Formación Profesional en CePETel 2023

Desde la Secretaria Técnica del Sindicato CePETel convocamos a participar del siguiente curso de formación profesional:

Big Data & Analytics – Parte 2

Clases: 8 de 3hs c/u de 18:00 a 21:00 hs.
Días que se cursa: los días martes 21 y 28 de marzo; 4, 11, 18 y 25 de abril; 2 y 9 de mayo.
Modalidad: a distancia (requiere conectarse a la plataforma Zoom en los días y horarios indicados precedentemente).
Docentes: María Trinidad Aquino y Raúl Alejandro Grassi

La capacitación es:

- **Sin cargo para afiliados y su grupo familiar directo.**
- **Sin cargo para encuadrados con convenio CePETel.**
- **Con cargo al universo no contemplado en los anteriores.**

Informes: enviar correo a tecnico@cepetel.org.ar

Inscripción (hasta el 16 de marzo): ingresar al formulario (se recomienda realizar el registro por medio de una cuenta de correo personal y ***no utilizar dispositivos de la empresa para acceder al link***).

https://forms.gle/zi9cgmHkgEN7gZyK8

Temario:

Módulo 1

Modelado Dimensional

(Clases 1, 2 y 3)

- Programación distribuida
- Ejemplos de frameworks de programación distribuida
- Definición del modelo dimensional
- Concepto DER
- Conceptos del modelo dimensional
- Arquitectura estándar del modelo dimensional

Ing. Daniel Herrero – Secretario Técnico – CDC

- Diferencias entre el modelo relacional y dimensional. ¿Cuál es más conveniente y por qué?
- Beneficios del modelado dimensional
- Componentes del Modelado Dimensional: Hechos, Atributos, Elementos y Dimensiones
- Toma de Decisiones Basada en Datos: Cultura Data Driven

TP Clase 1: Identifique la diferencia entre programación distribuida vs programación paralela

TP Clase 2: A partir de un modelo relacional construir un modelo dimensional con tablas de hechos y dimensiones (diagrama DER)

TP Clase 3: Investigar cuales son las características requeridas de un Data Warehouse (enumere al menos 4).

Módulo 2

Data Warehouse

(Clases 4 y 5)

- Definición de Data Warehouse. Objetivos y Características de un DW.
- Arquitectura estándar.
- Dimensiones conformadas
- Data Marts y Data Warehouse
- Modelado de Datawarehouse según enfoque: Snowflaked vs Star Schema.
- Proceso de modelado
- Tablas de Hechos y Dimensiones
- Nivel de Granularidad
- DW: Modelado vs. Desnormalización
- Proceso ETL. Sistemas más usados (PDI, Data Stage)
- Almacenamiento basado en filas vs columnas

TP Clase 4: A partir del modelo relacional construido en la clase 2 identificar las dimensiones y medidas. Si fuera necesario modificar el modelo siguiendo el enfoque estrella. Identificar las dimensiones conformadas.

TP Clase 5: Armar un modelo de ingesta de datos a un Data Warehouse. Incluir fuentes de datos internas y externas. Marcar como mínimo 5 diferencias entre procesos de ETL y ELT.

Módulo 3

Data Lake

(Clases 6, 7 y 8)

- ETL vs ELT – Cambio de Paradigma de BI
- Definición de Data Lake
- Diferencias y similitudes entre ambos sistemas (DW vs DL)
- Desafíos a la hora de implementar un Data Lake
- Recomendaciones para crear un Data Lake
- Tecnologías para un Data Lake
- Disaster Recovery y su importancia
- Arquitectura de un Data Lake
- Ingestas de Datos Batch, Micro Batch y Real Time
- Zonas en un Data Lake
- Ecosistema Hadoop
- Componentes de Hadoop
- Sistema de Almacenamiento Distribuido

TP Clase 6: Identificar si las siguientes características corresponden a un Data Warehouse o Data Lake.

TP Clase 7: Identificar a que framework del ecosistema Hadoop corresponden las características detalladas

Acerca de los docentes

María Trinidad Aquino: 2007-12–03 al momento Analista Senior Marketing – Región Patagonia Movistar, Neuquén.
* Proyecto Canal Presencial (Agentes y CEC) responsable de la construcción y disposición de los datos (KPI´s) para ver su evolución con aporte analítico, diagnóstico y sugerencia de planes de acción para su mejora.
*Proyecto Educador Digital País, referente de la región Patagonia con seguimiento de los KPI´s y evolución.
* Nuevos proyectos del área con análisis de datos, participación en el diseño, elaboración y difusión de lo implementado.

Formación académica:

- 2020-09 – 2021-10 (finalizada) Big Data, Data Engineer (Diplomatura) ITBA, CABA
- 1999-03 - 2003-12 (finalizada) Ciencias Sociales, Licenciada en Relaciones Públicas Universidad Nacional de Lomas de Zamora, Lomas de Zamora

Raúl Alejandro Grassi: desde 1995 hasta la fecha TELEFONICA DE ARGENTINA S.A. Puesto:Analista Senior - Sector Big Data Comercial.
Responsabilidades: Definición de inversiones anuales en capital (CapEx) en base a análisis de proyección comercial. Gestión de proyectos y seguimiento de inversiones. Diseño e implementación de modelos de aseguramiento de satisfacción de clientes. Planeamiento y ejecución de tableros de control y análisis del negocio basado en datawarehousing (heavy user) en los últimos 10 años, programando en SQL y modelado de datos. Análisis y Evaluación de acciones que impacten en cumplimiento de objetivos del Negocio B2C. Analista Senior BI, desarrollo en herramientas de explotación de BI (Microstrategy; Tibco Spotfire, Power BI, Tableau, etc.) y ecosistema Hadoop (Spark, Hive, SQL, procesos de ingestas ETL, ELT, etc.).
Desempeño durante 4 años en el sector Data Driven Comercial, promoviendo la cultura Data Driven y desarrollando tableros de control predictivos y prescriptivos con herramientas de explotación basadas en modelos relacionales/dimensionales.
Experiencia al menos 7 años como líder de proyectos, Manejo de Metodologías Agiles en
posiciones como Stakeholder, Scrum Master y PO.

Formación académica:

- 2020-2021 Licenciatura en Big Data – especialista en Data Engineer - ITBA (Instituto Tecnológico de Buenos Aires)
- 1999 Posgrado en Gestión Gerencial Avanzada (Managemente Executive Program) Universidad Argentina de la Empresa (UADE)
- 1986-1992 Ingeniero Electrónico Universidad de Buenos Aires

María Trinidad Aquino y Raúl Alejandro Grassi dictaron de manera virtual y para el Sindicato CEPETel Big Data & Analytics – Parte 1 durante el año 2022.

http://www.cepetel.org.ar tecnico@cepetel.org.ar Rocamora 4029 (CABA) (+54 11)35323201

CePETel
Sindicato de los Profesionales
de las Telecomunicaciones

SECRETARÍA TÉCNICA

IPEI
Instituto Profesional de
Estudios e Investigación

BIG DATA & ANALYTICS II

TEMARIO
Módulo 1: Modelado Dimensional
Módulo 2: Data Warehouse
Módulo 3: Data Lake

Disertantes: Lic. Maria Trinidad Aquino – Ing. Raúl Alejandro Grassi

BIG DATA & ANALYTICS II

Módulo 1: Modelado Dimensional

Programación Distribuida

Según Wikipedia
“La **programación distribuida** es un paradigma de programación enfocado en desarrollar **sistemas distribuidos, abiertos, escalables, transparentes y tolerantes de fallos**. Este paradigma es el resultado natural del uso de las computadoras y las redes.”

¿Por qué distribuir el sistema informático?

- El sistema interactúa con un entorno distribuido geográficamente
- Hay recursos que son compartidos por muchas aplicaciones
- Se prevé escalar el sistema en varios ordenes de magnitud
- Se necesita acumular la potencia de cálculo de muchos computadores
- El sistema requiere alta disponibilidad

CePETel

Programación Distribuida

Objetivos

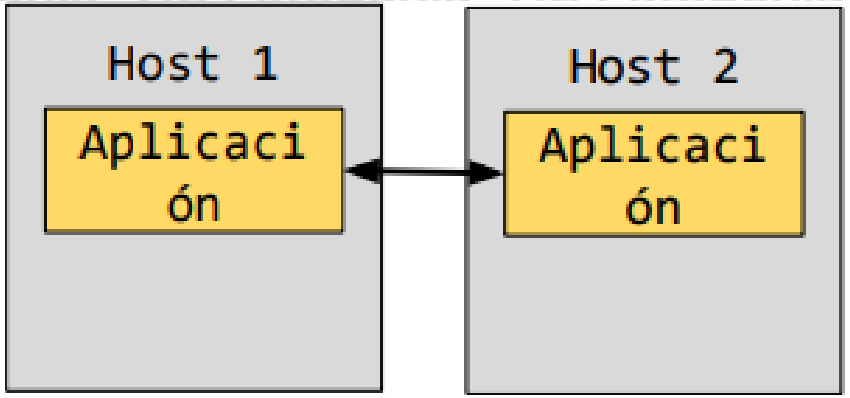
- Permitir el acceso a los recursos disponibles
  - busca optimizar la eficiencia
- Uso transparente de los recursos
  - acceso local/remoto, concurrente, reubicación, múltiples copias
- Tolerancia a fallos
  - los fallos afectan parcialmente al sistema distribuido
- Escalabilidad
  - aumento de recursos/usuarios
- Integración de sistemas heterogéneos

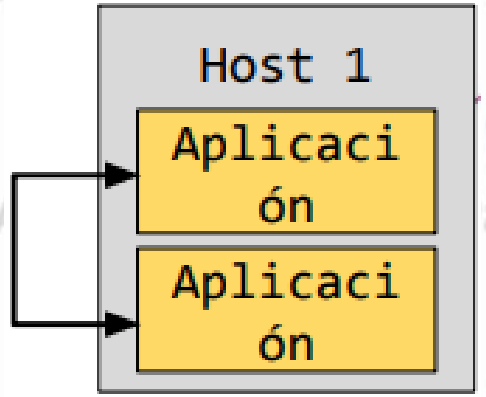
CePETel

Programación Distribuida

Técnica de programación ampliamente usada para **desarrollar aplicaciones** que **se comunican entre sí** mediante el **pase de mensajes** a través de una red.

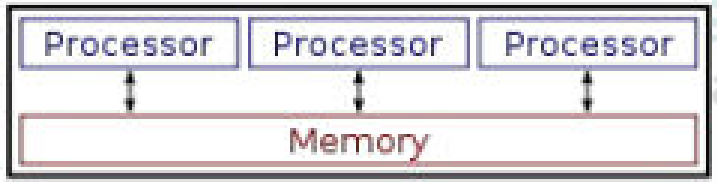
Con la programación distribuida se pueden construir sistemas distribuidos.



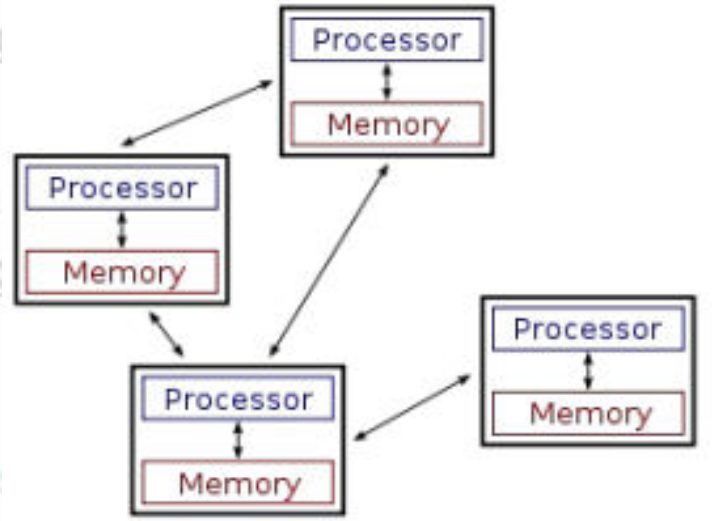


CePETel

Programación Distribuida



Parallel computing
Todos los procesadores tienen acceso a una memoria compartida.



Distributed computing
Cada procesador tiene acceso a su propia memoria.

CePETel

Progamación Distribuida

Múltiples procesadores -> Programación paralela

- C
- Java
- Scala
- MPI

Separación geográfica -> Programación distribuida

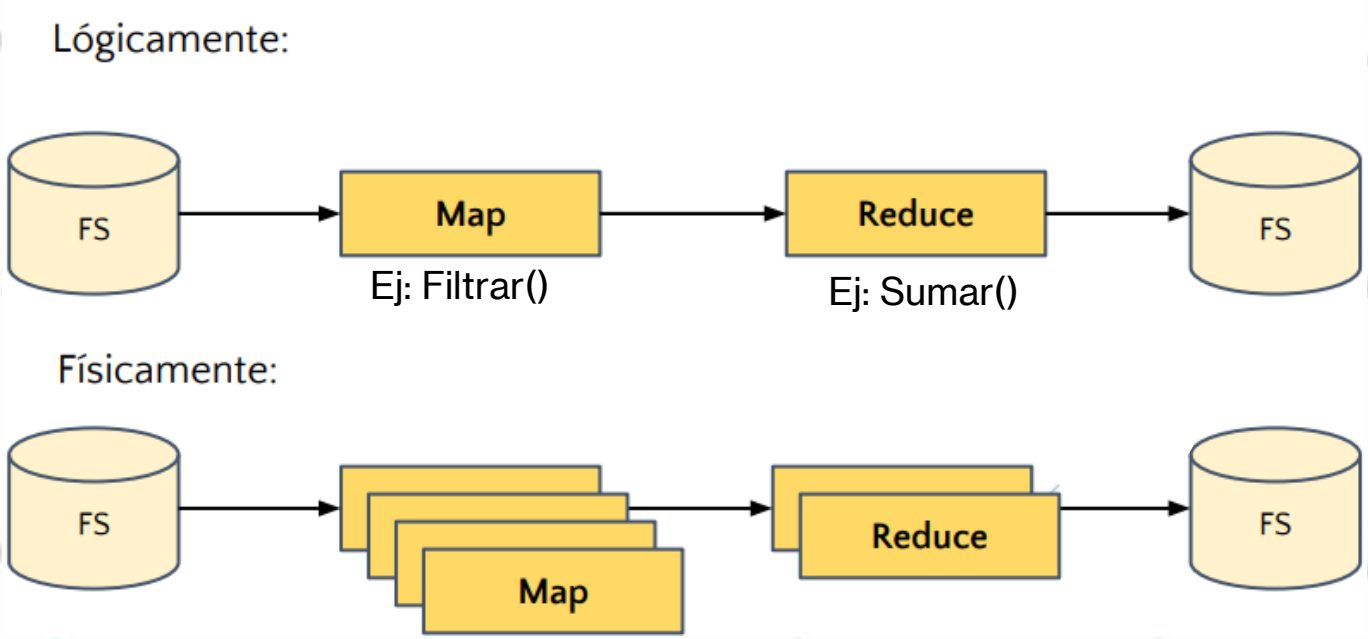
- Map reduce
- Spark
- Zookeeper

CePETel

Ejemplos de Frameworks de Programación Distribuida

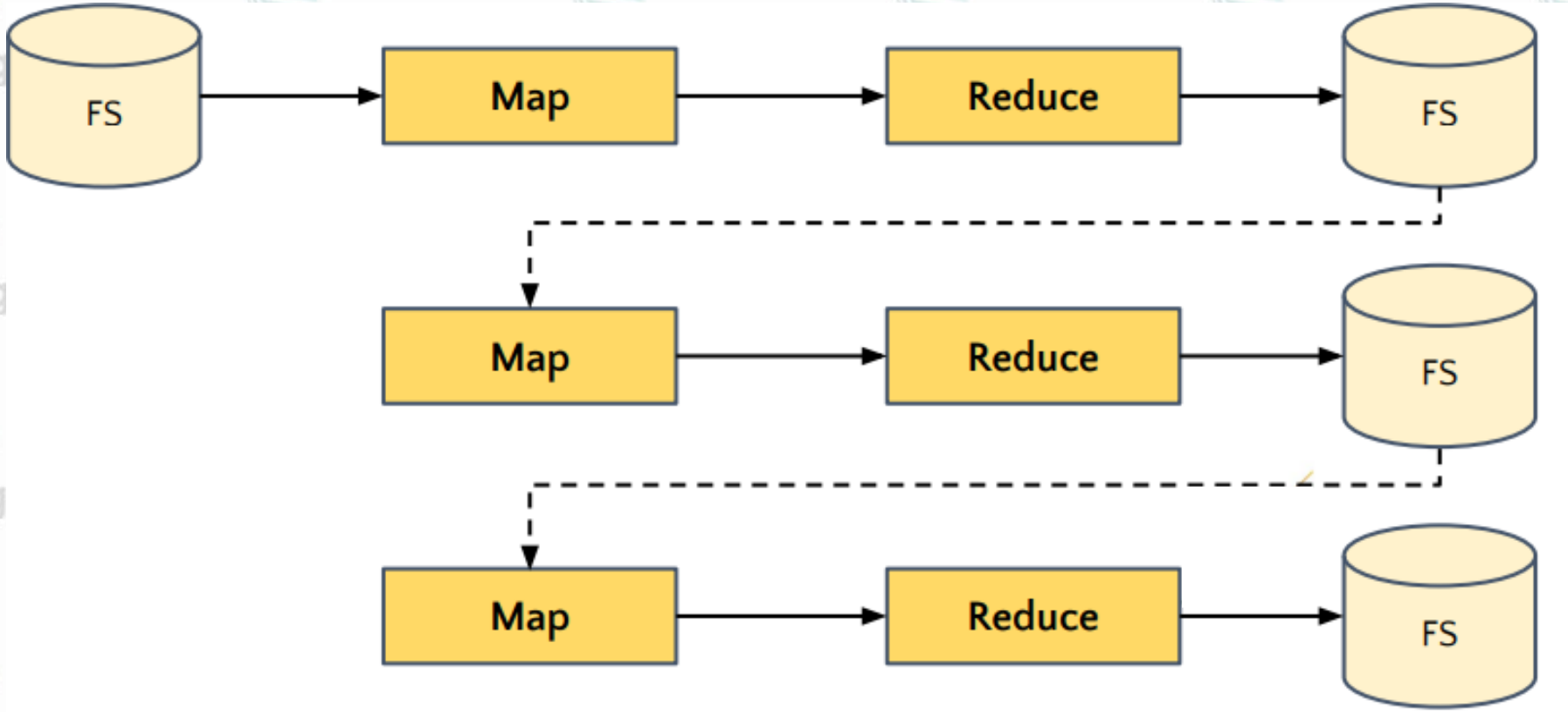
Map Reduce

- **Map reduce** es un **método de procesamiento distribuido.**
- Diseñado en **google y abierto al público en 2004** a través del paper MapReduce: Simplified Data Processing on Large Clusters
- Derivado del paradigma funcional y **compuesto por dos primitivas (Map y Reduce)**
- Permite al usuario **programar solamente las primitivas** (map/reduce) **y olvidarse de las problemáticas de la programación distribuida:** Sincronización, Coordinación, Distribución adecuada.



CePETel

Map Reduce



CePETel

Spark

- Un **framework de procesamiento distribuido** para el análisis de Big Data.
- Provee análisis de datos en memoria.
- Diseñado para ejecutar algoritmos iterativos y realizar análisis predictivos.
- Compatible con los medios de almacenamientos en Hadoop

Por qué usar Spark?

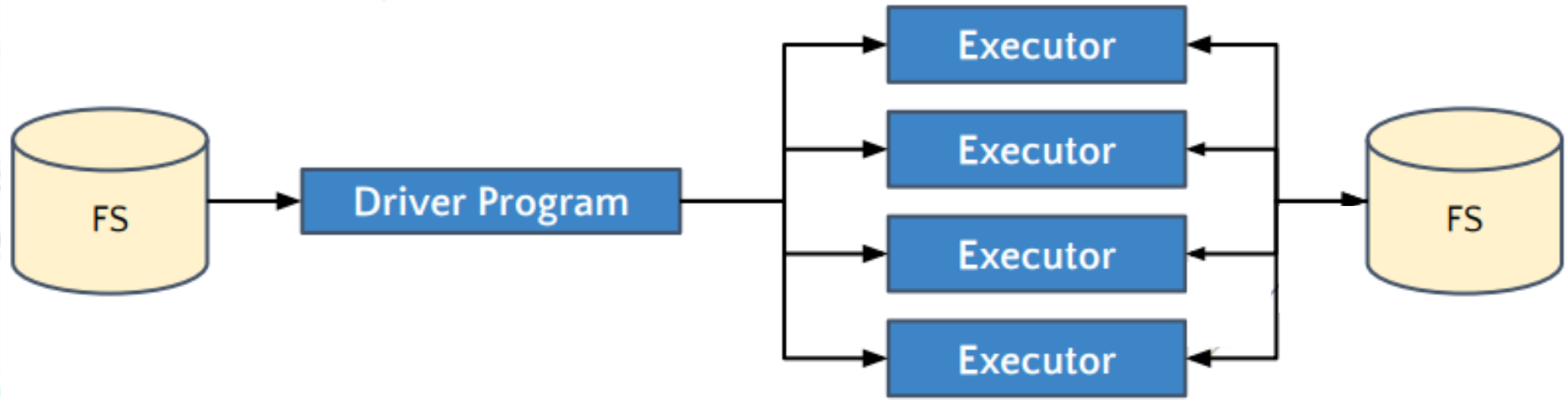
- **Rápido**: Batch y Streaming, DAG scheduler, optimización de queries, motor de ejecución
- **Fácil de usar**: Programable en Java, Scala, Python, R y SQL
- **Pensado para uso general**: Se pueden combinar SQL, streaming y analítica compleja, utilizando datos de múltiples fuentes
- **Versatilidad de implementaciones**: Deploy en Hadoop, la nube, standalone.

CePETel

Spark

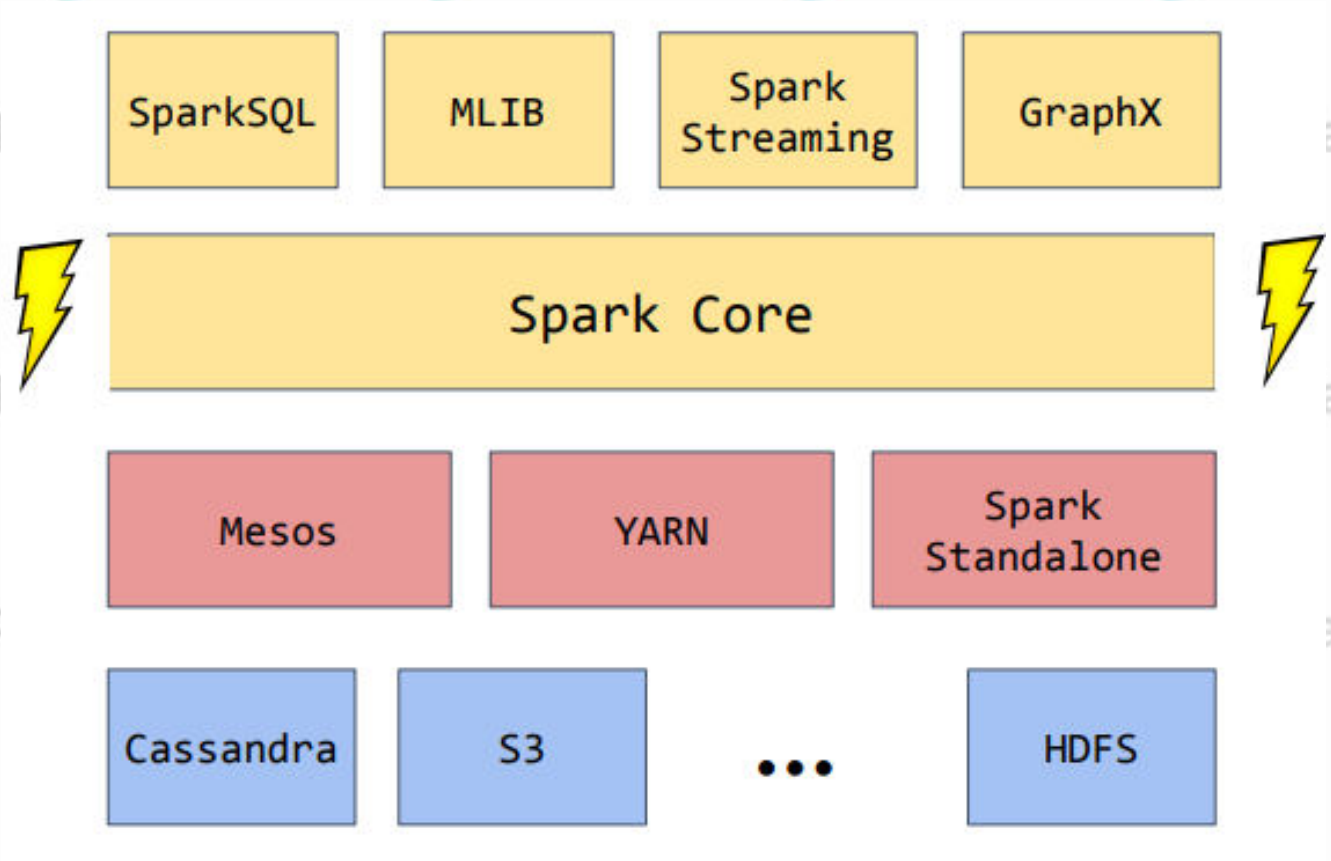
Procesamiento distribuido de propósito general:

- Map reduce
- Métricas
- SQL
- Machine learning



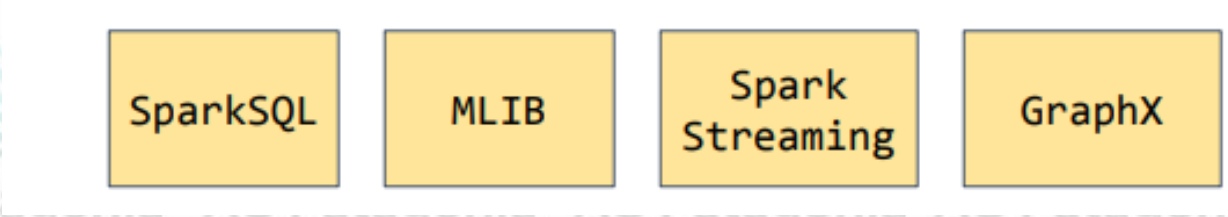
CePETel

Spark Stack



CePETel

Spark Stack



- **SparkSQL** Paquete de spark diseñado para trabajar con **información estructurada**. Utiliza una **variante de HQL** (Hive Query Language) y permite la conexión con **múltiples fuentes y formatos** (Hive, NoSQL, RDBMS, Parquet, Avro, JSON).
- **Mlib** Funcionalidad y modelos de **machine learning**. Incluye **modelos** que poseen capacidad de **aprendizaje distribuido** (clasificación, regresión, clustering, filtros colaborativos).
- **Spark Streaming** Componente que permite **procesar streams** de información en **tiempo real**. Implementa un mecanismo de **micro batches** y permite **interoperar** con spark-core, sparksql y mlib para procesamiento brindando las mismas características de tolerancia a fallas, escalabilidad y distribución que spark-core.
- **GaphX** Librería para **manipulación de grafos** de manera distribuida. Incluye **algoritmos** comunes como **PageRank, conteo de triángulos y componentes conexas.**

CePETel

Spark orientado a APPs

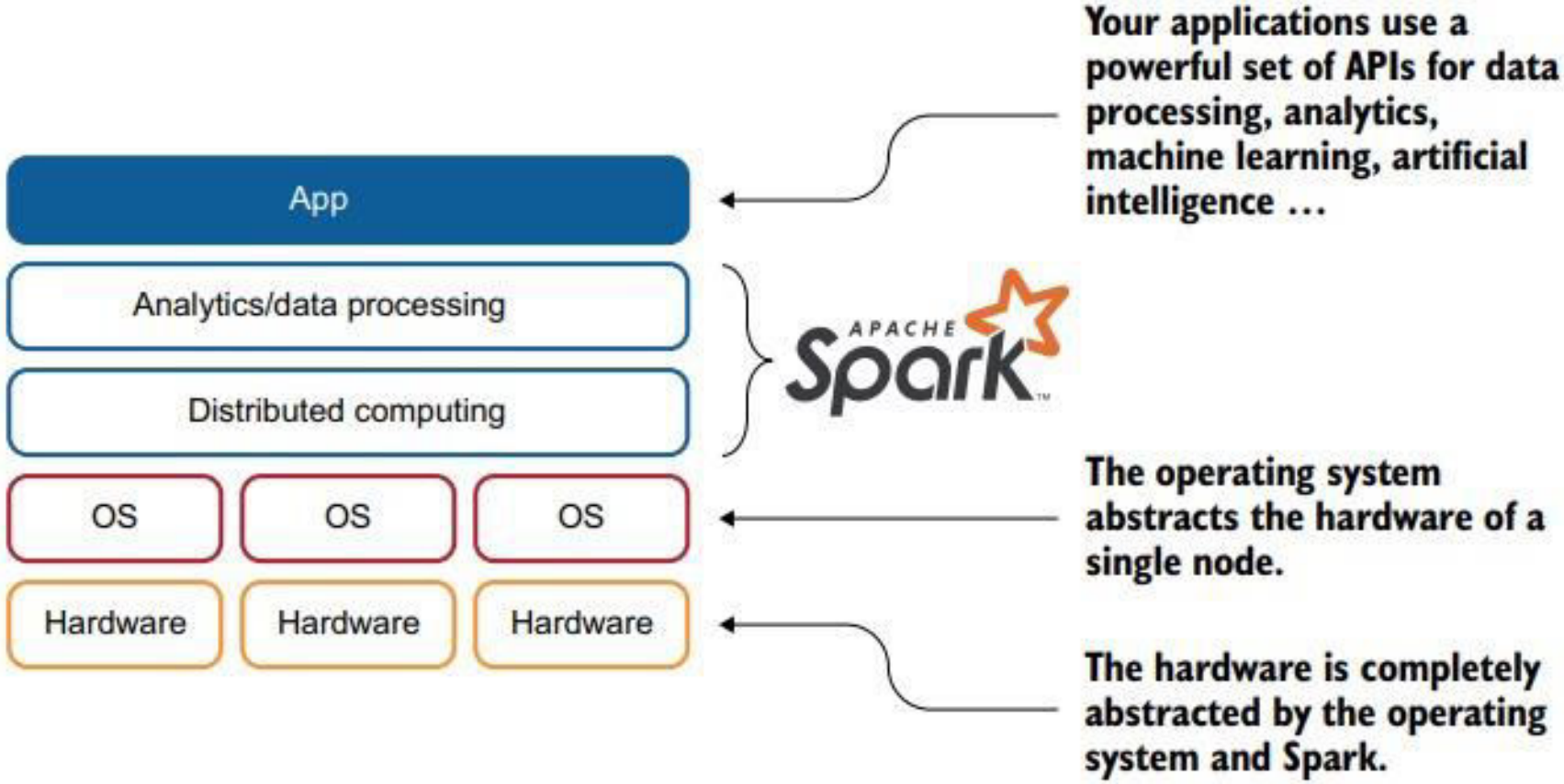


Figure 1.3 Apache Spark simplifies the development of analytics-oriented applications by offering services to applications, just as an operating system does.

