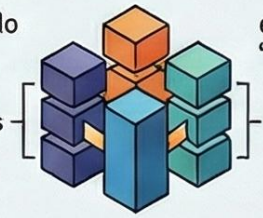
CONCEPTOS FUNDAMENTALES DE BUSINESS INTELLIGENCE

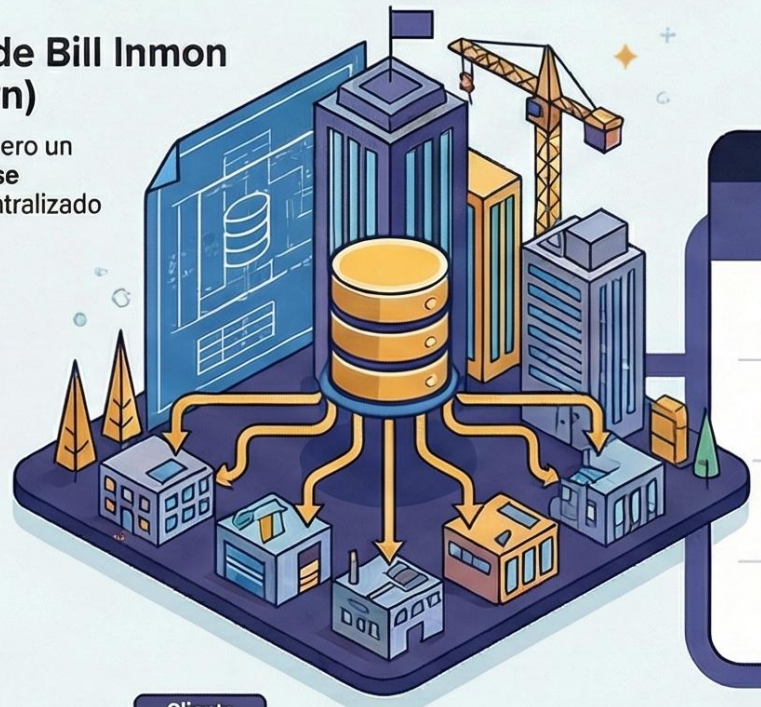
Claves del Diseño de un Data Warehouse

Un **Data Warehouse (DW)** es un repositorio centralizado para análisis y decisiones, basado en **"Hechos"** en **"Hechos"** (mediciones, e.g., ventas) y **"Dimensiones"** (contexto, e.g., quién, dónde, cuándo).



Enfoque de Bill Inmon (Top-Down)

Construye primero un **Data Warehouse** corporativo centralizado y normalizado.

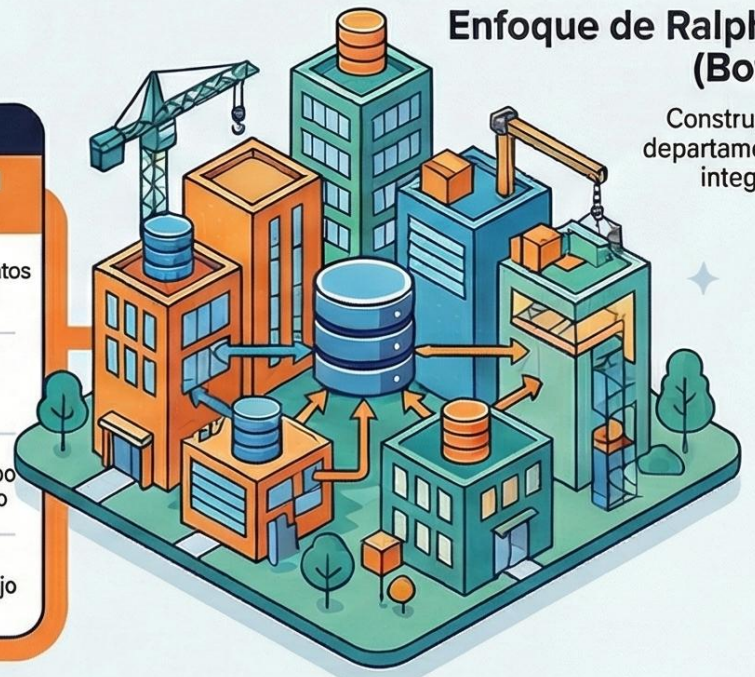


COMPARACIÓN DE FACTORES CLAVE

Enfoque Inmon (Top-Down)		Enfoque Kimball (Bottom-Up)
Toda la compañía	Alcance	Departamentos individuales
Alto	Coste Inicial	Bajo
Mayor tiempo de desarrollo	Plazos	Menor tiempo de desarrollo
Más fácil	Mantenimiento	Más complejo

Enfoque de Ralph Kimball (Bottom-Up)

Construye **Data Marts** departamentales que se integran en un DW.



Modelos de Datos: ¿Cómo estructurarlo?

Modelo en Estrella

Una tabla de hechos central conectada directamente a tablas de dimensiones desnormalizadas.

Key Advantage: Consultas más simples y rápidas, fácilmente entendible por los usuarios de negocio.

Modelo en Copo de Nieve

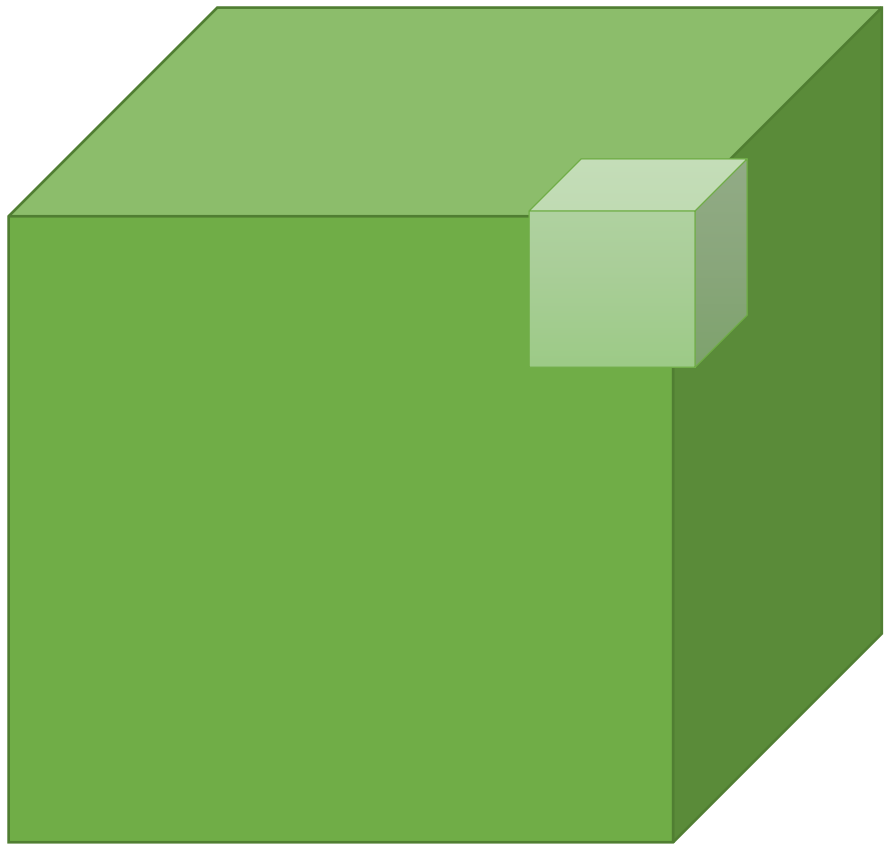
Similar al estrella, pero las tablas de dimensiones están normalizadas en varias tablas.

