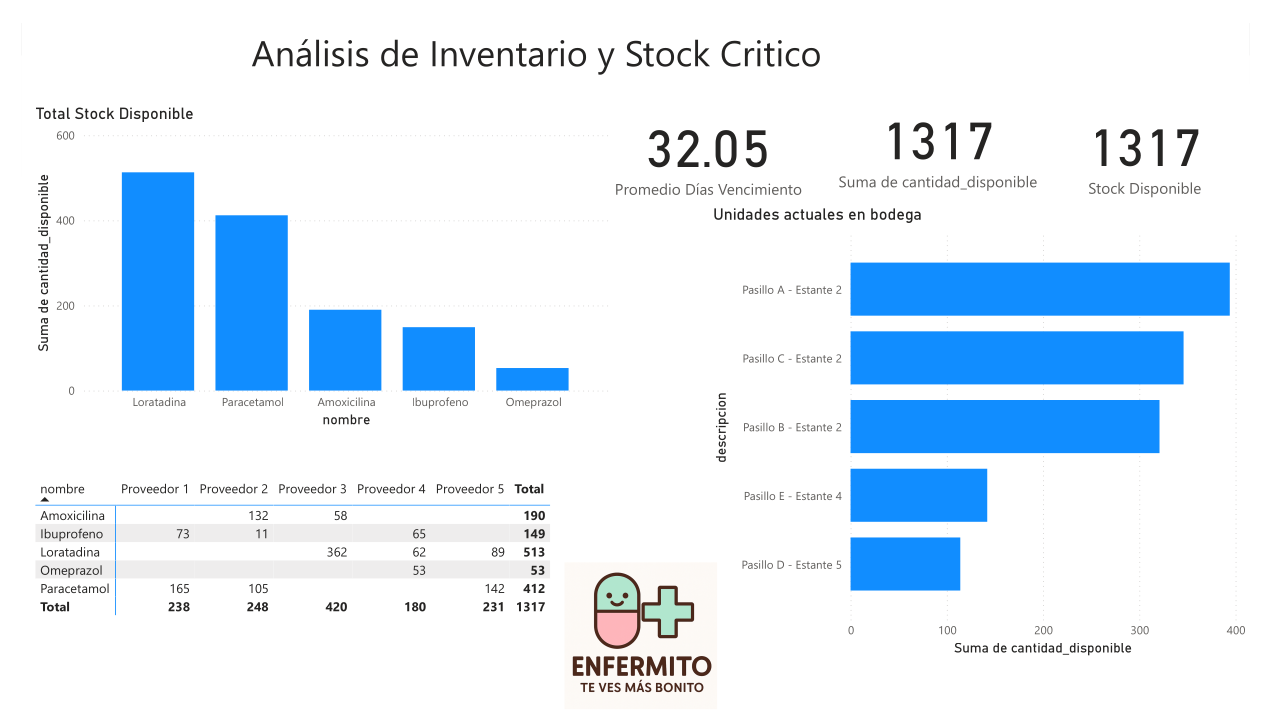
## **Dashboard de Análisis de Ventas**

Este informe presenta los indicadores clave de desempeño del área de ventas, donde se van a incluir el total vendido, los descuentos aplicados, el ticket promedio y el rendimiento por los productos. El gráfico en la parte inferior destaca los cinco productos más vendidos en el periodo que se anda analizando.

## **Dashboard de Pedidos**

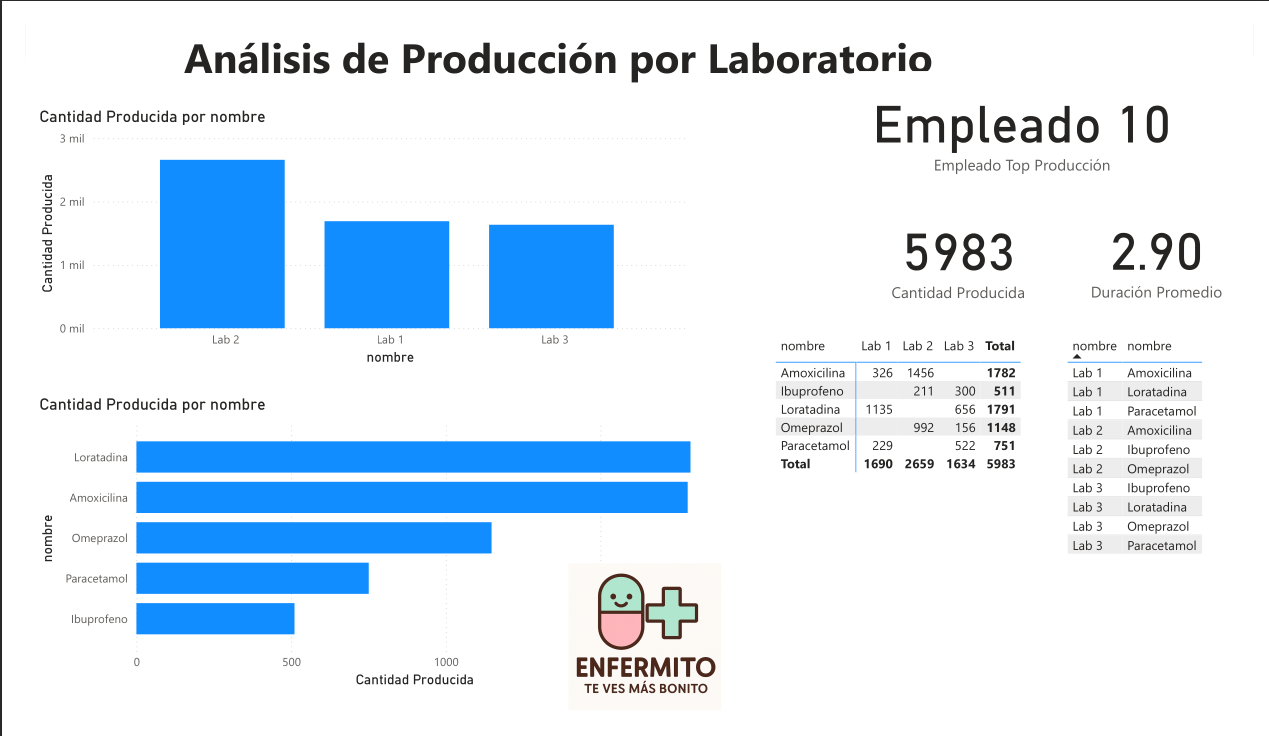
Este informe muestra el comportamiento de los pedidos realizados por las sucursales junto con la evaluación de las cantidades solicitadas y entregadas, y el nivel de cumplimiento y la urgencia de los pedidos. También se visualizan las sucursales con más solicitudes y los productos más requeridos.

## **Dashboard Inventario y Stock Crítico**

Este informe nos presenta los indicadores clave para mostrar el estado actual del inventario, poder identificar productos con bajo stock o con stock por vencerse junto con el análisis de la ocupación en bodega.

