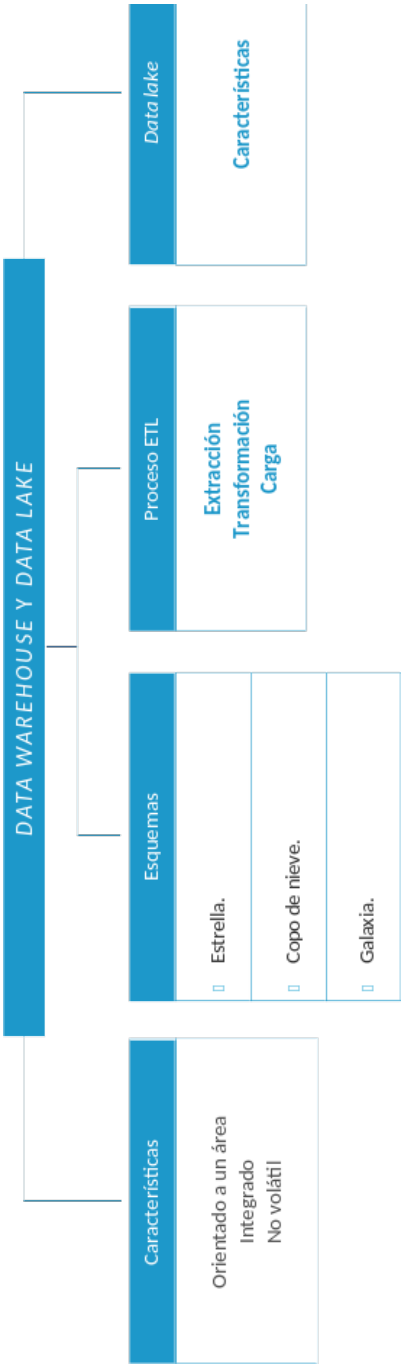
Gobierno del Dato y Toma de Decisiones

Tema 3. Data warehouse y data lake

Índice

Esquema



3.1. Introducción y objetivos

En el presente tema el estudiante podrá entender el proceso técnico que deben seguir los datos para transformarse de datos brutos a un *data warehouse* o un *data lake,* dependiendo de las necesidades empresariales.

Los objetivos de este tema son:

- Identificar cada uno de los pasos del proceso ETL: extracción, transformación y carga.
- Estudiar el concepto de *data warehouse* y diferenciar los tipos de esquemas.
- Comprender la diferencia entre un *data warehouse* y un *data lake.*

3.2. Procesos ETL

Como sus siglas indican, consiste en la extracción, transformación y carga de los datos, de modo que se puede afirmar que es una parte fundamental de este. Antes de guardar los datos, deben ser transformados, limpiados, filtrados y redefinidos. Como se mencionó anteriormente, la información que tienen las empresas en los sistemas no está preparada para la toma de decisiones (Ong et al., 2017).

El proceso de ETL consume entre el 60 y el 80 % del tiempo de un proyecto de *business intelligence,* por lo que es un proceso fundamental en el ciclo de vida del proyecto (Eckerson y White, 2003). Esta parte del proceso de construcción del *data warehouse* (DW) es costosa y consume una parte significativa de todo el proceso, razón por la que utilizan recursos, estrategias, habilidades especializadas y tecnologías. El proceso ETL va más allá del transporte de los datos de las fuentes a la carga dentro del DW, ya que añade un valor significativo a los datos. Una parte del proceso ETL se encarga de (Villanueva, 2011):

- Eliminar errores y corregir datos faltantes.
- Proporcionar medidas documentadas de la calidad de los datos.
- Supervisar el flujo de los datos transaccionales.
- Ajustar y transformar los datos de múltiples fuentes en uno solo.
- Organizar los datos para su fácil uso por los usuarios y las herramientas.

El proceso ETL es intuitivo y fácil de entender. La idea fundamental del proceso ETL es tomar los datos de las diferentes fuentes de información y depositarla sin errores en el *data warehouse.* Los procesos de limpieza y transformación de esa información son mucho más complejos de lo que se cree. Se pueden dividir en tareas específicas, dependiendo de las características de las fuentes de datos, los objetivos

de la empresa, las herramientas existentes y las características del DW final.

El desafío para un correcto desarrollo del proceso ETL es **planificar adecuadamente** la cantidad de tareas, para lo cual es preciso conservar la perspectiva sencilla e intuitiva del proceso.

El proceso ETL es obligatorio para acceder a los datos que formarán parte del *data warehouse.* El proceso ETL se divide en cuatro etapas:

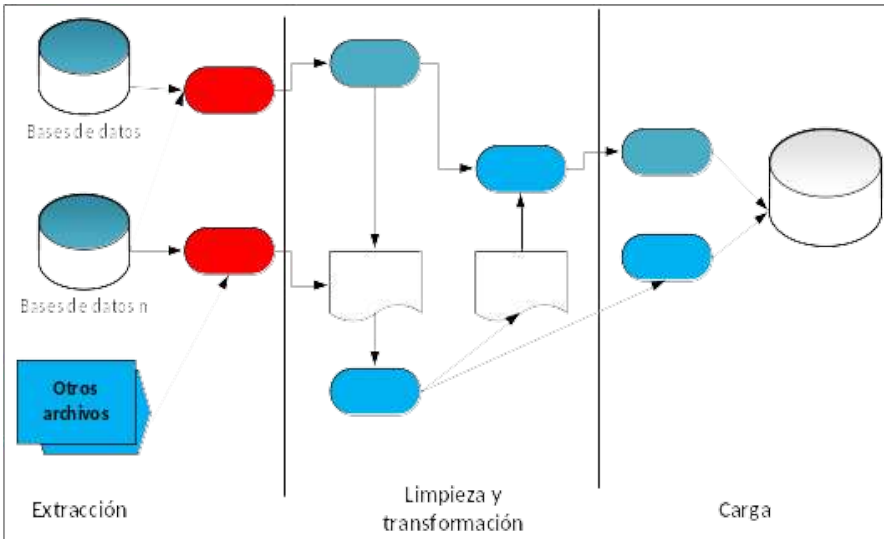


Figura 1. Etapas del proceso ETL.

Etapas

Extracción

Este proceso extrae los datos físicamente de las distintas fuentes de información. En este momento los datos están en la forma como se almacenan, en bruto. La extracción de los datos se puede realizar de forma manual o utilizando herramientas de ETL.

Durante el proceso de ETL, una de las primeras tareas que debe realizarse es la extracción de la información más relevante, es generalizar al *data warehouse* (Theodoratos et al., 2001). Para la extracción se pueden usar los siguientes métodos:

1. **La extracción estática,** que tiene lugar cuando el *data warehouse* necesita ser rellenado por primera vez. La detección de cambios se realiza físicamente mediante la comparación de dos imágenes (una correspondiente a la extracción anterior y la otra a la actual).

2. **La extracción incremental,** que es utilizada para actualizar los *data warehouse* de forma regular, aprovecha los cambios aplicados a los datos de origen desde la última extracción.

Finalmente, conviene recordar que el objetivo principal de esta etapa es extraer tan solo aquellos datos de los sistemas transaccionales que son necesarios y prepararlos para el resto de los subprocesos de ETL. Para ello, se deben determinar las mejores fuentes de información y de mejor calidad.