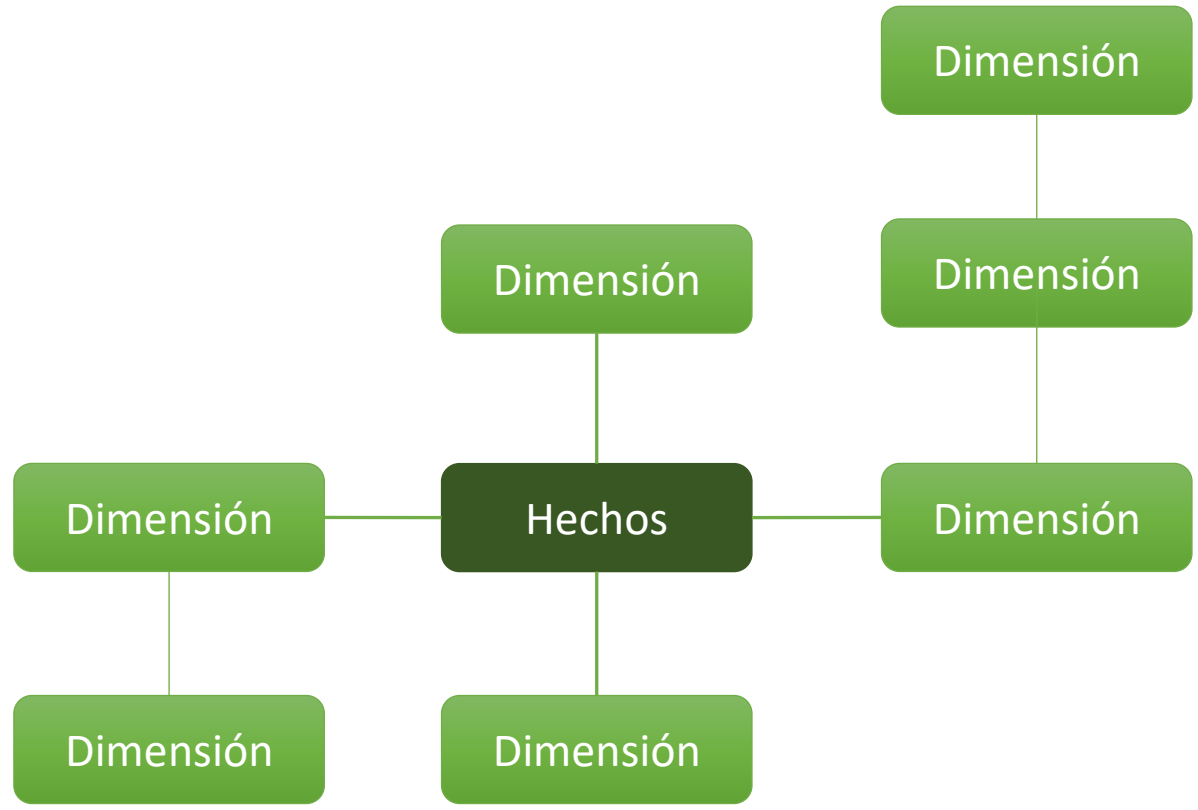
Key Advantage: Ahorra espacio de almacenamiento, ideal para dimensiones moy grandes y complejas.

¿Cuál elegir?

El modelo **en Estrella** es ideal para Data Marts y análisis simple; **Copo de Nieve** para DW corporativos donde el espacio es crítico.

CONCEPTOS FUNDAMENTALES DE BUSINESS INTELLIGENCE





Definición Data Mart / Data Warehouse

Diferentes definiciones…

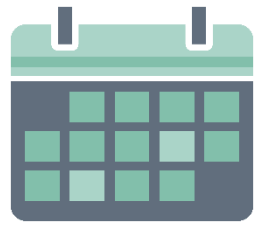
- **Repositorio unificado** para todos los datos que recogen los **diversos sistemas** de una empresa. El repositorio puede ser físico o lógico y hace hincapié en la captura de datos de diversas fuentes sobre todo para fines analíticos y de acceso.
- **Subconjuntos de datos** con el propósito de ayudar a que un área específica dentro del negocio pueda tomar mejores decisiones.
- **Una copia de los datos** de la transacción estructurados **específicamente** para preguntar y divulgar.
- **Proceso que integra información** de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de perspectivas y con grandes velocidades de respuesta.

Características de un Data Warehouse

- **Orientados a tema:** los datos Los datos que se analizan se organizan por departamentos, áreas o procesos del negocio que se quieren mejorar**.**
- **Integrados:** datos procedentes de diferentes fuentes de origen.
- **Variables en el tiempo:** datos relativos a un periodo que se incrementan periódicamente.
- **No volátiles:** datos almacenados que se añaden, no se actualizan ni se modifican.
- **Diseño tolerante al cambio.**
- Permite la **extracción y carga de datos de forma masiva**.

Conceptos Importantes

- Dentro del entorno de un **Data Warehouse** hay dos conceptos fundamentales que es importante entender antes de poder continuar.



  - **Dimensiones**
    - Representan aquellos conceptos desde los que se analizan los hechos y responden a la pregunta **¿QUIEN, DONDE, CUANDO?**
    - Algunas dimensiones habituales son la Fecha, País, Ciudad, Cliente…

  - **Hechos**
    - Representan aquello que se quiere medir y responden a la pregunta del **¿QUÉ?**
    - Entre los hechos mas típicos se encuentran datos como Ventas.

Dimensiones

Con respecto a las dimensiones:

- Contienen los descriptores textuales de los hechos.
- Cada una de las dimensiones esta contenida en una tabla diferente.
- La tendencia de crecimiento de estas tablas es a lo ancho, es decir, lo habitual es que se añada información adicional relacionada con la dimensión que ayude a filtrar los hechos con mayor detalle.

Mes	Clave Mes
Enero	1
Febrero	2
Marzo	3
...	...

Producto	Clave Producto
Libro	1
USB	2
Portatil	3
...	...

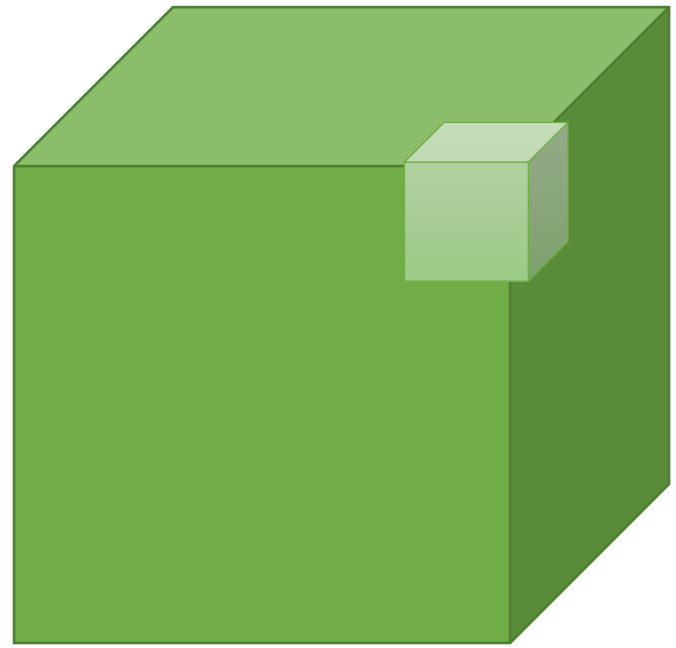
Localizacion	Clave Localizacion
Madrid	1
Barcelona	2
Sevilla	3
...	...

Granularidad

- La granularidad establece exactamente lo que **representa una sola fila de tabla de hechos**.
- Un ejemplo muy claro es el que ofrece la dimensión de tiempo. En este caso lo que hay que definir es si el análisis será a nivel de Año, Mes, Día o incluso inferior.
  - Declarar el **grano** es el paso fundamental en un diseño dimensional, es decir, se debe definir claramente hasta que punto se quiere profundizar en el análisis.