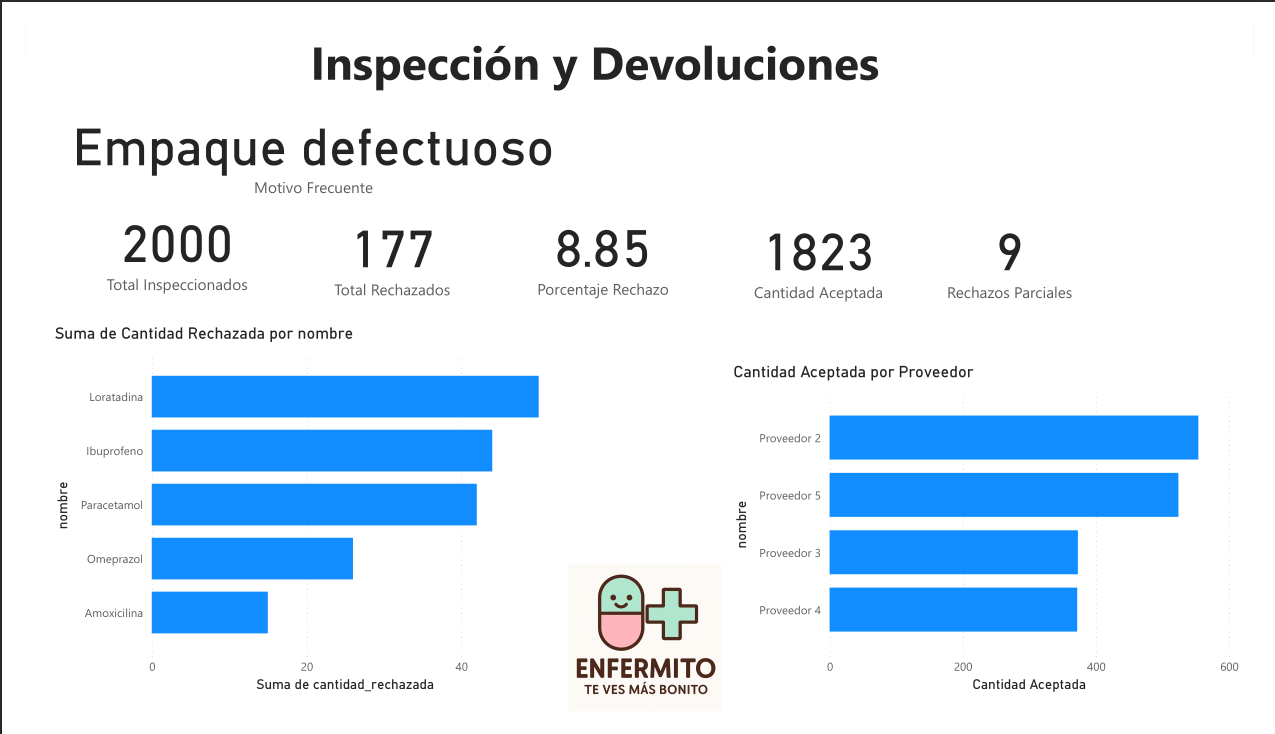
## **Dashboard de Producción**

Este informe permite analizar las actividades productivas por laboratorio, mostrando la cantidad fabricada por producto, el tiempo promedio de producción y los empleados más activos mediante también la evaluación de la eficiencia del proceso por laboratorio y por mes.



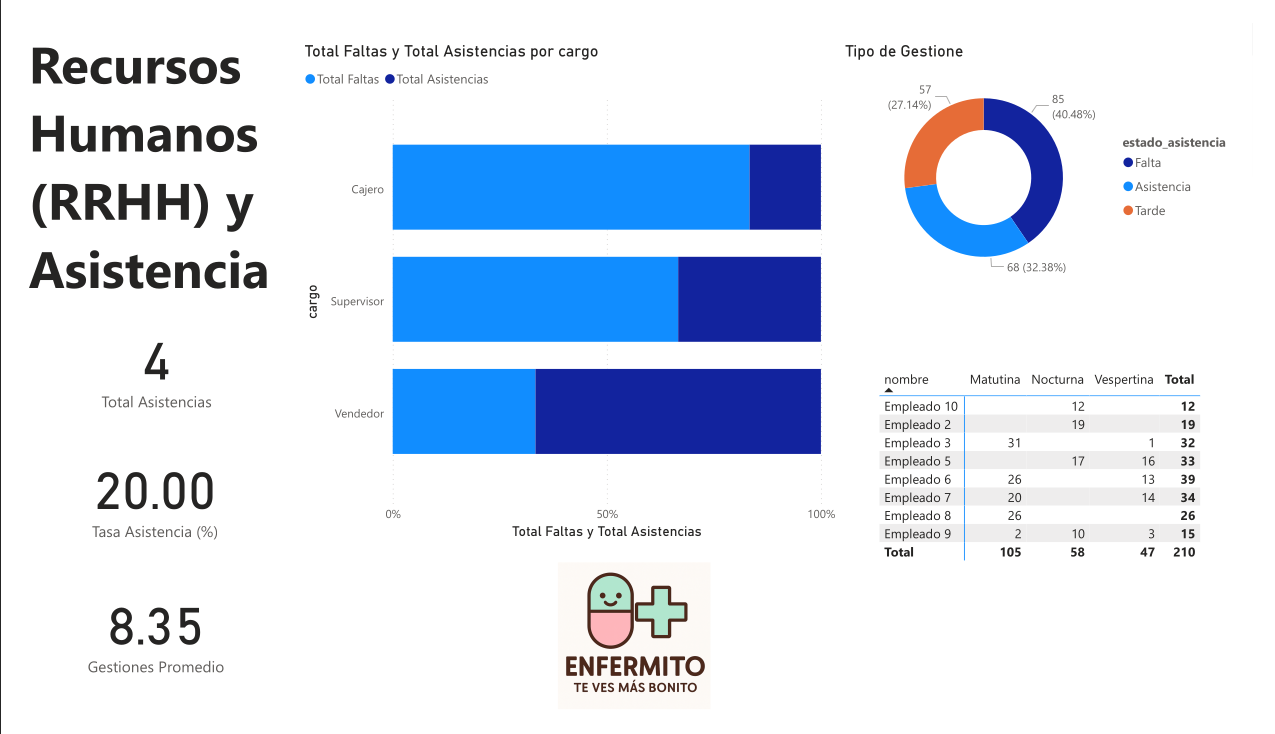
## **Dashboard de Inspección y Devoluciones**

Este informe permite poder presentar los resultados de las inspecciones realizadas a productos recibidos, destacando la cantidad aceptada y rechazada, el porcentaje de devoluciones y los motivos más frecuentes de devolución y se identifican los proveedores con mayores rechazos