Limpieza

Este proceso recupera los datos de la base de datos u otro tipo de fuente y comprueba la calidad, elimina los duplicados y, cuando es posible, corrige los valores erróneos y completa los valores incompletos, etc. Ejemplo de algunos errores más comunes:

- Datos duplicados: un cliente es registrado varias veces en la misma empresa.
- Inconsistencia en los datos: en la dirección de una persona, el código postal no corresponde a la ciudad donde vive.
- Inconsistencia de valores: aparece en primer lugar un valor y posteriormente aparece el mismo valor de otra forma. Por ejemplo: primero, escribir el país como USA y, luego, digitarlo completo (Estados Unidos de Norteamérica).

En particular, hay que tener en cuenta que estos tipos de errores son muy frecuentes cuando se manejan múltiples fuentes y se ingresan datos manualmente.

Las principales características de limpieza de datos que se encuentran en las herramientas de ETL son la rectificación y la homogeneización. Utilizan diccionarios específicos para rectificar errores de digitalización y para reconocer sinónimos, además de la limpieza basada en reglas para imponer normas específicas de dominio y definir asociaciones apropiadas entre valores.

Transformación

Este proceso recupera los datos limpios y de alta calidad, los organiza y resume en los distintos modelos de análisis. El resultado de este proceso es la obtención de datos limpios, consistentes, resumidos y útiles. La transformación incluye: cambios de formato, sustitución de códigos, valores derivados y agregados.

La transformación es el núcleo del proceso. Convierte los datos de su formato original a un formato de almacén de datos específico. Si se implementa una arquitectura de dos capas, esta fase genera su capa de datos conciliados.

Independientemente de la presencia de una capa de datos conciliados, establecer una correspondencia entre la capa de datos de origen y la de depósito de datos generalmente se dificulta, debido a la presencia de muchas fuentes diferentes y heterogéneas.

Los siguientes puntos deben rectificarse en esta fase:

- Los textos sueltos pueden ocultar información valiosa. Por ejemplo, Zapatos Zoe LTD no muestra explícitamente que se trata de una sociedad de sociedad limitada, ya que la sigla estándar en España es SL.
- Se pueden usar diferentes formatos para datos individuales. Por ejemplo, una fecha se puede guardar como una cadena de caracteres o como tres enteros.
- Seleccionar ciertas columnas para su carga (por ejemplo, que las columnas con valores vacíos no se carguen o se completen).

- Traducir códigos (por ejemplo, cuando se almacena una «H» para «Hombre» y «M» para «Mujer», pero luego se cambia a formato numérico: «1» para Hombre y «2» para Mujer). Otro ejemplo: «V» para vivo y «M» para muerto se cambia a «1» para vivo y «0» para muerto.

- Codificar valores libres, como, por ejemplo: convertir «Hombre» en «1», «Mujer» en «2» o «Niños» en «3».

- Obtener nuevos valores calculados (por ejemplo, el índice de masa corporal = peso/altura).

- Calcular totales de múltiples filas de datos (por ejemplo, el total de una población, total de años, etc.).

- Dividir una columna en varias (por ejemplo, la columna de «Diagnóstico: pasar a tres columnas Diagnóstico_1, Diagnóstico_2, Diagnóstico_3»).

- Datos erróneos: se pueden corregir o eliminar. Esto va a depender del valor que aporte las variables y los datos al *data warehouse.*

La carga y actualización

Es la última etapa del proceso y valida que los datos cargados en el DW sean consistentes con las definiciones y formatos; los integra en los distintos modelos de las distintas áreas de negocio que se han definido. Estos procesos suelen ser complejos, por tanto, es necesario tener personal experto que ayude en el proceso. Aquí es esencial comprobar que se ha desarrollado correctamente, ya que, en caso contrario, puede llevar a decisiones erróneas a los usuarios.

Esta etapa es el momento en el que se cargan los datos y se comprueba si los elementos que se cargaron son equivalentes a la información que había en el sistema transaccional, así como los valores que tienen los registros cargados corresponden a los definidos en el *data warehouse.* Es importante comprobar que se

ha desarrollado correctamente, ya que, de lo contrario, puede llevar a tomas de decisiones equivocadas. La carga en un almacén de datos es el último paso para seguir.

La diferencia fundamental entre carga y actualización radica en el hecho de que la carga se realiza cuando el DW está vacío, mientras que la actualización se hace cuando ya existen datos en el mismo. En cualquier caso, tanto la carga como la actualización se pueden llevar a cabo de dos maneras:

1. **Actualizar datos del almacén de datos completamente reescrito:** esto significa que los datos más antiguos se reemplazan. La actualización se usa normalmente en combinación con la extracción estática para poblar inicialmente un depósito de datos.

2. **Actualización de datos solo con los cambios aplicados a los datos fuente:** la actualización generalmente se lleva a cabo sin eliminar o modificar datos preexistentes. Esta técnica se usa en combinación con la extracción incremental para actualizar los almacenes de datos regularmente.

3.3. Almacén de datos (data warehouse o DW)

A través del *data warehouse,* conocido también como el almacén de datos en el diccionario de datos, se busca almacenar los datos de forma que facilite y maximice su **flexibilidad, facilidad de acceso** y **administración.** Surge como respuesta a las necesidades de los usuarios que necesitan información consistente, integrada, histórica y preparada para ser analizada y poder tomar decisiones. Al recuperar la información de los distintos sistemas (transaccionales, departamentales o externos) y almacenarlos en un entorno integrado de información diseñado por los usuarios, el *data warehouse* permitirá analizar la información contextualmente y relacionarla dentro de la organización.

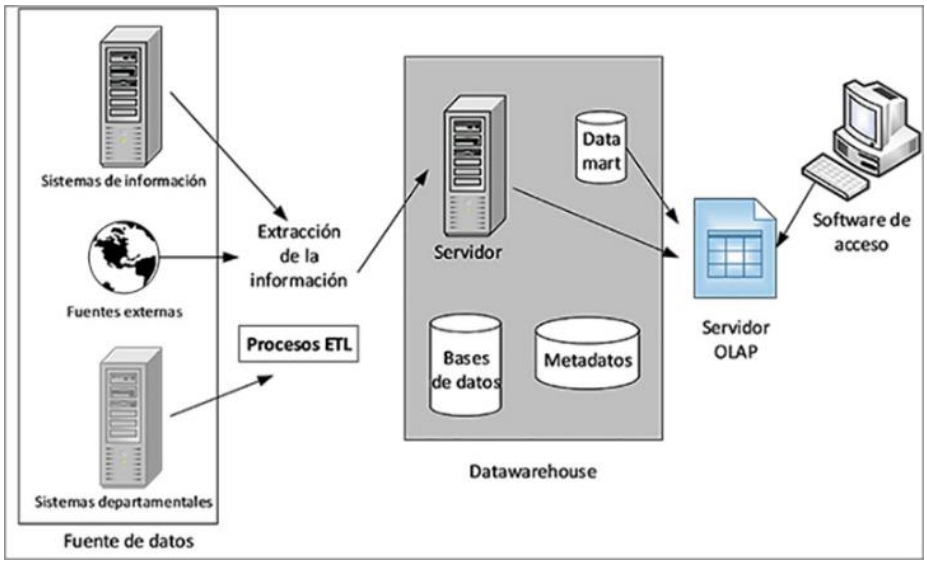


Figura 2. Componentes del *data warehouse*. Fuente: Cano (2007).