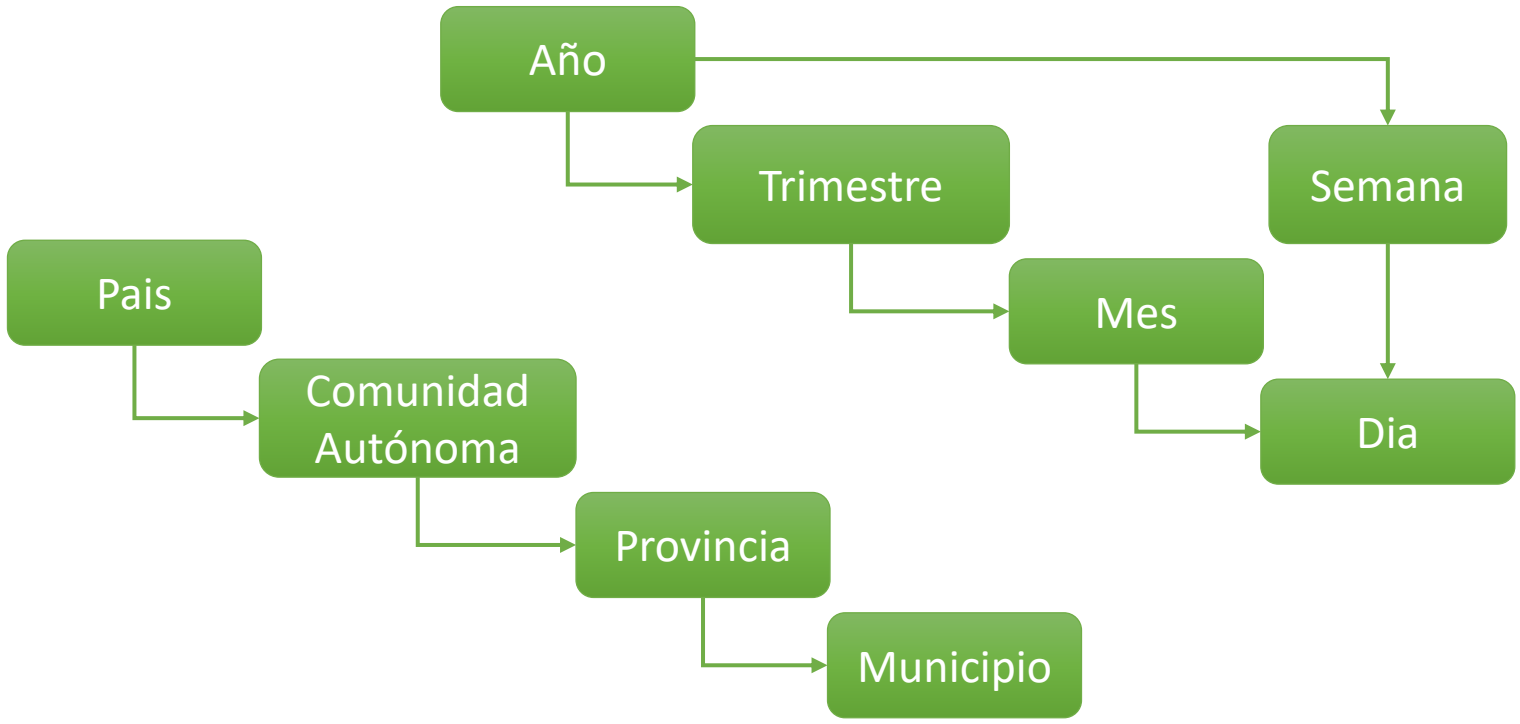
Granularidad

- El grano debe declararse antes de elegir las dimensiones o hechos porque cada dimensión o hecho candidato debe ser consistente con el grano.
- Esta **consistencia** impone una uniformidad en todos los diseños dimensionales que es **crítica** para el **rendimiento** de la aplicación BI y la **facilidad de uso.**
  - Cada grano de tabla de hechos propuesto da como resultado una tabla física separada; **diferentes granos no deben mezclarse en la misma tabla de hechos**.



Granuralidad. Jerarquías

- Las dimensiones puede contener varios puntos de vista: Jerarquías
- Una **jerarquía** esta compuesta de niveles que contienen miembros. Cada nivel tendrá una columna.
- Hay que formularse **¿A qué nivel se tiene la información en origen?** y **¿Hasta dónde queremos llegar?**



Año	Mes	Día
2008	Enero	1
2008	Enero	2
2008	Febrero	1

Dimensiones: Gestión de claves

Cada tabla de dimensiones tiene una sola columna de clave principal.

- Esta clave primaria **no** puede ser la **clave natural** del sistema transaccional (origen) porque habrá múltiples filas de dimensión para esa clave natural cuando se rastreen los cambios a lo largo del tiempo.
- Además, las claves naturales para una dimensión pueden ser creadas por más de un sistema fuente, y estas claves naturales pueden ser incompatibles o mal administradas.
- Las claves naturales creadas por los sistemas fuente operacionales están sujetas a reglas comerciales fuera del control del sistema DW / BI.
- Cuando el DW quiere tener una sola clave para ese empleado, se debe crear una nueva clave duradera que sea persistente y no cambie en esta situación.

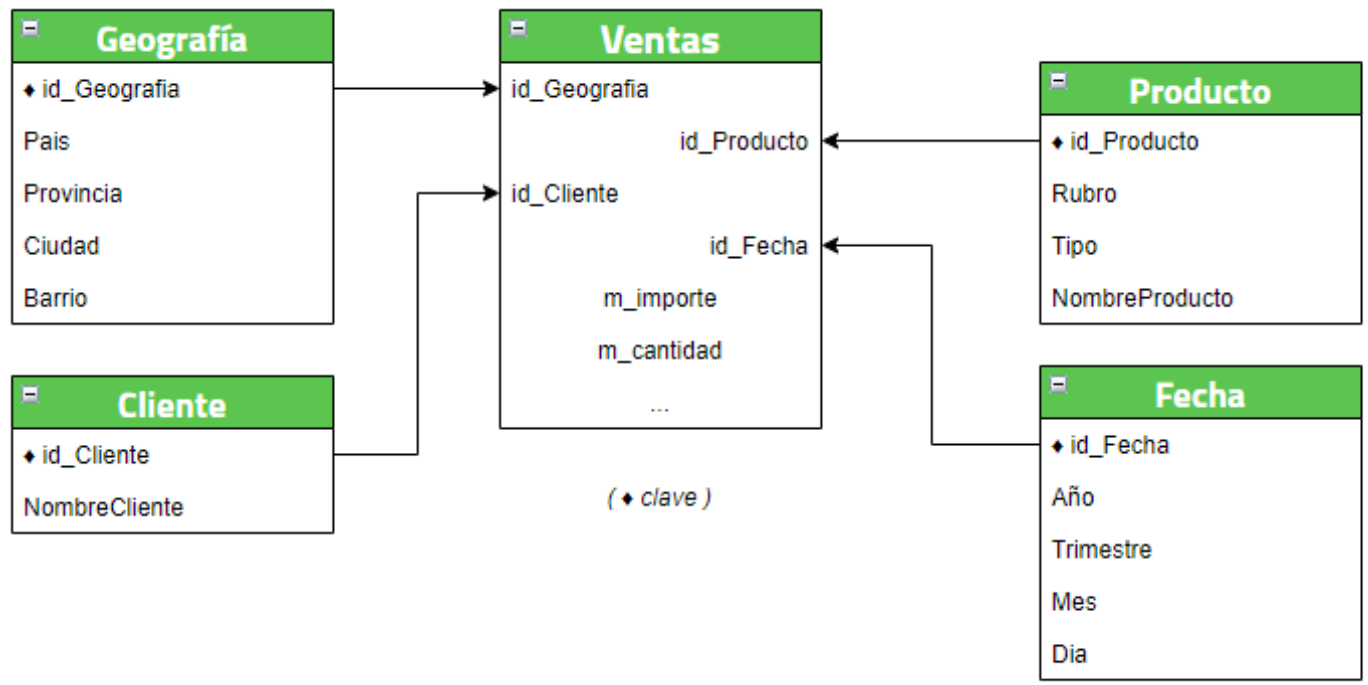
Dimensiones: Gestión de claves

El sistema DW en lugar de utilizar las claves naturales procedentes del sistema transaccional debe crear **claves sustitutas (subrrogadas o surrogated keys).**

- Estas claves sustitutas de dimensión son **números**, asignados en secuencia,
- La dimensión de fecha está exenta de la regla clave sustituta; esta dimensión altamente predecible y estable puede usar una clave principal más significativa.