CePETel

Apache Tez

Apache Tez es un **framework para la creación de un gráfico acíclico dirigido (DAG)** complejo de tareas para el procesamiento de datos.
En algunos casos, se utiliza como una alternativa a Hadoop MapReduce.
Tez **mejora** drásticamente el **paradigma de MapReduce** al ofrecer **mayor velocidad** sin renunciar a la capacidad de MapReduce para escalar a petabytes de datos.
Algunos proyectos importantes del ecosistema Hadoop, como Apache Hive y Apache Pig, usan Apache Tez, al igual que un número cada vez mayor de aplicaciones de acceso a datos de terceros desarrolladas para el amplio ecosistema Hadoop.

- Rendimiento de ejecución
- Mejoras de rendimiento sobre Map Reduce
- Gestión óptima de recursos
- Reconfiguración del plan en tiempo de ejecución
- Decisiones de flujo de datos físicos dinámicos

CePETel

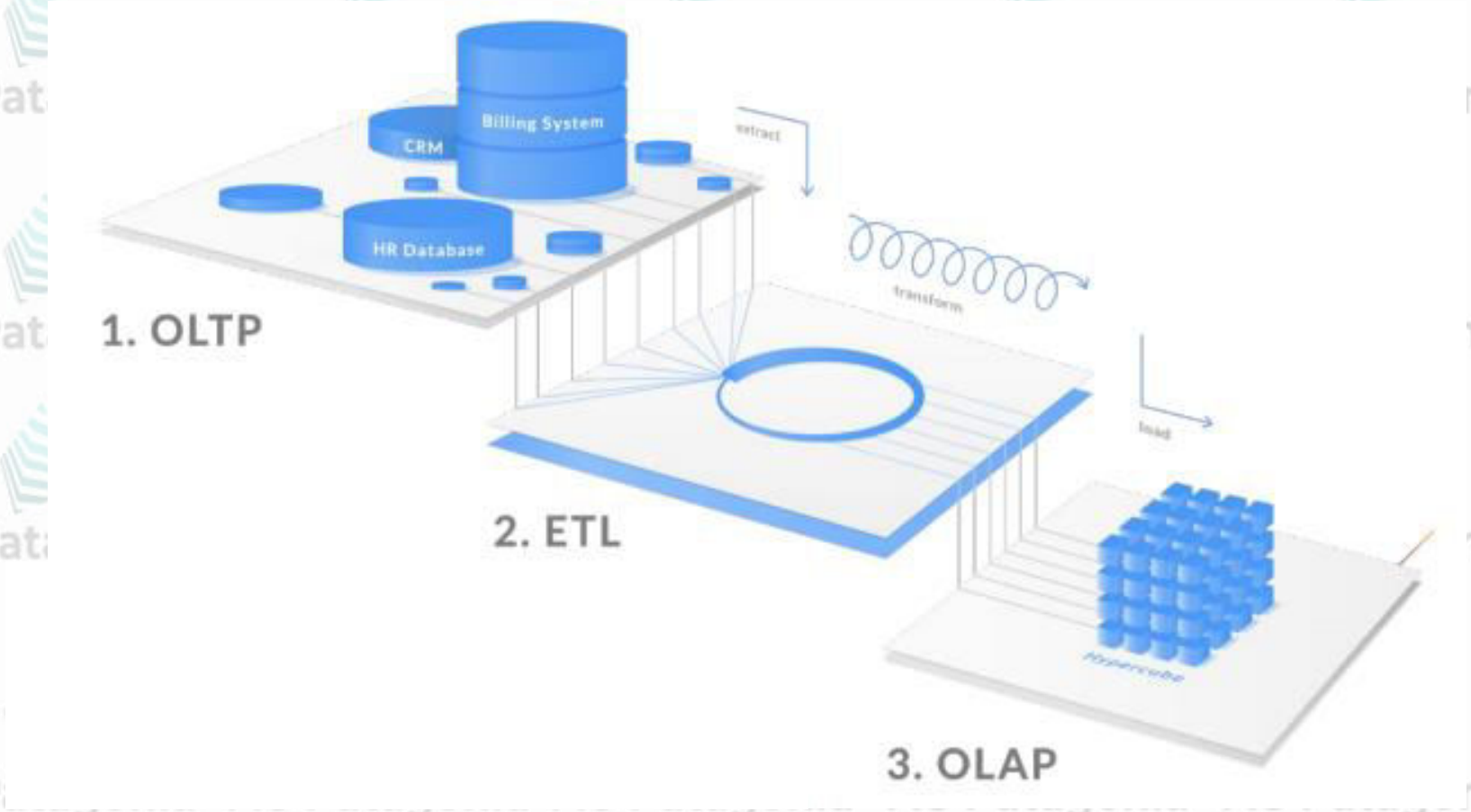
Modelado de Datos

Modelado Relacional (OLTP)

Modelo Relacional – Arquitectura Estándar DW

Las bases de datos **OLTP** registran transacciones una a la vez. El **DW** es diferente, **no necesita registrar detalles a nivel transaccional.**

DW necesita tener datos sobre diferentes criterios de su negocio y agregar (o permitir que los analistas agreguen) la información requerida para mejorar el negocio.



[Video: Modelo Relacional y Dimensional]

CePETel

Modelo Relacional

Su idea fundamental es el uso de relaciones. Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados tuplas. Pese a que esta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar, pensando en cada relación como si fuese una tabla que está compuesta por registros (cada fila de la tabla sería un registro o “tupla”) y columnas (también llamadas “campos”).

Es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente.

Modelo Relacional

El modelo relacional desarrolla un esquema de base de datos (data base schema) a partir del cual se podrá realizar el modelo físico o de implementación en el DBMS.
Este modelo esta basado en que todos los datos están almacenados en tablas (entidades/relaciones) y cada una de estas es un conjunto de datos, por tanto una base de datos es un conjunto de relaciones. La agrupación se origina en la tabla: tabla -> fila (tupla) -> campo (atributo)

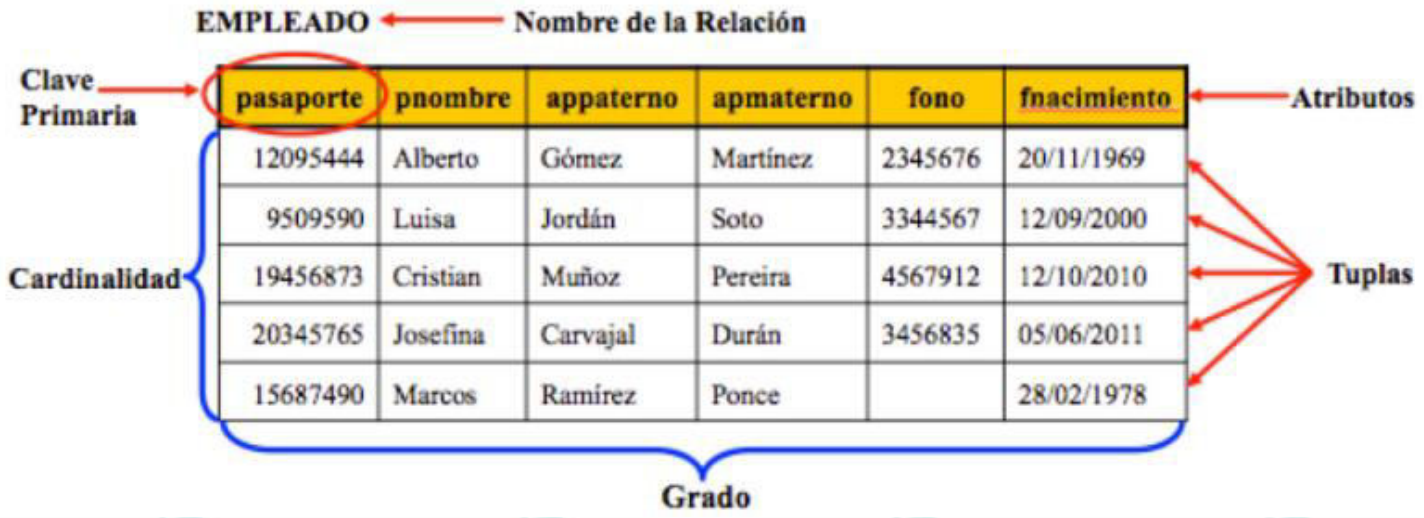
Terminología Relacional		Terminología de Tablas		Terminología de Archivo
Relación	=	Tabla	=	Archivo
Tupla	=	Fila	=	Registro
Atributo	=	Columna	=	Campo
Grado	=	Número de columnas	=	Número de campos
Cardinalidad	=	Número de filas	=	Número de registros

CePETel

Modelo Relacional

El Modelo Relacional se ocupa de:

- La estructura de datos
- La manipulación de datos
- La integridad de los datos



pasaporte	pnombre	appaterno	apmaterno	fono	fnacimiento
12095444	Alberto	Gómez	Martínez	2345676	20/11/1969
9509590	Luisa	Jordán	Soto	3344567	12/09/2000
19456873	Cristian	Muñoz	Pereira	4567912	12/10/2010
20345765	Josefina	Carvajal	Durán	3456835	05/06/2011
15687490	Marcos	Ramírez	Ponce		28/02/1978

Donde las relaciones estan formadas por :

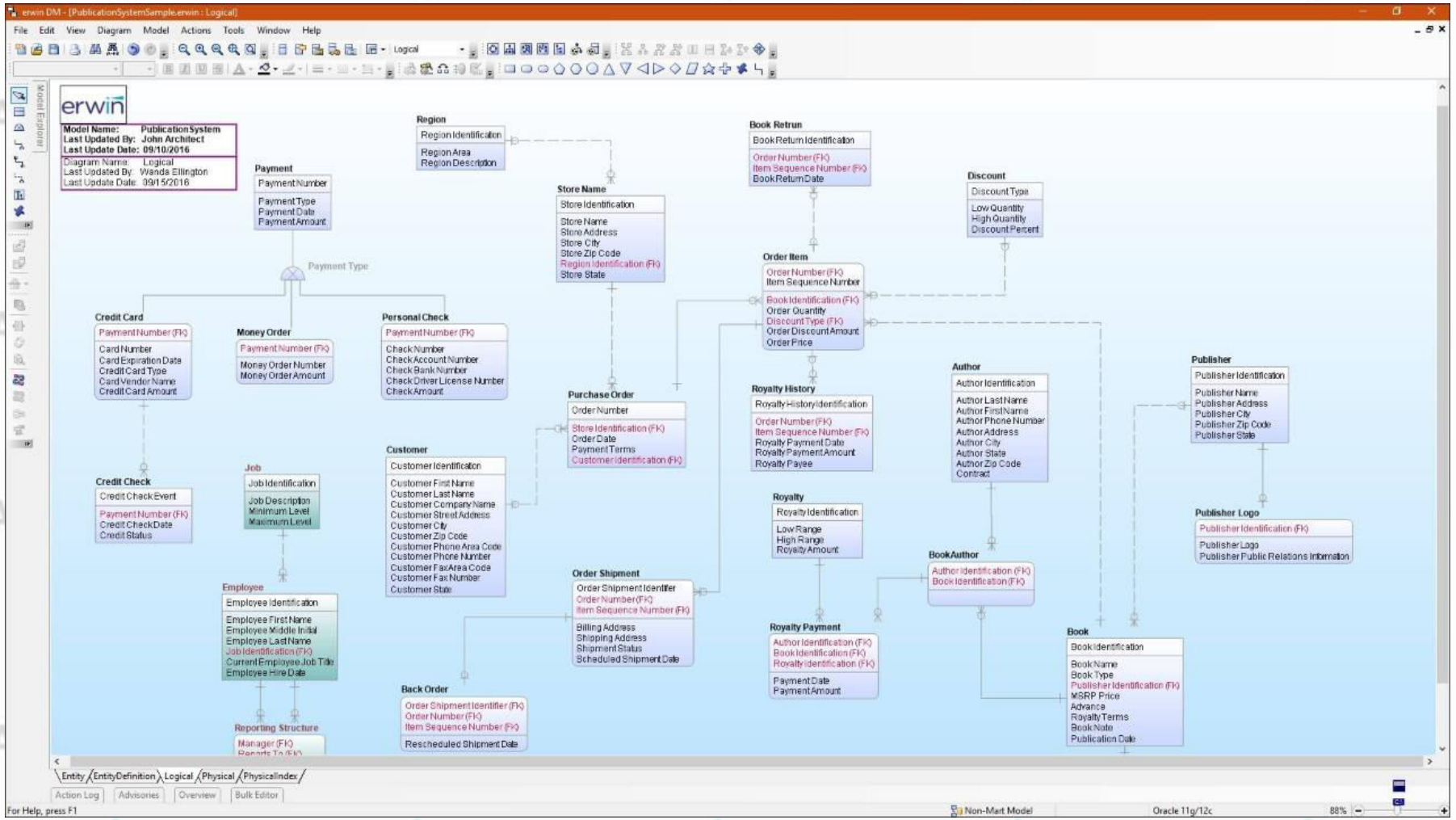
- Atributos (columnas)
- Tuplas (Conjunto de filas)

Existen dos formas para la construcción de modelos relacionales:

- Creando un conjunto de tablas iniciales y aplicando operaciones de normalización hasta conseguir el esquema más óptimo,
- O, convertir el modelo entidad relación (ER) en tablas, con una depuración lógica y la aplicación de restricciones de integridad.

CePETel

Modelo Relacional – Arquitectura Estándar DW



Desafíos que tiene que resolver - DER

Aunque el modelo entidad relacional orientado a transacciones es muy útil para la captura de transacción

se debe evitar en la entrega al usuario final.

CePETel

Modelado de datos – Modelo Relacional

En la mayoría de los paradigmas se separan las abstracciones de datos que representan **información** y los **procesos.**

Los **procesos** son abstracciones de control del flujo que manipulan esos datos.

En el diseño lógico de datos definimos

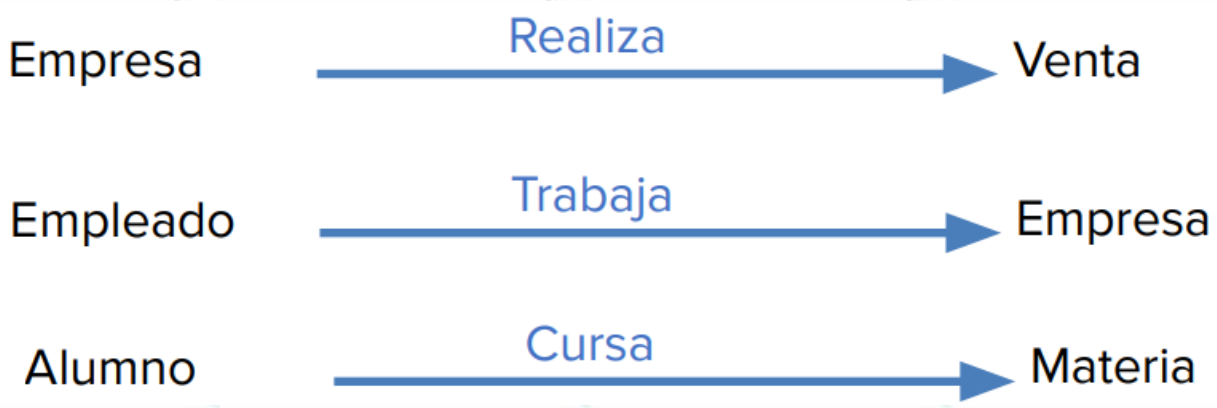
- las entidades que representan partes de un sistema
- sus relaciones.

CePETel

Modelado de datos – Modelo Relacional

Qué son las Relaciones?

- Representan asociaciones del mundo real entre entidades.
- Se puede representar con un verbo o preposición que conecta dos entidades.



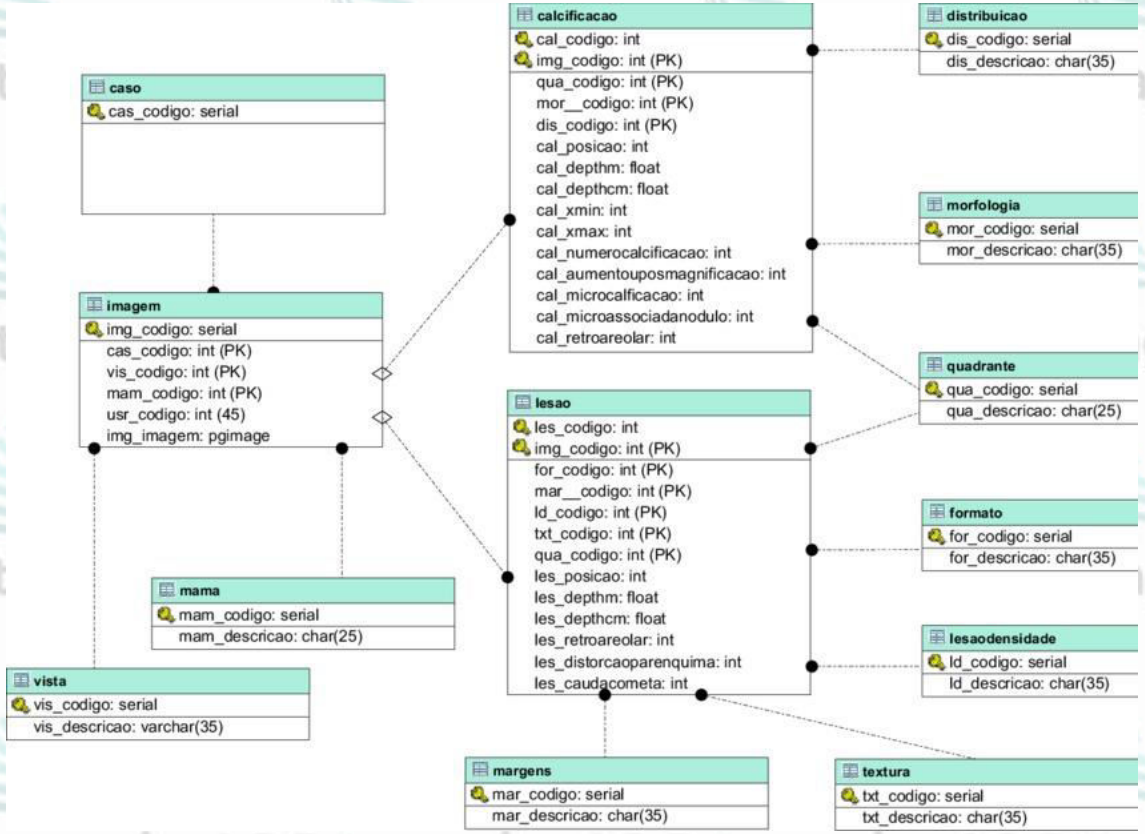
Modelado de datos – Modelo Relacional

¿Cómo podemos comunicar nuestro Modelo de Datos?

Diagrama de Entidad Relación DER.

- **Entidad**: cualquier cosa que tenga relevancia para nuestro sistema. Los proyectos, las tareas, los impuestos, las complejidades.

- **Atributos**: son las propiedades o características que describen una entidad. Un cliente tiene nombre y cuit, un auto tiene modelo, marca y color, etc.



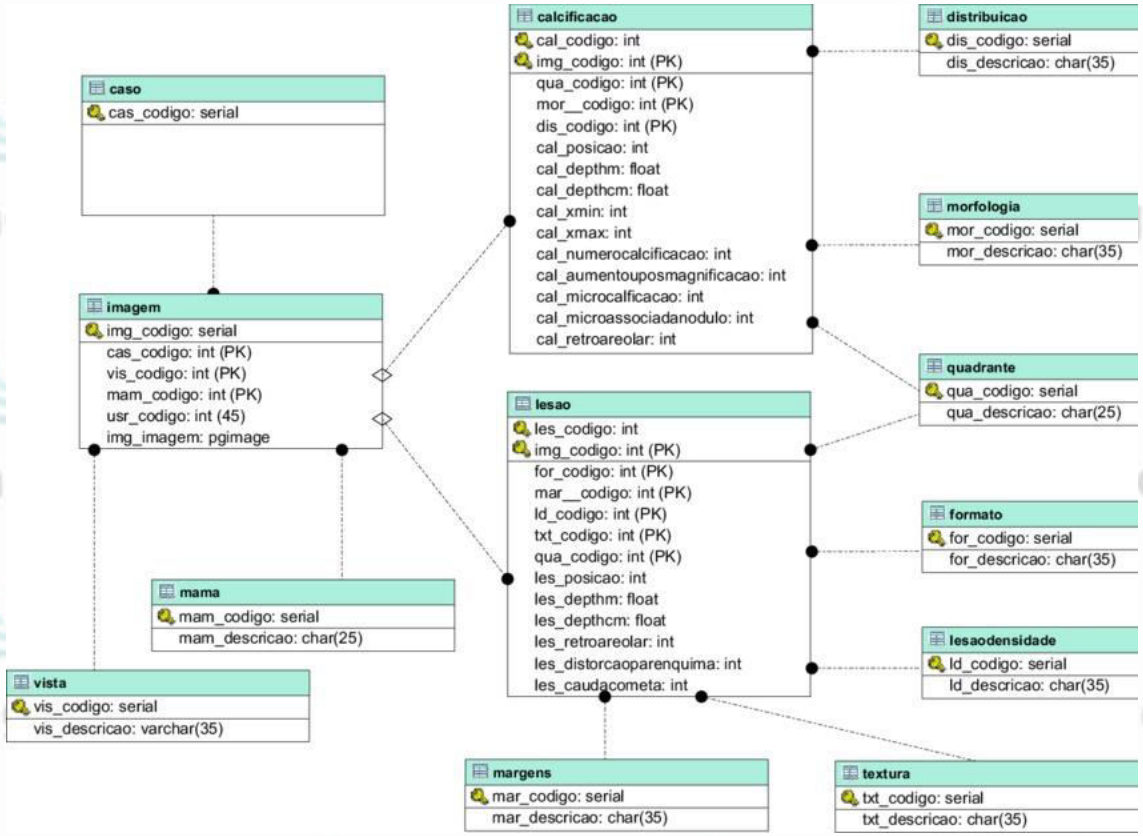
CePETel

Modelado de datos – Modelo Relacional

- **Atributos multivaluados**: son atributos que pueden tomar más de un valor (direcciones de mail de un empleado, subordinados de un jefe, entre otros)

- **Instancia de una entidad**: Ocurrencia particular de una entidad.

- **Relaciones entre entidades:**
    - Representan asociaciones del mundo real entre entidades
    - Se puede representar con un verbo o preposición que conecta dos entidades.



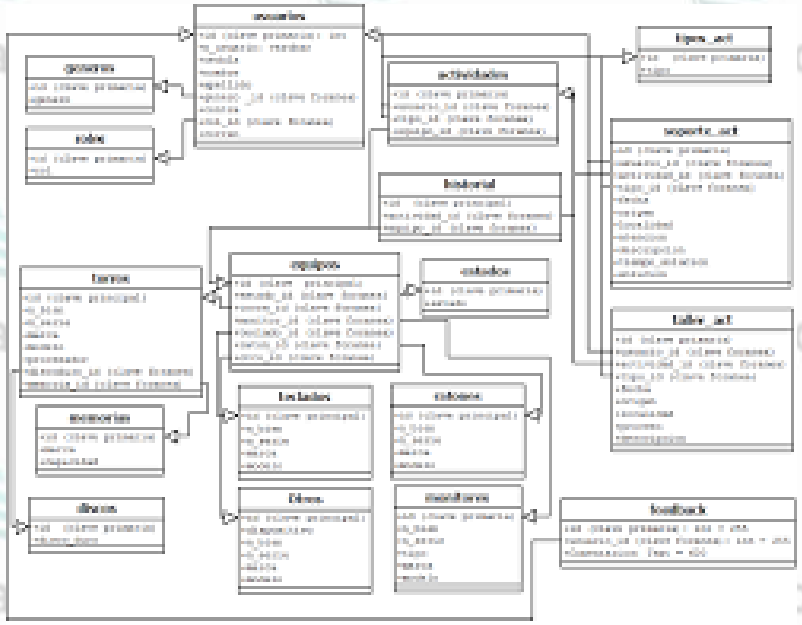
CePETel

Modelado Dimensional

Modelo Dimensional - Definiciones

"El modelado dimensional **es una técnica de diseño** que busca presentar los datos en un framework **estándar, intuitivo y escalable**, que permite un acceso a los datos altamente performante, basándose en el modelado relacional pero con algunas restricciones de diseño importantes"

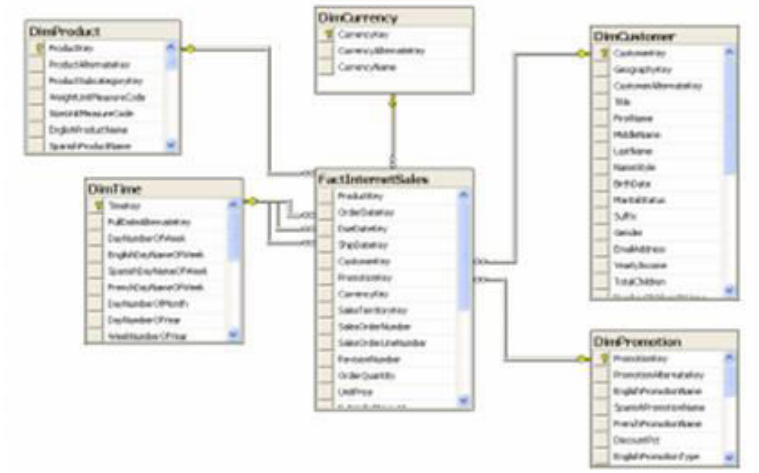
Modelo Relacional



"El **modelado dimensional** es una **técnica de diseño** que busca apoyar a las **consultas analíticas de los usuarios finales** y se orienta en torno a la **comprensibilidad, rendimiento y extensibilidad**."

Basado en el modelado relacional pero con algunas restricciones de diseño importantes. Una base de datos con "modelo dimensional" es una **base de datos** que tiene una **estructura adecuada** para resolver **consultas analíticas.**

Dimensional Model Example



CePETel

Modelado Dimensional - Beneficios

- **Comprensibilidad**: Facilitar la compresión, haciéndola intuitiva para usuarios de negocio (o no expertos técnicos) quienes deben hacer reportes y análisis de datos. La información se agrupa en dimensiones coherentes , por lo que es más fácil de leer e interpretar. La simplicidad también permite al software navegar las bases de datos de manera eficiente.

- **Alto rendimiento** en las búsquedas y lectura de datos. Los modelos dimensionales están más desnormalizados y optimizados para las consultas de datos, mientras que los modelos normalizados buscan eliminar redundancias de datos y están optimizados para la carga de transacciones y actualización.

- **Extensibilidad**: Son escalables y fácilmente acomodados a nuevos datos inesperados. Las tablas existentes pueden ser cambiados, ya sea por la simple adición de nuevas filas de datos en la tabla. Las consultas o aplicaciones montadas sobre el almacén de datos no necesitan ser reprogramadas para acomodarse a los nuevos cambios. Pero en los modelos normalizados cada modificación se debe considerar cuidadosamente, debido a las complejas dependencias entre las tablas de bases de datos.

CePETel

Modelado Dimensional - Beneficios

Comprensibilidad: Facilitar la compresión, haciéndola intuitiva para usuarios de negocio (o no expertos técnicos) quienes deben hacer reportes y análisis de datos.

- Presentar datos en tablas y con nombres que se corresponden bastante con el lenguaje y conceptos del negocio que poseen los analistas.

- Estructura de tablas simples, predecibles, estándar más fácil de leer, entender y más intuitivo
  - No presenta cambios inesperados. Todas las dimensiones son equivalentes, pueden ser pensadas como puntos simétricos de acceso a las fact.
  - En los modelos normalizados, los datos se divide en muchas entidades discretas e incluso un proceso de negocio simple podría resultar en docenas de tablas unidas entre sí de una manera compleja.

- Existen varios prototipos para los modelos de negocios ya existentes.

- Simplicidad también permite usar software para navegar las bases de datos.

CePETel

Modelado Dimensional - Beneficios

Alto rendimiento en las búsquedas y lectura de datos:

- Modelos sencillos que aseguran unos buenos tiempos de respuesta.

- Desnormalización y optimizaciones para las consultas analiticas.

- Modelos OTLP buscan eliminar redundancias y están optimizados para la carga de transacciones y actualización. Modelo dimensional no solo es más simple, es más predecible por lo que permite a la base de datos hacer fuertes supuestos sobre los datos, teniendo un impacto positivo en el rendimiento.