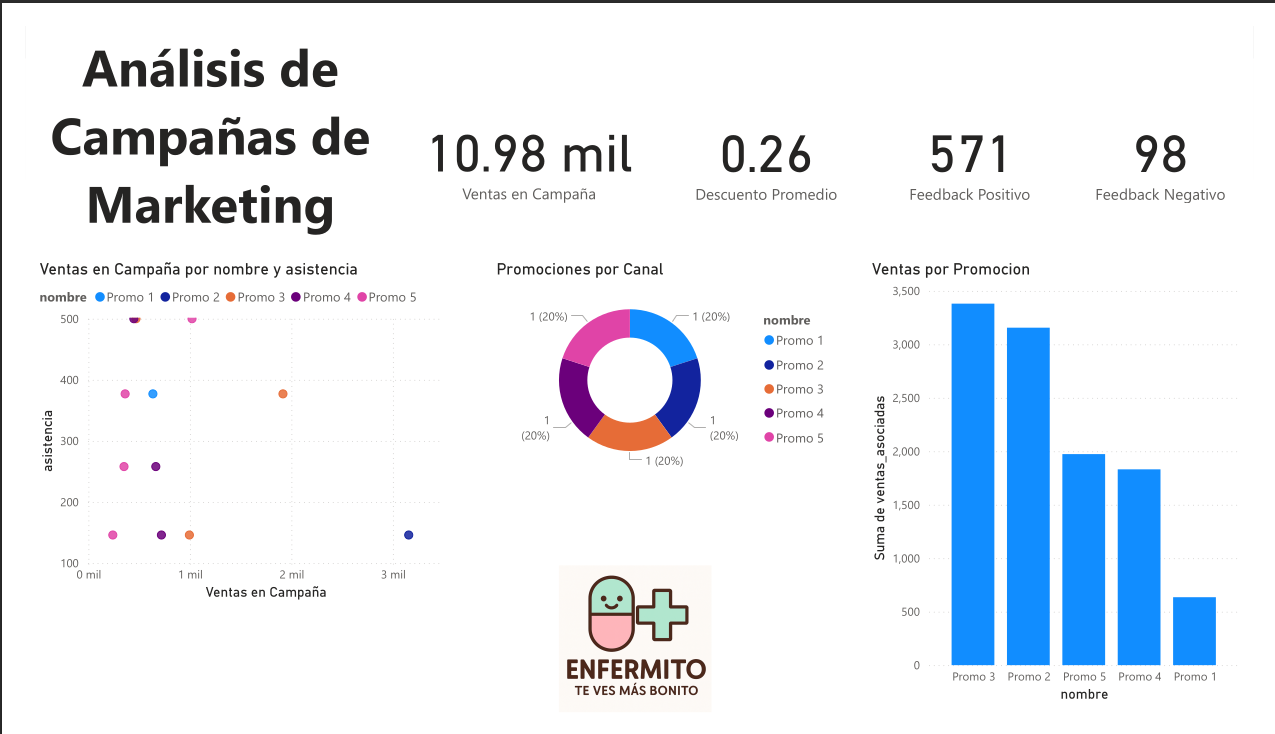


## **Dashboard de Recursos Humanos**

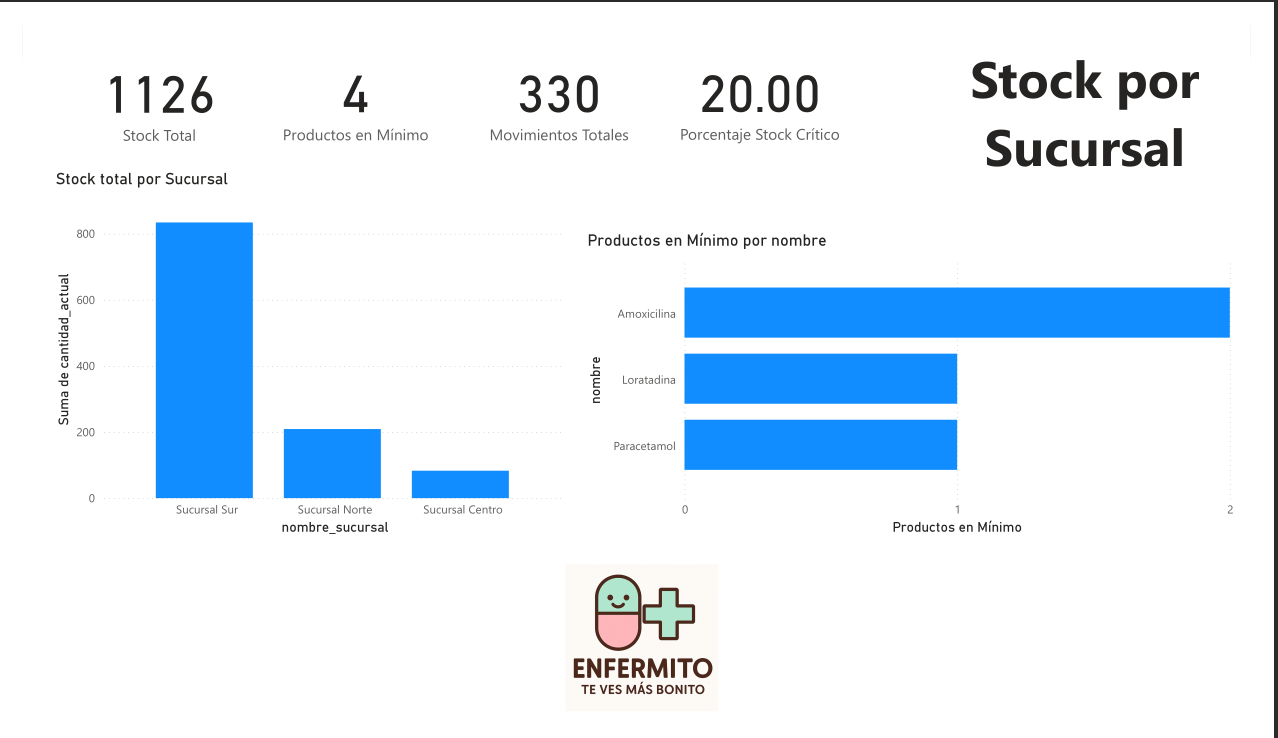
Este informe muestra el desempeño del personal en cuanto a asistencias, faltas y volúmenes de las gestiones diarias y se detallan los empleados con más ausencias, el comportamiento pro jornada y la evolución de asistencia en el tiempo