Fuentes de datos

Se parte de las fuentes para sostener la información del *data warehouse.* Las fuentes de información externas en algunos casos son compradas a otras empresas que

gestionan información comercial, encuestas de satisfacción y estudios de mercado, entre otros. Las fuentes de información externas son esenciales para enriquecer la información que se tiene de los clientes. En otras ocasiones es favorable para la empresa incorporar información como, por ejemplo: la población, el número de habitantes y los presupuestos públicos.

El autor Bill Inmon definió las **características** que debe cumplir un *data warehouse:* debe estar orientado sobre un área, integrado e indexado en el tiempo; es un conjunto no volátil de información que soporta la toma de decisiones (Inmon, 1992).

- **Orientado a un área:** significa que cada parte del DW está construida para resolver un problema de negocio, que ha sido definido por quienes toman las decisiones. Por ejemplo, entender los hábitos de compra de los adolescentes, analizar la calidad de los productos o analizar la productividad de una línea de producción. Para poder analizar un problema de negocio se necesita información que pueda venir de distintos sistemas: ventas, clientes y elementos de transporte, entre otros.

- **Integrado:** la información debe ser convertida en medidas comunes, códigos y formatos comunes para que pueda ser útil. La integración permite a las organizaciones implementar la estandarización de conceptos, por ejemplo: la moneda, las fechas, etc.

- **Indexado en el tiempo:** significa que la información histórica se mantiene y se almacena en determinadas unidades de tiempo, tales como horas, días, semanas, meses, trimestres o años. Ello nos permitirá analizar, por ejemplo, la evolución de las ventas, los inventarios en los períodos que se definan.

- **No volátil:** esta información no es mantenida por los usuarios, como se realizaría en los entornos transaccionales. La información se almacena para la toma de decisiones. La actualización no se realiza de forma continua, sino periódicamente, como lo defina la empresa.

El *data warehouse* debe cumplir con algunos **objetivos.** Ralph Kimball (1996) define

los siguientes:

- Acceder a la información de la empresa o del área funcional.
- Ser consistente.
- Separar la información para ser analizada a nivel individual o de manera conjunta.
- Utilizar herramientas de presentación de la información.
- Facilitar la publicación de la información.
- Tener alta calidad para soportar procesos de reutilización.

Los usuarios de negocio necesitan tomar decisiones basadas en la información del DW, por lo que se deben asegurar las siguientes características según Barrer (1998).

- Alta disponibilidad.
- Rendimiento.
- Copias de seguridad y recuperación.
- Recuperación física en caliente.

Esquemas de un *data warehouse*

Existen varias estructuras bajo las cuales se construye un DW, las más utilizadas son los modelos estrella y copo de nieve, sus nombres se basan en el dibujo que forman al crearse.

Esquema estrella

Este modelo es el más sencillo. Está formado por una tabla central de «hechos» y

varias «dimensiones», incluida una dimensión de «tiempo». Lo más representativo de la arquitectura de estrella es que solo existe una tabla de dimensiones para cada dimensión. Esto quiere decir que la única tabla que tiene relación con otra es la de hechos, esto es, que toda la información relacionada con una dimensión debe estar en una sola tabla. En la Figura 3 se observa un ejemplo de este modelo.

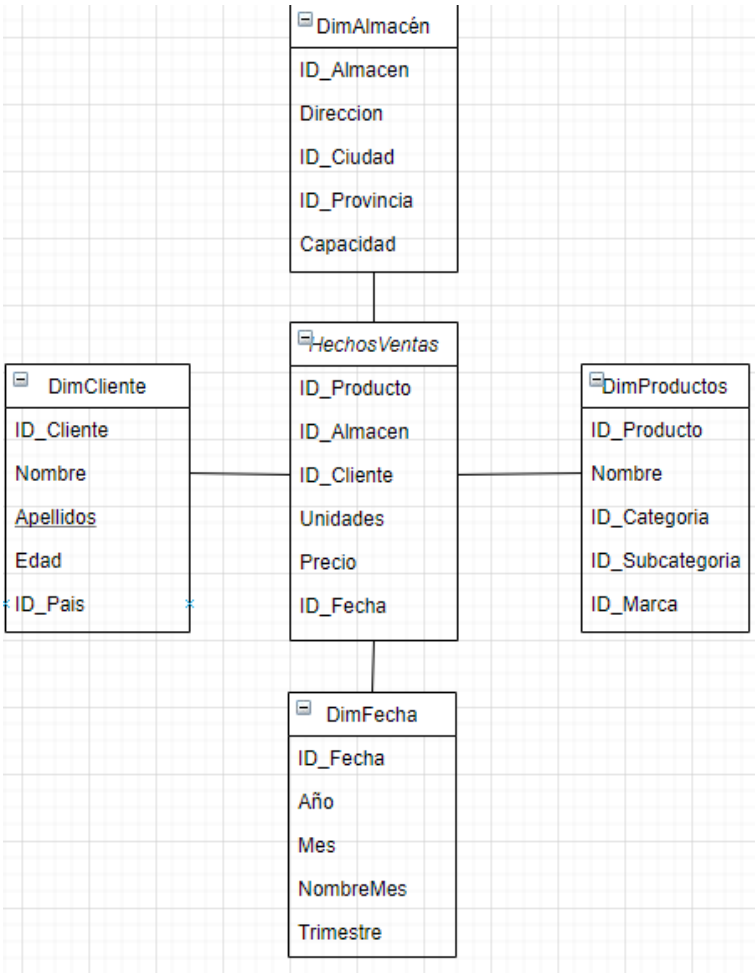


Figura 3. Ejemplo de modelo estrella. Fuente: adaptado de Esquema en estrella, 2021.

En un *data warehouse* de ventas, los hechos son las ventas. En uno financiero, los elementos del balance. En uno de análisis de la bolsa, los hechos serían los conceptos de apertura y precio de cierre. En la tabla de hechos, la clave está conformada por las claves foráneas que apuntan a las dimensiones: ID_Producto,

ID_Almacen, ID_Cliente, ID_Fecha. Es decir, para un almacén, un día, un producto y un cliente, solo puede existir un registro de unidades y precio.

Un modelo estrella es un modelo desnormalizado, ya que lo que se busca es una mejora en el rendimiento de las consultas. Los ***join*** en las bases de datos relacionales pueden ser muy pesados.

Ventajas y desventajas de este modelo:

- Simple y rápido para un análisis multidimensional. Permite consultar datos agregados y detalles.
- Permite implementar la funcionalidad de los datos multidimensionales y, a la vez, las ventajas de una base de datos relacional.
- En cuanto a rendimiento es la mejor opción, ya que permite indexar las dimensiones de forma individualizada sin que el rendimiento de la base de datos se vea afectado.

Esquema copo de nieve

Es una variante del modelo anterior. En este modelo la tabla de hechos ya no es la única que se relaciona con otras tablas, existen otras tablas que se relacionan con las dimensiones y que no tienen relación directa con la tabla de hechos. El modelo fue concebido para facilitar el mantenimiento de las dimensiones, sin embargo, esto permite que se vinculen más tablas a las secuencias SQL. Este modelo es complejo de mantener, ya que permite la vinculación de muchas tablas.