Cada dimensión es el equivalente a un punto de entrada en la tabla de hechos, y esta estructura simétrica permite un manejo eficaz de procesamiento en consultas complejas.
La optimización de consulta se hace simple, predecible y controlable.

CePETel

Modelado Dimensional - Beneficios

Extensibilidad: Son escalables y fácilmente acomodados a nuevos datos inesperados.

- Las tablas existentes pueden ser cambiadas, ya sea por la simple adición de nuevas filas de datos en la tabla o ejecutar en SQL algún comando de tipo "Alter Table".

- Las consultas y aplicaciones antiguas continúan funcionando sin producir resultados diferentes. Las consultas o aplicaciones montadas sobre el datawarehouse no necesitan ser reprogramadas para acomodarse a los nuevos cambios.

- Pero en los modelos normalizados cada modificación se debe considerar cuidadosamente, debido a las complejas dependencias entre las tablas de bases de datos.

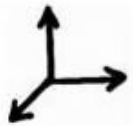
CePETel

Modelado Dimensional Conceptos

Hechos (Facts)

Métricas/Indicadores

Atributos



Dimensiones

Elementos

CePETel

Modelado Dimensional Conceptos

Hechos (Facts)

Evento concreto y específico del proceso de negocio, y **de interés** para la organización.

- Asociados al tiempo, a un instante
- Valores **Cuantitativos**... de tipo numérico (Especialmente con decimales)
- No se conoce de antemano

Ejemplo: la cantidad de ventas de un comercio.

CePETel

Modelado Dimensional Conceptos

Métricas/Indicadores

- Cálculos realizados a partir de los Hechos.
- Usualmente agregaciones como sumatorias y promedios.
- Características, al igual que los hechos:
  - Valores Cuantitativos
  - De tipo numérico (Especialmente con decimales)
  - No se conoce de antemano

CePETel

Modelado Dimensional Conceptos

Atributos

- Generalmente del tipo texto (o pueden tratarse como tal).
- Valores cualitativos.
- Describe características de una entidad.
- Proveen contexto a los hechos.
- Proveen nivel de detalle a las métricas

Modelado Dimensional Conceptos

Elementos

- Instancia o valor que puede tomar un atributo.

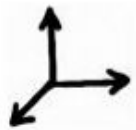
Estado Civil *(Atributo)*
- Soltero *(Elemento)*
- Casado *(Elemento)*
- Viudo *(Elemento)*

Nombre de Producto *(Atributo)*
- Televisor *(Elemento)*
- Buzo *(Elemento)*
- Campera *(Elemento)*

Categoría *(Atributo)*
- Microcentro *(Elemento)*
- San Telmo *(Elemento)*
- Recoleta *(Elemento)*

CePETel

Modelado Dimensional Conceptos



Dimensiones

- Las dimensiones son los diferentes puntos de vista por los que queremos analizar la información.
- Agrupaciones de atributos que estén altamente correlacionados entre sí.
- Incluyen los diferentes atributos que queremos analizar.
- Se estructuran de forma jerárquica, conforme a diferentes niveles de detalle.

Cliente *(Dimensión)*

- Estado Civil *(Atributo)*: Soltero *(Elemento)*
- Sexo *(Atributo)*: Masculino *(Elemento)*
- Nacimiento *(Atributo)*: 3/7/85 *(Elemento)*

CePETel

Modelado Dimensional - Proceso

El proceso consiste en cuatro pasos:

1. **Elegir el proceso de negocio**: Consiste en elegir el área a modelar. Es una decisión de la dirección, y depende fundamentalmente del análisis de requerimientos y de los temas analíticos anotados en la etapa anterior.

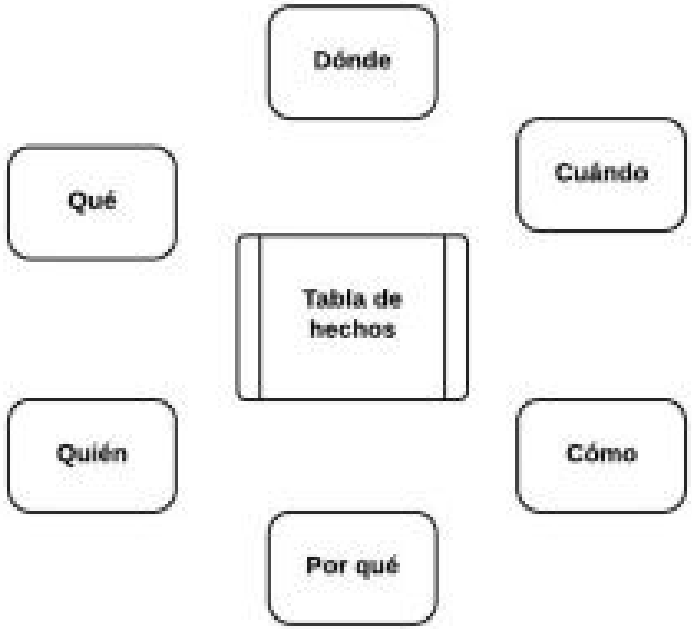
2. **Establecer el nivel de granularidad**: La granularidad significa especificar el nivel de detalle. La elección de la granularidad depende de los requerimientos del negocio y lo que es posible a partir de los datos actuales. La sugerencia general es comenzar a diseñar el Datawarehouse al mayor nivel de detalle posible, ya que se podrían realizar agrupamientos posteriores, al nivel deseado.

3. **Elegir las dimensiones**: Las dimensiones surgen naturalmente de las discusiones del equipo, y facilitadas por la elección del nivel de granularidad. Las tablas de dimensiones tienen un conjunto de atributos (generalmente textuales) que brindan una perspectiva o forma de análisis sobre una medida en una tabla hechos. Una forma de identificar las tablas de dimensiones es que sus atributos son posibles candidatos para ser encabezado en los informes, tablas pivot, cubos, o cualquier forma de visualización, unidimensional o multidimensional.

CePETel

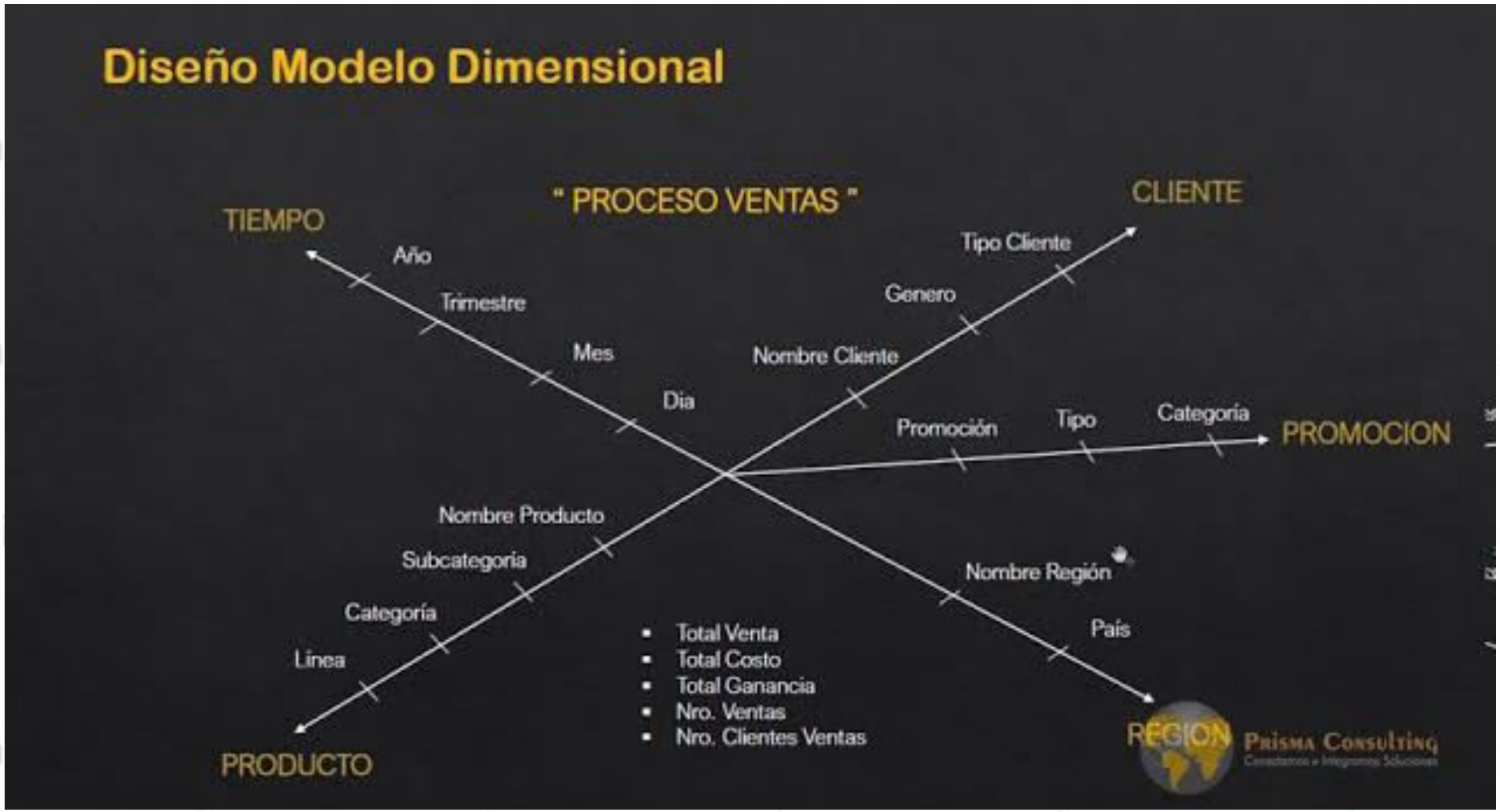
Modelado Dimensional - Proceso

4. **Identificar medidas y las tablas de hechos:** Este paso consiste en identificar las medidas que surgen de los procesos de negocios. Una medida es un atributo (campo) de una tabla que se desea analizar, sumando o agrupando sus datos y usando los criterios de corte conocidos como dimensiones. Las medidas habitualmente se vinculan con el nivel de granularidad. Un registro contiene una medida expresada en números, como ser cantidad, tiempo, dinero, etc., sobre la cual se desea realizar una operación de agregación (promedio, conteo, suma, etc.) en función de una o más dimensiones. La granularidad, en este punto, es el nivel de detalle que posee cada registro de una tabla de hechos



CePETel

Modelado Dimensional - Proceso



CePETel

Modelado Dimensional

Jerarquías

- Representadas por un ordenamiento lógico dentro de la dimensión, se encuentran formadas por los diferentes tipos de relaciones entre los atributos de una misma dimensión.

- Como convenciones del modelado, la jerarquía principal se dibuja verticalmente desde el atributo más agregado (arriba) hasta el más atómico (abajo) y las jerarquías características se adicionan por los costados.

Ejemplo
- Empresa: Cadena de supermercados
- Actividad objeto de análisis: ventas de productos
- Información registrada sobre una venta: ”Se vendió 5 unidades del producto A, en el almacén número 1, el día 20/01/2007 por un total de 250.000 pesos”

Para hacer el análisis no interesa la venta individual realizada por un cliente, si no las ventas diarias de productos en los distintos almacenes de cadena

CePETel

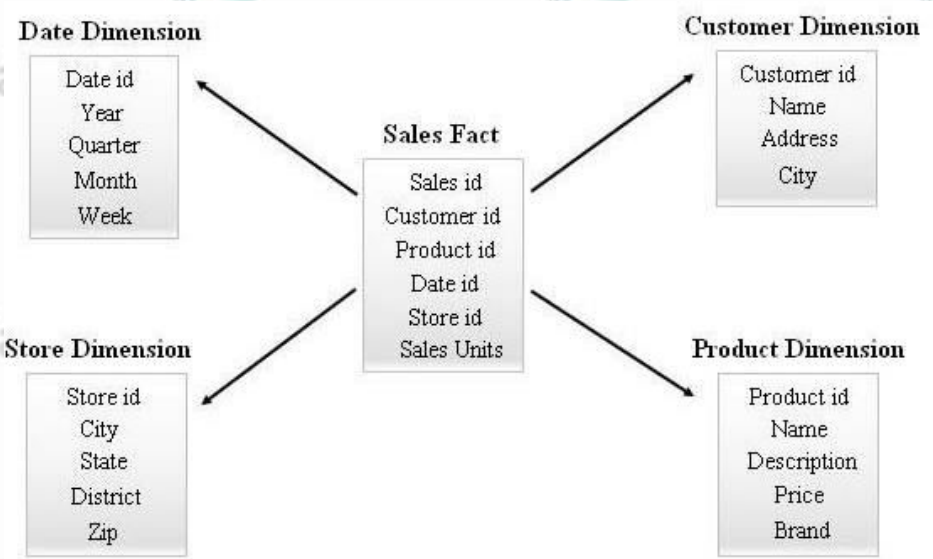
Dimensiones Conformadas

- Es una dimensión que tiene el **mismo significado** para **todos los equipos** de la organización (y los datamarts que se generan).

- Claves para poder obtener tener **visión integral** de la organización.

- **Integra hechos**. Sin una estricta adherencia de las dimensiones conformadas, el DW no podrá funcionar como un todo integrado.

- La mayor responsabilidad del equipo que diseña un DW es establecer, publicar, mantener las dimensiones conformadas

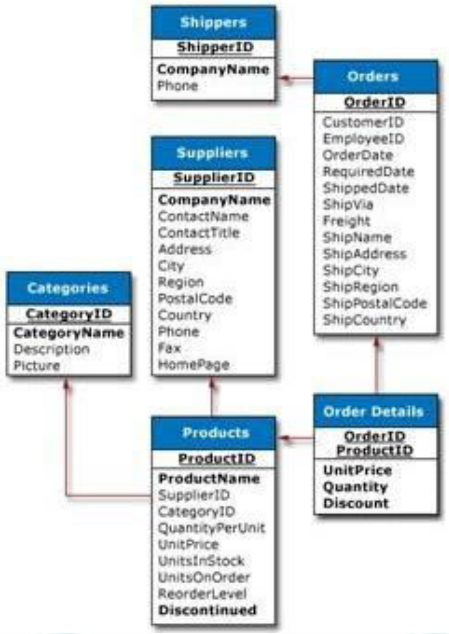


CePETel

Modelado Relacional vs Dimensional

Modelo E-R

- Entidades
- Atributos
- Relaciones



Modelo dimensional

- Hechos
- Dimensiones
- Medidas

CePETel

Modelado Relacional vs Dimensional

Ventajas Modelo Relacional

- Provee herramientas que garantizan evitar la duplicidad de registros.
- Garantiza la integridad referencial, así, al eliminar un registro elimina todos los registros relacionados dependientes.
- Favorece la normalización por ser más comprensible y aplicable.

Ventajas del Modelado Dimensional

- Recuperación de datos más rápida
- Mejor comprensión de los procesos comerciales: Los principios del modelado dimensional se basan en tablas de hechos y dimensiones.
- Flexible para cambiar: El marco de modelado dimensional hace que el proceso de almacenamiento de datos sea extensible. El diseño se puede modificar fácilmente para incorporar nuevos requisitos comerciales o realizar ajustes en el repositorio central.

CePETel


Cultura Data Driven

Cultura Data Driven

La **cultura Data Driven** consiste, básicamente, en reemplazar la experiencia, la intuición y la especulación, por un conocimiento sólido, fundado principalmente sobre datos. Cuando la toma de decisiones estratégicas en materia de negocios se apoya en el análisis y la interpretación de los datos, las perspectivas de una empresa cambian radicalmente, abriéndose a nuevos horizontes.

Sin embargo, el **mayor obstáculo** a la hora de crear negocios basados en datos no es técnico ni tecnológico, sino estrictamente cultural. Es mucho más fácil realizar una descripción acerca de cómo inyectar datos en un proceso de toma de decisiones, que lograrlo efectivamente debido al cambio de mentalidad que esto supone.

CePETel

Cultura Data Driven

Los **cuatro pilares fundamentales** para una implementación exitosa de la cultura Data Driven son los siguientes:

1. Liderazgo basado en datos: aunque toda la empresa debe comprometerse en asumir este cambio de paradigma, la dirección debe dar el ejemplo, no tanto con palabras sino con hechos concretos.
2. Alfabetización en materia de datos: la formación de todos los miembros de una organización en lo que respecta al aprovechamiento de los datos, debe ser asumida de forma permanente, como un asunto de máxima prioridad.
3. Madurez en el empleo de los datos: cada individuo dentro de una empresa debe tener la posibilidad de acceder, de forma ágil y sencilla, a datos limpios y precisos que ofrezcan una mayor comprensión del trabajo cotidiano.
4. Decisiones y revisiones fundadas en datos: debe establecerse un proceso sistemático que permita tomar decisiones prospectivas y revisiones retrospectivas, incorporando ambas acciones como parte del funcionamiento de la organización.

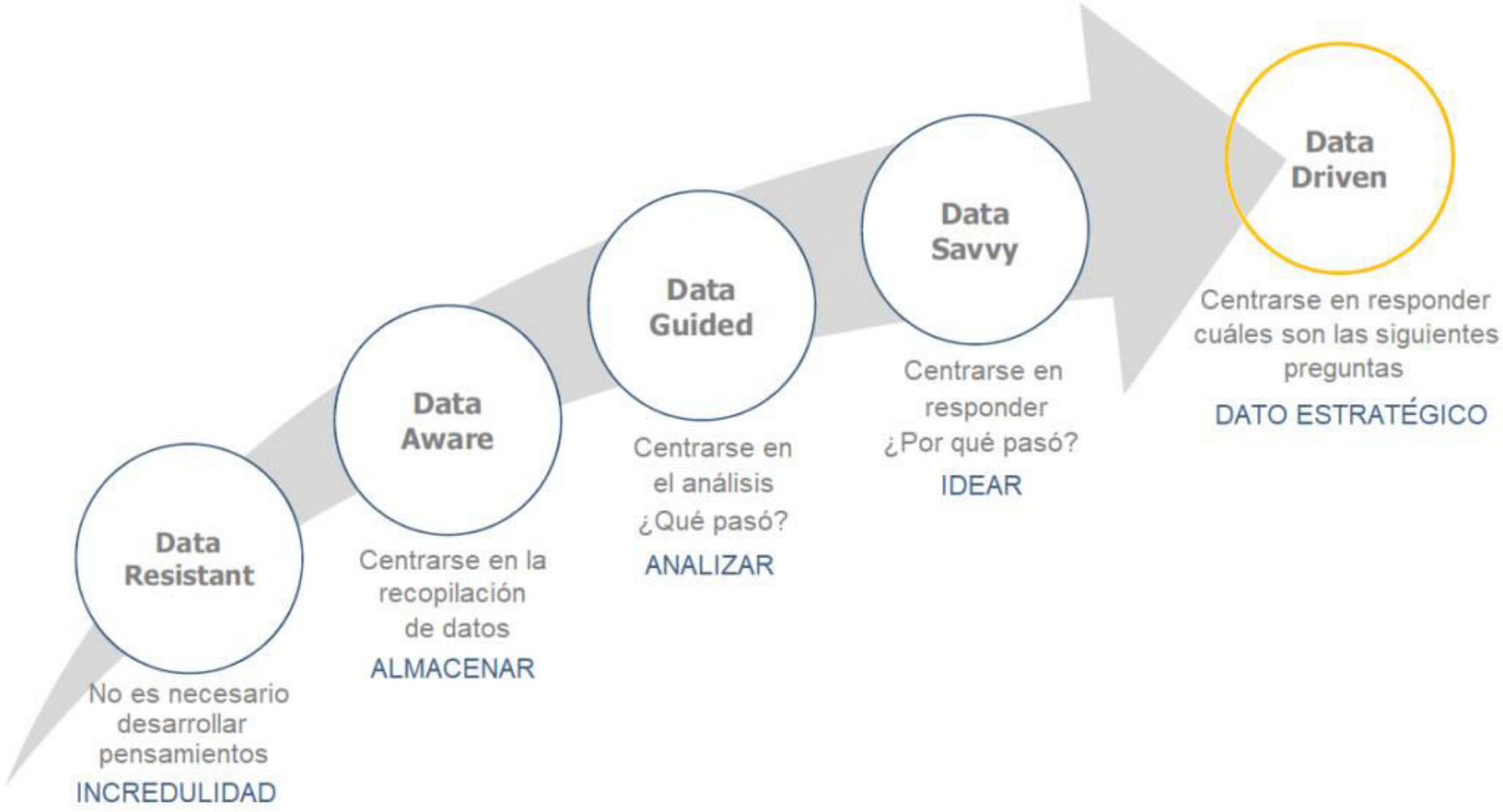
A fin de cumplimentar estos pasos y lograr que la cultura Data Driven fructifique, debe fomentarse el pensamiento crítico y el trabajo colaborativo. Se trata de una transformación progresiva que, además de flexibilidad para el cambio, requiere de un esfuerzo sostenido en el tiempo.

CePETel

Video: Cultura Data Driven

Toma de Decisiones basada en datos

Evolución hacia una Cultura Data Driven



CePETel

Toma de Decisiones basada en datos

CePETel

Toma de Decisiones basada en datos

6 Pasos hacia la Transformación del Data Driven

1: Establece una Estrategia de Datos.

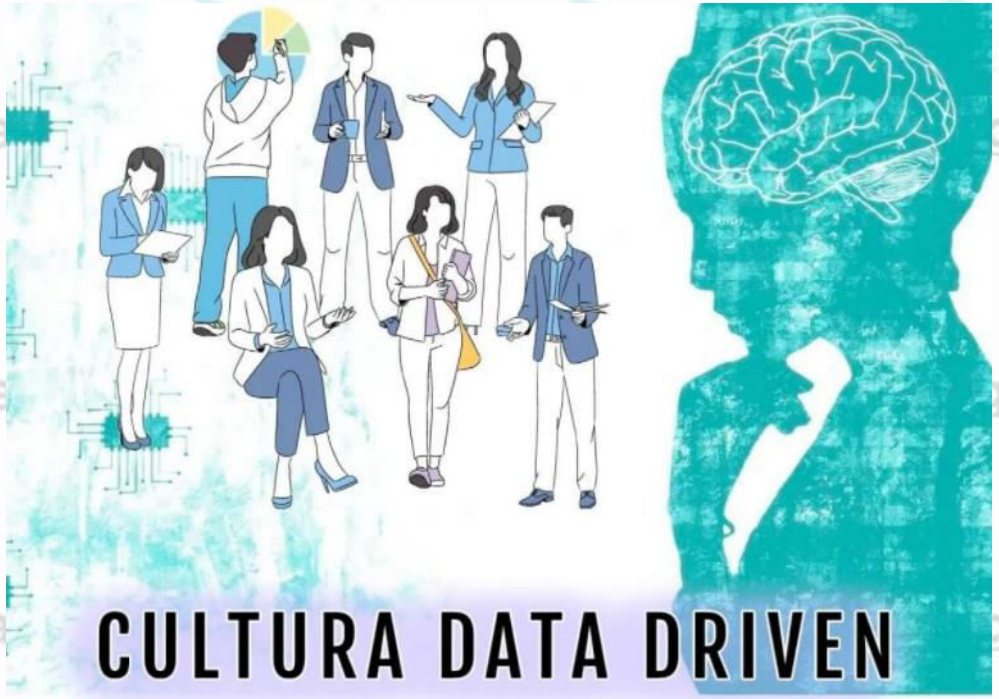
2: Democratiza tus Datos.

3: Construye una Cultura de Recolección de Datos.

4: Acelera la velocidad de comprensión.

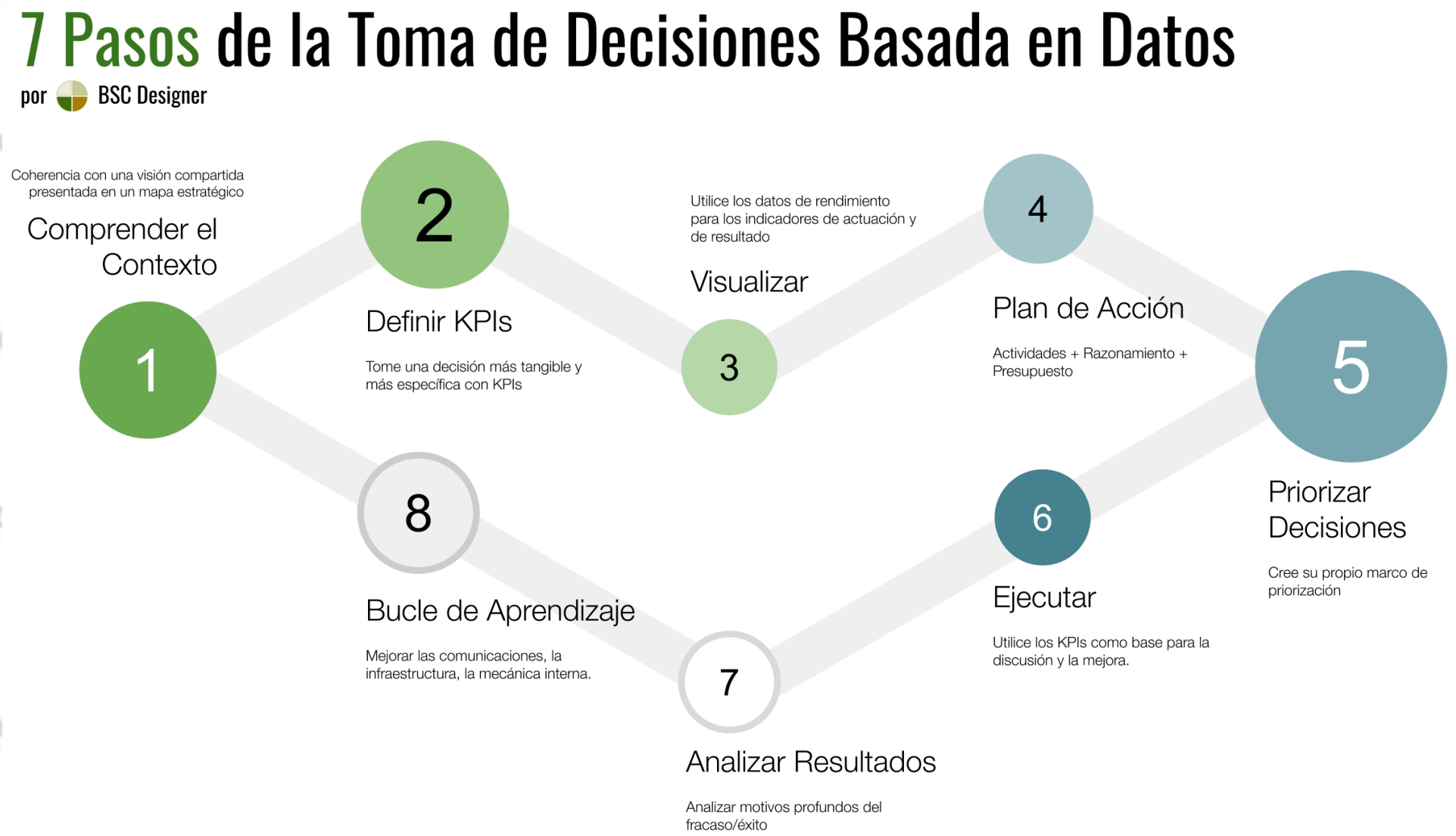
5: Mide el valor del **Data** Science.

6: Aplica una estructura del gobierno del dato.



CePETel

Toma de Decisiones basada en datos



CePETel

TP N° 1: Tipos de Bases de Datos

- Identifique la diferencia entre Modelo Relacional vs. Modelo Dimensional.

Armar una presentación en 1 hoja en base a lo solicitado para exponer en clase. Éxitos!!

CePETel

TP N° 2: Tipos de Bases de Datos

- Busque ejemplos de tablas de hechos y dimensiones de algún caso que conozca o ficticio.

-¿Para que sirven las dimensiones conformadas? ¿Se le ocurre algún ejemplo?

Armar una presentación en 1 hoja en base a lo solicitado para exponer en clase. Éxitos!!