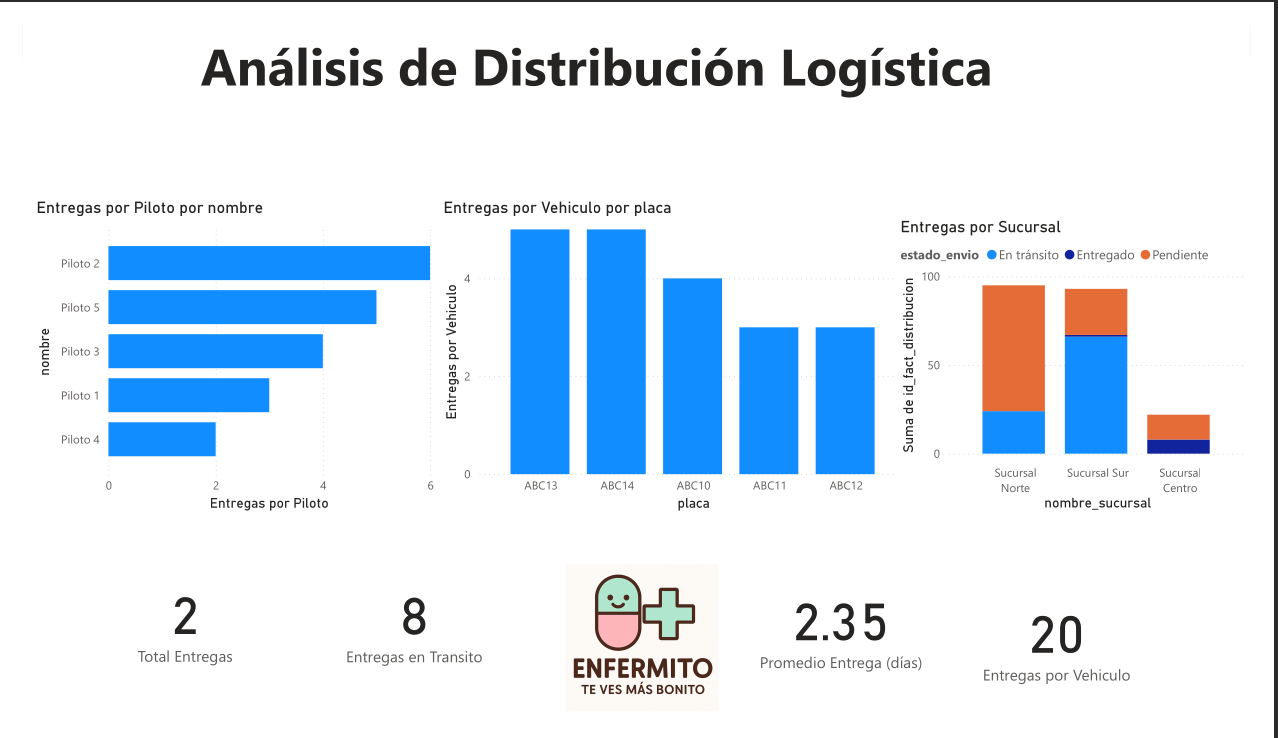
## **Dashboard de Marketing**

Este informe analiza el impacto de las campañas promocionales en las ventas, midiendo los productos vendidos durantes las promociones junto con las actividades realizadas, el descuento promedio aplicado y la retroalimentación positiva o negativa de los clientes y se destacan las promociones más exitosas

## **Dashboard de Stock por Sucursal**

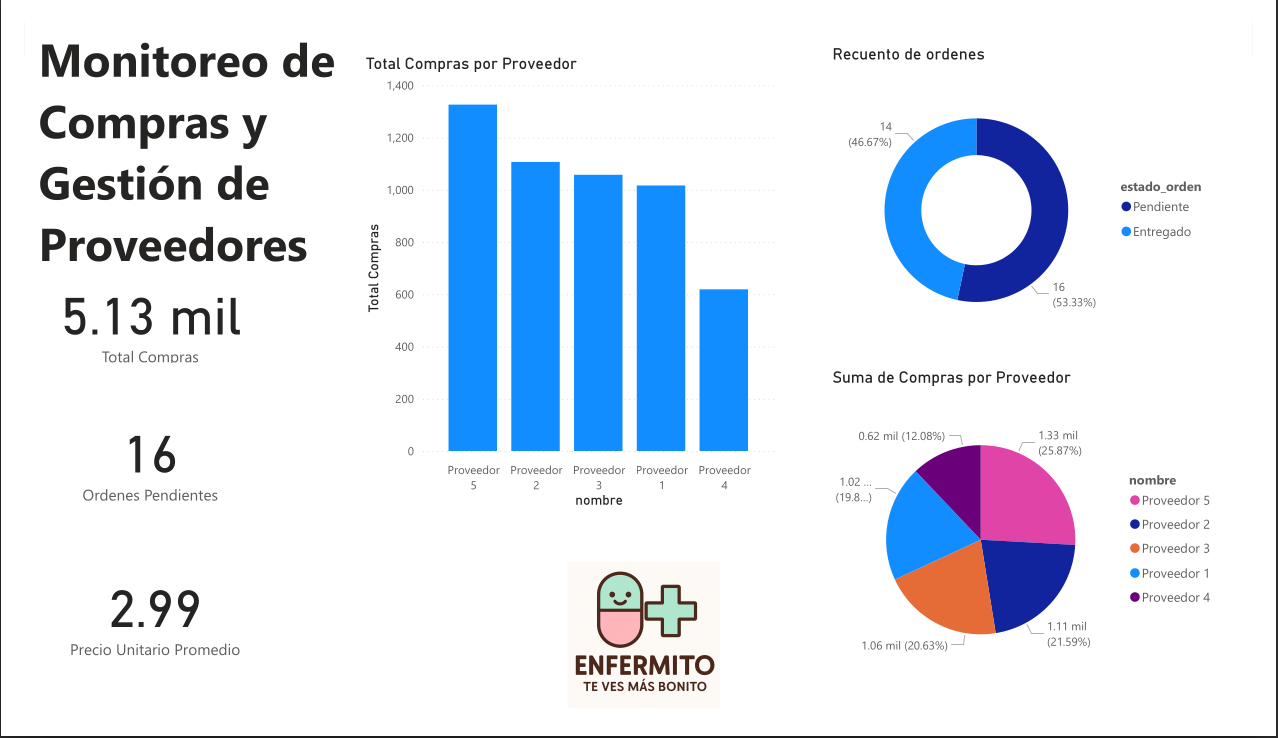
Este informe permite visualizar la cantidad de productos disponibles por sucursal, identificar aquellas con inventario por debajo del mínimo (establecido en 10) y analizando también los diferentes tipos de movimientos registrados (ingresos, salidas, ajustes)

## **Dashboard de Distribución Logística**

Este informe presenta el desempeño de la logística de la distribución desde la bodega central destacando la cantidad de entregas realizadas, el tiempo promedio de entrega y los envíos en tránsito y también se visualiza el rendimiento por piloto y por cada vehículo

## **Dashboard de Compras y Proveedores**

Este informe muestra el comportamiento de las compras realizadas a proveedores, incluyendo el total gastado, el número de ordenes, el precio promedio por unidad y los proveedores más frecuentes y también se identifican además las órdenes aún pendiente de entrega



Estos reportes fueron generados a partir de métricas basadas en lo que hemos estado integrando y trabajando, teniendo en cuenta que en esta fase ya tenemos los datos centralizados y una arquitectura de BI bastante desarrollada y los datos bastante maduros. Estos dashboards son de utilidad para poder tomar decisiones basadas en estos datos, como por ejemplo:

* Identificar los productos más vendidos para poder enfocarse en su promoción y stock
* Redefinir el mix de productos por sucursal
* Automatización de pedidos urgentes para poder reducir tiempos de espera
* Reubicar productos con sobrestock a otras sucursales
* Optimizar el uso de espacio en bodega ( para evitar expansiones innecesarias)
* Implementar las mejoras en procesos de inspección y empaque

Si lo dividimos, de acuerdo a cada Dashboard obtenemos:

1. **Dashboard de Ventas**
   1. Decisiones posibles:
      1. Focalizar esfuerzos de marketing en los productos más vendidos.
      2. Eliminar o reemplazar productos con bajo rendimiento.
      3. Establecer metas o bonificaciones basadas en desempeño por empleado.
      4. Optimizar el catálogo según lo más vendido por sucursal o región.
2. **Dashboard de Pedidos**
   1. Decisiones posibles:
      1. Establecer políticas para centralizar y calendarizar los pedidos por sucursal.
      2. Rediseñar rutas logísticas para optimizar entregas urgentes.
      3. Priorizar atención a sucursales con altos volúmenes o retrasos frecuentes.
      4. Automatizar la generación de pedidos según demanda histórica.
3. **Dashboard de Inventario y Stock Crítico**
   1. Decisiones posibles:
      1. Reubicar productos con bajo movimiento hacia sucursales con mayor demanda.
      2. Definir nuevos puntos de reorden para productos sensibles o próximos a vencer.
      3. Evaluar proveedores que entregan lotes con bajo rendimiento.
      4. Mejorar la planificación de compras y uso del espacio en bodega.
4. **Dashboard de Producción**
   1. Decisiones posibles:
      1. Redistribuir la producción entre laboratorios según eficiencia y capacidad.
      2. Aumentar producción de medicamentos con alta rotación en sucursales.
      3. Implementar indicadores de productividad por empleado y laboratorio.
      4. Prever tiempos reales de fabricación para planificar entregas internas.
5. **Dashboard de Inspección y Devoluciones**
   1. Decisiones posibles:
      1. Cambiar proveedores con altos índices de devoluciones.
      2. Identificar y eliminar productos con reincidencia en fallas de calidad.
      3. Diseñar controles adicionales para las categorías más afectadas.
      4. Fortalecer los criterios de inspección interna y seguimiento post-entrega.
6. **Dashboard de Recursos Humanos**
   1. Decisiones posibles:
      1. Ajustar turnos o jornadas con mayores niveles de ausentismo.
      2. Reconocer y retener empleados con mayor volumen de gestiones o asistencia.
      3. Detectar áreas críticas por inasistencias reiteradas.
      4. Mejorar la cobertura operativa en días o turnos de mayor carga.
7. **Dashboard de Marketing**
   1. Decisiones posibles:
      1. Reasignar presupuesto a campañas con mayor retorno en ventas o participación.
      2. Descontinuar promociones con poco impacto o feedback negativo.
      3. Replicar campañas exitosas por canal o región.
      4. Optimizar el calendario promocional con base en el comportamiento estacional.
8. **Dashboard de Stock por Sucursal**
   1. Decisiones posibles:
      1. Redistribuir inventario entre sucursales para evitar desabastecimiento.
      2. Mejorar abastecimiento preventivo en productos críticos.
      3. Identificar sucursales con alta rotación para priorizar logística.
      4. Analizar el tipo de movimiento más común para ajustar procedimientos.
9. **Dashboard de Distribución Logística**
   1. Decisiones posibles:
      1. Rediseñar rutas de distribución basadas en desempeño y distancia.
      2. Asignar pilotos y vehículos de forma más eficiente.
      3. Detectar causas de retrasos o entregas incompletas.
      4. Optimizar frecuencia de envíos según volumen y urgencia de pedidos.
10. **Dashboard de Compras y Proveedores**
    1. Decisiones posibles:
       1. Negociar precios y condiciones con proveedores más frecuentes.
       2. Priorizar la entrega de órdenes pendientes de mayor valor o urgencia.
       3. Analizar tendencias de precios para tomar decisiones presupuestarias.
       4. Evaluar desempeño de proveedores por cumplimiento y costo.

# **Fase de Operacionalización**

### **1. Mantenimiento y Actualización de la Solución BI**

Para garantizar el funcionamiento continuo y la utilidad de la solución de BI implementada, se establecen las siguientes estrategias de mantenimiento:

* **Automatización del ETL:** El proceso de Extracción, Transformación y Carga (ETL) será automatizado utilizando herramientas programables (como Python con pandas y sqlalchemy, o procesos de cronjobs) que se ejecuten de forma diaria o semanal.
* **Actualización de Datos:** Se definirá un calendario de carga de datos para integrar ventas, pedidos, compras, producción y RRHH. Los archivos de entrada (Excel, JSON) serán almacenados en una ubicación controlada y procesados según formato estándar.
* **Mantenimiento del Data Warehouse:** Se revisarán periódicamente las dimensiones y hechos para asegurar su integridad, se optimizarán índices y se eliminarán datos redundantes o inconsistentes.
* **Respaldo de datos:** Se realizarán copias de seguridad diarias o semanales del DW y de los scripts principales.
* **Revisión de dashboards:** Cada trimestre se validará que los indicadores siguen alineados con los objetivos de negocio y se adaptarán según nuevos requerimientos.

### **2. Roles de Mantenimiento y Gobierno de Datos**

Para sostener el sistema de BI, se identifican los siguientes roles clave como se mencionó anteriormente:

* **Administrador de BI:** Encargado de asegurar la disponibilidad y rendimiento del sistema, gestionar permisos y coordinar los procesos de carga y actualización de dashboards.
* **Ingeniero de Datos:** Responsable de los procesos ETL, calidad de datos y mantenimiento de la base de datos.
* **Analista de Datos:** Se encarga de crear nuevos dashboards, generar reportes estratégicos y adaptar indicadores según necesidades del negocio.
* **Responsable de Calidad de Datos:** Verifica que los datos provengan de fuentes confiables, auditables y con estándares de integridad.

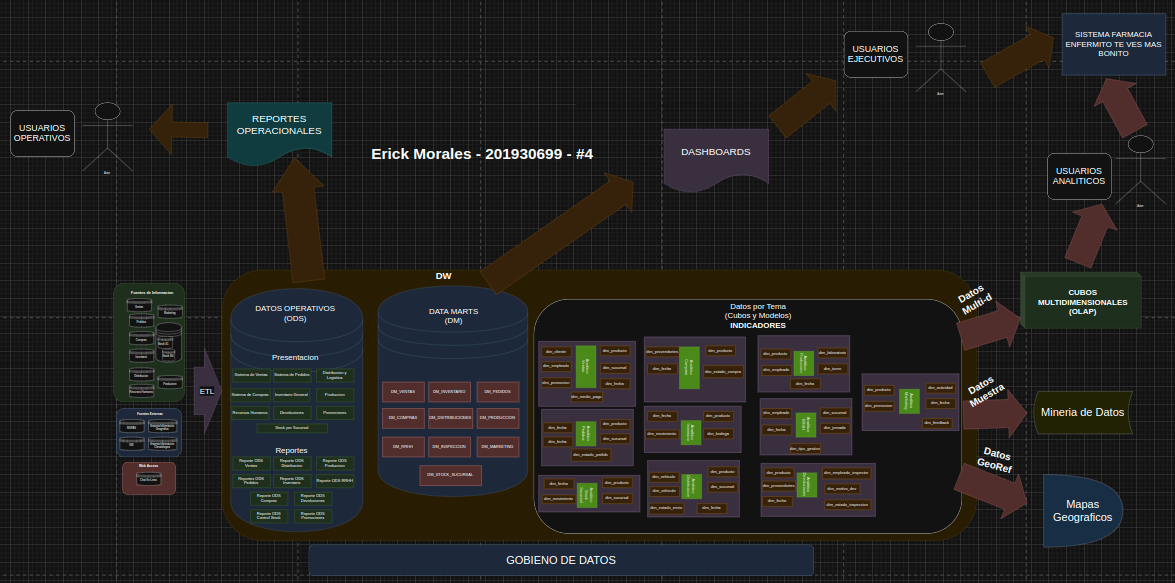
### **3. Estrategia de Escalabilidad**

Para garantizar que la solución pueda crecer con la empresa, se proponen las siguientes estrategias:

* **Adición de nuevas sucursales o laboratorios:** El modelo dimensional ya está normalizado para admitir nuevas sucursales y laboratorios sin modificar el diseño.
* **Nuevos indicadores:** Los dashboards y cubos OLAP están diseñados con flexibilidad para incorporar nuevas medidas, hechos y dimensiones.
* **Nuevas fuentes de datos:** Se pueden integrar otras fuentes (como sistemas ERP, sensores IoT, reportes MSPAS) mediante la misma arquitectura ETL ya establecida.
* **Migración futura a la nube:** La estructura puede trasladarse a una solución como BigQuery, Snowflake o Azure SQL si se desea escalar horizontalmente

# **Arquitectura general del sistema de BI**

Se presenta un gráfico que permite poder observar la arquitectura general de BI que se tendría al final el proceso