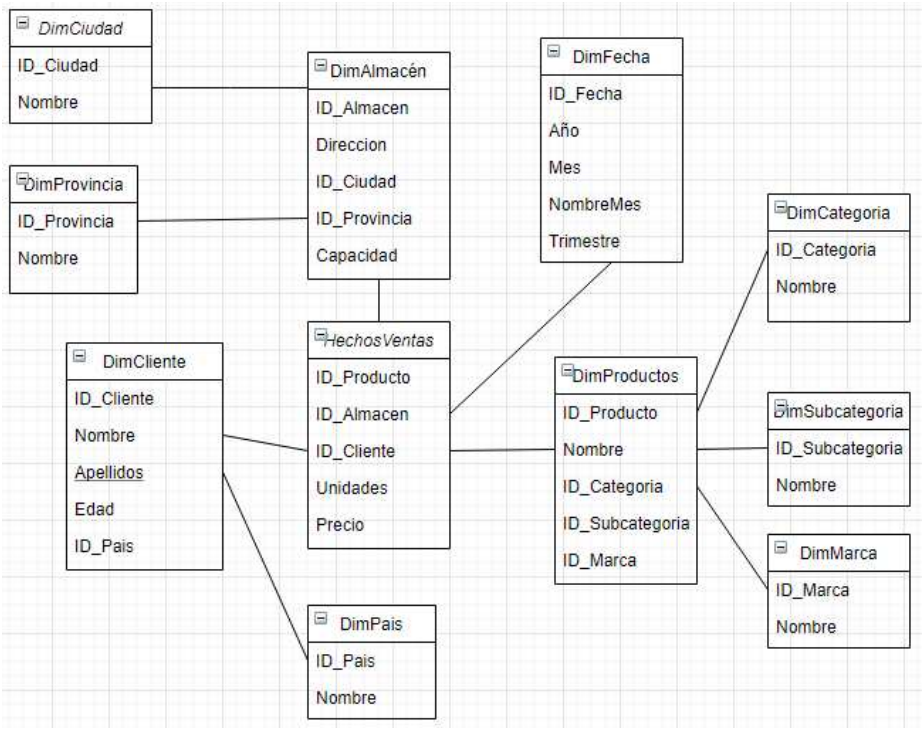


Figura 4. Ejemplo de modelo copo de nieve. Fuente: adaptado de Esquema en copo de nieve, 2020.

Ventajas y desventajas de este modelo:

- Al estar normalizado se evita la redundancia de datos.
- El tiempo de respuesta es muy elevado, por lo que, si es necesaria una respuesta rápida y es crítico para el sistema, puede no ser la mejor opción.

Los *data warehouse* se representan normalmente como una gran base de datos, que en algunas ocasiones pueden estar distribuidas en distintas bases de datos, es decir, centralizar toda la información que posee la empresa en un solo sitio, esto permite manejar la información fácilmente (ver Figura 4). El trabajo de construir un DW colectivo puede generar inflexibilidades, o ser costoso y requerir plazos de tiempo elevados.

Esquema galaxia

Este esquema contiene varias tablas de hechos que comparten dimensiones. Es muy común encontrar este tipo de esquema, incluso es recomendable compartir dimensiones. El esquema se ve como una colección de estrellas, por eso su nombre.

Por ejemplo, pueden existir dos tablas de hechos: inventario y ventas que podrían compartir las dimensiones de producto y fecha.

Veamos algunas arquitecturas:

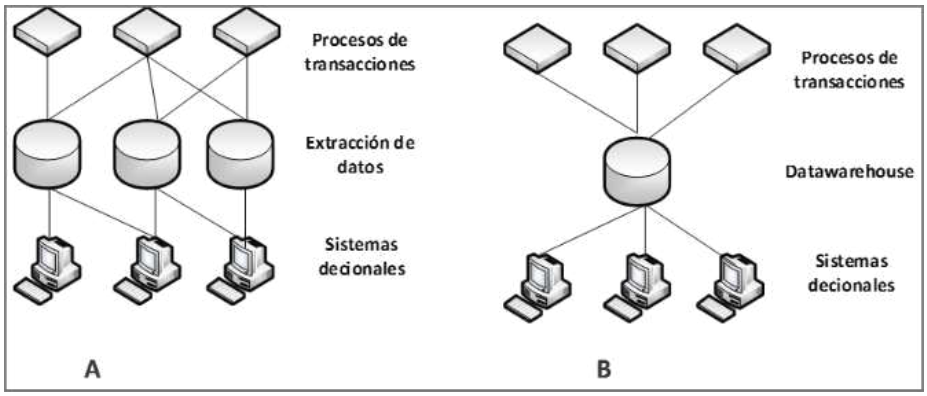


Figura 5. Almacenes de datos antes (A) y después de aplicar *data warehouse* (B). Fuente: Abella et al. (2000).

Arquitecturas

Para la realización del *data warehouse* se adoptan dos clasificaciones diferentes para su arquitectura:

- La primera clasificación está orientada a la estructura y depende del número de capas utilizadas por la arquitectura.
- La segunda clasificación depende de cómo se empleen las diferentes capas para crear vistas orientadas a los departamentos.

Arquitectura de una sola capa: no se utiliza con frecuencia en la práctica. Su objetivo es minimizar la cantidad de datos almacenados. Para alcanzar este objetivo,

se eliminan las redundancias de datos. Esto significa que un almacén de datos se implementa como una **vista multidimensional** de datos operacionales creados por un *middleware* específico o una capa de procesamiento intermedio (Devlin, 1997).

La debilidad de esta arquitectura radica en que no cumple con los requisitos de separación entre procesamiento analítico y transaccional. Las consultas de análisis se envían a los datos operativos después de que el *middleware* los interpreta. De esta manera, las consultas afectan a las cargas de trabajo transaccionales regulares. Además, aunque esta arquitectura puede cumplir los requisitos de integración y exactitud de los datos, no puede registrar más que las fuentes.

Por estas razones, un enfoque de este tipo para los almacenes de datos puede ser exitoso solo si las necesidades de análisis son particularmente restringidas y el volumen de datos a analizar es enorme (Rizzi y Golfarelli, 2009).

Arquitectura de dos capas: aunque normalmente se nombra «arquitectura de dos capas» para destacar la separación entre las fuentes físicamente disponibles y los almacenes de datos, en realidad consta de cuatro etapas de flujo de datos posteriores (Hüsemann et al., 2000).

- **Capa de origen:** es el sistema de almacén de datos que utiliza fuentes heterogéneas de datos. Los datos se guardan originalmente en bases de datos relacionales corporativas o pueden provenir de sistemas de información fuera de los muros corporativos. La prioridad en este tipo de sistema es la actualización y se mantienen pocos datos históricos.

- **Capa de almacenamiento de datos:** los datos almacenados en las diferentes fuentes deben extraerse, limpiarse para eliminar inconsistencias y rellenar espacios, e integrarse para convertirlas en fuentes heterogéneas en un esquema común, proceso ETL. Pueden combinar esquemas heterogéneos, extraer, transformar, limpiar, validar, filtrar, quitar duplicados, archivar y cargar los datos fuente para ser utilizados en el *data warehouse* (Jarke et al., 2013).