CePETel

CePETel
Sindicato de los Profesionales de las Telecomunicaciones

SECRETARÍA TÉCNICA

IPEI
Instituto Profesional de Estudios e Investigación

BIG DATA & ANALYTICS II

TEMARIO
Módulo 1: Modelado Dimensional
Módulo 2: Data Warehouse
Módulo 3: Data Lake

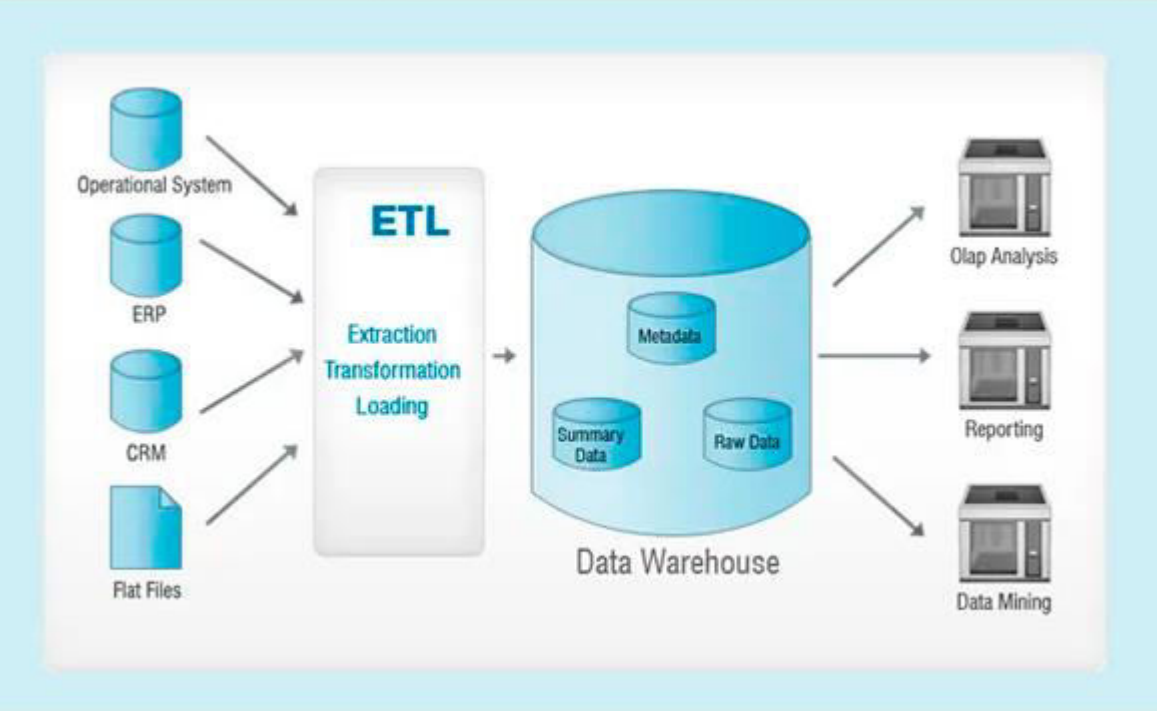
Disertantes: Lic. Maria Trinidad Aquino – Ing. Raúl Alejandro Grassi


DATA
WAREHOUSE

Data Warehouse

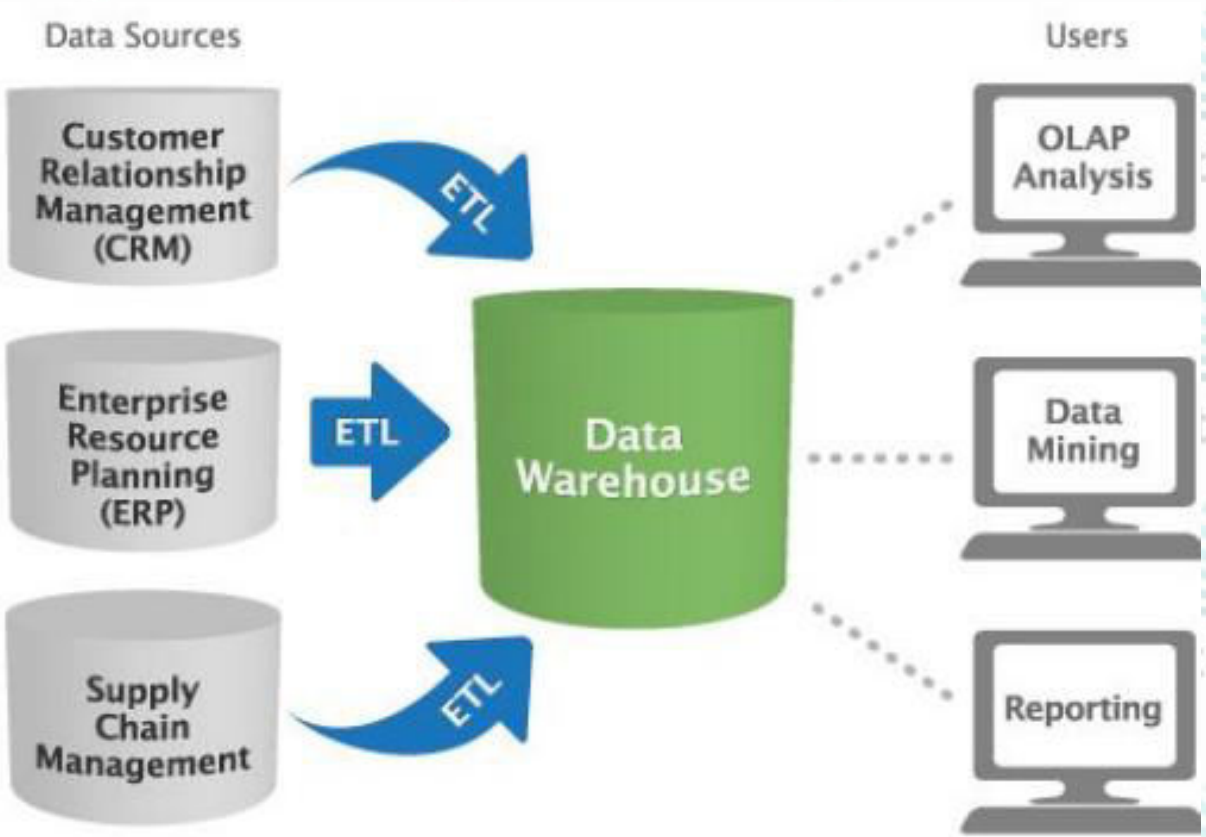
Video: ¿Qué es un Datawarehouse?

Es un sistema que agrega y combina información de diferentes fuentes en un almacén de datos único y centralizado; consistente para respaldar el análisis empresarial, la minería de datos, inteligencia artificial (IA) y Machine Learning.



CePETel

Data Warehouse – Arquitectura Estándar



Un Data Warehouse es una almacén/base de datos corporativa que se caracteriza por integrar y depurar información de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de perspectivas y con grandes velocidades de respuesta.

La creación de un Data Warehouse representa en la mayoría de las ocasiones el primer paso, desde el punto de vista técnico, para implantar una solución completa y fiable de Business Intelligence.

CePETel

Data Warehouse

Data Warehouse vs. Bases de Datos convencionales

El Data Warehouse concentra y almacena de **forma estructurada** toda la información obtenida a partir de las **múltiples fuentes de datos** en nuestra organización, permitiendo así una **rápida integración** con herramientas de minería de datos, análisis y reportes (dashboards).

Funciona un poco diferente a las **bases de datos convencionales (*OLTP - OnLine Transactional Processing*).** Como su nombre lo indica, manejan los datos transaccionales y los datos de cara a los procesos principales de la organización. Al ser transaccionales comúnmente manejan segundo a segundo operaciones de consulta, inserción, borrado y actualización de datos según los requerimientos del usuario (por ejemplo, una reserva de cine).

CePETel

Data Warehouse

Data Warehouse vs. Bases de Datos convencionales

Los **Data Warehouse** usan **OLAP (*OnLine Analytical Processing*).** Son datos que, aunque no están disponibles en tiempo real, pueden ser **analizados de forma rápida y masiva sin interrumpir los procesos** del usuario. Esto le otorga a los científicos de datos una **perspectiva más amplia para tomar decisiones** (por ejemplo, total de ventas a través del tiempo). Como el Data Warehouse está diseñado con una finalidad analítica, este proceso puede llegar a ser hasta 1000 veces más rápido que una base de datos convencional.

CePETel

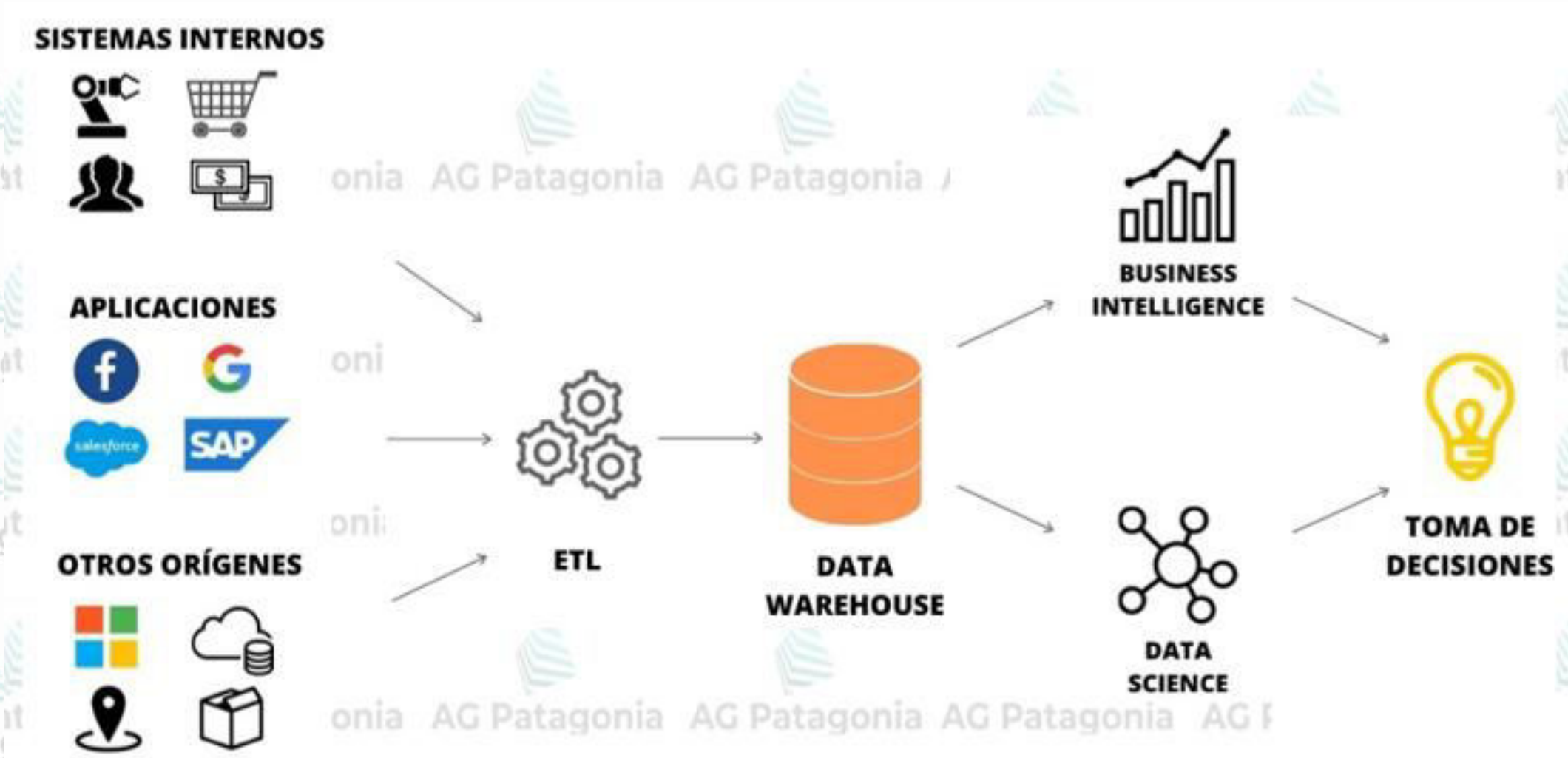
Data Warehouse

Data Warehouse vs. Bases de Datos convencionales

BASE	DATA WAREHOUSE	DATABASE
Definición	Una especie de base de datos optimizada para recopilar información de diferentes fuentes para análisis e informes comerciales.	Almacenamiento o recopilación de datos de manera organizada para almacenar, actualizar, acceder y recuperar datos.
Estructura de datos	La estructura de datos desnormalizados se utiliza para mejorar el tiempo de respuesta analítica.	La estructura de datos normalizada está en una base de datos en tablas separadas.
Cronología de datos	Los datos históricos se almacenan para análisis, mientras que los datos actuales también se pueden usar para análisis en tiempo real.	El procesamiento diario y la transacción de datos se realizan en una base de datos.
Mejoramiento	Warehouse está optimizado para realizar procesamiento analítico en grandes datos a través de consultas complejas.	Optimizado para la actualización rápida de datos para maximizar el acceso mejorado a los datos.
Análisis	Se realiza un análisis dinámico y rápido de los datos.	La función transaccional se lleva a cabo, aunque el análisis es posible pero es difícil de realizar debido a la complejidad de los datos normalizados.

CePETel

Data Warehouse



CePETel

Data Warehouse

Objetivos

• Proporciona una herramienta para la **toma de decisiones** en cualquier área funcional, basándose en información integrada y global del negocio, aportando una ventaja competitiva en la organización.

• Facilita la aplicación de **técnicas estadísticas de análisis y modelización** para encontrar relaciones ocultas entre los datos del almacén; obteniendo un valor añadido para el negocio de dicha información.

• Proporciona la capacidad de **aprender de los datos del pasado y de predecir situaciones futuras** en diversos escenarios.

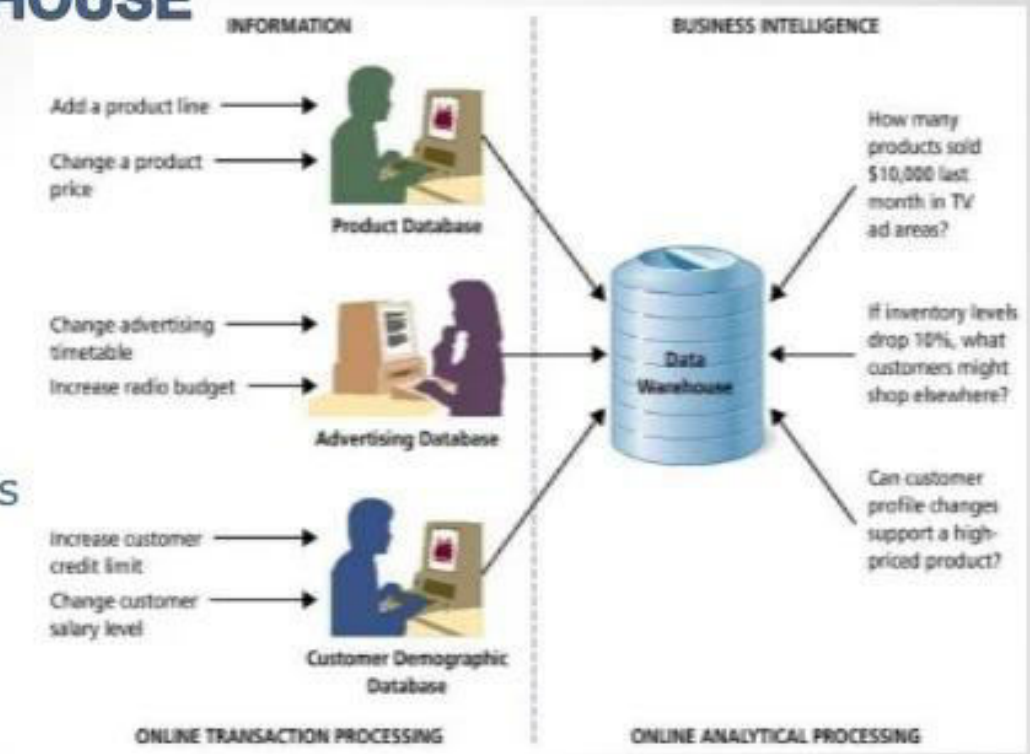
CePETel

Data Warehouse

Objetivos

OBJETIVOS ESPECIFICOS DE UN SISTEMA DE DATAWAREHEHOUSE

- Organiza y orienta los datos desde la perspectiva del usuario final, mientras que los sistemas operacionales organizan sus datos desde la perspectiva de la aplicación, para lograr eficiencia en el acceso a datos.

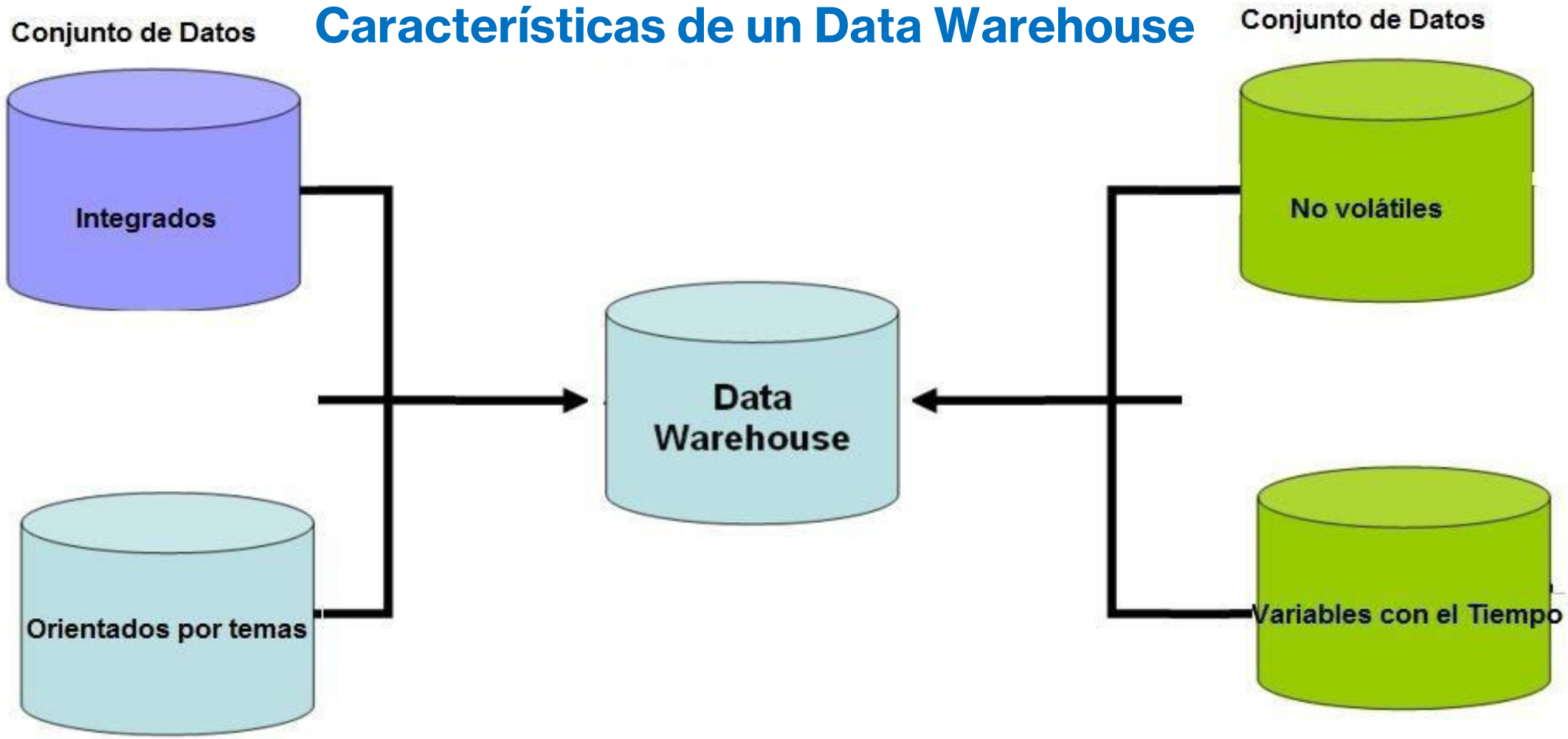


- El Datawarehouse surgió con el objetivo de hacer consultable la información que se tiene de una empresa tanto de meses como de años anteriores.

CePETel

Data Warehouse

Características de un Data Warehouse



CePETel

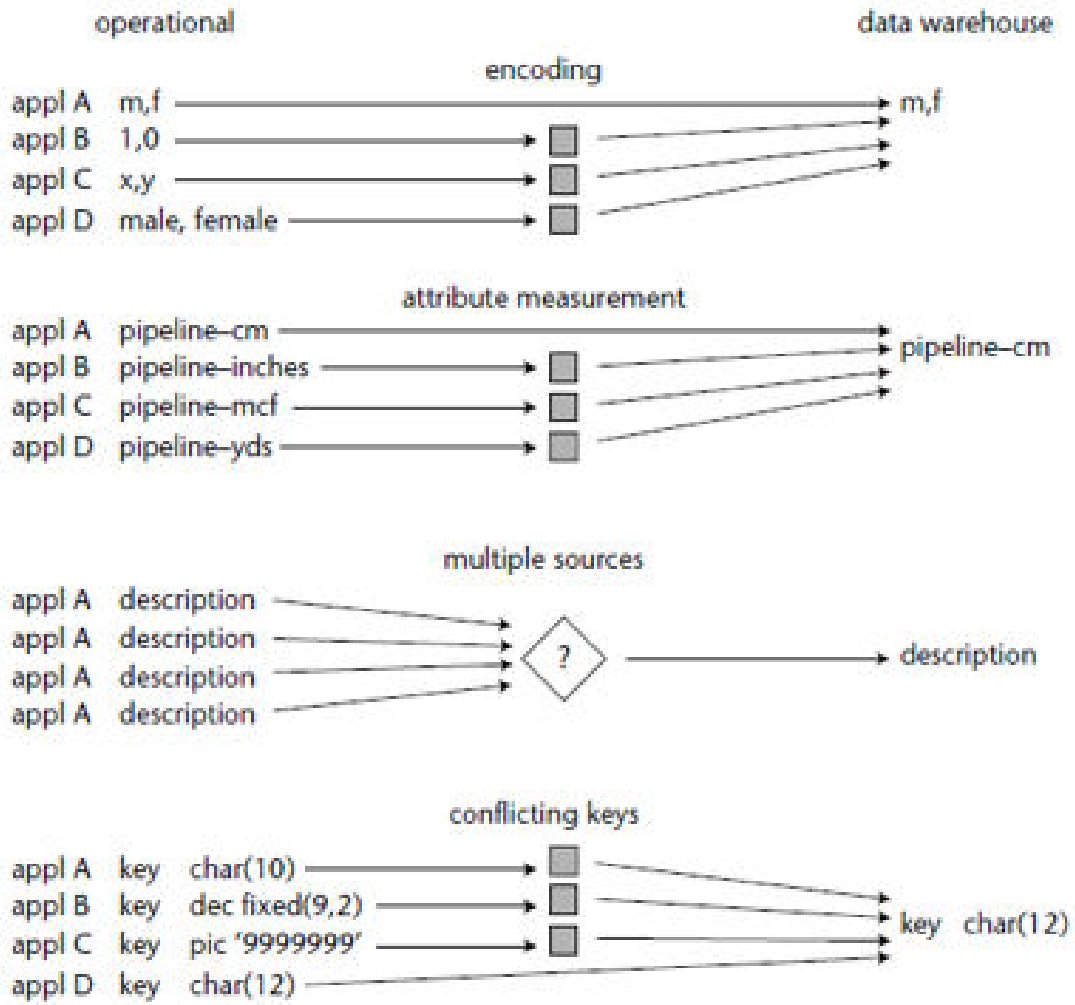
Data Warehouse

Características

Integrado

• Información proveniente de sistemas heterogéneos. (BD, excels, archivos planos, etc.)

• Procesos de integración de datos y limpieza de información. (unificación de formatos, códigos, etc.)



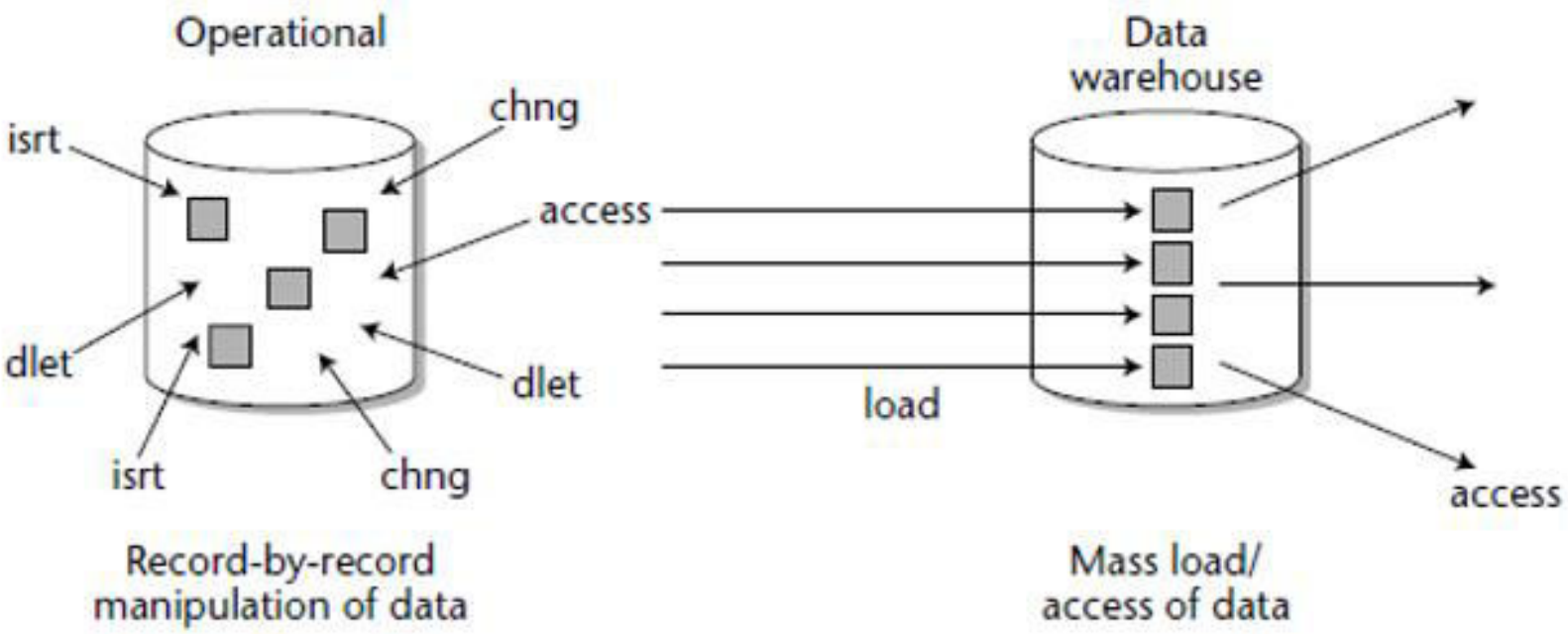
CePETel

Data Warehouse

Características

No volátil

- Pesado para ser leído.
- Los datos perduran en el tiempo. Sólo inserts y updates.



CePETel

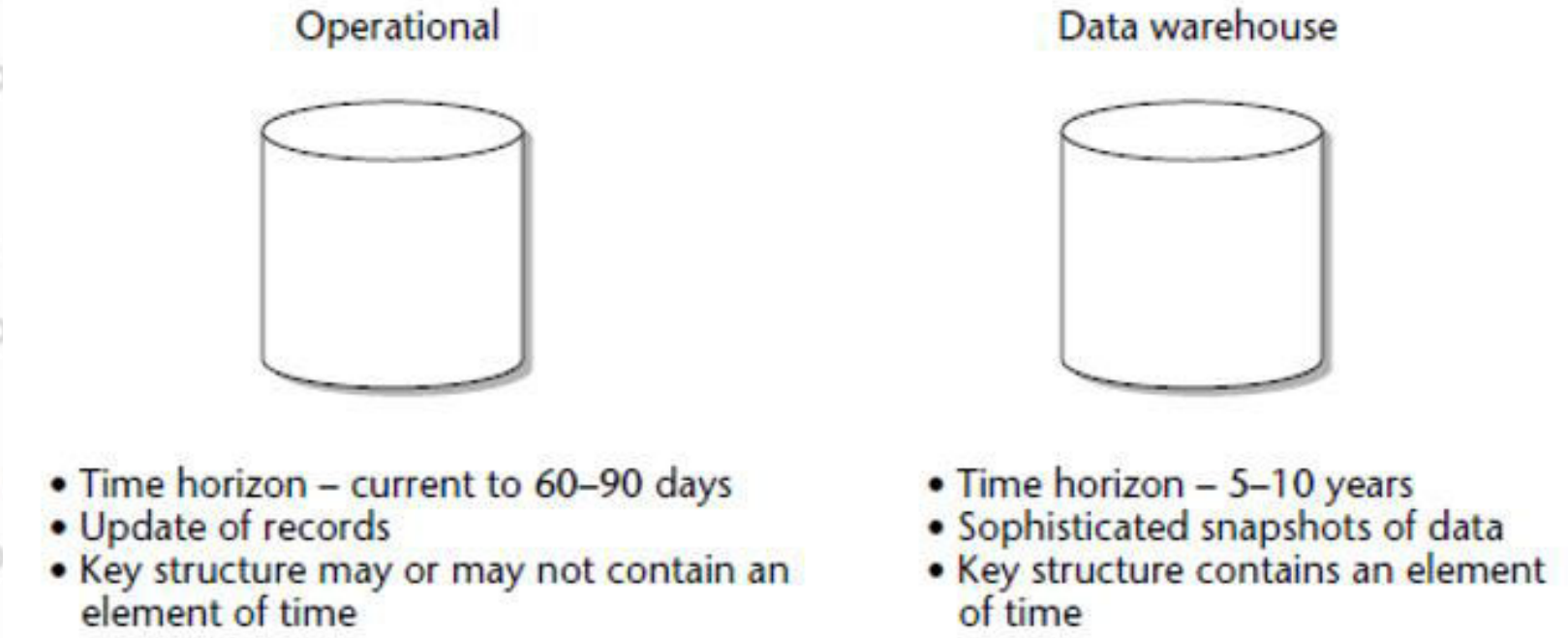
Data Warehouse

Características

Histórico / Variable en el tiempo

• Es un repositorio de datos históricos (tiempo es parte implícita de la información contenida)

• El tiempo de conservación de los datos es mayor que en sistemas transaccionales.