Dimensiones: Estructura de las tablas

- Las **claves primarias** están incrustadas como una **clave externa en cualquier tabla de hechos asociada.**
- Las tablas de dimensiones suelen ser anchas, tablas desnormalizadas con muchos atributos de texto de baja cardinalidad.
- Los atributos de dimensión se completan con descripciones detalladas.



Hechos

Con respecto a los hechos:

- Almacena datos números e indicadores clave (KPIs).
- Es la tabla central en un modelo multidimensional.
- Los datos se encuentran en un nivel de detalle (grano) determinado y uniforme.
- La tendencia de crecimiento de estas tablas es a lo alto, es decir, no se añaden indicadores nuevos sino sucesos nuevos.

Clave Mes	Clave Producto	Clave Localización	…	Euros	Unidades
2	1	3	…	300	5
3	2	1	…	1000	7
1	3	1	…	227	8
…	…		…	…	…

Consideraciones

Con respecto a las dimensiones:

- Cuantos mas atributos (columnas) tenga una dimensión, mayor es la amplitud en el análisis que podemos realizar.

Otras consideraciones:

- Cuándo algo es un hecho y no una dimensión? (edad, peso,…)
  - ¿Participa en cálculos?
  - ¿Participa en restricciones?
- **Tiempo y Fecha**
  - ¿Cada cuanto se mide o se quiere analizar?
- Definición clara de la **Granularidad** de los datos.

Con respecto a los hechos:

- **Error típico:** Replicar el modelo operacional (¡Quiero tenerlo todo!)
- Hechos calculados directamente en el DW

vs.

- Hechos calculados al vuelo

Comparativa entre sistemas OLTP y DW

Transaccional (OLTP)

- Orientado a operaciones
- Orientados al negocio
- Datos detallados
- Datos aislados
- Lectura /Escritura
- Sin redundancia: normalizada (3FN)

Data Warehouse

- Orientado a tema (Departamentos)
- Analizar el negocio: Toma de decisiones
- Datos resumidos y agregados
- Lectura, consultas complejas
- Se prima la rapidez al tamaño

Comparativa entre sistemas

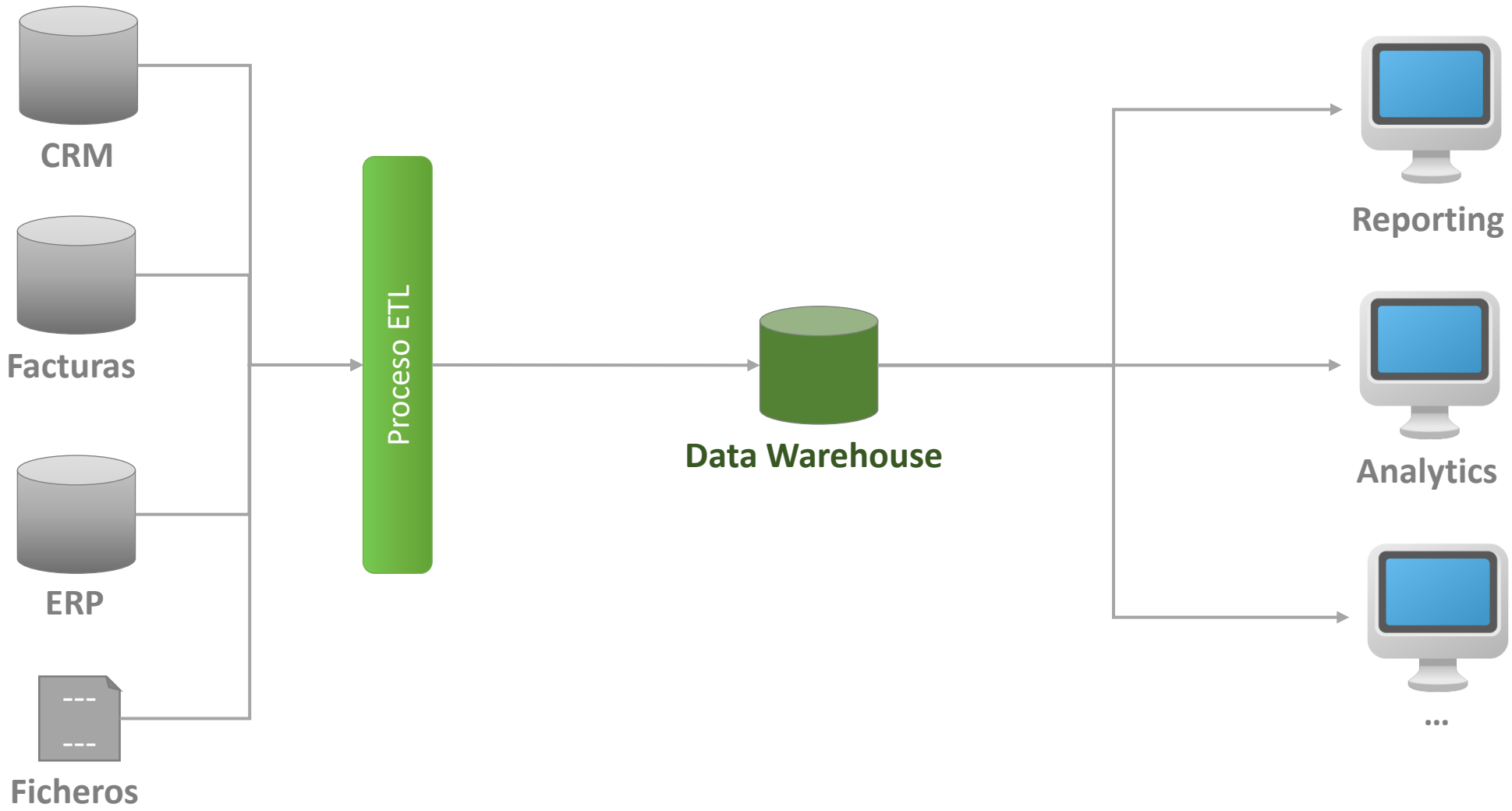
Transaccional (OLTP)

- Datos actuales
- Consultas predefinidas
- Nivel detalle
- Alta, baja, modificación y consulta
- Orientado a procesos

