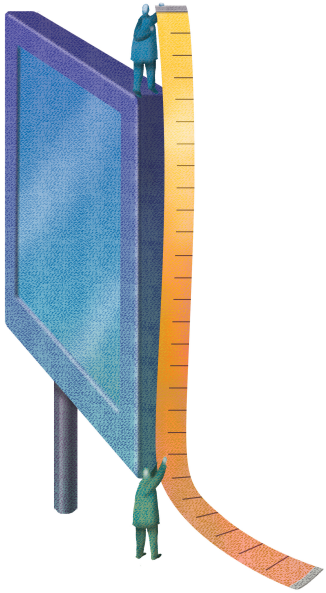
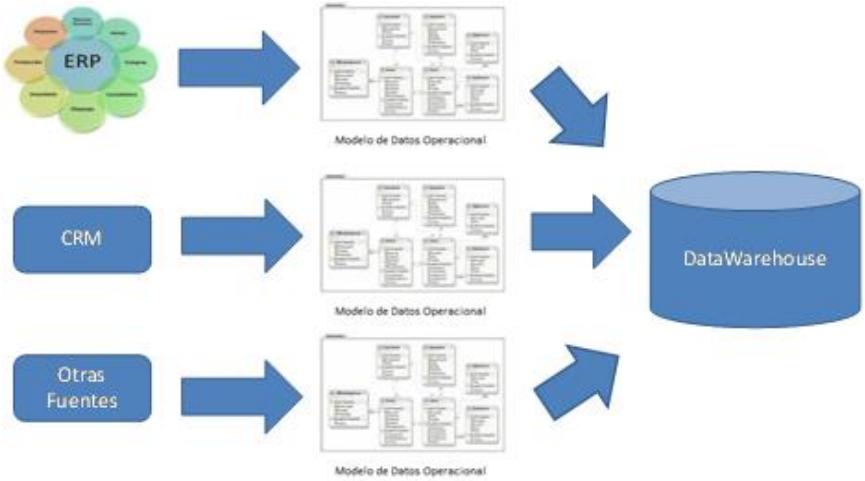
# ¿Qué se entiende por DataWarehouse y Datamart?

Generar un repositorio de datos consolidado suele requerir una importante inversión, pero es un proyecto que produce un alto impacto en la empresa. Si el diseño es robusto, permitirá rentabilizar esta inversión y potenciará el crecimiento de la compañía.



Si su empresa ya abordó un proyecto de desarrollo de un Datawarehouse (DWH), es muy probable que, por lo menos alguna vez, usted ya haya recuperado información de éste. Tal vez ni siquiera sabía que lo estaba haciendo o nadie habla de ese "lugar" de la empresa en esos términos. Pero lo cierto es que cuando un DataWarehouse fue desarrollado exitosamente comienza a difundirse rápidamente por todos los usuarios de las distintas áreas de la empresa.

El DataWarehouse (en español, Repositorio de Datos) es aquel centro de información de la empresa que posee los datos que son volcados desde fuentes transaccionales (CRM, ERP y otros) y que son guardados de manera estructurada en un modelo de datos diseñado para la realización de consultas y análisis **(ver Figura 1).**




En general, su diseño y desarrollo involucra una cantidad de recursos - tiempo y dinero- muy importantes. Dado que a partir del DWH se generarán una cantidad enorme de procesos, análisis, bases de datos, informes y decisiones, no es extraño que sean proyectos con inversión millonaria y de un alto impacto para la empresa.

## Aspectos fundamentales de un DataWarehouse

Aunque no es el objetivo de este curso profundizar en la teoría del DataWarehousing, es importante conocer los aspectos que lo definen:

1. **Orientado al objeto de análisis:** Los DWH están diseñados para ayudar a analizar datos. Por ejemplo, para aprender mejor sobre los datos de venta de su empresa, usted puede construir un DWH que tiene como foco las ventas. Usando este DWH podrá responder a preguntas como ¿cuál es la razón de la caída de ventas de este trimestre? ¿qué productos son los que aportan en mayor porcentaje a la venta? ¿quiénes son los mejores clientes? Luego, es necesario considerar el objetivo del DWH para diseñarlo con foco en el sujeto del análisis (financiero, RRHH, ventas, operaciones, logística, etc.)

2. **Integrado:** El DWH debe insertar datos de fuentes dispares en un formato consistente. Debe resolver problemas, por ejemplo, conflictos de nombres de tablas y campos, e inconsistencias entre las unidades de medidas. Cuando logran resolver estos conflictos se dice que el DWH está integrado.