- **Capa de depósito de datos:** la información se almacena en un solo depósito lógicamente centralizado. Se puede acceder directamente al almacén de datos, pero también se puede utilizar como fuente para crear nuevos productos de datos, que replican parcialmente los contenidos del almacén de datos y están diseñados para departamentos empresariales específicos. Los repositorios de metadatos almacenan información sobre fuentes, procedimientos de acceso, usuarios, esquemas de *data mart* (estos y los metadatos se amplían más adelante). Un DW está constituido por la integración de varios *data marts.*

- **Capa de análisis:** se accede de manera eficiente y flexible a los datos integrados para emitir informes, analizar la información y representar escenarios hipotéticos de negocios (adecuados para cada empresa). Tecnológicamente hablando, aquí se utilizan diferentes herramientas de visualización de datos, optimizadores de consultas para el apoyo para la toma de decisiones.

Impacto del *data warehouse* (Mendez et al., 2003)

El éxito del *data warehouse* está enfocado en mejorar los procesos empresariales, operacionales y de toma de decisiones. Para que esto funcione se deben tener en cuenta los impactos producidos en los diferentes ámbitos de la empresa:

Impacto en las personas

La construcción del *data warehouse* requiere de la participación de quienes lo utilizarán, depende de la realidad de la empresa y de las condiciones que existan en el momento de la creación, las cuales determinarán cuál será su contenido.

Como se ha visto, el *data warehouse* provee los datos que posibilitarán a los usuarios acceder a la propia información en el momento en que la necesiten. Para

que se realice esta entrega hay que tener en cuenta:

- Los usuarios deberán adquirir nuevas destrezas; por lo tanto, van a necesitar programas de capacitación adecuados.
- Los largos tiempos de análisis y programación se reducen para usuarios pertenecientes a las áreas de tecnología, y se reduce también el tiempo de espera para los usuarios de negocio.
- Como la información estará lista para ser utilizada, es probable que aumenten las expectativas. Se reducirá considerablemente la gran cantidad de reportes en papel.

Impactos en los procesos empresariales y de toma de decisiones

- Mejora del proceso para la toma de decisiones, ya que facilita la disponibilidad de la información. Las decisiones son tomadas más rápidamente y la gente entiende más del porqué de las decisiones.
- Los procesos empresariales se optimizan, se elimina el tiempo de espera de la información al encontrarse almacenada en un solo sitio.
- Se reducen los costos de los procesos, una vez desarrollado el *data warehouse* y en múltiples ocasiones se esclarecen sus conexiones y dependencias, lo que aumenta la eficiencia en dichos procesos.
- El *data warehouse* permite que los datos de los sistemas sean utilizados y examinados al estar organizados para tener un significado para la empresa.
- Aumenta la confianza en las decisiones tomadas con base en la información del DW , tanto los responsables de la toma de decisiones como los afectados conocen la información, que tendrá que ser de buena calidad, clara, precisa y concisa.
- La información que se comparte lleva a un lenguaje común, conocimiento común y mejora de la comunicación en la empresa.

Data mart

El *data warehouse* es una gran estructura. En muchas ocasiones, para facilitar el manejo de los datos, es necesario utilizar estructuras de datos más pequeñas llamadas *data mart* (ver Figura 6). El propósito es ayudar a que un departamento específico dentro de la empresa pueda tomar mejores decisiones. Los datos existentes en este contexto pueden ser resumidos, agrupados y explotados de múltiples formas para diversos grupos de usuarios.

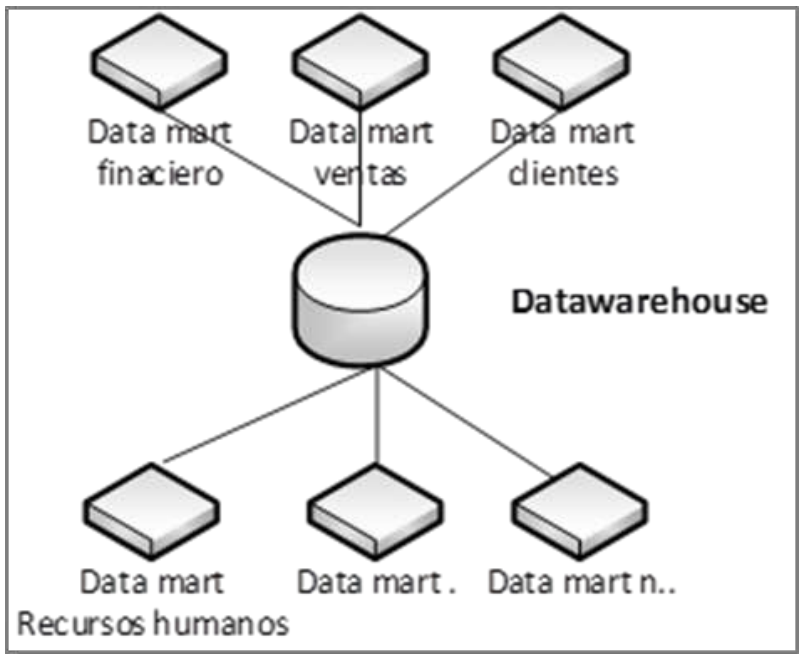


Figura 6. Ejemplo de *data mart.*

Los *data mart* están dirigidos a un conjunto de usuarios dentro de la empresa, que puede estar formado por los miembros de un departamento, por los usuarios de un determinado nivel administrativo o por un grupo de trabajo multidisciplinar con objetivos comunes.

Los *data mart* están compuestos por partes del DW primario, que en algunos casos

pueden ser:

- **Dependientes:** utilizan los datos y metadatos del *data warehouse* directamente en lugar de obtenerlos de los sistemas de producción.
- **Independientes:** los datos son tomados de cada área de la empresa, siempre manteniendo los datos alineados con el DW, si este existe. Aunque los *data mart* no son estrictamente necesarios, son muy útiles para los sistemas de almacenamiento de datos en medianas y grandes empresas debido a que:
  - Se usan como bloques de construcción mientras se desarrollan depósitos de datos de forma incremental.
  - Marcan la información requerida por un grupo específico de usuarios para resolver consultas más rápidas por el menor volumen de datos.
  - Pueden ofrecer un mejor rendimiento porque son más pequeños que los *data warehouse* primarios. Por lo tanto, son más fáciles de implementar.
  - Al ser pequeños los conjuntos de datos consumen menos recursos.