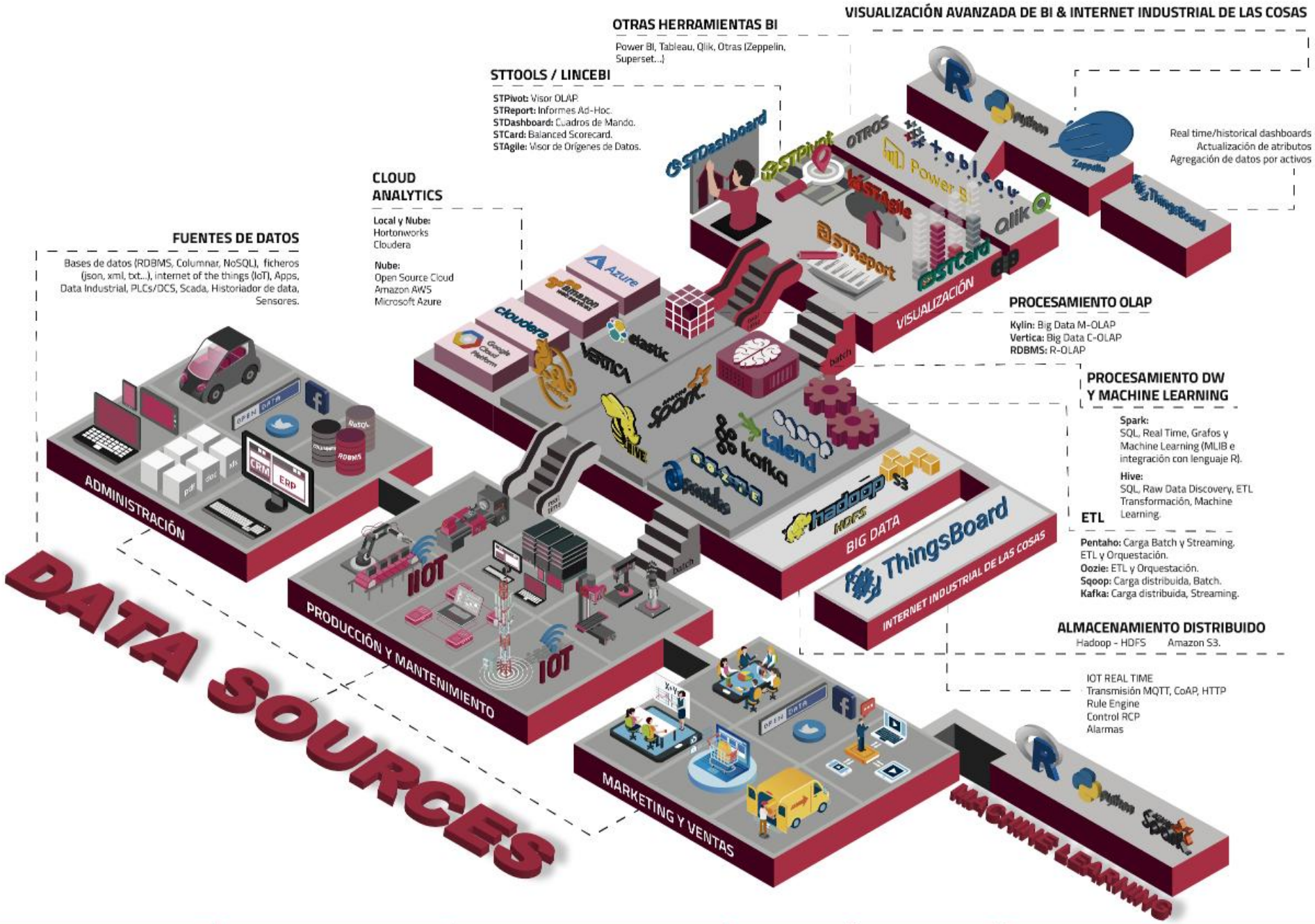
Data Warehouse

- Datos actuales e históricos
- Consultas dinámicas
- Nivel detalle y sumarizado
- Carga y consulta
- Orientado al análisis

Diseño Básico Solucion BI



Diseño Complejo Solucion BI



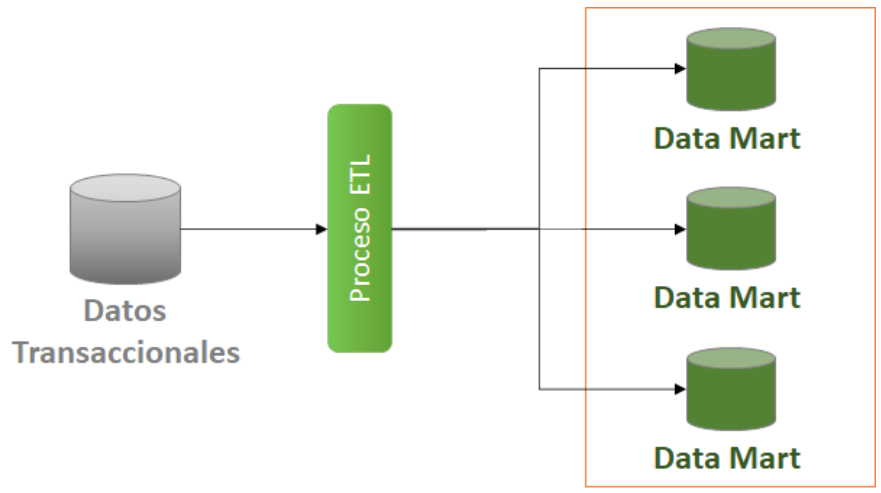
INTELIGENCIA ARTIFICIAL	INTERNET INDUSTRIAL DE LAS COSAS	ROBÓTICA AUTOMATIZADA DE PROCESOS	BIG DATA	MACHINE LEARNING	BUSINESS INTELLIGENCE

Kimball - Data Warehouse

Modelo de **Ralph Kimball**.

El Data Warehouse es un conglomerado de todos los Data Marts dentro de una empresa

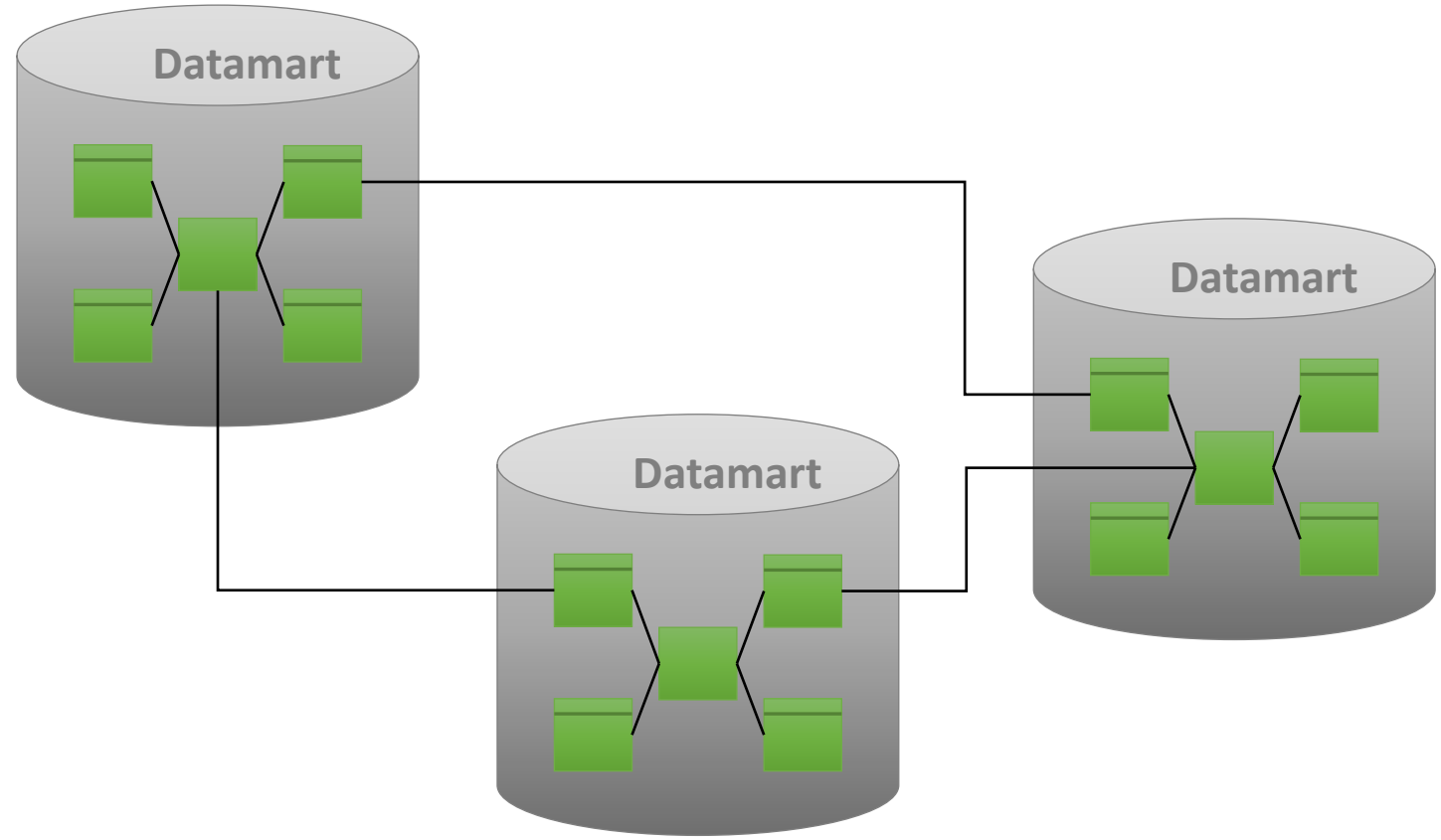
1. Basado en datos transaccionales.
2. Estructurados de una forma especial para el análisis.
3. De acuerdo al Modelo Dimensional (no normalizado), que incluye, las dimensiones de análisis y sus atributos, su organización jerárquica, así como los diferentes hechos de negocio que se quieren analizar.
4. **Orientado a departamento.**
5. **Difícil de integrar** los distintos Data Marts.
6. **Datos sumarizados por negocio.**
7. **No** es **tolerante** a **cambio.**