CePETel

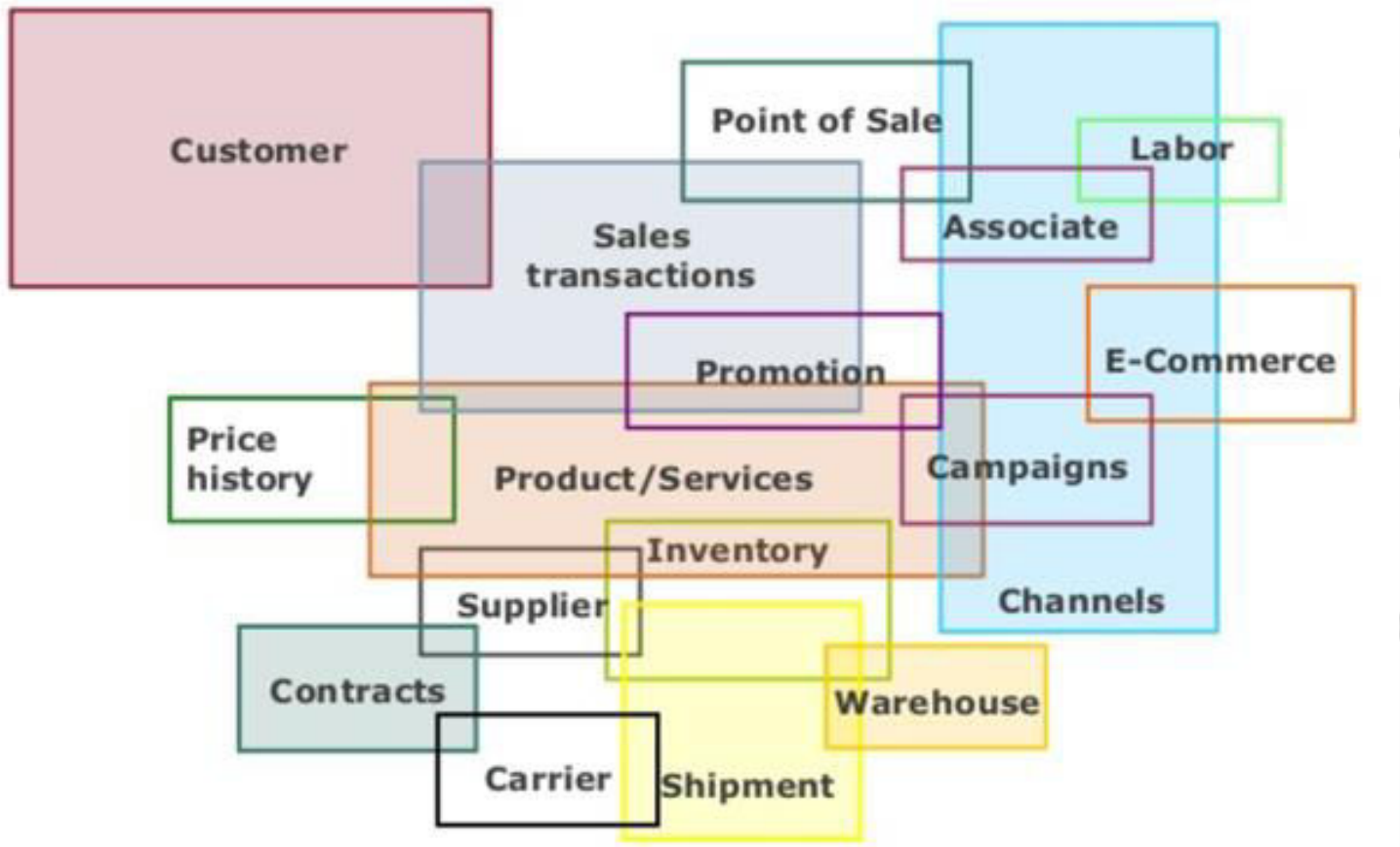
Data Warehouse

Características

Orientado al negocio (por temas)

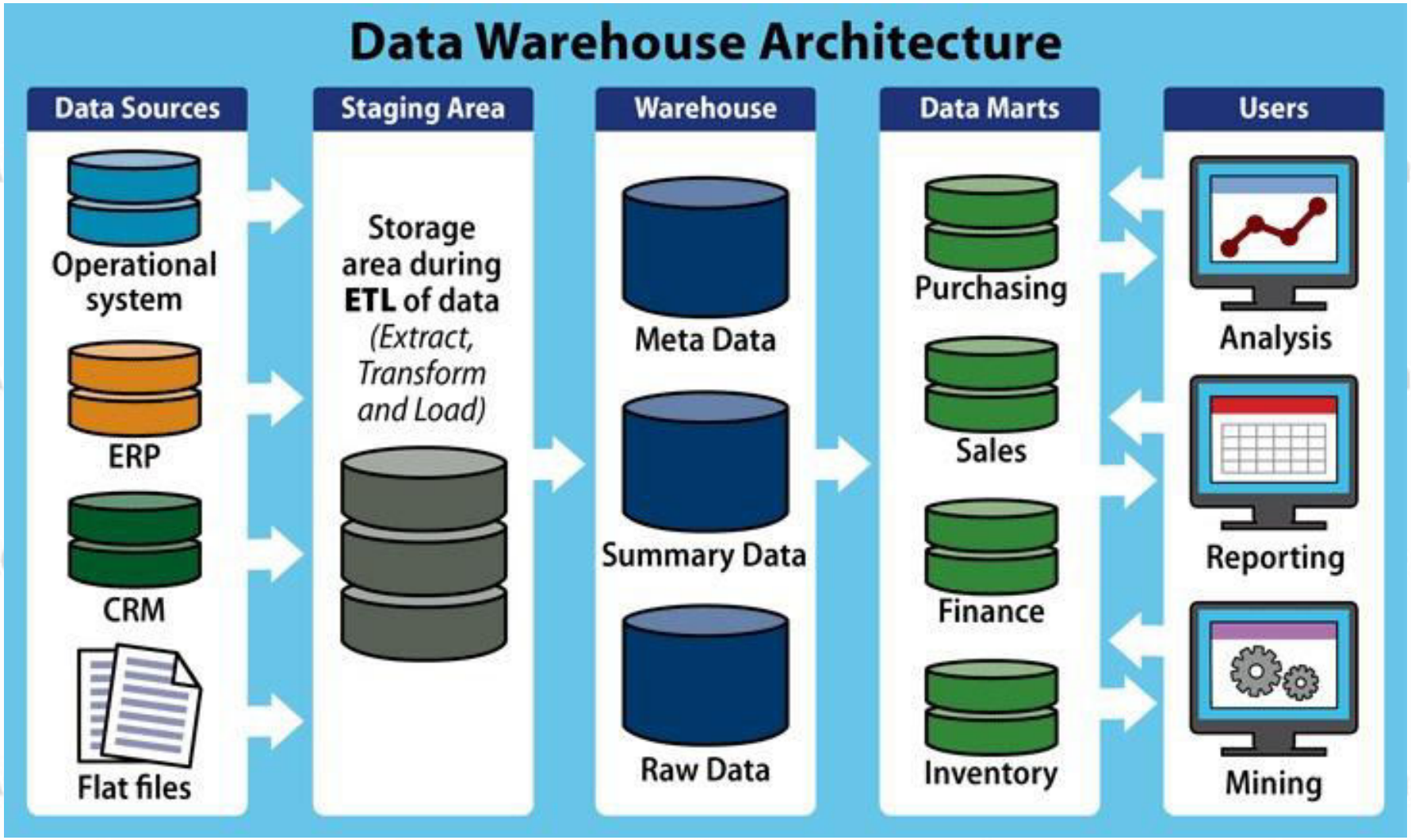
• Los datos están organizados por temas y presentados como se manejan en el negocio (facilitar su acceso y entendimiento por parte de los usuarios finales).

• Solo datos necesarios para la toma de decisiones y tienen el nivel de detalle y estructura adecuados.



CePETel

Data Warehouse

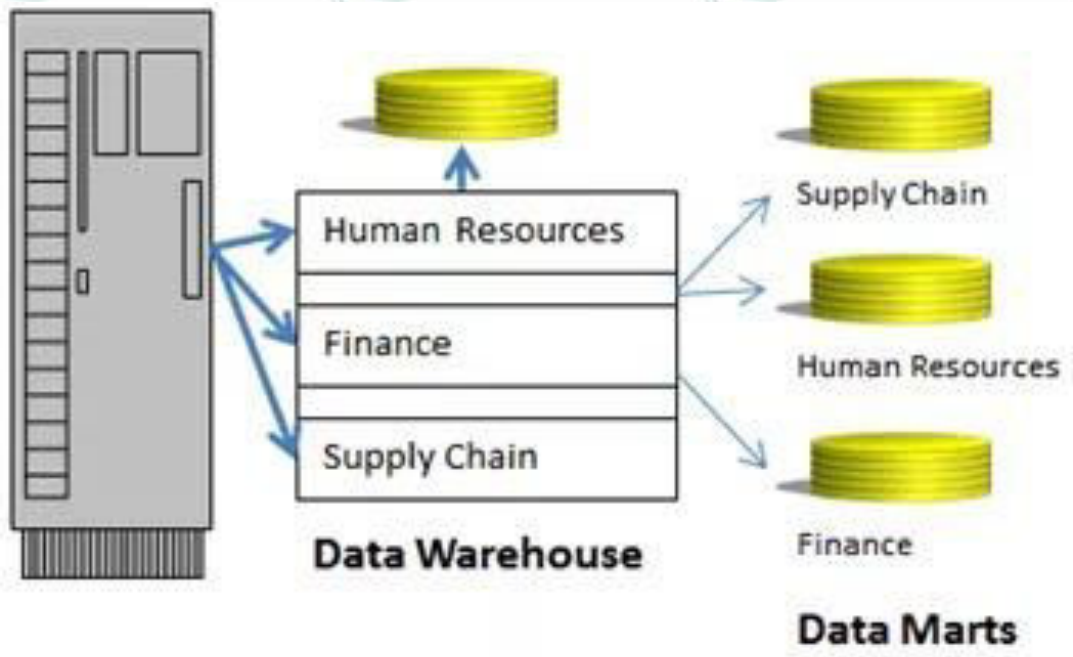


CePETel

Data Warehouse

¿Qué es un data mart?

Un data mart es un **subconjunto de una base de datos** – habitualmente un data warehouse – donde los datos son almacenados **para una área del negocio concreta**. Es decir, en un data mart **se almacenan conjuntos de datos concisos y específicos** dispuestos al análisis para un departamento o línea de negocio concreto como, por ejemplo, el departamento comercial.

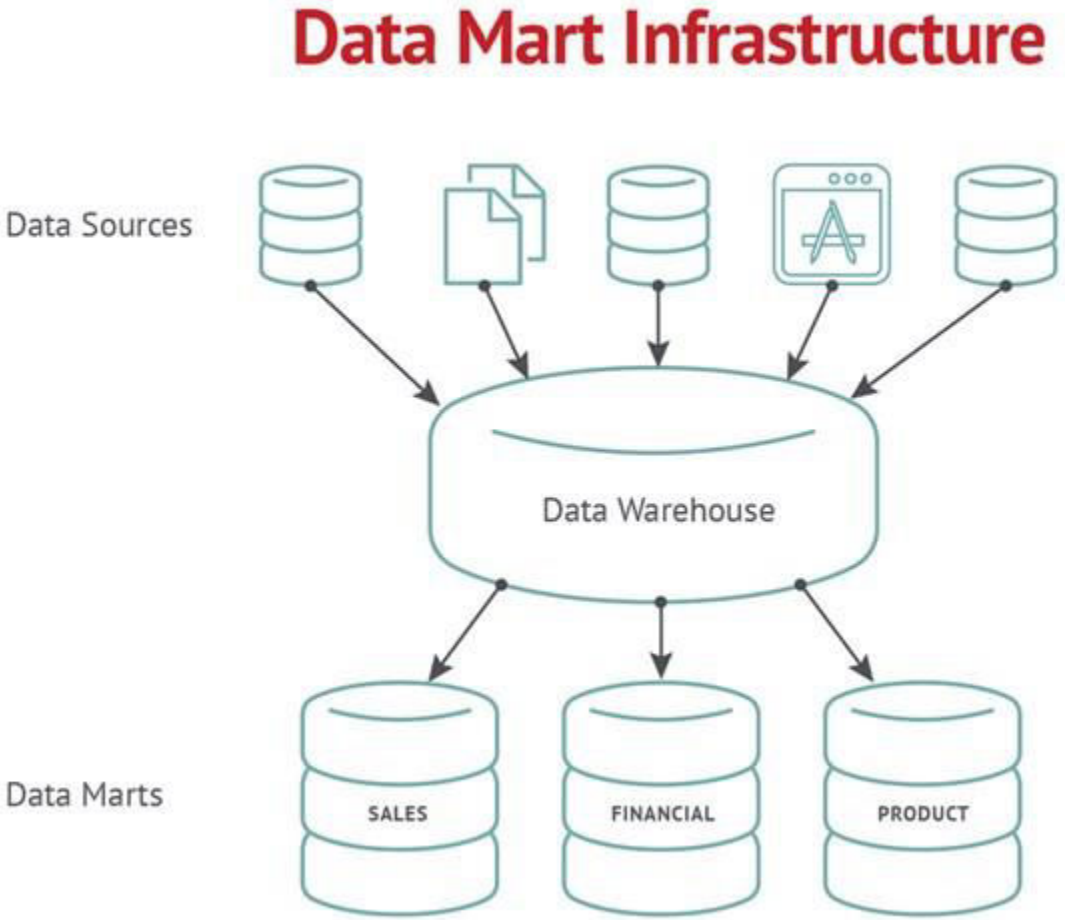


CePETel

Data Warehouse

¿Qué es un data mart?

El data mart está **orientado a la consulta específica** y, igual que en un data warehouse, los datos tienen una **estructura clara** –también habitualmente en modelos dimensionales de estrella o copo de nieve–. La intención del uso del data mart es indexar datos para facilitar las queries sobre áreas específicas del negocio y **satisfacer las necesidades de un grupo concreto de usuarios** dentro de la organización como, por ejemplo, los miembros del equipo de ventas o de finanzas.



CePETel

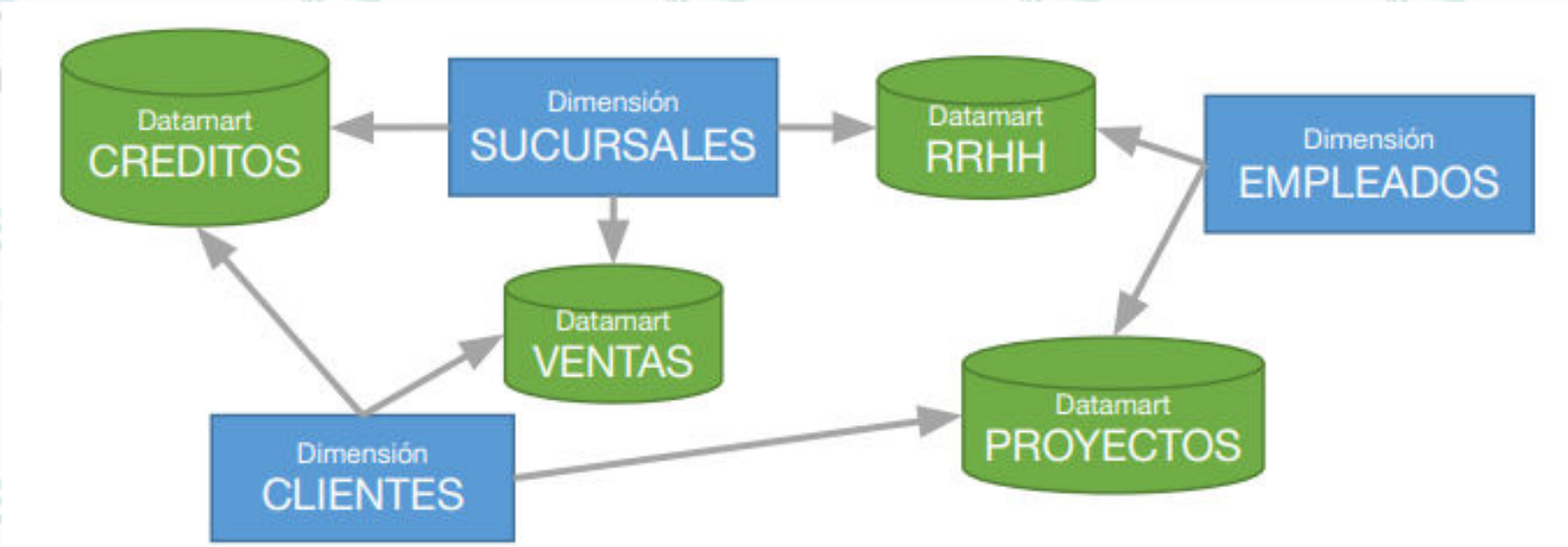
Data Warehouse vs Data Mart

	DATA WAREHOUSE	DATA MART
01. Uso	Facilita la integración de datos y el proceso de toma de decisiones empresariales.	Asiste en la toma de decisiones estratégicas concretas.
02. Objetivo	Proporciona un entorno integrado y coherente para todos los activos de datos de la empresa.	Proporciona un entorno integrado para los datos referentes a un departamento empresarial concreto.
03. Diseño	Proceso de diseño difícil. No tiene por qué estar basado en un modelo dimensional.	Proceso de diseño fácil basado en un modelo dimensional.
04. Enfoque	General. Información relativa a toda la compañía.	Específico. Información relativa a un departamento o área de negocio concreta.
05. Tipos de datos	Datos detallados de estructura no volátil y variantes en el tiempo.	Principalmente datos consolidados preparados para satisfacer las necesidades informativas de los responsables del área temática.
06. Alcance	Almacena datos vinculados a toda la empresa y a cualquier aspecto de la actividad empresarial. Ejerce de fuente de información para cualquier área de la organización.	Almacena datos relativos a un departamento o un aspecto de la actividad empresarial concreto. Cada departamento o área de negocio puede disponer de un data mart propio.
07. Fuentes de origen	Recopila datos procedentes de una gran cantidad de fuentes.	Recopila datos de una cantidad de fuentes reducida, normalmente del propio data warehouse.
08. Tamaño	Habitualmente de 100 GB a 1 TB.	Habitualmente menos de 100 GB.
09. Tiempo de implementación	Su implementación suele durar entre varios meses y varios años.	El tiempo de implementación suele ser de unos pocos meses.

CePETel

Data Warehouse vs Data Mart

- **Data warehouse** es la unión de **todos los datamarts**.
- El **uso de dimensiones conformados** permite usar los datamarts como un data warehouse integrado.



CePETel

Modelado Dimensional Conceptos

Recordando!

Dimensiones

- Las dimensiones son los diferentes puntos de vista por los que queremos analizar la información.
- Agrupaciones de atributos que estén altamente correlacionados entre sí.
- Incluyen los diferentes atributos que queremos analizar.
- Se estructuran de forma jerárquica, conforme a diferentes niveles de detalle.

Cliente *(Dimensión)*

- Estado Civil *(Atributo)*: Soltero *(Elemento)*
- Sexo *(Atributo)*: Masculino *(Elemento)*
- Nacimiento *(Atributo)*: 3/7/85 *(Elemento)*

CePETel

Dimensiones Conformadas

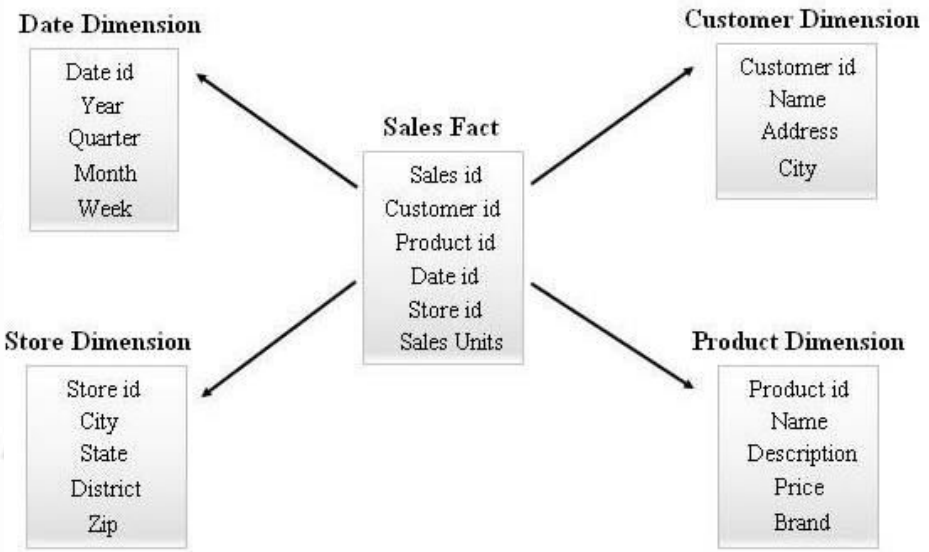
Recordando!

- Es una dimensión que tiene el **mismo significado** para **todos los equipos** de la organización (y los datamarts que se generan).

- Claves para poder obtener tener **visión integral** de la organización.

- **Integra hechos**. Sin una estricta adherencia de las dimensiones conformadas, el DW no podrá funcionar como un todo integrado.

- La mayor responsabilidad del equipo que diseña un DW es establecer, publicar, mantener las dimensiones conformadas

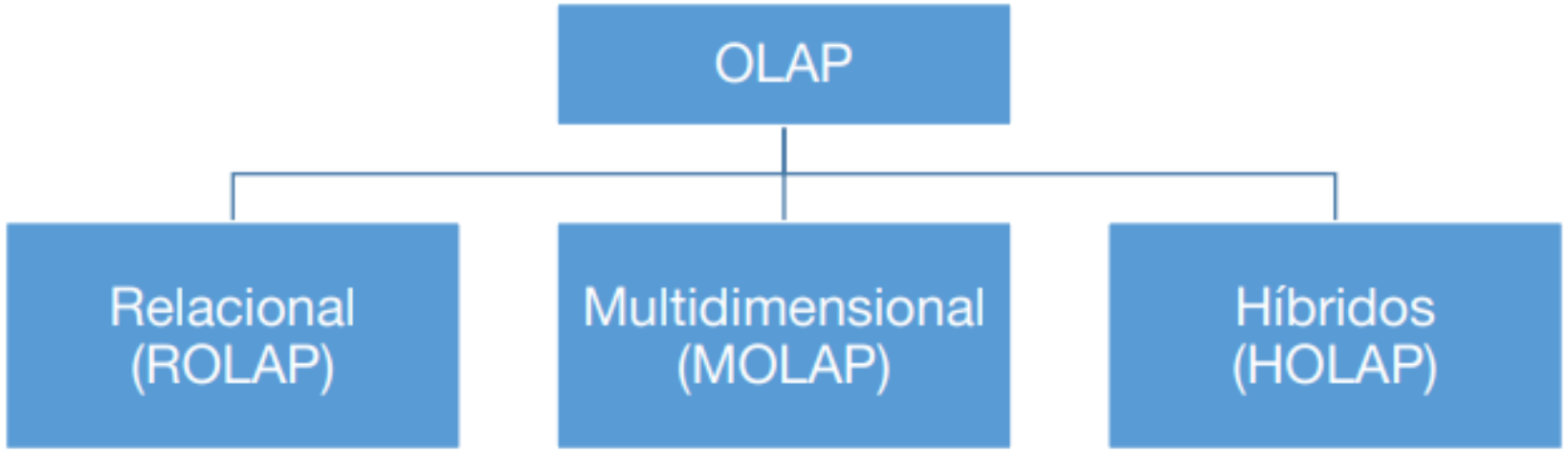


CePETel

DATA CONCEPTS

KIMBALL & INMON

Tecnologías OLAP



…almacenamiento en una base de datos relacional.

- MicroStrategy
- OBIEE (Oracle)
- Microsoft Analysis Services

…almacenamiento en un vector multidimensional o cubo optimizado.

- icCube Server
- Oracle OLAP
- Microsoft Analysis Services

ROLAP + MOLAP

- MicroStrategy
- SAP AG
- Microsoft Analysis Services

CePETel

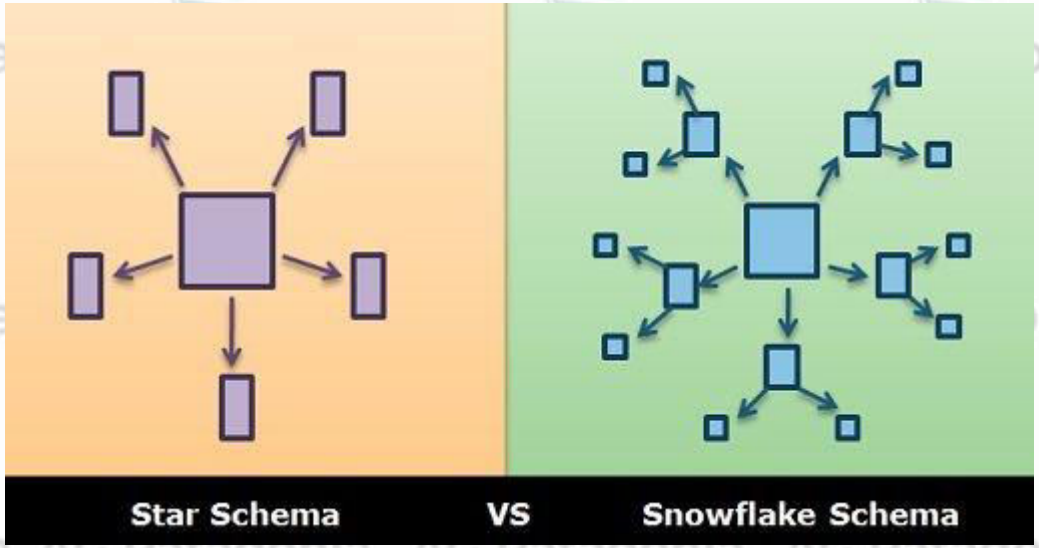
Data Warehouse - Modelado

[Video: Modelado Estrella vs. Copo de Nieve: Dimensiones]

- Base de datos relacional (ROLAP) -> “modelo dimensional”.

Existen **2 tipos de modelados** para el armado de cubos:

- **Estrella (star schema)**
- **Copo de nieve (snowflaked)**



Parte del proceso de modelado dimensional consiste en elegir cuál es el mejor modelo a elegir al momento de plantear las estructuras de las dimensiones.

CePETel

Data Warehouse - Enfoques

Modelo Estrella (Start Schema)

Ralph Kimball

Enfoques

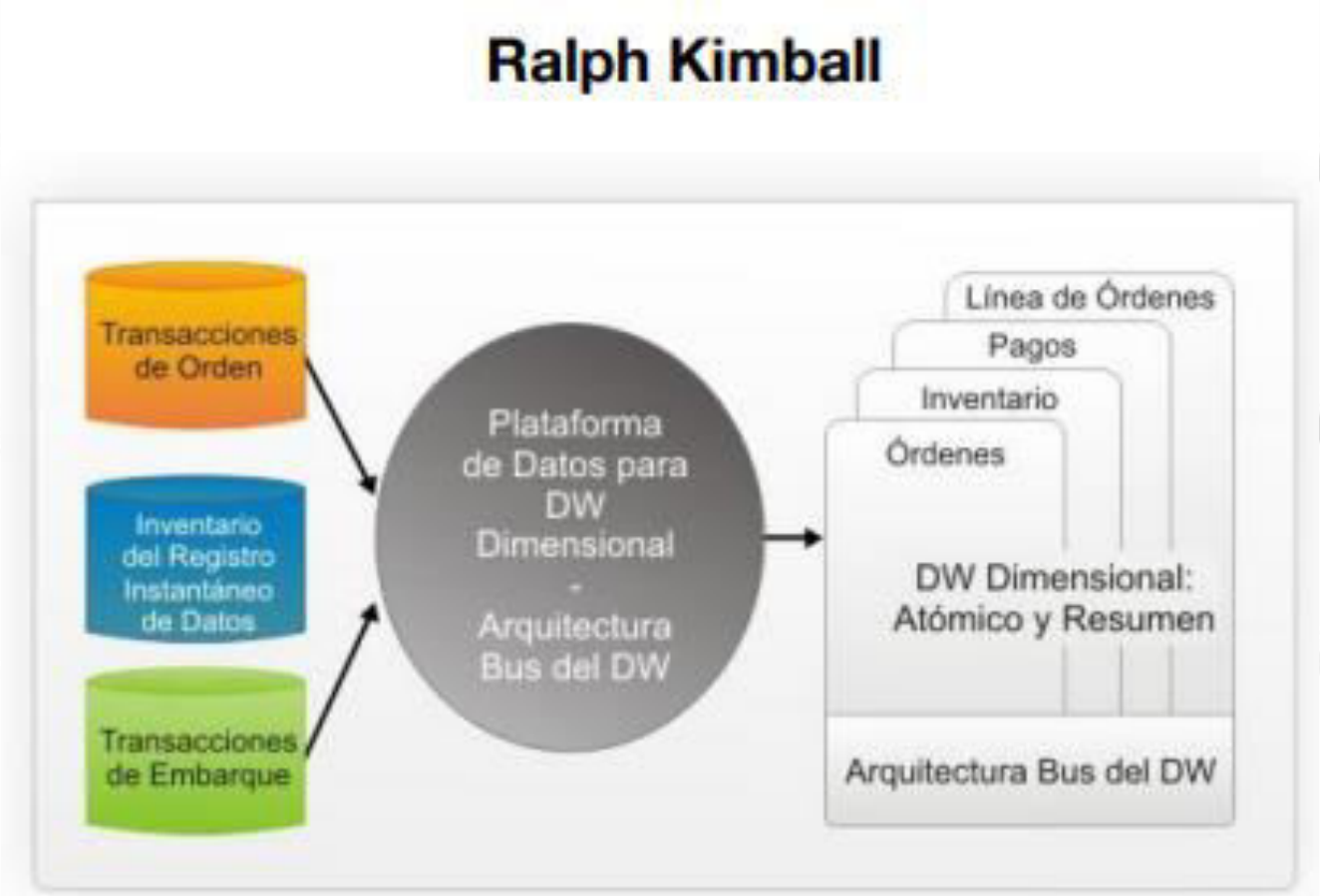
Modelo Copo de Nieve (Snowflake)

Bill Inmon

El concepto de Datawarehouse (DW) llegó de la mano de Bill Inmon y Ralph Kimball. Ambos pensaron en un **único repositorio de información** para poder integrar y explotar información **de diversos sistemas fuentes**. Pero, más allá de esta generalización conceptual, cada uno decidió hacerlo a su manera.