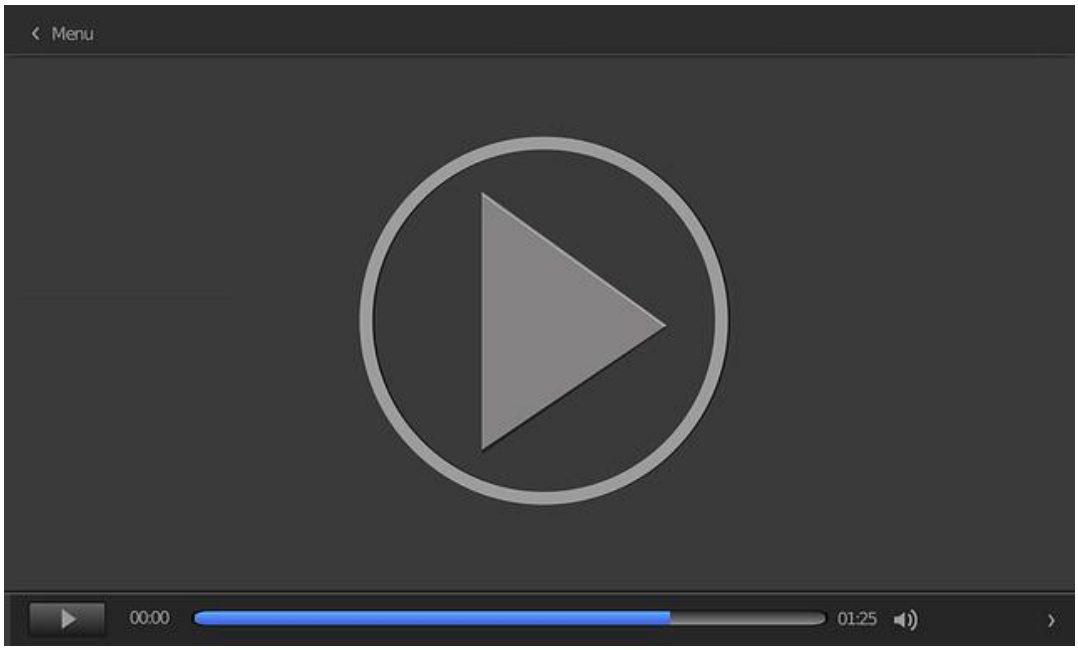
Los metadatos

Un componente esencial de un *data warehouse* son los metadatos. Es el repositorio central de información que abarca todos los niveles. Da el significado de cada uno de los componentes, variables y atributos que residen en el DW o *data mart.* La información que contiene los metadatos es útil para los departamentos y los propios usuarios. Incluye localizaciones, estructura, definiciones de negocio, descripciones minuciosas de los tipos de datos, significado, formatos, la cantidad y otras características, como los valores máximos y mínimos de los datos. En otras palabras, mapean los datos.

La información más importante va dirigida hacia:

- **El usuario:** información sobre el significado de los datos utilizados y su localización en el *data warehouse.*
- **Equipo responsable de los procesos de transformación de los datos:** información sobre la ubicación del dato en los sistemas de producción y los procesos de transformación.
- **Equipo responsable de los procesos de creación de nuevos datos a partir de los datos detallados.**

A continuación, accede al vídeo *Metadatos.*



Accede al vídeo:

https://unir.cloud.panopto.eu/Panopto/Pages/Embed.aspx?id=29e9caea-5899-41de-9e0b-ad66008e96d7

3.4. Lago de datos (data lake)

Puede definirse como un almacén de datos o un repositorio de grandes cantidades de datos que son útiles para realizar análisis. Los datos se almacenan en una arquitectura plana en lugar de una forma jerárquica, como se hace con los almacenes de datos o DW. Los datos almacenados pueden ser de cualquier tipo: datos estructurados (filas y columnas), semiestructurados (CSV, JSON, XML) y no estructurados (PDF, documentos, fotos, vídeos, correos). Es necesario crear metadatos para poder tener información adicional de cada dato almacenado. Si un lago de datos no proporciona valor para los usuarios o es inaccesible, se denomina pantano de datos.

Es necesario implementar un esquema de lectura para que los científicos y analistas de datos puedan realizar análisis predictivos, descubrir conocimiento y generar herramientas de visualización, entre otros procesos posibles. La transformación de datos se realiza en la etapa en la que se leen los datos.

Cuando se crea un *data lake,* el proceso ETL (extracción, transformación y carga) cambia a ELT (extracción, carga y transformación). Los datos se almacenan sin procesar (Nair, 2018).

En la siguiente tabla se encuentran las diferencias entre ETL y ELT.

	ETL	ELT
Procesamiento	Los datos se transforman en un servidor de almacenamiento intermedio antes de subirse al *data warehouse*.	Los datos se suben al *data lake* y allí permanecen. Las transformaciones se realizan en el sistema de destino.
Tiempo de carga	Los datos se cargan en un almacén intermedio y luego se mueven al sistema destino objetivo. Tiempo de carga intensivo.	Los datos se cargan una sola vez. Tiempo de carga muy rápido.
Tiempo de mantenimiento	Altos niveles de mantenimiento.	Bajo nivel de mantenimiento.
Complejidad de implementación	Lo complejo es tener la estructura y los procesos que llenarán esa estructura.	Se debe tener claro qué herramientas se van a utilizar y los *skills* necesarios.
Madurez	Este proceso se utiliza desde hace más de dos décadas, bien documentado y mejores prácticas disponibles fácilmente.	Es nuevo y complejo de implementar.

Tabla 1. Diferencias entre ETL y ELT. Fuente: adaptado de Ladrero, 2020.

3.5. Referencias bibliográficas

Barrer, R. (1998). *Managing a datawarehouse.*

Cano, J. L. (2007). *Business intelligence: competir con información.* ESADE Business School. http://itemsweb.esade.edu/biblioteca/archivo/Business_Intelligence_competir_con_informacion.pdf

Devlin, B. (1997). *Data warehouse: From architecture to implementation.* Addison-Wesley.

Eckerson, W., y White. C. (2003). *Evaluating ETL and Data Integration Platforms.* TDWI Report Series.

Esquema en copo de nieve. (7 de junio de 2020). En *Wikipedia.* https://es.wikipedia.org/wiki/Esquema_en_copo_de_nieve

Esquema en estrella. (2 de mayo de 2021). En *Wikipedia.* https://es.wikipedia.org/wiki/Esquema_en_estrella

Hüsemann, B., Lechtenbörger, J., y Vossen, G. (2000). Conceptual Data Warehouse Design. *Proc. of the International Workshop on Design and Management of Data Warehouses.*

Inmon, W. H. (1992). *Building the data warehouse.* Wiley. https://epdf.pub/building-the-data-warehouse.html

Jarke, M., Jeusfeld, M. A., Quix, C. J., Vassiliadis, P., y Vassiliou, Y. (2013). Data warehouse architecture and quality: impact and open challenges. En J. Bubenko, J. Krogstie, O. Pastor, B. Pernici, C. Rolland y A. Sølvberg (eds.), *Seminal contributions to information systems engineering* (pp. 183-189). Springer.

Kimball, R. (1996). *The data warehouse toolkit.* Wiley.

Ladrero, I. (12 de noviembre de 2020). ELT o ETL, ¿qué es mejor? [Página web]. *Baoss.* https://www.baoss.es/elt-o-etl-que-es-mejor/

Mendez, A., Mártire, A., Britos, P. y García-Martínez, R. (2003). Fundamentos de data warehouse. *Reportes técnicos en ingeniería del software, 5*(1), 19-26.

Nair, S., y Poornima, S. (2018). *Data lake: AWS & AZURE data lake, big data solutions & security.*

Ong, T. C., Kahn, M. G., Kwan, B. M., Yamashita, T., Brandt, E., Hosokawa, P., Uhrich, C., y Schilling, L. M. (2017). Dynamic-ETL: a hybrid approach for health data extraction, transformation and loading. *BMC Medical Informatics and Decision Making,* 17, 134.