Kimball - Data Warehouse

Arquitectura de Bus Empresarial de Data Warehouse

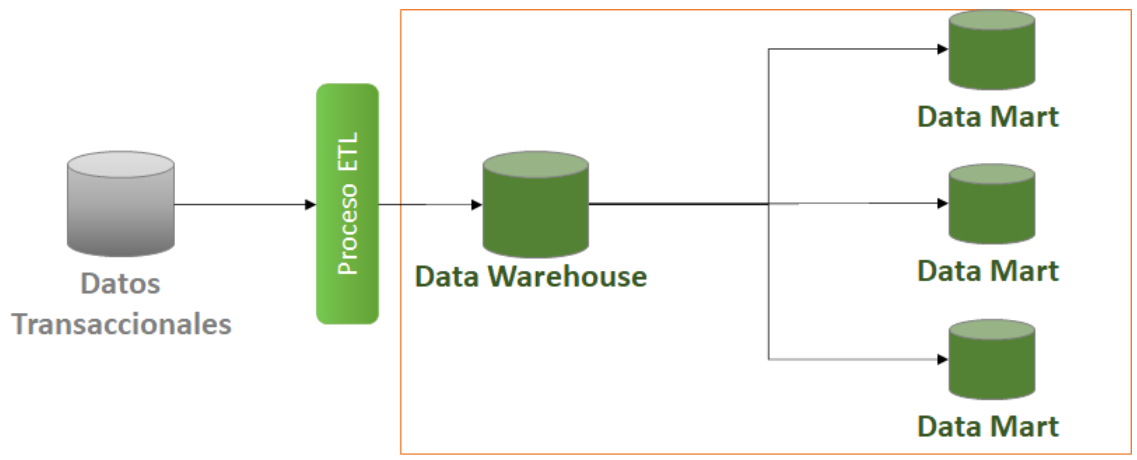
- Modelo dimensional.
- Datos atómicos y agregados.
- Uso de dimensiones conformadas.
- Diseñado para almacenar datos permitiendo su explotación y análisis.



Bill Immon - Data Warehouse

Modelo de Bill **Immon**.

- Transferir la información de los diferentes OLTP (Sistemas Transaccionales) de las organizaciones a un lugar centralizado donde los datos puedan ser utilizados para el análisis.
- La información ha de estar a los máximos niveles de detalle.
- Los DWs departamentales (datamarts) son tratados como **subconjuntos** de este **DW** corporativo
- Cubren las necesidades individuales de análisis de cada departamento
- Siempre a partir de este **DW Central**, del que también se pueden construir los ODS (Operational Data Stores).

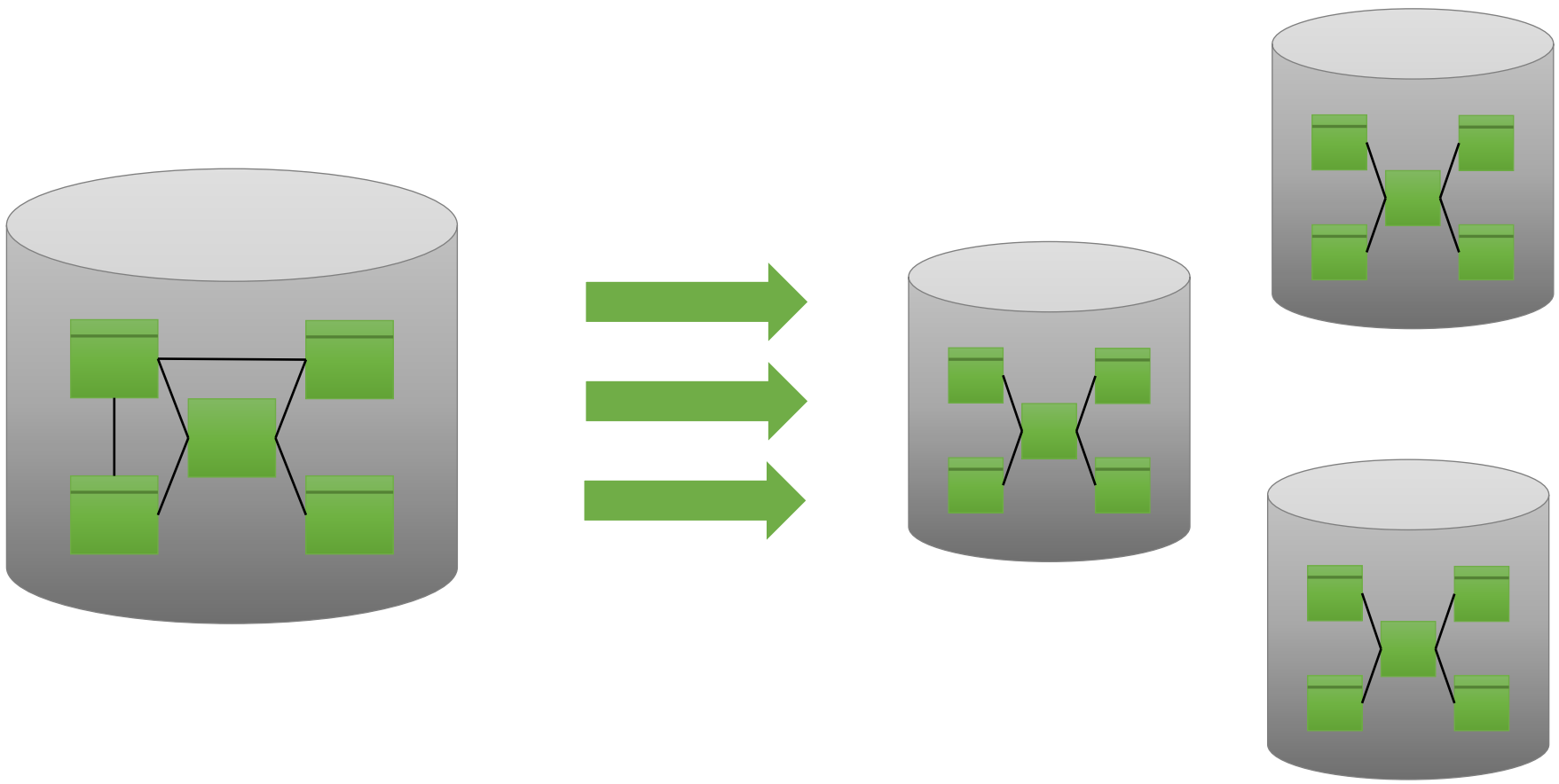


Bill Immon - Data Warehouse

Características:

- **Orientado a temas.** Los datos en la base de datos están organizados de manera que todos los elementos de datos relativos al mismo evento u objeto del mundo real queden unidos entre sí.
- **Integrado.** La base de datos contiene los datos de todos los sistemas operacionales de la organización, y dichos datos deben ser consistentes.
- **No volátil.** La información no se modifica ni se elimina, una vez almacenado un dato, Datos sumarizados por negocio.
- **Variante en el tiempo.** Los cambios producidos en los datos a lo largo del tiempo quedan registrados para que los informes que se puedan generar reflejen esas variaciones.

Bill Immon. Data Warehouse



Data Warehouse Empresarial

- Modelo normalizado (3FN).
- Datos atómicos al máximo nivel de detalle.
- Diseñado para almacenar datos.