# **Conclusión**

La implementación de una arquitectura de BI (Business Intelligence o Inteligencia de negocios) en la empresa “Enfermito te ves más bonito” va a permitir resolver de una manera óptima e integral las problemáticas detectadas durante el diagnóstico de la empresa. Con la centralización de los datos, la integración de sistemas y el desarrollo de dashboards, esta empresa ahora cuenta con una plataforma robusta para poder realizar una toma de decisiones basadas en las evidencias de sus datos, cabe mencionar que entre los principales logros de la solución propuesta se encuentra:

* Reducción de los costos en su logística gracias al análisis de inventario, stock y distribución
* Mejora en la eficiencia de los laboratorios para poder alinear la producción con la demanda real
* Disminución del desabastecimiento mediante el análisis estacional y de pedidos
* Mayor control sobre el temas de las devoluciones e inspecciones gracias a la trazabilidad de los mismos
* Mejora en la productividad del recurso humano con los reportes de asistencia y gestiones de los mismos
* Unificación de la información proveniente de todas las sucursales del país

El sistema propuesto sigue siendo escalable, sostenible con el tiempo y puede seguir siendo mejorado constantemente a medida que la empresa siga madurando su cultura basada en datos.

**Situación Posterior a la implementación de la solución:**

Tras haber implementado nuestra solución de BI en la empresa vemos:

* Que la empresa cuenta con un Data Warehouse centralizado y estructurado con la información integrada
* La empresa a mejorado su capacidad de respuesta ante problemas operativos
* La empresa cuenta con dashboards para monitorear en tiempo real las ventas, inventario, distribución, producción, marketplace y RRHH.
* La empresa puede planificar campañas, producción, compras y distribución basado en los datos históricos
* La empresa dispone de una base sólida para escalar su operación e incluso conectar con datos de fuentes externas

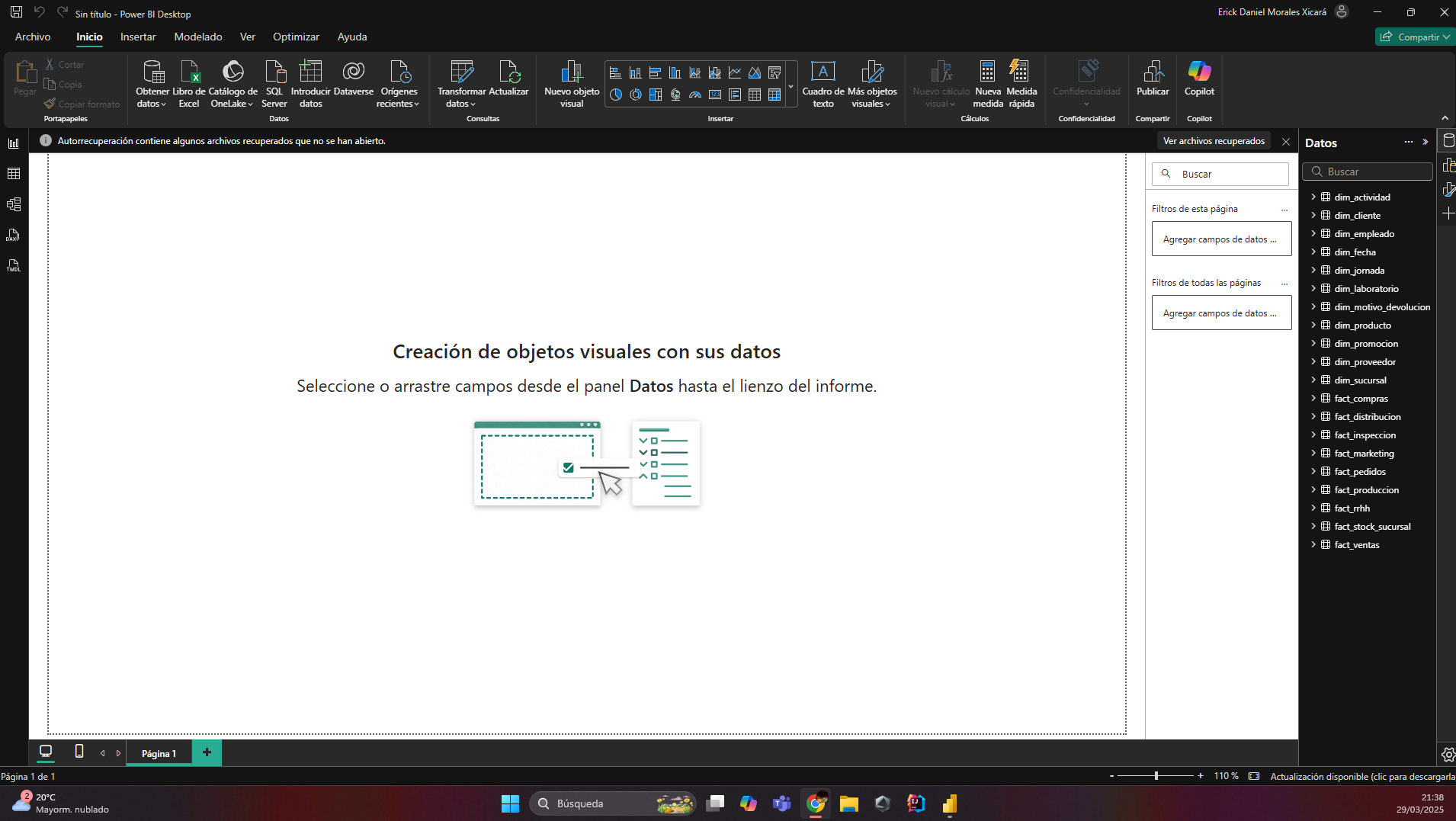
Y como respuesta final, **SE RECOMIENDA** ubicar una nueva bodega cercana a la región Norte y Sur dado que con esto se estará priorizando aquellas sucursales que tengan