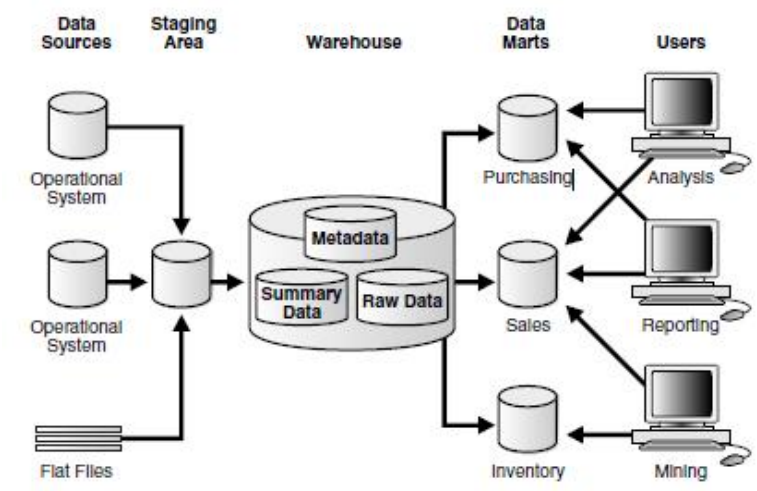
3. **No volátil:** Esta condición apunta a que una vez que el dato ha entrado en el DWH, éste no debe cambiar; es decir, el dato es invariante en el tiempo. Esto es lógico al considerar que el propósito del DWH es permitirle analizar aquello que ha ocurrido a través del tiempo.

4. **Contiene series de tiempo:** Un DWH debe enfocarse en el cambio a través del tiempo. Para descubrir tendencias en el negocio, los analistas requieren de grandes volúmenes de datos. Por ello es necesario que el DWH no sólo considere el momento actual del negocio, sino que su historia.

En general, las empresas que han desarrollado DWH (proyecto bastante costoso en dinero y tiempo), poseen un área especializada en su administración y mantención. Esta área es la encargada de velar porque los procesos funcionen y lo hagan a tiempo, que el hardware utilizado sea el adecuado y que el desempeño en la respuesta sea satisfactorio. Es decir, es el área responsable del correcto funcionamiento del DWH. Esta tarea, llevada a cabo por los DBA (Database Administrators o Administradores de Base de Datos), no es en absoluto trivial y requiere de gran conocimiento técnico.

Es muy común encontrar que los usuarios no extraen la información directamente del DWH, sino que lo hacen a través de repositorios que son alimentados por éste. A estos repositorios, que están en una fase posterior, se les denomina DataMart (**ver Figura 2).**

Los DataMart son también DataWarehouse, pero poseen un sujeto de análisis distinto. Responden a necesidades de conocimiento específicas para un área o producto determinado. En general, las áreas de Inteligencia de Negocios son los creadores de DataMarts alimentados, en gran medida, por el DataWarehouse. Pero para efectos del diseño y conceptualización, este DataMart representa el DataWarehouse desde el cual se realizará el análisis. Por ello, hablaremos indistintamente de DataWarehouse y DataMart, sin entrar en consideraciones sobre las diferencias técnicas existentes ya que no son objeto de este curso.




## Extracción, Transformación y Carga (ETL)

Hay tres pasos que son fundamentales para el diseño y generación de un DataWarehouse: extracción, transformación y carga (ETL). Determinarán el flujo correcto de información desde el origen hasta las bases de destino. Un buen diseño de estos procesos es básico para lograr un crecimiento orgánico del repositorio.

1. **Extracción:** este proceso inicial es el encargado de recuperar la información de la base o archivo de origen, con el fin levantar aquel formato necesario para el proceso posterior de transformación. En esta etapa se realizan procesos de verificación de la información recuperada (formatos de fecha, valores monetarios, largos de strings, etc.). Se requiere que no genere grandes demandas en los sistemas de origen al momento de solicitar la extracción de la información.

2. **Transformación:** aunque muchos de los datos no requieren ser transformados con respecto a su origen, existen otros que necesitan procesos de transformación antes de ser cargados en las bases finales. Este proceso es el más “creativo” de los tres, dado que es aquí donde se aplican las reglas del negocio que darán riqueza al dato extraído.

Ejemplos de transformación son: calcular el promedio de compra de los tres últimos meses, calcular la edad a partir de la fecha de nacimiento, transformar un string a otro (Sexo M a Masculino), trasponer o pivotear filas en columnas y muchas

otras transformaciones que estarán al servicio de quien las analice.