CePETel

Data Warehouse

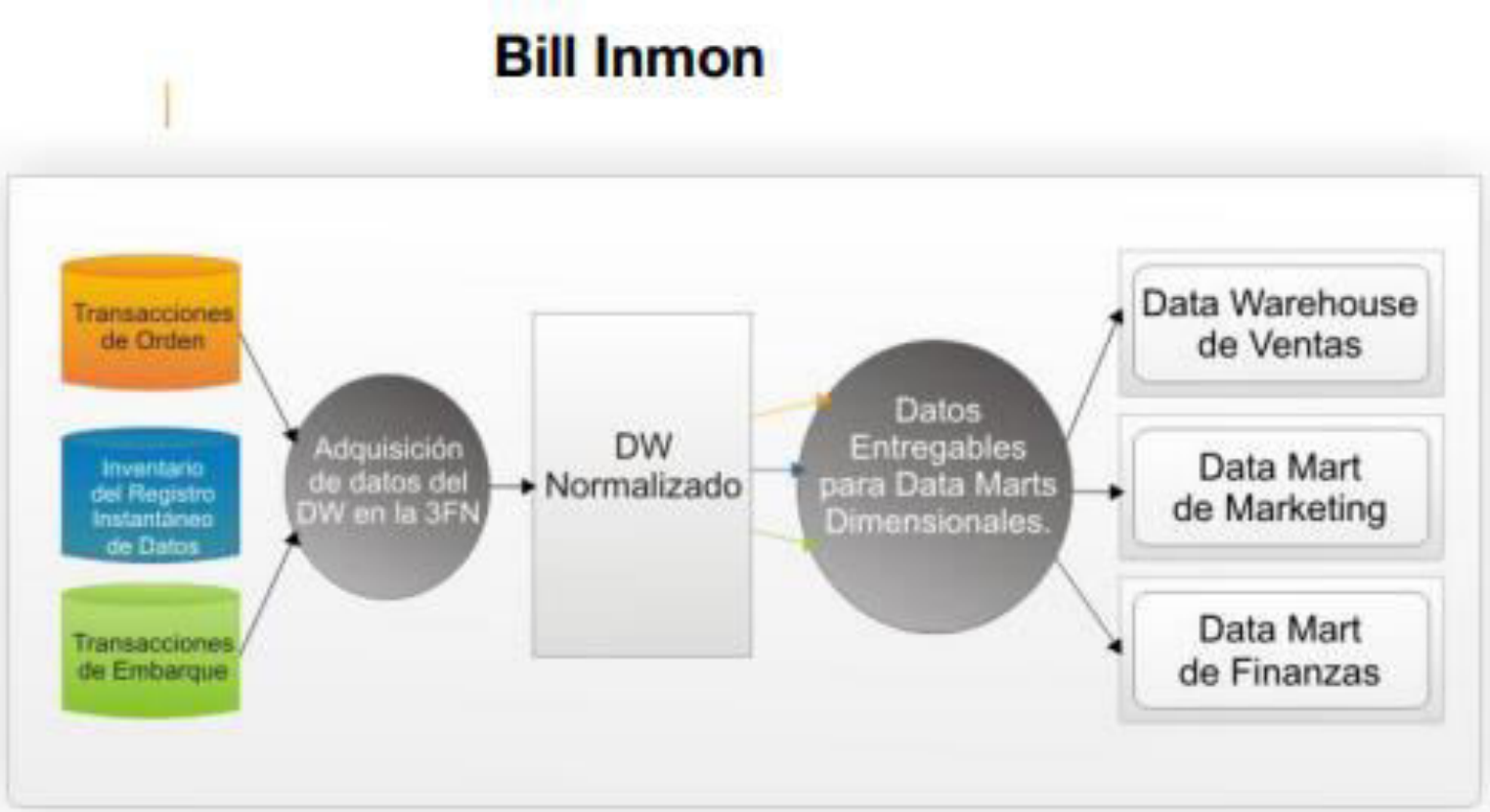


Kimball sugiere utilizar una metodología **Bottom-Up**, donde la información se extrae de los sistemas transaccionales para ser cargada en diferentes **Data Marts** cada uno de los cuales son **independientes**, están modelados en forma dimensional y tienen **foco por proceso.**

CePETel

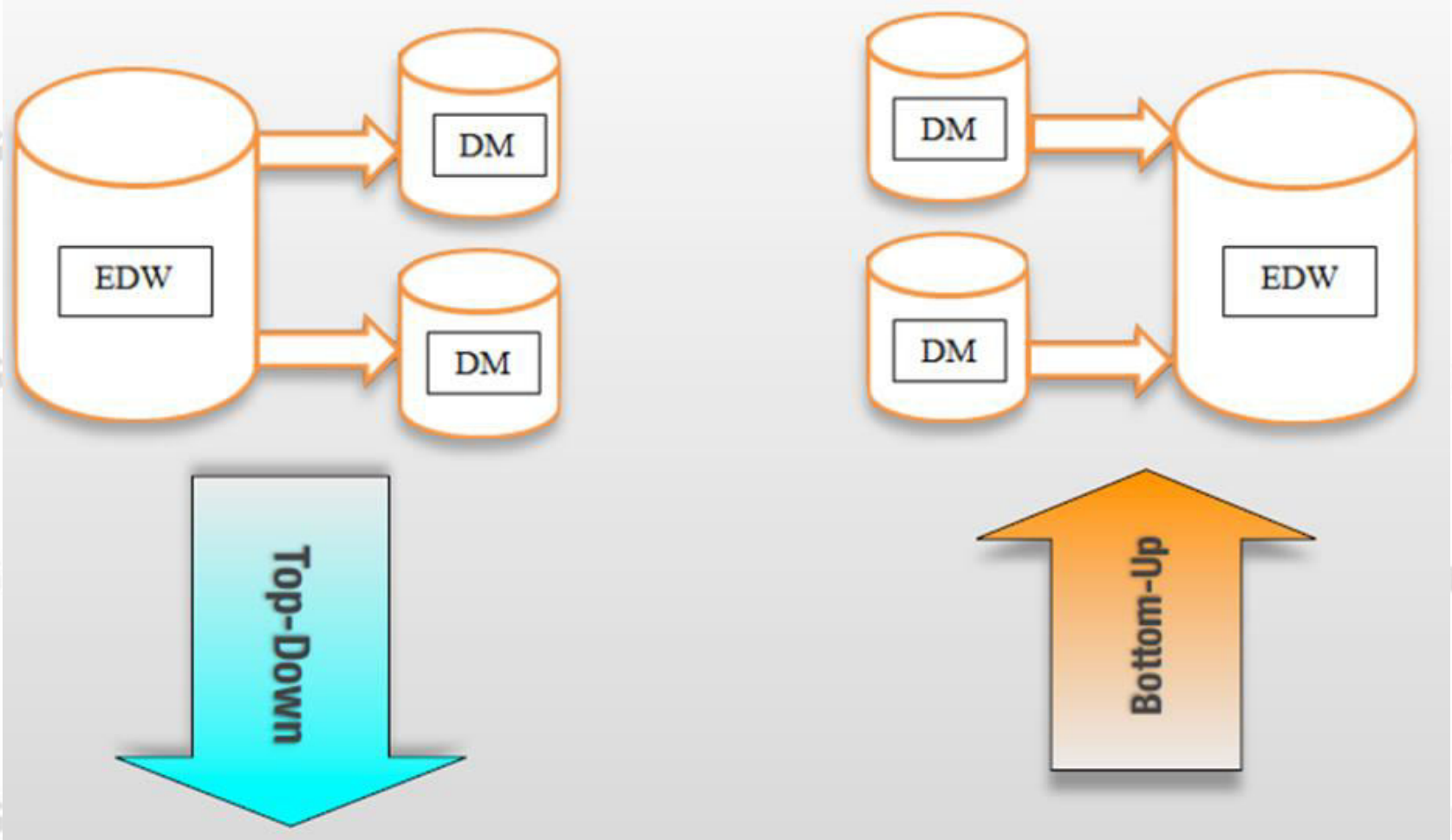
Data Warehouse

Con **Inmon**, el **DW** no está modelado dimensionalmente, sino que tiene el formato de un **modelo relacional**. Una vez que tenemos el DW generado de esta manera, se pueden crear los Datamarts para las áreas de negocio que necesitemos, y además lo podríamos utilizar para cualquier otro tipo de sistema decisional como por ejemplo sistemas expertos, o minería de datos.



CePETel

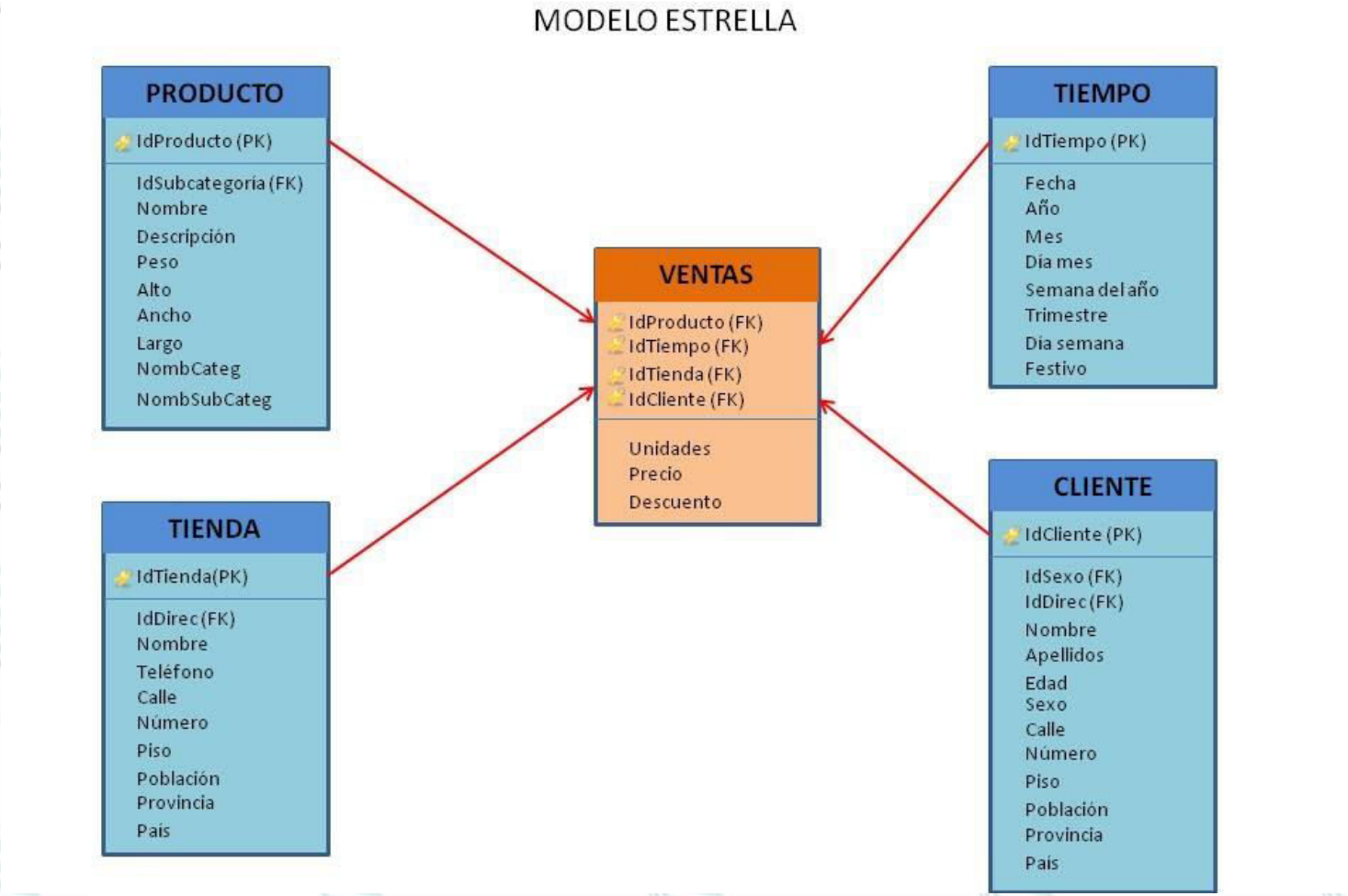
Data Warehouse



CePETel

Data Warehouse

Modelo Estrella



Esquema de estrella es un modelo de datos formado por **una tabla de hechos**, que contiene los datos para el análisis, **rodeada de las tablas de dimensiones**.

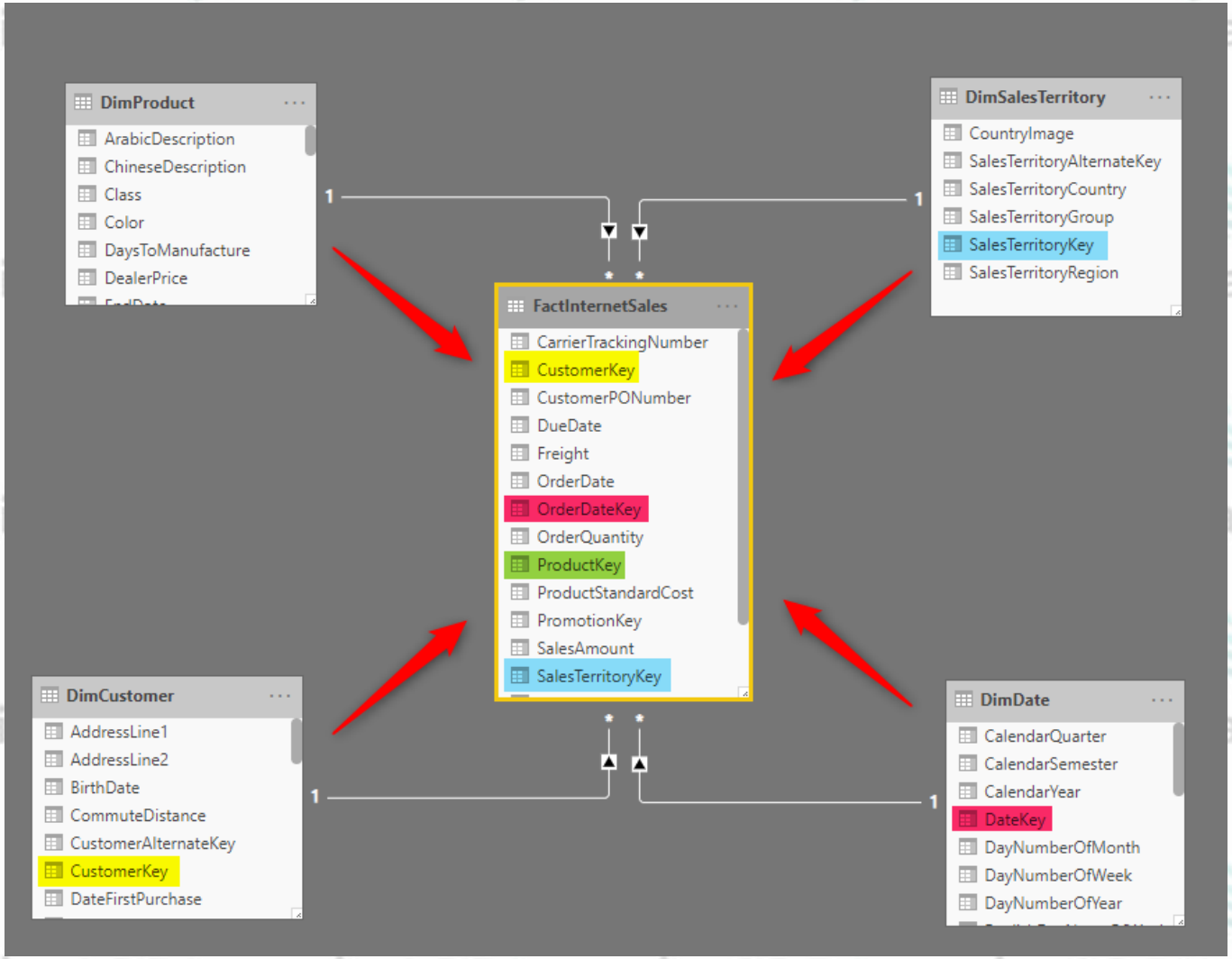
CePETel

Data Warehouse

Modelo Estrella

El modelo estrella separa los datos del proceso de negocio en: hechos y dimensiones.

- Los **hechos** contienen datos medibles, cuantitativos, y
- Las **dimensiones** contienen los atributos que describen los datos indicados en los hechos.

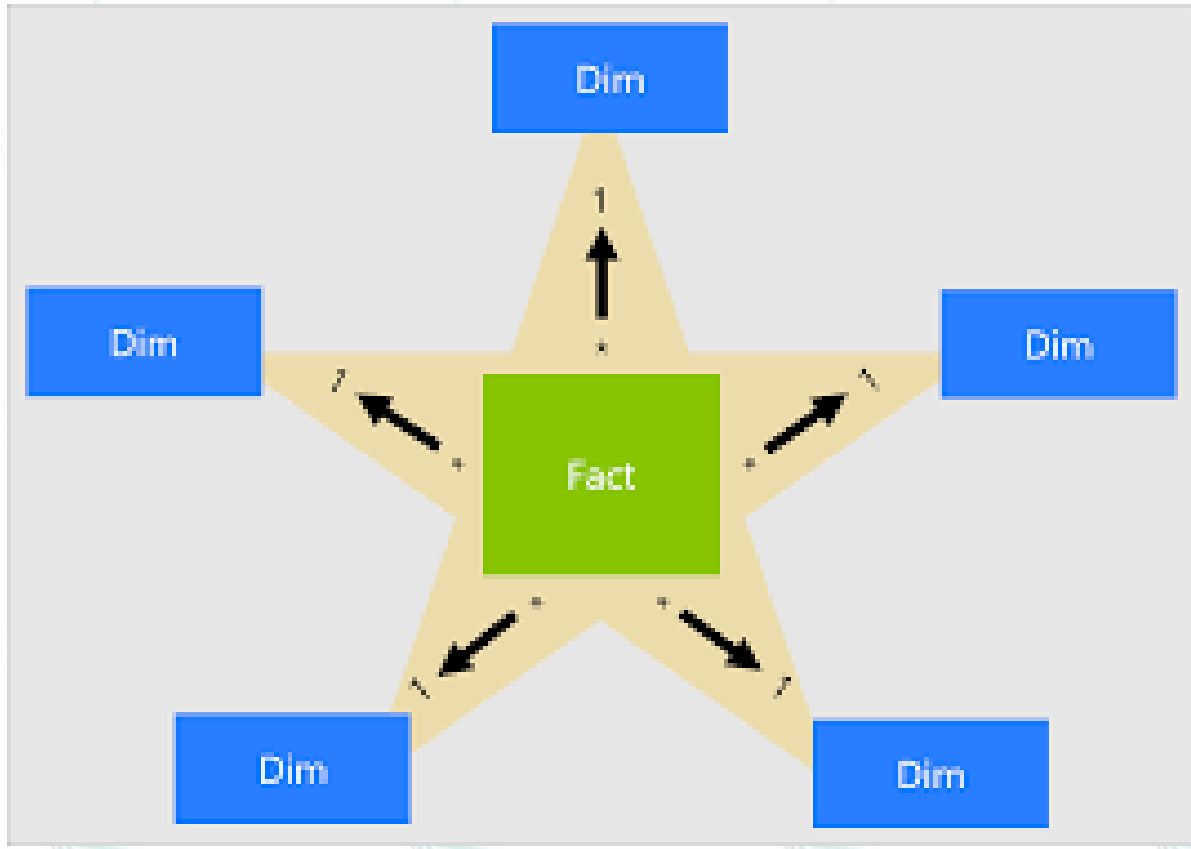


CePETel

Data Warehouse

Modelo Estrella

- ✓ Puede estar centralizado o distribuido
- ✓ El EDW puede implementarse en una base de datos relacional o en una Multidimensional
- ✓ Se construye aplicando la técnica de modelado dimensional
- ✓ Centrado en los procesos de negocio
- ✓ Los usuarios acceden directamente



CePETel

Data Warehouse

Modelo Estrella

Ventajas:

- Menos cantidad de tablas (una por cada dimensión).
- Queries simples. Las uniones y cruces son más sencillos.
- Mejoras en el rendimiento de las consultas. Menos usos de JOINs, aunque un poco más de usos de agregaciones y distintcs.

Desventajas:

- Poco flexible. Los esquemas en estrella son construidos para una vista de los datos en particular
- Más difícil de actualizar los datos en dimensiones por desnormalización.
- Requiere más almacenamiento de datos por desnormalización

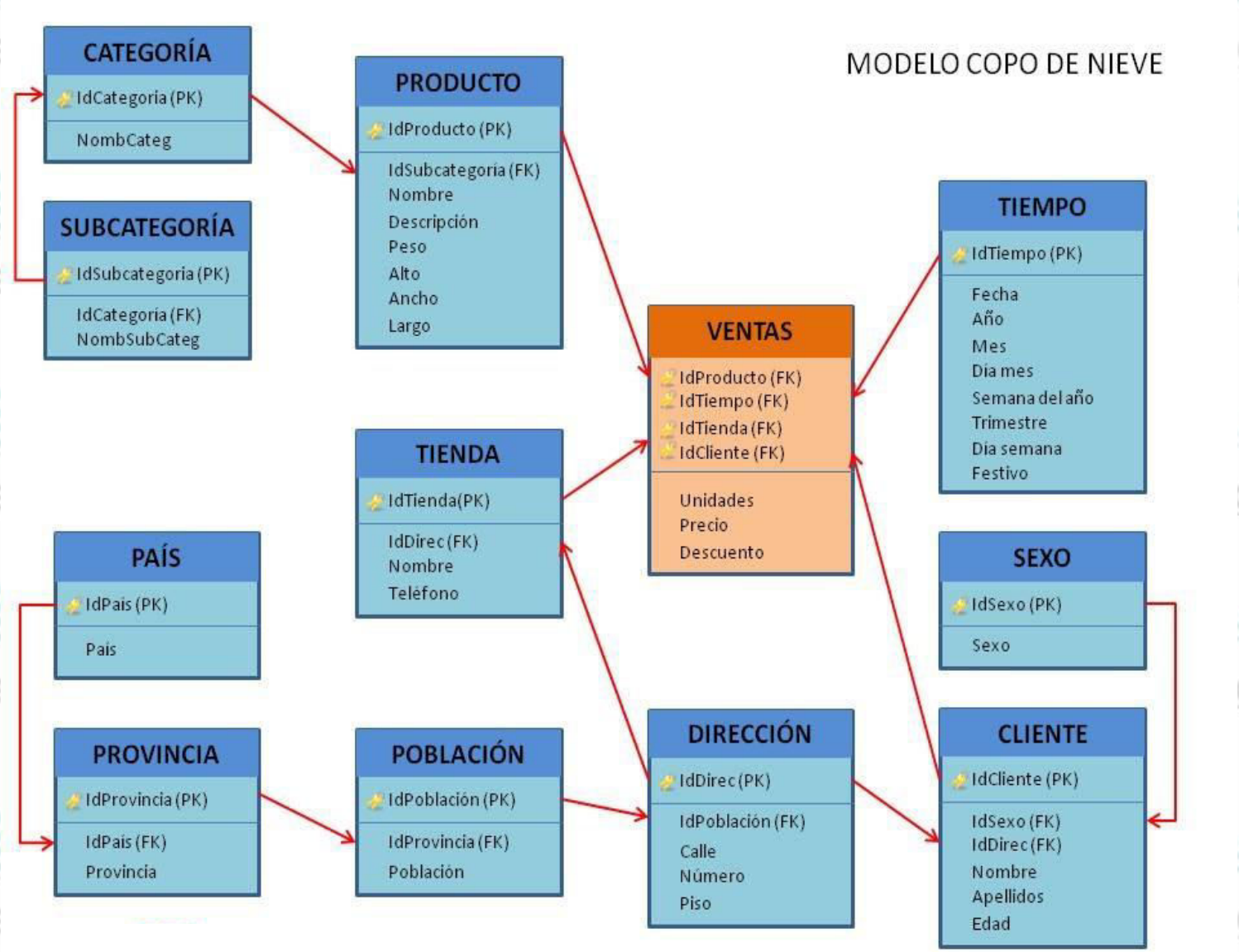
CePETel

Data Warehouse

Modelo Copo de nieve

Estructura **más compleja** que el esquema de estrella.

Se da cuando alguna de las dimensiones se implementa con más de una tabla de datos.

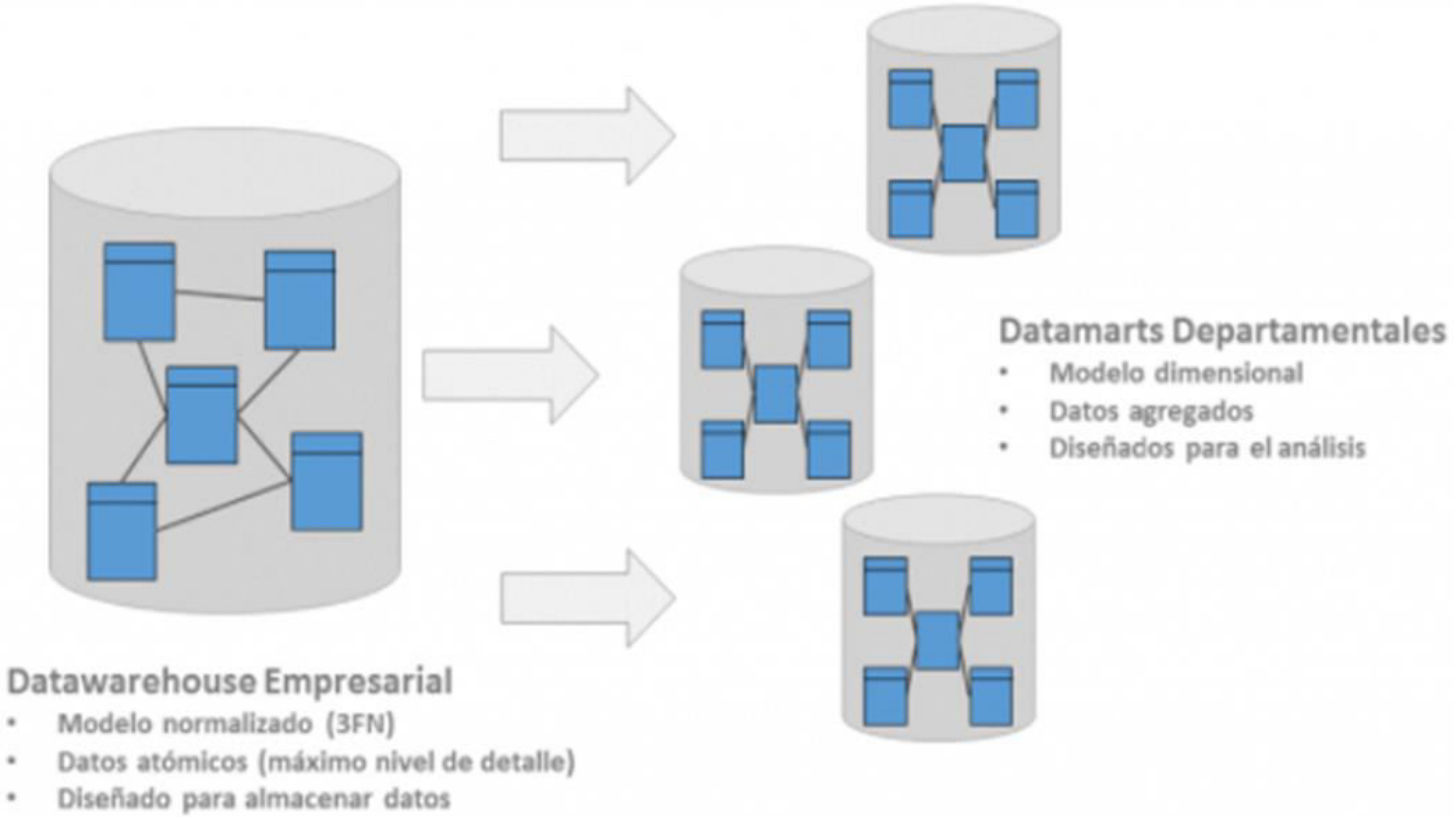


CePETel

Data Warehouse

Modelo Copo de nieve

El objetivo es normalizar estas tablas se representa como una tabla de hechos conectada con dimensiones anidadas. Al normalizar por completo las dimensiones el resultado parece un copo de nieve



CePETel

Data Warehouse

Modelo Copo de nieve

Ventajas:

- La normalización de los atributos reduce el almacenamiento de datos.
- Fácil actualización de datos gracias a normalización.
- Algunas herramientas de modelado de bases de datos multidimensional OLAP se optimizan

Desventajas:

- Mayor cantidad de tablas, modelo más difícil de entender para el usuario final.
- Queries complejas debido a la normalización (implica un mayor número de tablas y de cruces)
- Bajo rendimiento debido a la normalización

CePETel

Modelo Estrella vs. Snowflake

	Estrella	Snowflake
Mantenimiento	Tiene redundancia. Poco mantenimiento	No hay redundancia. Fácil mantenimiento
Facilidad de uso	Queries menos complejas. Fácil uso	Queries complejas. Difícil de entender
Rendimiento de las queries	Ejecuciones más rápidas	Más tiempo de ejecución debido a los cruces
Tipo de DW	Data Mart	Data Warehouse
Joins	Bajo número de joins	Alto número de joins
Tablas de dimensión	Una tabla de dimensión por cada dimensión	Más de una tabla de dimensión por dimensión
Cuando usarlo	Cuando las tablas de dimensión tienen pocas filas	Cuando las tablas de dimensión tienen un tamaño bastante elevado
Normalización / Desnormalización	Tablas de dimensión y de hechos desnormalizadas	Tablas de dimensión normalizadas. Tablas de hechos desnormalizadas
Modelo de datos	Top - Down	Bottom - Up

CePETel

Data Warehouse - Proceso de Modelado

Recordando!