Datamarts Departamentales

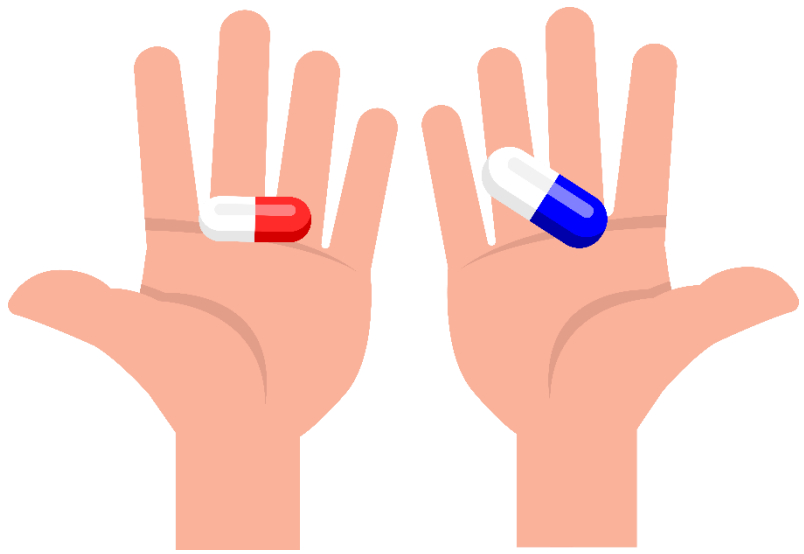
- Modelo dimensional.
- Datos agregados.
- Diseñado para el análisis.

Comparativa Immon vs. Kimball

	Immon	Kimball
Presupuesto	Coste inicial alto	Coste inicial bajo
Plazos	Requiere mas tiempo de desarrollo	Tiempo de desarrollo inferior
Expertise	Equipo con alta especialización	Equipo con especialización media
Alcance	Toda la compañía	Departamentos individuales
Mantenimiento	Fácil mantenimiento	Mantenimiento más complejo

Comparativa Immon vs. Kimball

1) Tanto **Kimball** como **Inmon** comparten la necesidad de establecer un sistema de almacenamiento de datos integrado y estable que garantice la explotación de la información.
2) **Immon** orientado a toda la compañía, gran volumen de datos. **Kimball** orientado a departamento, explotación rápida y sencilla.
3) **Kimball** se ajusta más a proyectos pequeños: sistema fácilmente explotable y entendible por el usuario y de rápido desarrollo. **Immon** orientado a proyectos grandes y más complejos.
4) Existen casos en los que se han implantado soluciones intermedias, logrando así sistemas híbridos que permiten conjugar con éxito las ventajas de ambas perspectivas.



Comparativa Immon vs. Kimball

Aspectos que habrá que analizar antes de decantarnos por una de las opciones:

- **Presupuesto** para acometer el proyecto.
- **Plazos** disponibles para la construcción del Data Warehouse.
- ***Expertise*** requerido para el equipo de desarrolladores.
- **Alcance** del Data Warehouse, ya sea para albergar los datos de toda la compañía o de determinadas áreas de negocio o departamentos.
- **Complejidad** de las labores de mantenimiento.

Comparativa Immon vs. Kimball

Conclusiones:

- **Ambas arquitecturas** son perfectamente **aplicables** al desarrollo de los DW corporativos.
- El nivel de impacto de factores como la **urgencia** de resultados, la **disponibilidad económica** y la **plataforma** tecnológica preexistente influirán decisivamente en la elección de una u otra visión.
- **Normalmente** se empieza por **Kimball** (DataMart), presupuestos ajustados, desconfianza o punto de entrada, y se debe aplicar el bus de dimensiones.
- A medida que una organización **crece** en BI, lo mejor sería pasar a **Immon** para ahorrar mantenimiento de ETL, crear un repositorio único.
- **Alcance** del Data Warehouse, ya sea para albergar los datos de toda la compañía o de determinadas áreas de negocio o departamentos Complejidad de las labores de mantenimiento.

Diseño del Data Warehouse

- Tipos de Modelos

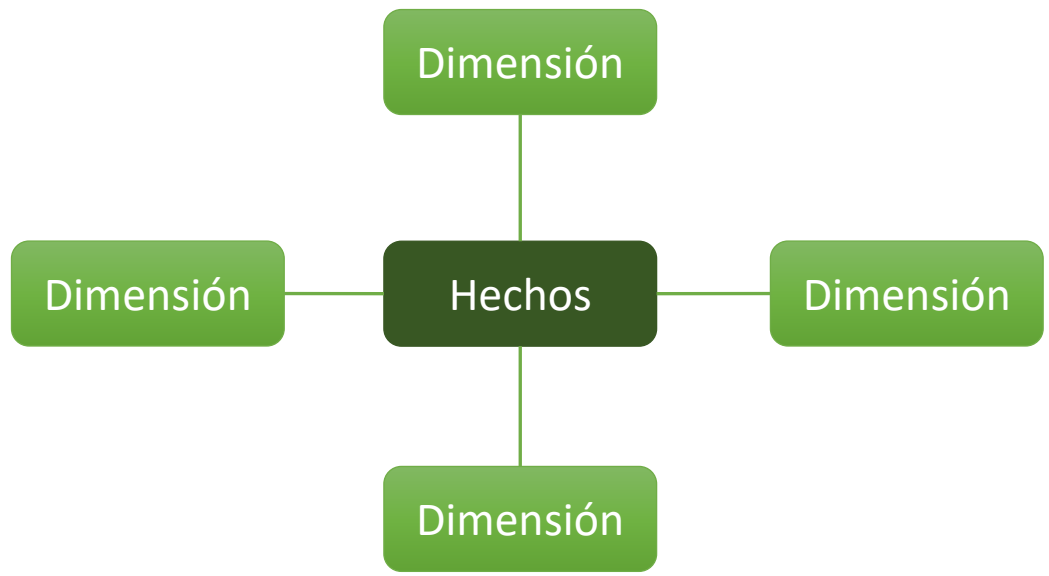
Recordamos...

- Las dimensiones proporcionan el contexto **“QUIÉN, DÓNDE, CUÁNDO“** y los hechos nos indican el **“QUÉ”.**
- Las tablas de dimensiones contienen los atributos descriptivos utilizados por las aplicaciones de BI para filtrar y agrupar los hechos.
- Con el grano de una tabla de hechos , se pueden identificar todas las dimensiones posibles.
- Siempre que sea posible, una dimensión debe tener un **valor único** cuando se asocia con una fila de hechos determinada.
- **Declarar el grano es el paso fundamental en un diseño dimensional.**

Modelo en Estrella

El modelo en estrella es la forma de modelado mas habitual y mas sencilla.

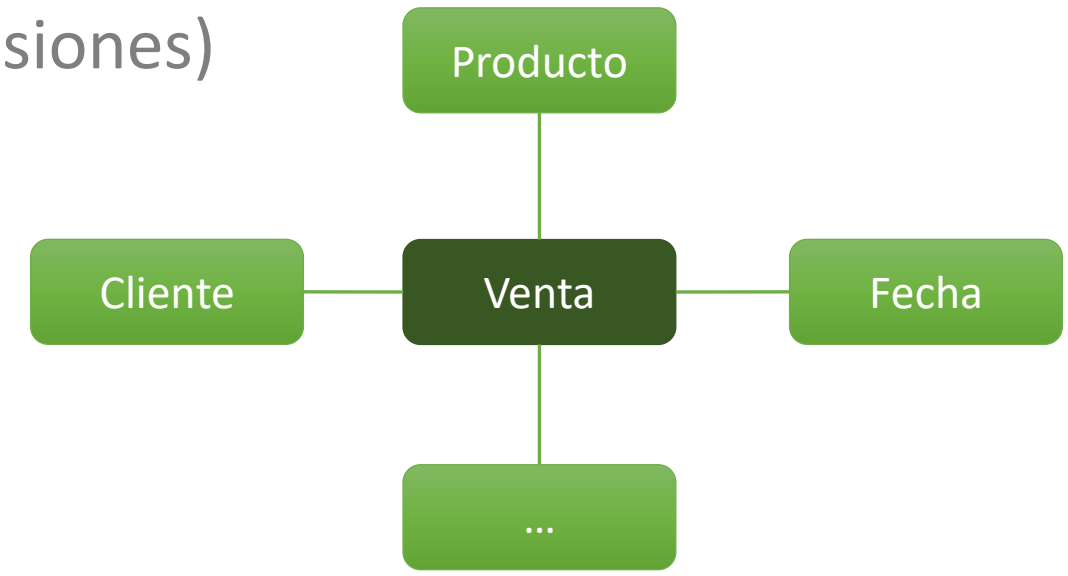
- Recibe su nombre por su estructura en la que aparece una tabla de hechos central relacionada con múltiples tablas de dimensiones.
- En él la información se encuentra **desnormalizada**, es decir, si tenemos una dimensión de **Localización**, todos los datos a los distintos niveles estarán en la misma dimensión: **Continente, País, Ciudad…**