Rizzi, S., y Golfarelli, M., (2009). *Data warehouse design: modern principles and methodologies.* McGraw-Hill Education.

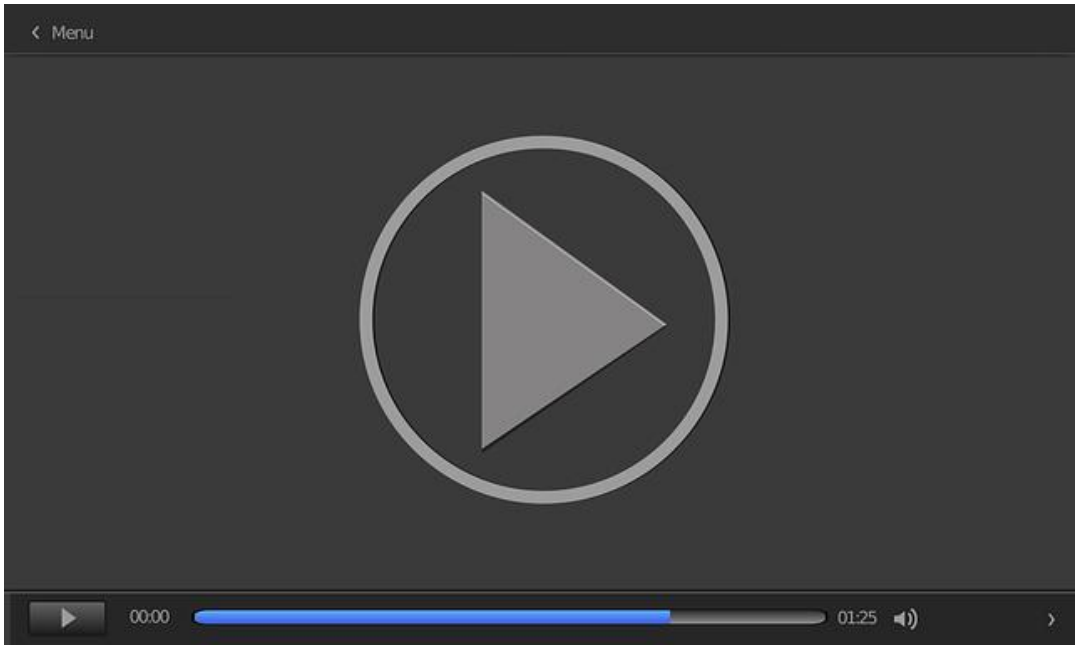
Theodoratos, D., Ligoudistianos, S., y Sellis, T. (2001). View selection for designing the global data warehouse. *Data & Knowledge Engineering, 39*(3), 219-240.

Villanueva, J. (2011). *Marco de trabajo basado en ontologías para el proceso ETL* (Trabajo Fin de Máster). Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, México.

Creando una ETL con las herramientas de Pentaho 6

Joseph Reyes. (6 de mayo de 2016). *Creando una ETL con las herramientas de Pentaho 6* [Vídeo]. Youtube. https://www.youtube.com/watch?v=a6nMj6M7IUU&t

Vídeo tutorial demostrativo para crear una ETL a partir de una base de datos transaccional, tomando como modelo un negocio de tipo tienda.



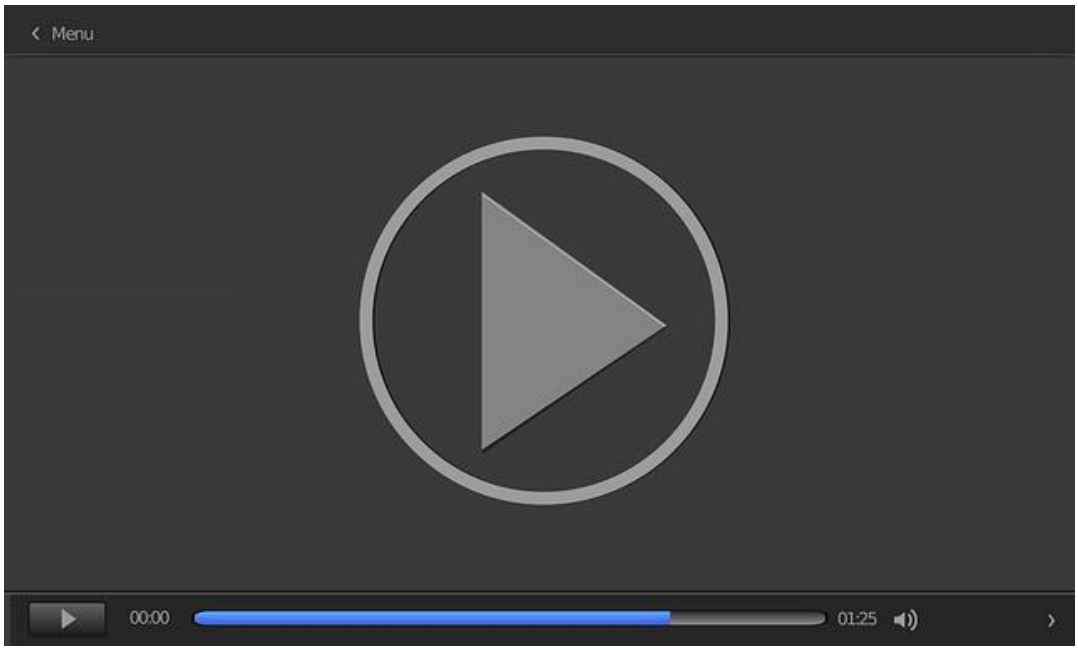
Accede al vídeo:

https://www.youtube.com/embed/a6nMj6M7IUU?si=V6Jgq4UW2KlIKP14

Desarrollo de un cubo OLAP con Schema Workbench de Pentaho

Auribox Training. (17 de junio de 2017). *Desarrollando un CUBO OLAP con Schema Workbench de Pentaho | Tutorial* [Vídeo]. Youtube. https://www.youtube.com/watch?v=eYAgvsT5dd4

En este vídeo podrás observar paso a paso la creación de un cubo con la herramienta Pentaho, de tipo *open source,* que integra todas las etapas de una estrategia BI.