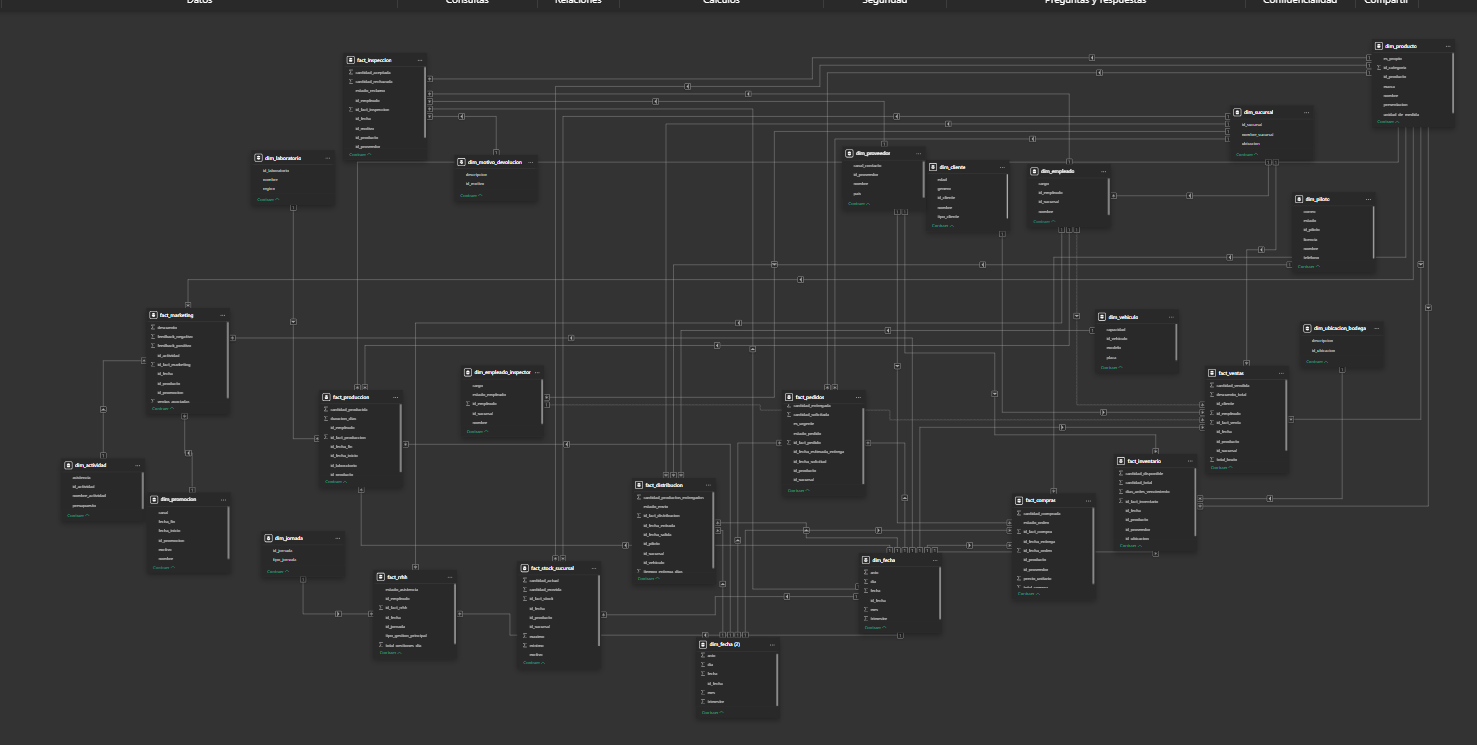
* Mayor cantidad de pedidos pendientes
* Mayor frecuencia de entregas en tránsito
* Dificultades visibles en cumplimiento logístico

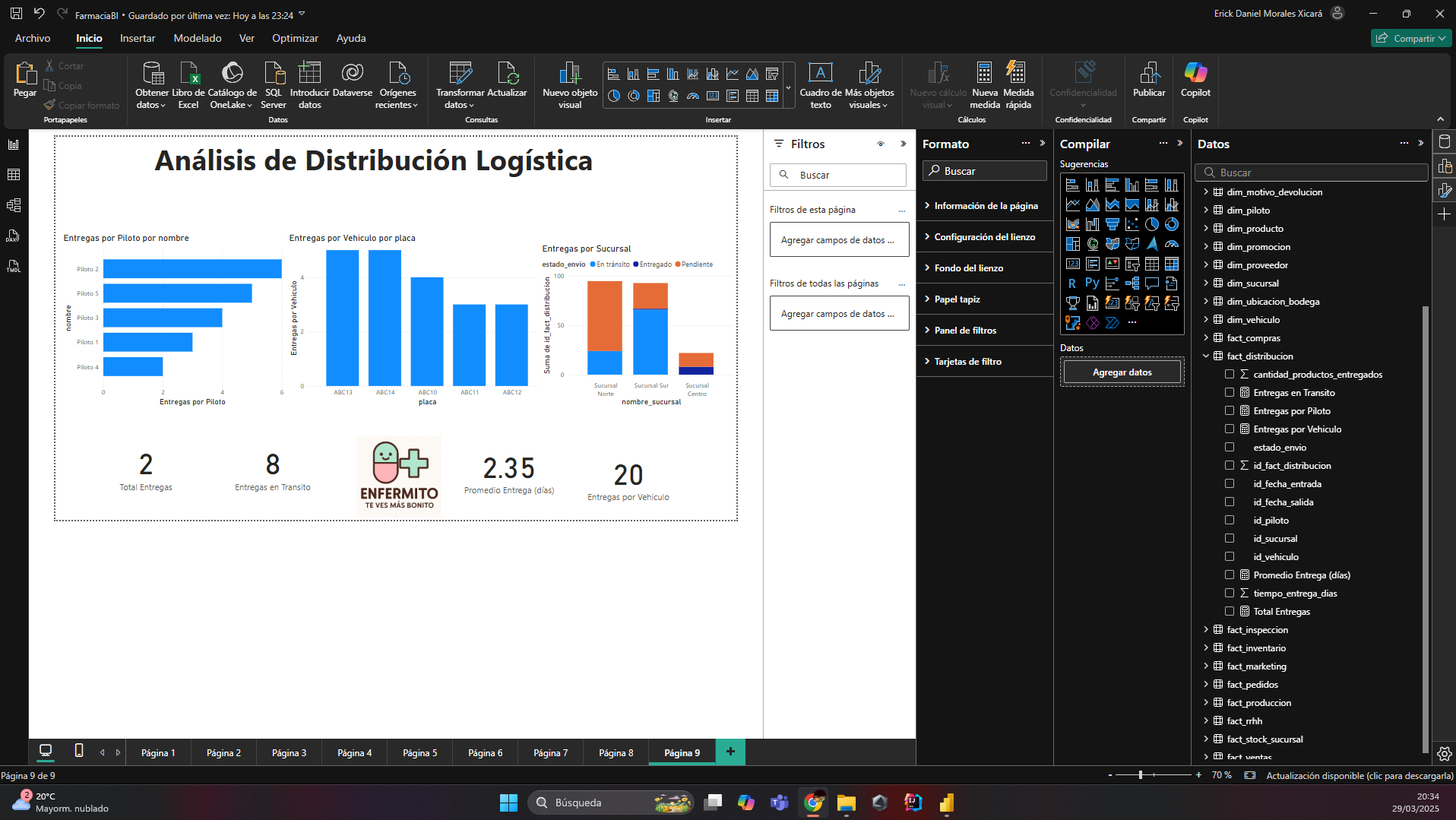
Cabe recalcar que esto puede ir cambiando conforme se siga haciendo la carga de los datos pero los criterios para tomar la decisión se basa en las tablas:

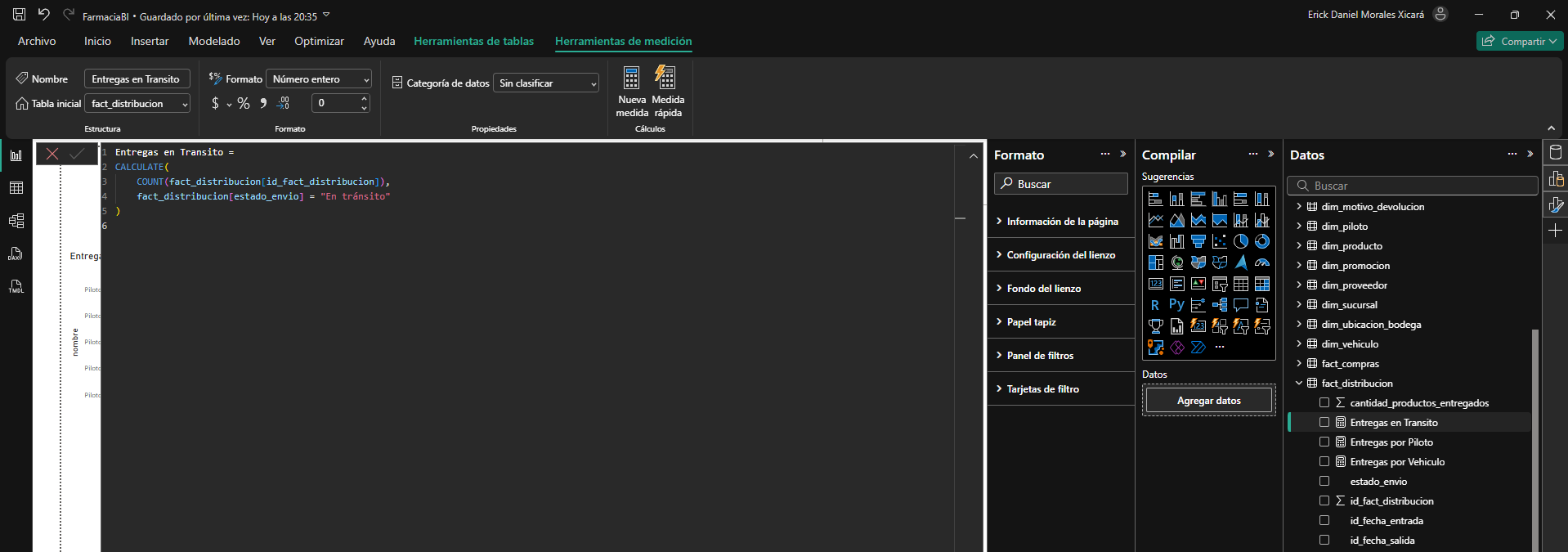
* fact\_distribucion
  + **Indicador:** Total de entregas realizadas por sucursal.
  + **Justificación:** Una sucursal que recibe muchos productos desde la bodega central podría beneficiarse si se reduce la distancia de despacho.
* fact\_distribucion
  + **Indicador:** Cantidad de entregas no finalizadas por sucursal.
  + **Justificación:** Muestra saturación en el canal logístico actual, indicando la necesidad de descongestionar esa ruta.
* fact\_distribucion
  + **Indicador:** Tiempo medio de entrega por sucursal o región.
  + **Justificación:** Tiempos altos pueden deberse a distancias largas o rutas ineficientes; abrir una bodega puede acortar este tiempo.
* fact\_pedidos
  + **Indicador:** Volumen de productos solicitados por sucursal.
  + **Justificación:** Las sucursales con más demanda podrían recibir productos desde una bodega más cercana.
* fact\_stock\_sucursal
  + **Indicador:** Frecuencia de quiebres de stock por sucursal.
  + **Justificación:** Si una sucursal está constantemente desabastecida, necesita una cadena de suministro más rápida y directa.
* dim\_sucursal
  + **Campo**: ubicación o región (si lo tienes cargado)
  + **Justificación**: Permite segmentar la operación por zonas geográficas para optimizar cobertura.

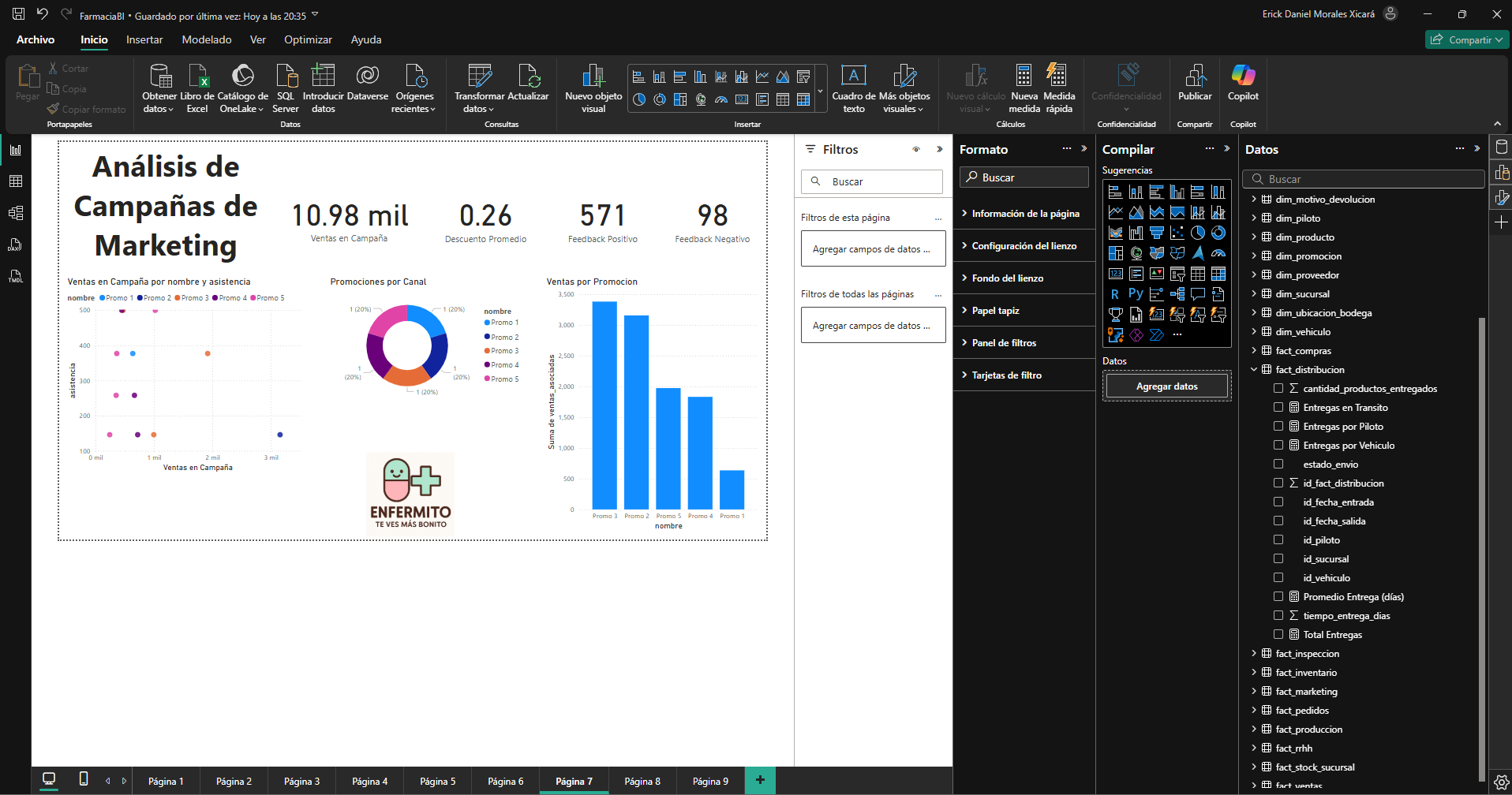
# **Anexos**











# **Bibliografía**

* <https://app.powerbi.com/home?experience=power-bi>
* <https://srlabs.a2hosted.com/rs-humhub/index.php?r=wiki%2Fpage%2Fview&title=5.+Arquitectura+de+una+soluci%C3%B3n+de+Business+Intelligence&cguid=a9423a44-8ee7-4719-9034-48f8bf104ccd>
* <https://srlabs.a2hosted.com/rs-humhub/index.php?r=wiki%2Fpage%2Fview&title=4.+El+ciclo+de+vida+de+los+datos&cguid=a9423a44-8ee7-4719-9034-48f8bf104ccd>
* <https://learn.microsoft.com/en-us/dax/>