Modelo en Estrella

Entre sus características destacan:

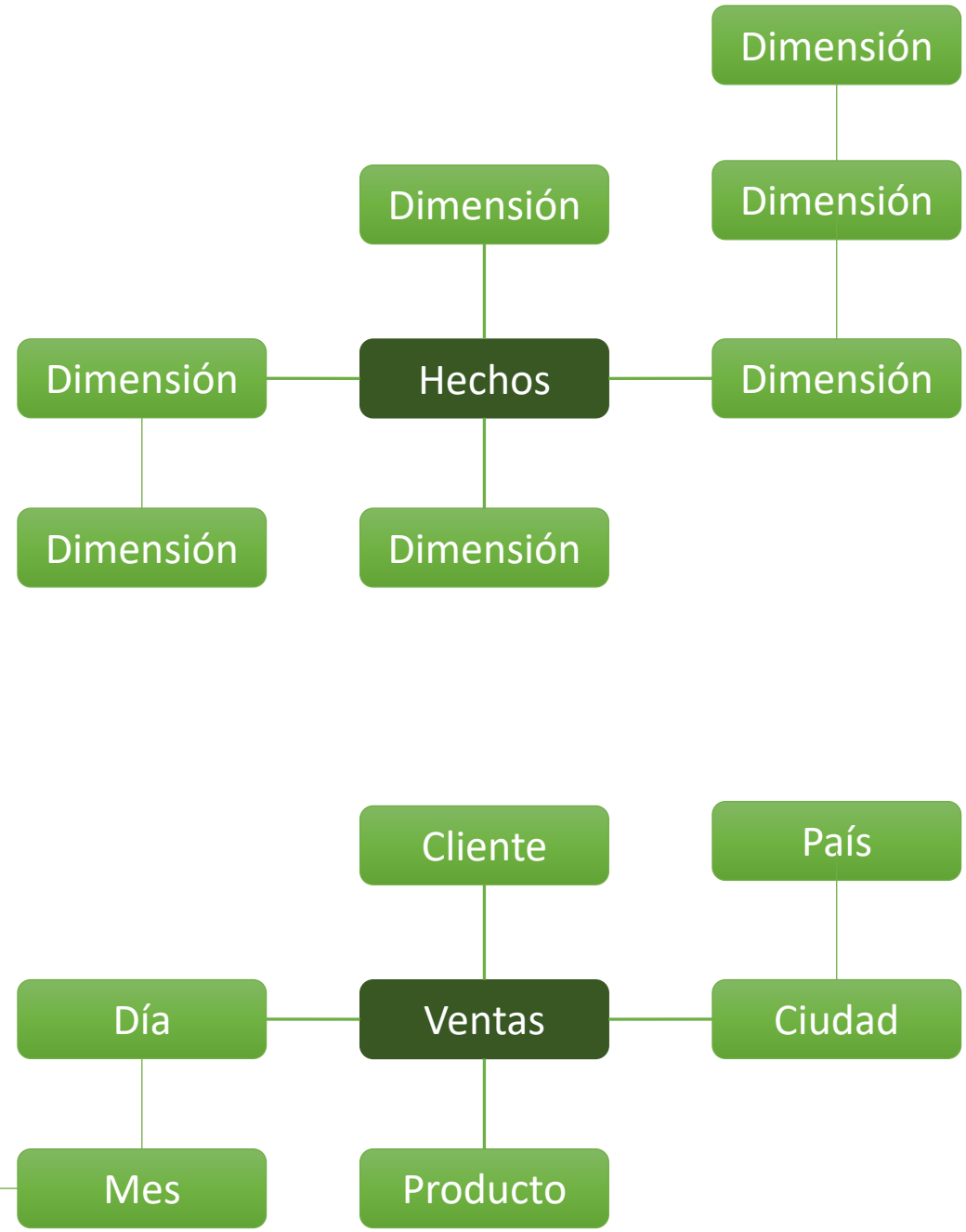
- Fácilmente reconocible por los usuarios de negocio.
- Adaptable a cambios (nuevos hechos, dimensiones, jerarquías)
- Beneficios en el rendimiento a nivel de base de datos.
- Número de tablas reducido.
  (90 % de las ocasiones < 25 dimensiones)
- Mayor número de filas. (Hechos)
- Mayor número de columnas. (Dimensiones)



Modelo de Copo de Nieve

El modelo de copo de nieve es una **forma normalizada del modelo en estrella.**

- En este caso se compone por una estructura similar con una tabla central de hechos y dimensiones relacionadas pero también aparecen dimensiones relacionadas entre sí.
- Aquí la información está normalizada, para el mismo caso anterior, tendremos una dimensión para cada nivel: **Dimensión País, Dimensión Ciudad…**



Diferencias en los modelos

- El modelo en **Copo de Nieve** utiliza **menos espacio** para almacenar las dimensiones porque como la información está normalizada hay menos duplicidad en los datos.
- Por otra parte, el **modelo en estrella** al estar desnormalizado debe ser tratado mas cuidadosamente para evitar **problemas de integridad** y puede complicar el mantenimiento en el largo plazo.
- Para perfiles mas avanzados, el **esquema en copo de nieve** sigue unas pautas de diseño mas lógica y organizadas.

Diferencias en los modelos

A nivel de consulta se ve que para obtener el mismo resultado, la consulta utilizada en el modelo en **Copo de Nieve es mas compleja** que la del **Modelo en Estrella** ya que hay que ir accediendo a todas las dimensiones que participan.

```
SELECT
  dim_store.store_address,
  SUM(fact_sales.quantity) AS quantity_sold

FROM
  fact_sales
  INNER JOIN dim_product ON fact_sales.product_id = dim_product.product_id
  INNER JOIN dim_prod_type ON dim_product.prod_type_id = dim_prod_type.prod_type_id
  INNER JOIN dim_time ON fact_sales.time_id = dim_time.time_id
  INNER JOIN dim_year ON dim_time.year_id = dim_year.year_id
  INNER JOIN dim_store ON fact_sales.store_id = dim_store.store_id
  INNER JOIN dim_city ON dim_store.city_id = dim_city.city_id

WHERE
  dim_year.action_year = 2016
  AND dim_city.city = 'Berlin'
  AND dim_product_type.product_type_name = 'phone'

GROUP BY
  dim_store.store_id,
  dim_store.store_address
```

Copo de Nieve

```
SELECT
  dim_store.store_address,
  SUM(fact_sales.quantity) AS quantity_sold

FROM
  fact_sales
  INNER JOIN dim_product ON fact_sales.product_id = dim_product.product_id
  INNER JOIN dim_time ON fact_sales.time_id = dim_time.time_id
  INNER JOIN dim_store ON fact_sales.store_id = dim_store.store_id

WHERE
  dim_time.action_year = 2016
  AND dim_store.city = 'Berlin'
  AND dim_product.product_type = 'phone'

GROUP BY
  dim_store.store_id,
  dim_store.store_address
```

Estrella

La decisión final

Cuando usar un Modelo en Estrella:

- **En Data Marts** (subconjuntos de un Data Warehouse) donde el ahorro de espacio no sea prioritario.
- Cuando se necesite un **análisis lo mas simple posible** para facilitar acciones futuras a usuarios de negocio.
- Para **herramientas especificas** que necesitan este modelo en concreto para poder ser utilizadas.

Cuando usar un Modelo en Copo de Nieve:

- En **Data Warehouses** principalmente para ahorrar espacio.
- Para **tablas de dimensión muy grandes** que requieran gran cantidad de espacio.
- Para herramientas especificas que necesitan este modelo en concreto para poder ser utilizadas.

Continuará...