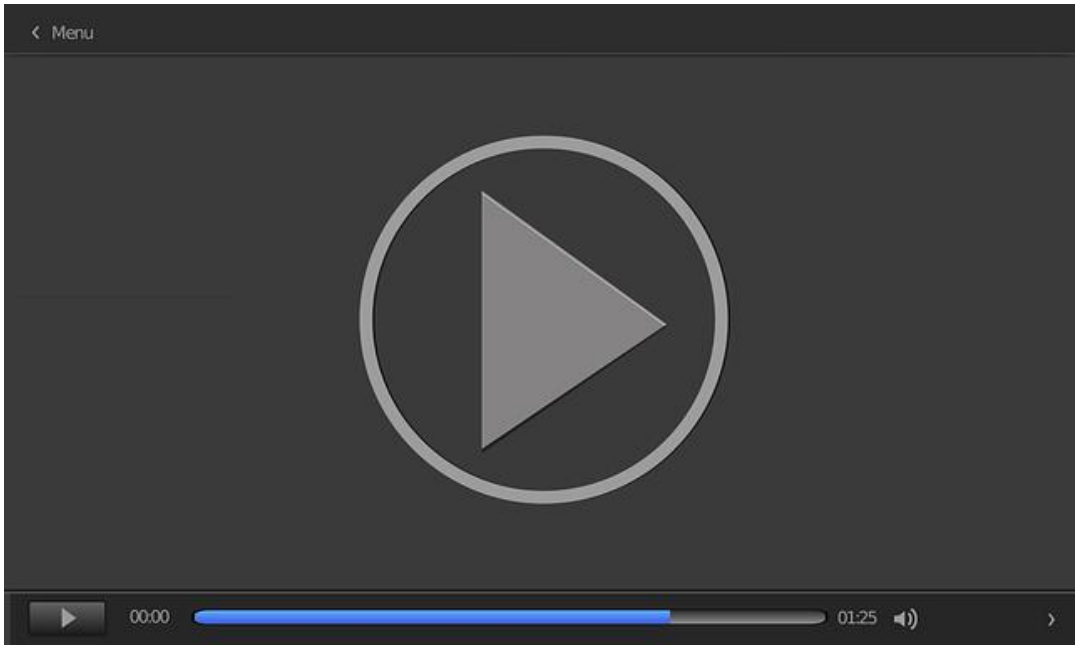
Accede al vídeo:

https://www.youtube.com/embed/eYAgvsT5dd4?si=ddgGezU7TVGfRuGr

Azure data lake storage tutorial

Adam Marczak - Azure for Everyone. (12 de diciembre de 2019). *Azure Data Lake Storage (Gen 2) Tutorial | Best storage solution for big data analytics in Azure* [Vídeo]. Youtube. https://www.youtube.com/watch?v=2uSkjBEwwq0

En este vídeo podrás ver una introducción a lo que sería construir un *data lake* en Azure, cómo trabaja y cómo aprovechar las ventajas de este tipo de almacenamiento en la nube.



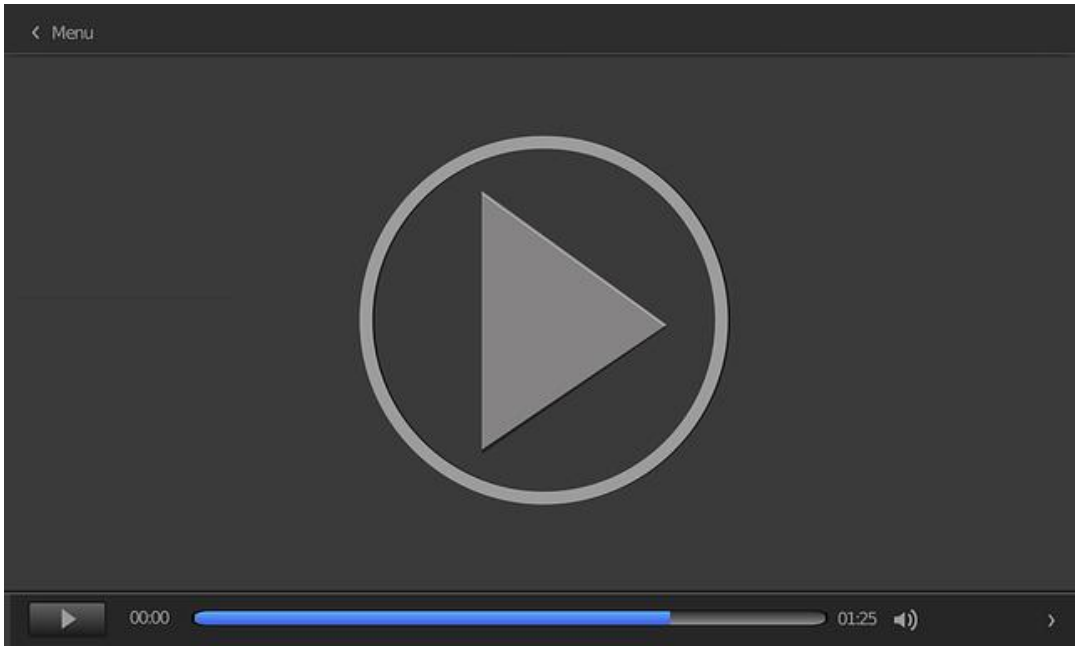
Accede al vídeo:

https://www.youtube.com/embed/2uSkjBEwwq0?si=unI5tWJI1haqtpm2

ETL vs. ELT

Astera Software. (28 de noviembre de 2019). *[WEBINAR]: ETL vs. ELT: A Data Integration Showdown* [Vídeo]. Youtube. https://www.youtube.com/watch?v=YOn9hGCwmrA

En este webinar hablan sobre las capacidades de cada uno de estos enfoques, cómo pueden usarse individualmente y combinarlos para un mejor rendimiento.



Accede al vídeo:

https://www.youtube.com/embed/YOn9hGCwmrA?si=xue1U8ehelyqQlnk

Test

1. ¿Cuáles son etapas del proceso ETL?

   A. Extracción.

   B. Transformación.

   C. Subida de datos brutos.

   D. A y B son correctas.

2. Los data mart:

   A. Son los metadatos del data warehouse.

   B. Son estructuras de datos específicas para un departamento, el conjunto de data marts compone un data warehouse.

   C. Permiten acceder directamente al data warehouse.

   D. Son una fuente de datos.

3. Son arquitecturas para implementar un data warehouse:

   A. Arquitectura mecánica.

   B. Arquitectura de una sola capa.

   C. Arquitectura de dos capas.

   D. B y C son correctas.

4. ¿Cuáles pueden ser dos posibles fuentes de datos para un data warehouse?

   A. Bases de datos relacionales y archivos de texto plano.

   B. Archivos XML y codificación de archivos HTML.

   C. Archivos PDF y documentos en papel.

   D. Ninguna de las anteriores.

5. ¿Cuáles pueden ser posibles fuentes de datos para un data lake?

A. Bases de datos relacionales y archivos de texto plano.

B. Archivos XML y codificación de archivos HTML.

C. Archivos PDF y fotos.

D. Todos las anteriores.

6. El autor Bill Inmon definió las características que debe cumplir un data warehouse. ¿Cuáles son?

A. Orientado a un área e integrado.

B. Portátil y fácil de manejar.

C. Indexado en el tiempo y no volátil.

D. Consistente y fácil.

7. ¿Cuál es la función del data warehouse y del data lake?

A. Aumentar el trabajo de los usuarios.

B. Ayudar en la toma de decisiones.

C. Centralizar los datos para facilitar el manejo.

D. Ninguna de las anteriores.

8. Es falso si hablamos de ETL:

A. Los datos se transforman en un servidor intermedio antes de subir al DW.

B. El tiempo de carga, sobre todo la primera vez, es muy rápido.

C. Altos niveles de mantenimiento.

D. Las estructuras pueden llegar a ser complejas.

9. Es falso si hablamos de ELT:

A. Los datos se cargan y se transforman en un servidor intermedio antes de subir al DW.

B. El tiempo de carga es muy rápido.

C. Bajo nivel de mantenimiento.

D. Es nuevo y complejo de implementar.

10. Es cierto si hablamos de metadatos:

A. Son un repositorio central de información.

B. Da significado a cada componente, variable y atributo que reside en el DW.

C. Contiene información sobre la estructura del data lake.

D. A y B son verdaderos.