El proceso consiste en cuatro pasos:

1. Elegir el proceso de negocio: Consiste en elegir el área a modelar. Es una decisión de la dirección, y depende fundamentalmente del análisis de requerimientos y de los temas analíticos anotados en la etapa anterior.

2. Establecer el nivel de granularidad: La granularidad significa especificar el nivel de detalle. La elección de la granularidad depende de los requerimientos del negocio y lo que es posible a partir de los datos actuales. La sugerencia general es comenzar a diseñar el **DW** al mayor nivel de detalle posible, ya que se podrían realizar agrupamientos posteriores, al nivel deseado.

3. Elegir las dimensiones: Las dimensiones surgen naturalmente de las discusiones del equipo, y facilitadas por la elección del nivel de granularidad. Las tablas de dimensiones tienen un conjunto de atributos (generalmente textuales) que brindan una perspectiva o forma de análisis sobre una medida en una tabla hechos. Una forma de identificar las tablas de dimensiones es que sus atributos son posibles candidatos para ser encabezado en los informes, tablas pivot, cubos, o cualquier forma de visualización, unidimensional o multidimensional.

Data Warehouse - Proceso de Modelado

Recordando!

4. Identificar medidas y las tablas de hechos: Este paso consiste en identificar las medidas que surgen de los procesos de negocios. Una medida es un atributo (campo) de una tabla que se desea analizar, sumando o agrupando sus datos y usando los criterios de corte conocidos como dimensiones.
Las medidas habitualmente se vinculan con el nivel de granularidad. Un registro contiene una medida expresada en números, como ser cantidad, tiempo, dinero, etc., sobre la cual se desea realizar una operación de agregación (promedio, conteo, suma, etc.) en función de una o más dimensiones. La granularidad, en este punto, es el nivel de detalle que posee cada registro de una tabla de hechos.

CePETel

Tablas de Hechos y Dimensiones

Implementación Modelo Dimensional

Tablas de hechos / Fact tables

- **Tabla principal** del modelo dimensional
- Contienen campos claves que **se unen a las tablas de dimensión**, dan contexto y nivel de detalle al dato.
- Contiene métricas o también llamadas **medidas** y es **aquello que queremos medir o analizar**. Generalmente son **valores numéricos** que se suelen agregar.

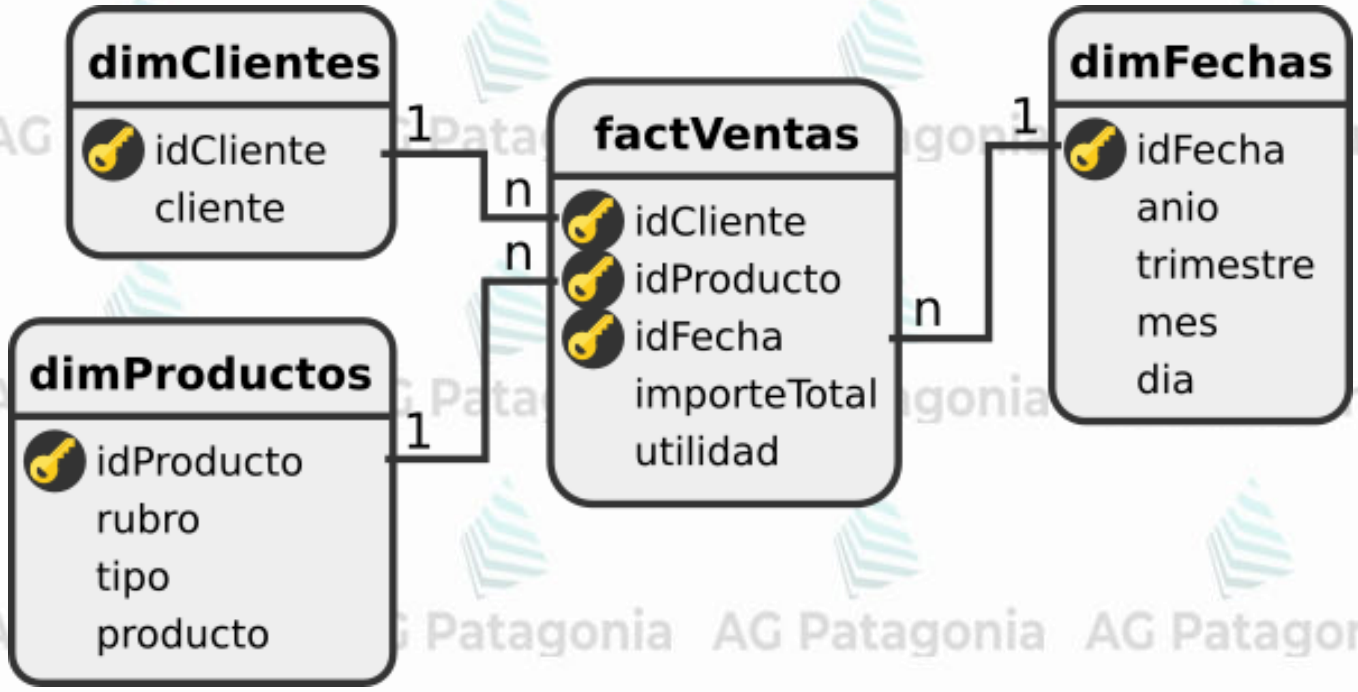
Ejemplo: Ventas de coches de un concesionario, ganancia en euros de una empresa, el número de materias aprobadas de un colegio, etc.

CePETel

Implementación Modelo Dimensional

Tablas de hechos / Fact tables

- Cada **fila** de la tabla **corresponde** a un **evento ocurrido**.
- Evitan la redundancia de atributos por estas estos en las tablas de dimensiones.
- Normalmente tienen **muchos** (millones) **registros** por ejemplo: ventas, compras, movimientos de contabilidad, y es donde ocurre la mayoría de escrituras al modelo.



CePETel

Tablas de hechos

Medidas

Son los valores que recogen el proceso de una actividad o los resultados de la misma. Estas medidas proceden del resultado de la actividad del negocio.

Medidas en tablas de hecho, se pueden clasificar por:

- Naturaleza aditiva
- Tipo de agregación
- Nivel de detalle/granularidad

Pasos para el diseño de una tabla de hechos:

1. Seleccionar un proceso de negocio para modelar.
2. Declarar el nivel de granularidad.
3. Elegir las dimensiones.
4. Identificar los hechos.

CePETel

Tablas de hechos

Nivel de Granularidad

El primer paso para definir la tabla de hechos del Data Warehouse es definir la granularidad de la tabla, la granularidad indica el nivel de detalle que se mide por cada fila de datos. Por ejemplo, cada fila representa una venta de un producto por cliente por día o una fila representa el total de venta por cliente por día.

- Por ejemplo en la dimensión Sucursal:

Dimensión Sucursal	
*	Sucursal
**	Tipo Sucursal

Dimensión Sucursal	
*	Sucursal
**	Tipo Sucursal
***	País

Dimensión Sucursal	
*	Sucursal
**	Tipo Sucursal
***	País
****	Provincia

Dimensión Sucursal	
*	Sucursal
**	Tipo Sucursal
***	País
****	Provincia
*****	Ciudad

CePETel

Tablas de hechos

Nivel de Granularidad

- Define que representa cada fila.
- Niveles de detalle no se pueden mezclar (medidas con distintos niveles de detalle -> tablas físicas distintas)
- Debe ser consistente con las dimensiones asociadas a la tabla.
- Registrar hechos al mayor nivel de detalle posible.
  - Flexibilidad para responder consultas del usuario impredecibles
  - Siempre se puede subir en la jerarquía.
  - Mayor espacio de almacenamiento y cantidad de registros a procesar.

FactWeekly

WeekKey	Measure1
1	200
2	200
3	200
4	200
5	200
6	200

DimWeek

WeekKey	Week	StartDate	EndDate
1	Week 1	27 Dec 2010	2 Jan 2011
2	Week 2	3 Jan 2011	9 Jan 2011
3	Week 3	10 Jan 2011	16 Jan 2011
4	Week 4	17 Jan 2011	23 Jan 2011
5	Week 5	24 Jan 2011	30 Jan 2011
6	Week 6	31 Jan 2011	6 Feb 2011

FactMonthly

MonthKey	Measure1
1	4500
2	4000

DimMonth

MonthKey	Month
1	40544
2	40575

CePETel

Decidir la granularidad

- La granularidad:
  - Es el nivel de detalle al que se desea almacenar información sobre la actividad a modelar.
  - Define el nivel atómico de datos en el almacén de datos.
  - Determina el significado de las tuplas de la tabla de hechos.
  - Determina las dimensiones básicas del esquema.

Tablas de hechos

Tipos de medidas

Valores cuantitativos dentro de una tabla de hecho o medidas, se clasifican en 3 tipos:

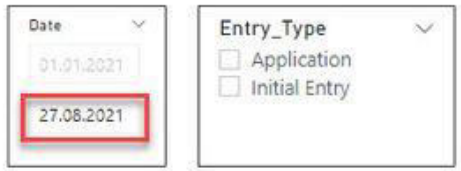
- **Aditivas**: se pueden **sumar** a lo largo de **cualquier dimensión** relacionadas a la tabla de hechos, dando siempre un valor correcto.

CePETel

Tablas de hechos

Tipos de medidas

- **Semi-aditivas:** se pueden **sumar** a lo largo de **algunas dimensiones** asociadas a la tabla de hecho, pero **no de todas**. Sumar a lo largo de estas dimensiones daría un resultado incorrecto o que carece de sentido.
  - Casos típicos: Balance, stock, objetivos de ventas, presupuestos.



Index	Entry_No	Entry_Type	Posting_Date	Document_Type	Initial_Entry_Due_Date	Amount_LCY	mx Original Amount	mx Remaining Amount2	mx Remaining Amount3	Customer PastDue	Age Group	Customer Sold per Aging Grup
1	460697	Initial Entry	20.07.2021	Invoice	04.08.2021	290,28	290,28	290,28	290,28	26	Overdue (1-30)	290,28
2	462686	Initial Entry	23.07.2021	Invoice	07.08.2021	6.351,91	6.351,91	6.351,91	6.351,91	23	Overdue (1-30)	6.351,91
3	462706	Initial Entry	23.07.2021	Invoice	07.08.2021	735,49	735,49	735,49	735,49	23	Overdue (1-30)	735,49
4	433155	Initial Entry	30.07.2021	Invoice	29.08.2021	165,84	165,84		0,00	1	Overdue (1-30)	165,84
5	433181	Initial Entry	30.07.2021	Invoice	29.08.2021	191,49	191,49		0,00	1	Overdue (1-30)	191,49
6	433728	Initial Entry	30.07.2021	Invoice	29.08.2021	4.822,44	4.822,44		4.822,44	1	Overdue (1-30)	4.822,44
7	441974	Initial Entry	30.07.2021	Invoice	14.08.2021	642,60	642,60	642,60	642,60	16	Overdue (1-30)	642,60
8	464932	Initial Entry	30.07.2021	Invoice	14.08.2021	1.083,58	1.083,58	1.083,58	1.083,58	16	Overdue (1-30)	1.083,58
9	434565	Initial Entry	03.08.2021	Payment	03.08.2021	-268,21	-268,21	0,00	0,00	27	Overdue (1-30)	-268,21
10	433711	Initial Entry	04.08.2021	Invoice	19.08.2021	78,31	78,31	0,00	0,00	11	Overdue (1-30)	78,31
11	464216	Initial Entry	09.08.2021	Invoice	24.08.2021	1.338,75	1.338,75	1.338,75	1.338,75	6	Overdue (1-30)	1.338,75
12	464236	Initial Entry	09.08.2021	Invoice	24.08.2021	734,46	734,46	734,46	734,46	6	Overdue (1-30)	734,46
13	464952	Initial Entry	13.08.2021	Invoice	28.08.2021	3.347,74	3.347,74		3.347,74	2	Overdue (1-30)	3.347,74
14	464983	Initial Entry	13.08.2021	Invoice	28.08.2021	44,86	44,86		44,86	2	Overdue (1-30)	44,86
15	441097	Initial Entry	20.08.2021	Payment	20.08.2021	-78,31	-78,31	0,00	0,00	10	Overdue (1-30)	-78,31
16	434565	Application	03.08.2021	Payment	03.08.2021	268,21				27	Overdue (1-30)	268,21
17	433711	Application	20.08.2021	Invoice	19.08.2021	-78,31				11	Overdue (1-30)	-78,31
18	441097	Application	20.08.2021	Payment	20.08.2021	78,31				10	Overdue (1-30)	78,31
19	433155	Application	30.08.2021	Invoice	29.08.2021	-165,84				1	Overdue (1-30)	-165,84
20	433181	Application	30.08.2021	Invoice	29.08.2021	-191,49				1	Overdue (1-30)	-191,49
Total						**19.392,11**	**19.481,23**	**19.392,11**	**19.392,11**			**19.392,11**

CePETel

Tablas de hechos

Tipos de medidas

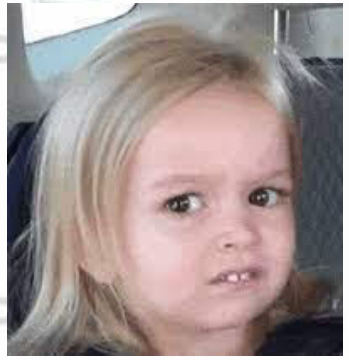
Valores cuantitativos dentro de una tabla de hecho o medidas, se clasifican en 3 tipos:

- **No aditivas**: **no** se pueden **sumar** a lo largo de **ninguna dimensión** asociadas a la tabla de hechos.
  - Casos típicos: ratios, descuentos, promedios, temperaturas y velocidad.

CePETel

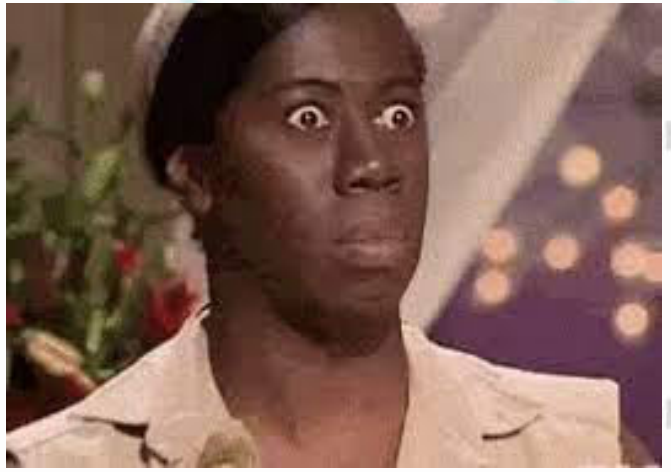
Tipos de Medidas

Tipos de medidas



Tengo 500

Gasto 200	Quedan 300
Gasto 150	Quedan 150
Gasto 90	Quedan 60
Gasto 60	Quedan 0
500	**510**



CePETel

Tipos de Medidas

Tipos de medidas

Tengo 500

Gasto 200	Quedan 300
Gasto 150	Quedan 150
Gasto 90	Quedan 60
Gasto 60	Quedan 0
500	**510**

- "Gasto" -> Medida aditiva, sumable a lo largo de cualquier dimensión.
- "Quedan" -> Medida semi-aditiva, no sumable a lo largo del tiempo.

CePETel

Tipos de Medidas

Tipos de medidas

Tengo 500

Gasto 200	Quedan 300
Gasto 150	Quedan 150
Gasto 90	Quedan 60
Gasto 60	Quedan 0
500	**510**

¿Por qué es importante?

- Evitar cálculos erróneos en las queries de consulta.
- Diseñar modelos dimensionales que permitan recomputar datos a distintos niveles de detalle.
  Muy útil para tabla agregadas.

CePETel

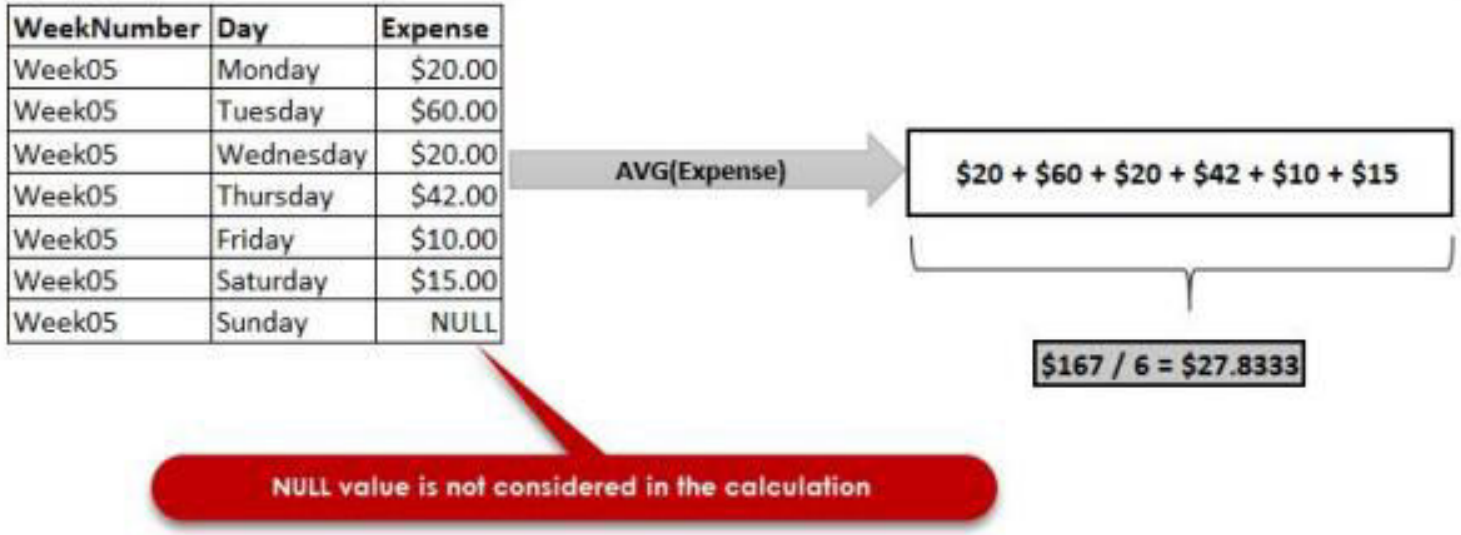
Tipos de Medidas

Medidas con nulos

En general, no hay problema con valores “null”s en las medidas dentro de una tabla de hecho.

- Funciones analiticas o de agregacion (SUM, AVG, COUNT, MIN, MAX) suelen computar los valores nulos de forma correcta.

- No suele ser así con valores nulos en las dimensiones.



CePETel

Implementación Modelo Dimensional

Tablas de Dimensiones / Lookup Tables

- Tablas **simples**, usualmente **anchas** y que tienden a la **desnormalización.**
- Son **usadas como punto de entrada** a los datos (como filtros agrupadores, etc.) y se **unen a las tablas de hechos a través de un campo clave.**
- Los atributos de la tabla de dimensión ofrecen información característica de las tablas de hechos, **ofreciendo contexto y nivel de detalle**.
- No hay **límite** de tablas de dimensión.

CePETel

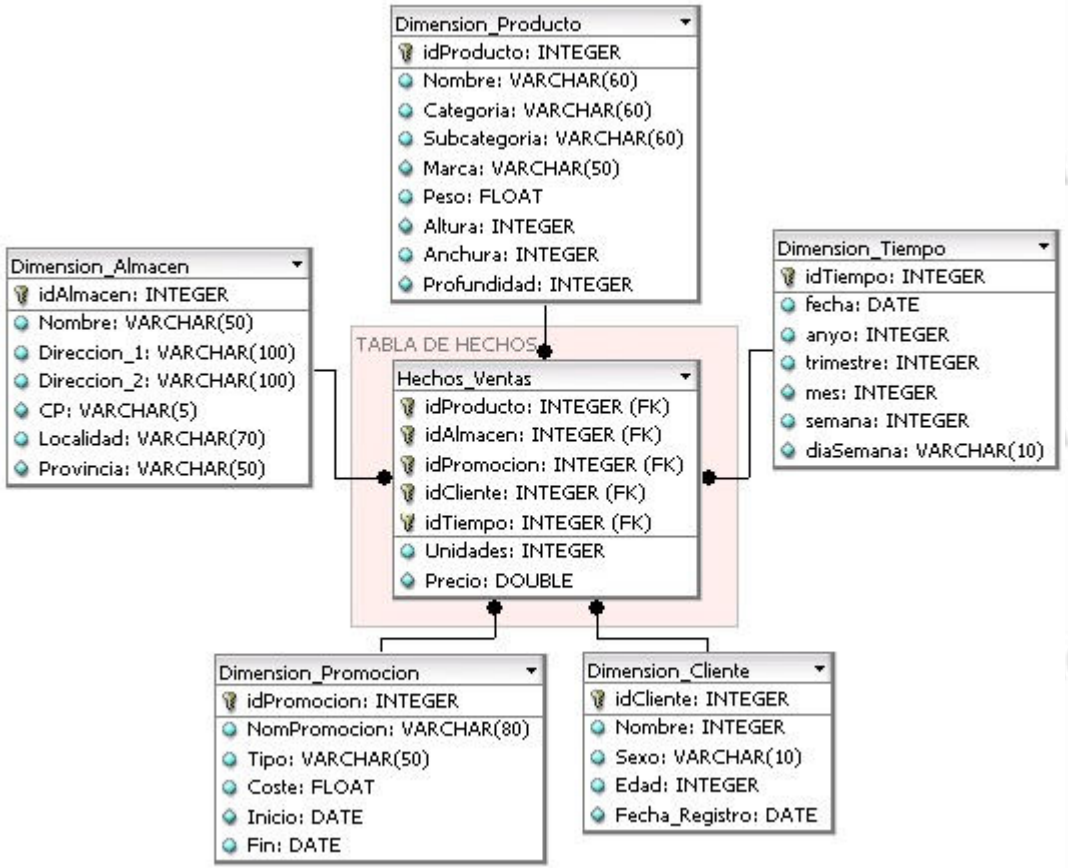
Implementación Modelo Dimensional

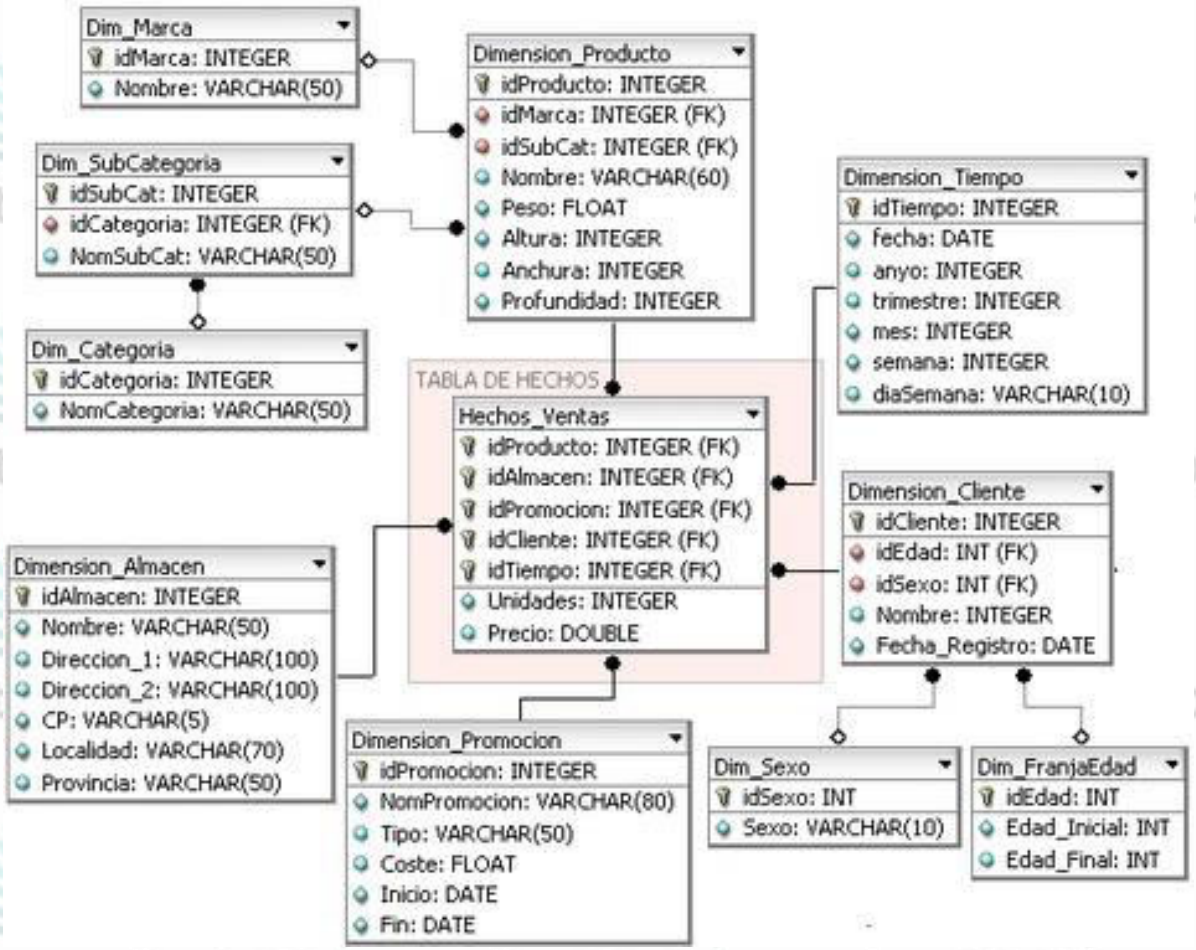
Tablas de Dimensiones / Lookup Tables

- Las dimensiones pueden contener una o varias relaciones jerárquicas.
- Grado de normalización/desnormalización de las dimensiones determina si se esta armando un modelo Start o Snowflake.
- Normalmente tiene **pocos** (miles) **registros** y sus datos **no se actualizan tan frecuentemente** como las tablas de hecho.
- Contiene **atributos** de **tipo texto**, con **baja cardinalidad**.
- por ejemplo: clientes, productos, almacenes, proveedores, calendario...

CePETel

Implementación Modelo Dimensional





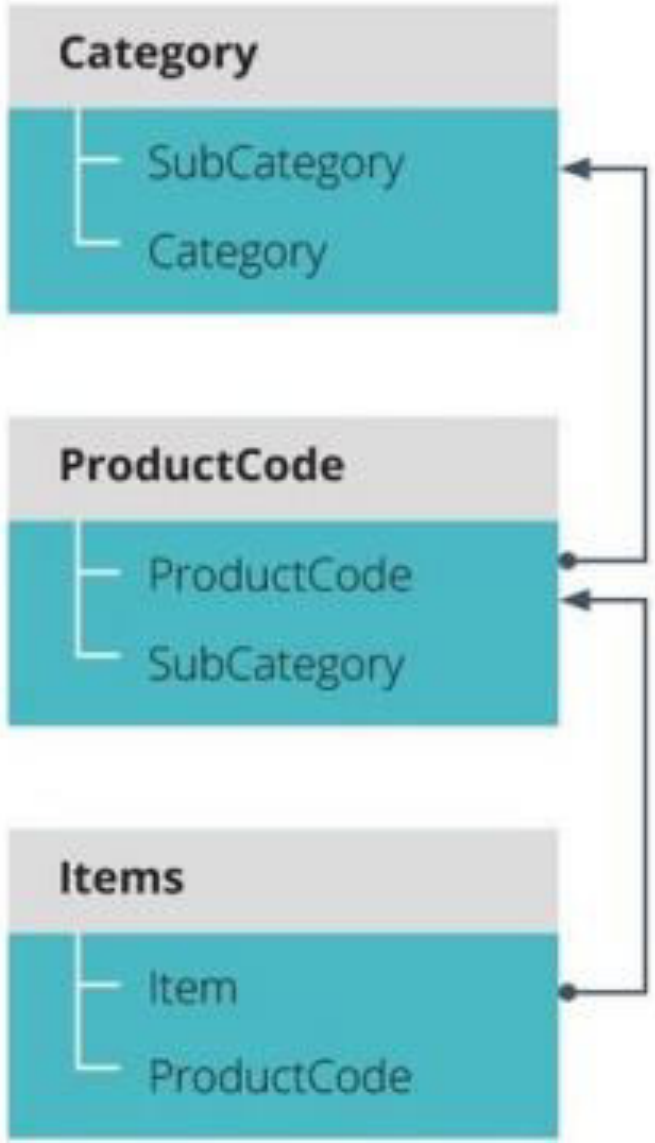
CePETel

Data Warehouse

Modelado & Desnormalización

En diseño de base de datos, la **normalización** es un proceso que se implementa para **organizar y garantizar la integridad** y estructura de los datos.