**3. Carga:** Dependiendo del diseño inicial, el proceso de carga ejecutará distintas acciones al momento de inyectar los datos en las bases de destino. Existen diversas acciones que pueden realizarse al momento de incorporar nueva información:

- ***Carga con agregación:*** aplica a la información actual existente un nivel de agregación (suma o conteo) de manera de no mantener grandes volúmenes de información histórica. Luego, la nueva información es cargada con el nivel de granularidad mínimo (nivel de mayor detalle existente en una base) definido.

- ***Carga con granularidad múltiple:*** a partir de las fuentes de origen se crean diversos niveles de agregación al momento de realizar la carga. Luego, en este proceso no se afecta la información histórica existente.

## Características de un buen DataMart

En esta sección abordaremos aquellos aspectos que debe considerar en el diseño del DataMart orientado al uso del BI para que sea un repositorio ágil, con capacidad de crecimiento y estructurado:

- **Automatice, automatice, automatice:** Hay una regla informal que se aplica al diseñar los procesos de carga de información: “Si lo realiza manualmente más de tres veces, es necesario automatizarlo”. Actualmente los software de ETL (extracción, transformación y carga) permiten niveles de automatización robustos, donde es posible - incluso sin necesidad de programación- entregar las instrucciones precisas para que no sea necesario agregar ninguna manualidad. Es muy común encontrar tablas en los DataMart que han sido “olvidadas” por los procesos de carga y quedan desactualizadas mucho tiempo provocando un arrastre de errores que puede ser perjudicial al momento de la entrega de información o análisis.

- **Cree un log de carga y sistemas de alarmas:** Es importante generar una tabla donde queden registrados los resultados de los procesos ETL y que generen un aviso al usuario. De esta forma logrará controlar el éxito o falla de sus procesos, sin tener la necesidad de revisarlos uno a uno. Una práctica utilizada para dar aviso al usuario es el envío de un email describiendo el resultado del proceso.

- **Organice la información en bases de datos distintas:** A medida que el número de tablas crecen en su DataMart, requerirá organizar la información de manera lógica (tal como lo hacemos con las carpetas en nuestro computador). Cada DataMart tiene sus particularidades que hacen que la forma de organizar la información en distintas bases no tenga una regla única. Destine tiempo a encontrar aquella lógica, que permitirá que sea trivial para los usuarios saber dónde guardar la información.

- **Proyecte el crecimiento de espacio:** Antes de comenzar a integrar información es necesario que considere el espacio que requerirá para lograr incorporar los datos necesarios. En general, los crecimientos son muy fuertes al inicio y después se mantienen en niveles estables. Luego, considere el crecimiento en un horizonte de tiempo prudente para poder reaccionar con tiempo a las necesidades de crecimiento de capacidad de almacenamiento.

- **Cree índices sobre las tablas:** Los índices son objetos de bases de datos utilizados para mejorar el rendimiento de las consultas. Generalmente se crean sobre aquellas columnas sobre las que se generan los cruces con otras tablas (RUT, ID Cliente, Código de Transacción, etc.). Más información sobre índices en bases de datos en el siguiente link.

- **Nombre las columnas iguales de la misma forma:** Aunque parece trivial, esto no siempre se realiza de manera exhaustiva. Por ejemplo, no es extraño observar DataMarts donde se nombra el RUT de un cliente de dos formas distintas (RUT_CLIENTE y RUT). Esto provoca que el usuario tenga que recordar o revisar en el diccionario de datos qué nombre utilizar al momento de usar las distintas tablas.

- **Designe a un DBA:** Con el fin de administrar y ordenar el uso del DataMart es muy recomendable designar a una persona como Administrador de la Base de Datos (DBA por sus siglas en inglés). El DBA será el encargado de administrar permisos sobe las bases, asignar los espacios, recuperar información, coordinar las ventanas de procesamiento de usuarios y eliminar/recuperar procesos, entre otros.

- **Cree un diccionario de datos:** Si posee un diccionario de datos de las tablas y bases de su DataMart, hará muy simple la tarea del usuario de familiarizarse con el mundo de información existente. Además, podrá permitir la incorporación de usuarios de información que no necesariamente pertenecen al equipo de BI.

- **El DataMart debe responder al negocio:** Tal como hemos mencionado en clases anteriores, el BI debe estar al servicio del negocio. Por lo tanto, el centro del BI, el DataMart/DataWarehouse también debe estarlo. El DataMart es la materia prima sobre la cual será posible generar el conocimiento para las decisiones de negocio. Si el DataMart diseñado es robusto, generaremos conocimiento robusto y permitirá seguir profundizando en áreas que no hubiera sido posible descubrir sin haber invertido tiempo y dinero en el repositorio de datos.