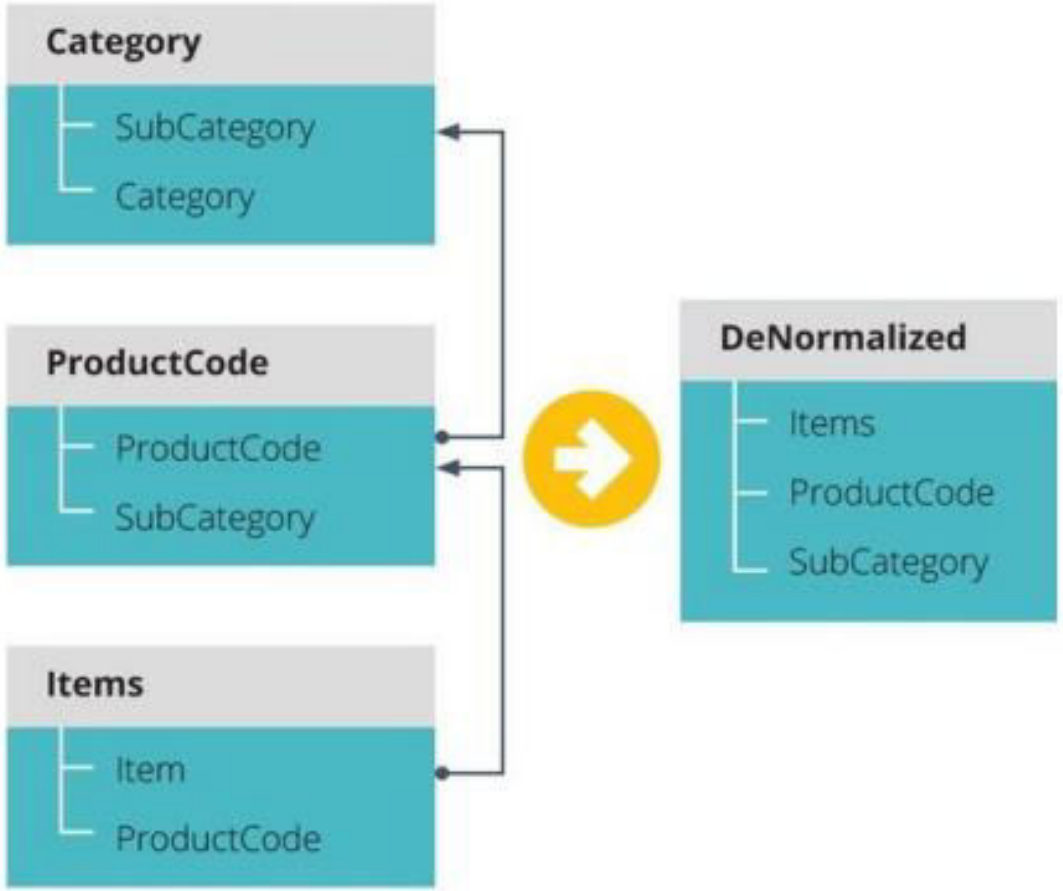
Su meta es **eliminar datos redundantes o duplicados** en la base de datos y asegurar que las **dependencias tengan sentido**. De esta manera, se hace **más fácil el mantenimiento y la escalabilidad del sistema**.



CePETel

Data Warehouse

Modelado & Desnormalización



Hay momentos donde normalizar no es lo más apropiado, como por ejemplo: DW.

El **propósito** es **proveer** la información al usuario de la **manera más rápida posible**, ya sea para informes, estadísticas, métricas, etc.

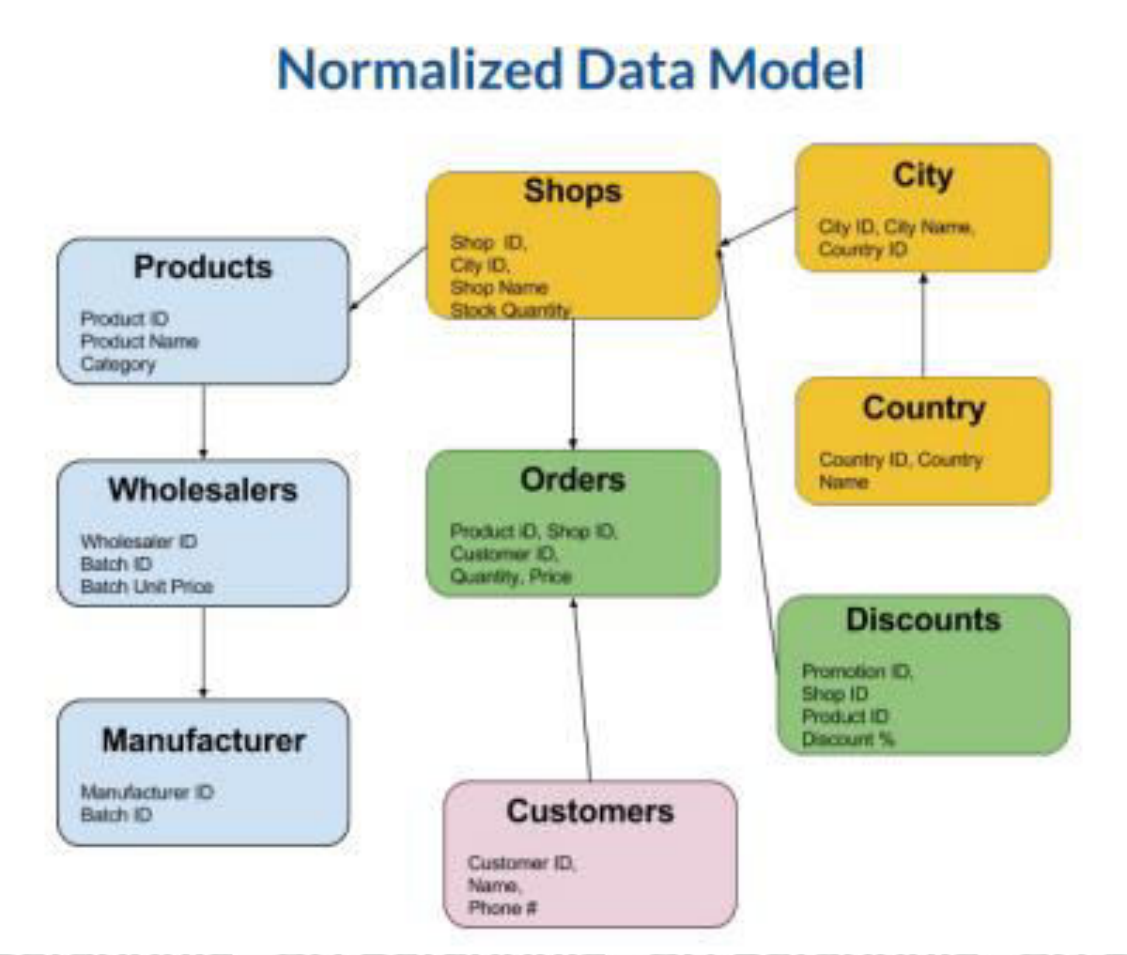
En este modelo de base de datos, la mejor opción es **desnormalizar**.

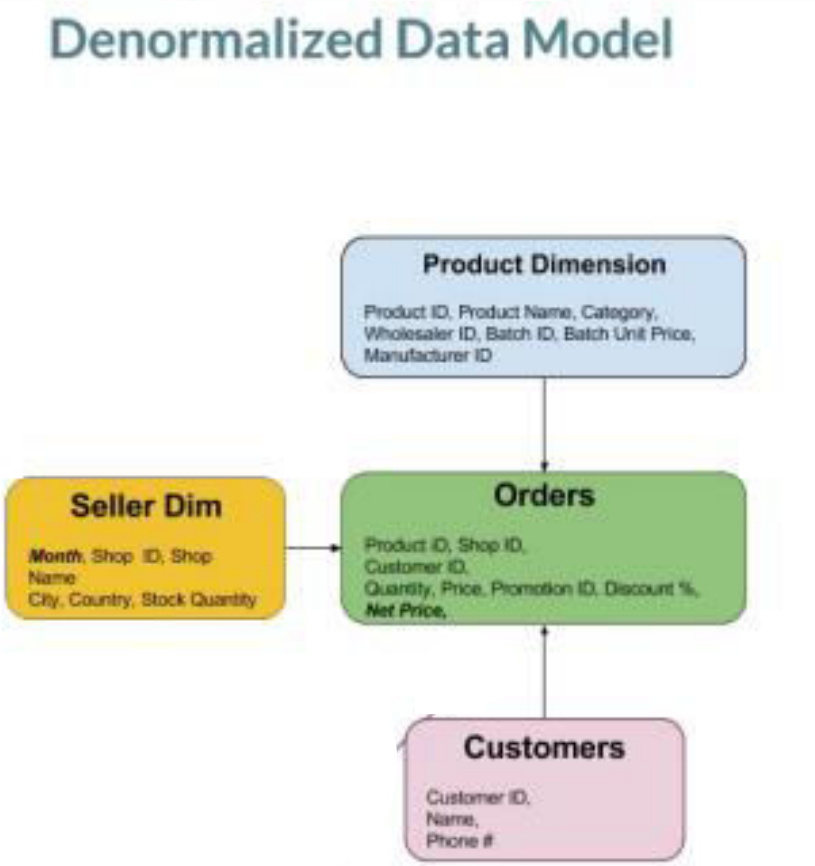
Otro propósito para **desnormalizar** es tener datos históricos, en donde es necesario tener redundancia de datos.

CePETel

Data Warehouse

Modelado vs. Desnormalización

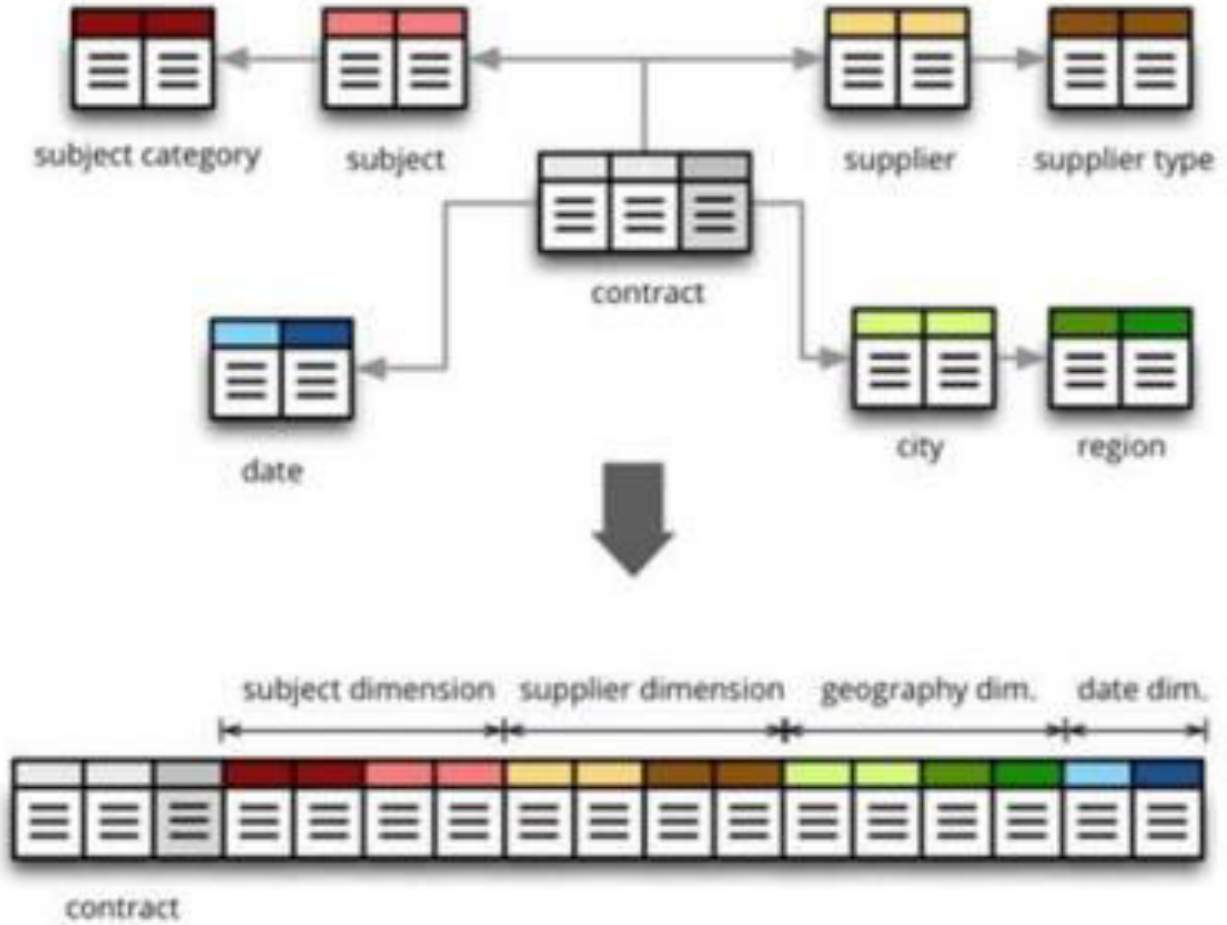


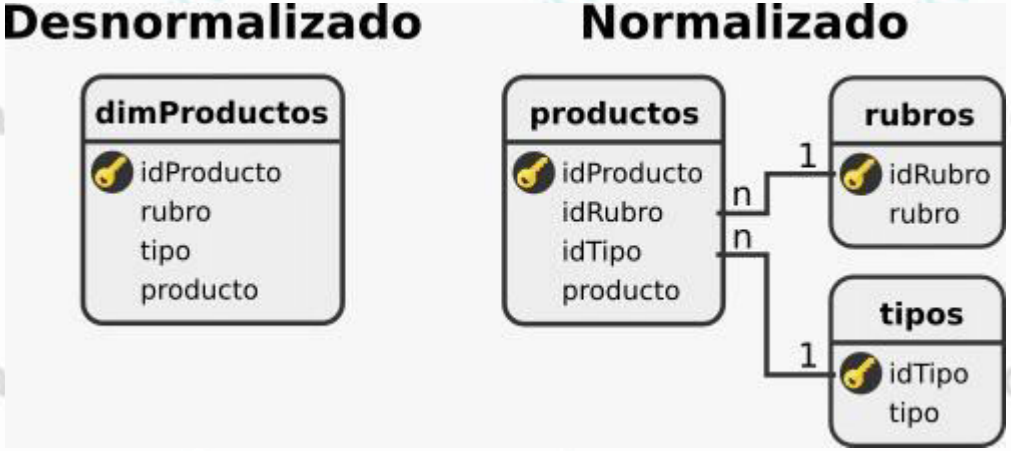


CePETel

Data Warehouse

Modelado vs. Desnormalización





CePETel

Data Warehouse

Video: ¿Qué son los procesos ETL?

Ingestas & ETLs

El **Data warehouse** se alimenta mediante **procesos ETL** (extracción, transformación y carga), desde diferentes fuentes heterogéneas.
El proceso ETL se **comunica** con los sistemas Transaccionales, archivos propios de los usuarios, bases de datos internas/externas, aplicaciones para extraer los datos y cargarlos.

- **Extracción**: obtención de información de las distintas fuentes tanto internas como externas.

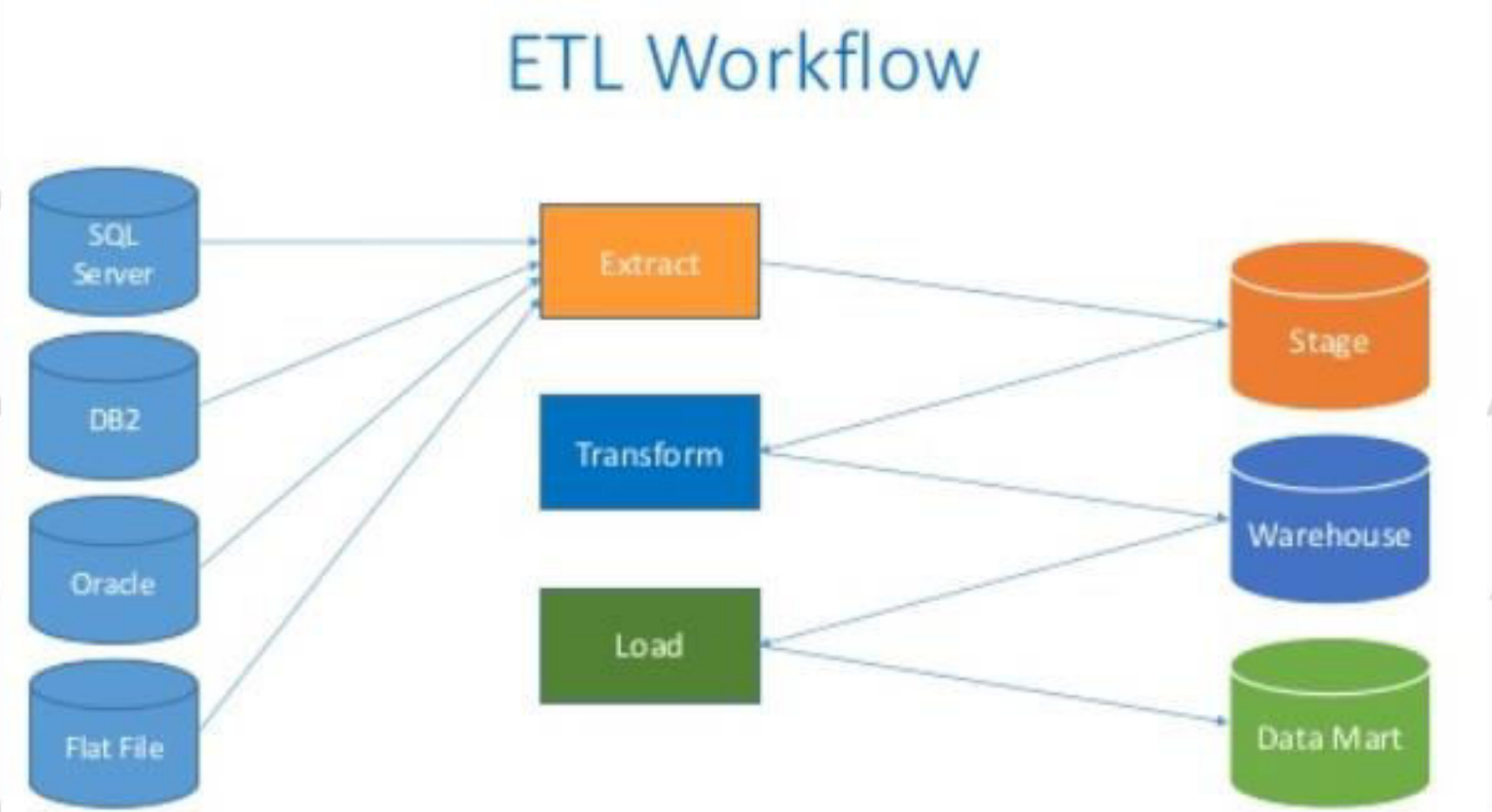
- **Transformación**: filtrado, limpieza, depuración, homogeneización y agrupación de la información.

- **Carga**: organización y carga de los datos y los metadatos en la base de datos.

Antes de realizar la carga final, se pueden reformatear, limpiar, filtrar y posteriormente cargarlos en otra base de datos, en archivos de texto, o en diferentes tecnologías.

CePETel

Data Warehouse



CePETel

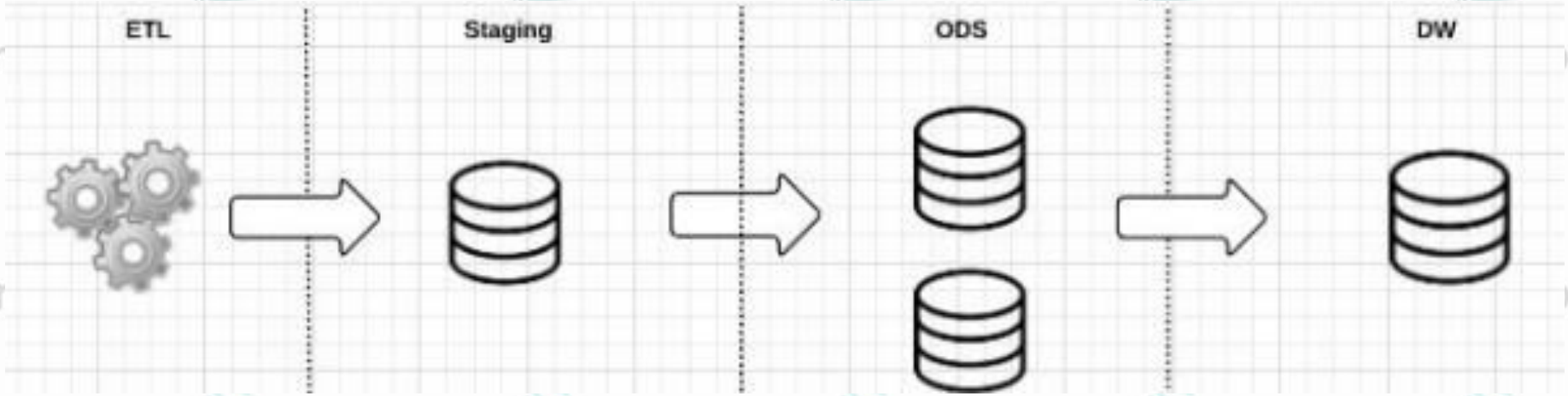
Data Warehouse – ETL - Procedimiento

El procedimiento de extracción consiste en alojar la extracción de datos en un área **Staging**, con el objetivo de bajar los **datos de forma cruda y sin transformaciones** para lograr una **óptima performance.**

El área **Staging** es un área **volátil**, que sólo aloja la extracción en curso (se borra en cada ejecución) y no almacena historia.

El siguiente paso (puede ser opcional) consiste en realizar las transformaciones y formateos correspondientes y almacenar la historia de las extracciones en el área **ODS** (Operational Data Store).

Por último el DW es el área donde se alimenta a los modelos/cubos/datamarts, puede contener toda la historia o sólo un rango específico para reportes (dado que se puede regenerar partiendo del área ODS).

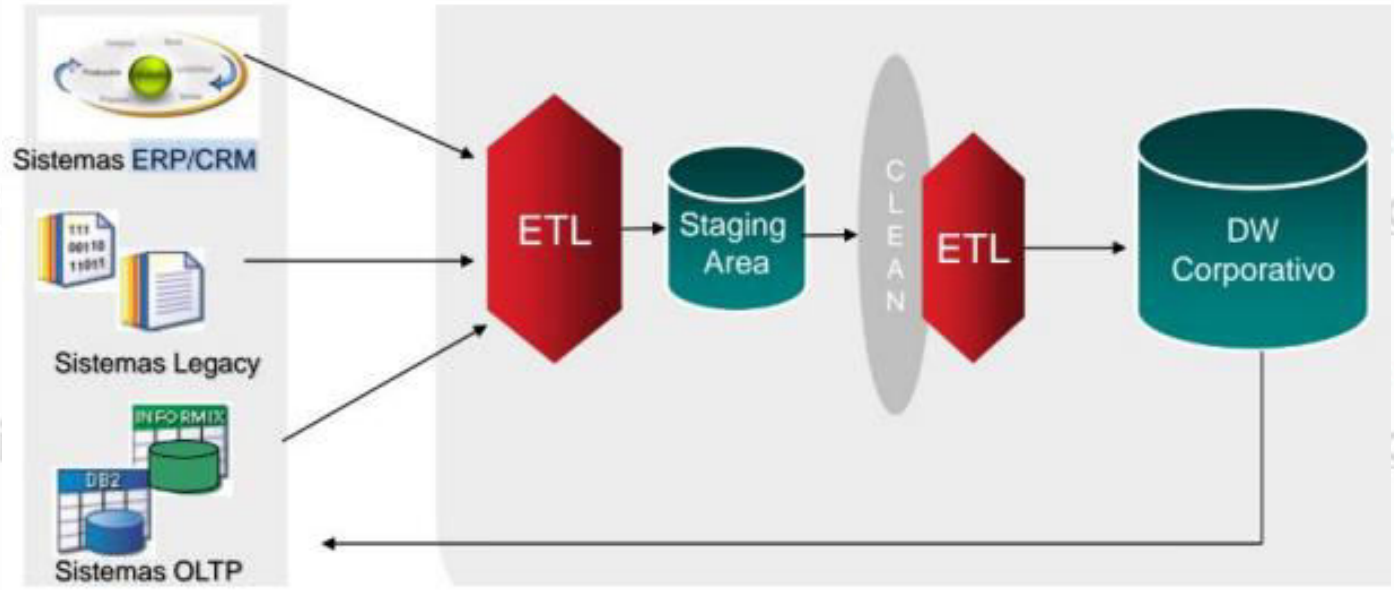


CePETel

Data Warehouse

Staging Area

Es un área de almacenamiento temporal que servirá como punto intermedio para la carga final del DW.



CePETel

Data Warehouse

Staging Area

Usar o no área de Staging dependerá principalmente de:

- Tener los datos en el destino final lo antes posible.
- En caso de falla evitar empezar todo el proceso de su comienzo.

Considerar:

- **Recover**: tener la información en un área de Stage permite retomar desde un punto intermedio en caso que haya una falla.
- **Audit**: Al tener el ETL separado en más partes hace más fácil la auditoría de la carga de información.
- **Backup**: en muchos casos los grandes volúmenes de datos podrían impedir hacer un backup, entonces si la Stage está en un file system se comprimen los archivos y se guardan.

CePETel

Data Warehouse

Staging Area - Tipos

Transitoria

Datos se guardan solamente durante el proceso de ejecución del ETL, una vez finalizado son borrados.

- Requieren de **menos espacio** de almacenamiento
- Son **más simples** de mantener
- **No** permite hacer **auditorías** posteriores a la ejecución

CePETel

Data Warehouse

Staging Area - Tipos

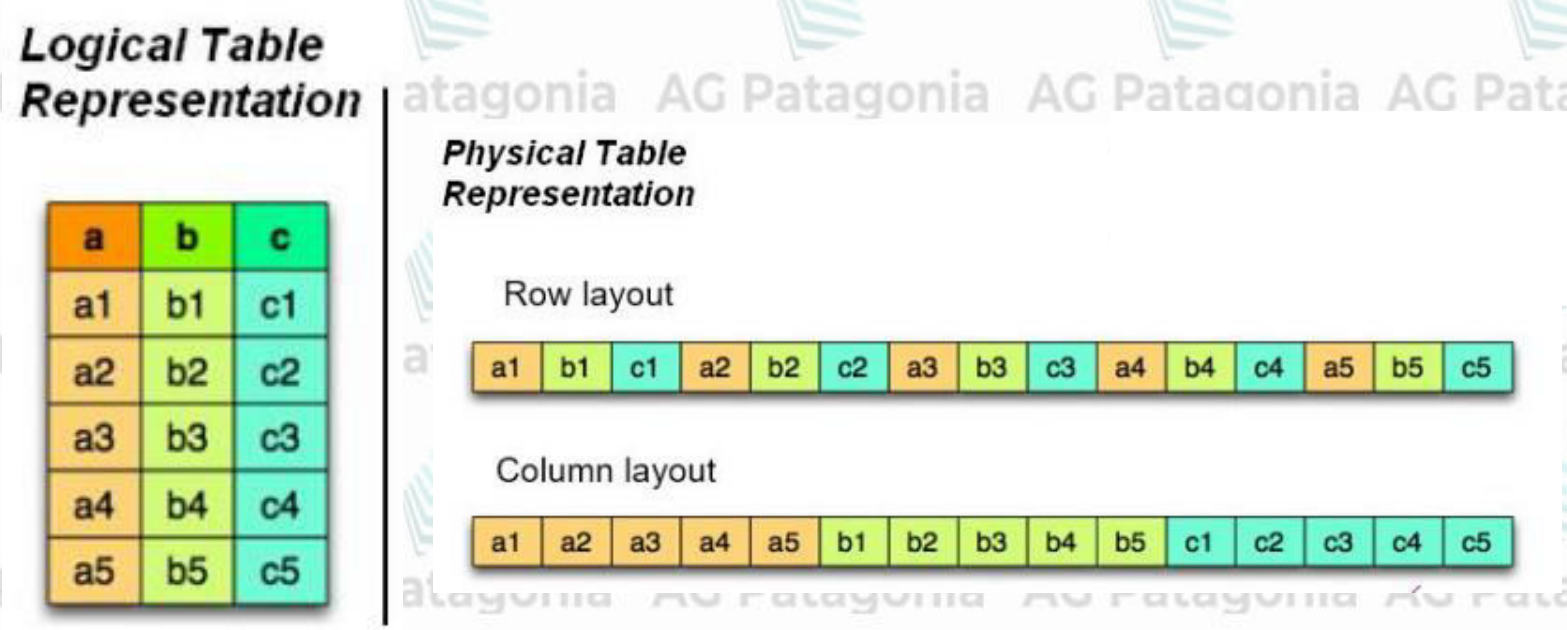
Persistente

Datos siguen guardándose en el área de stage una vez finalizada la ejecución del ETL

- Requieren **mayor espacio** de almacenamiento
- Son **más complejos** de mantener
- Permite realizar **auditorías** y **detección de errores**.
- Permite **mayor flexibilidad** en el **proceso** de **ETL**

Data Warehouse

Almacenamiento basado en filas vs columnas



Row Layout
- Más fácil para escribir o leer registros individuales.
- Se tiene a leer datos innecesarios.
- Solo se leen datos relevantes.
- Escritura es un proceso más costoso y complejo.

Column Layout
Solo se leen datos relevantes.
- Escritura es un proceso más costoso y complejo.

Recomendable para lecturas frecuentes, intensas y masivas (sobre grandes volúmenes de datos).

CePETel

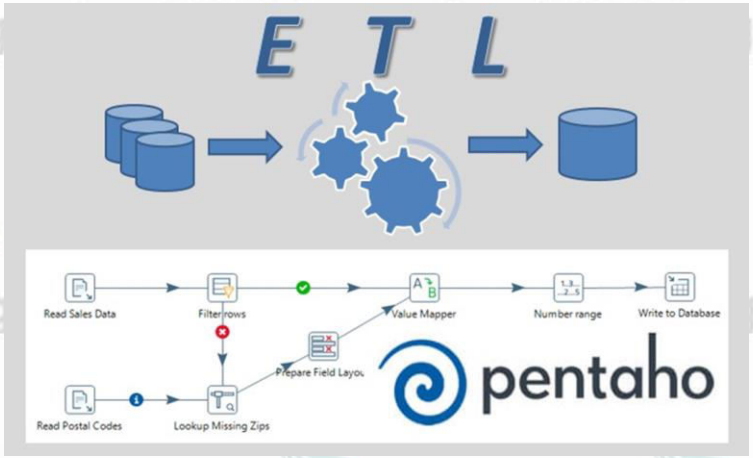
Data Warehouse

Sistemas mas usados de ETL

PENTAHO DATA INTEGRATION (PDI)

Líder Mundial en software de BI Open Source

Fundado en **Orlando en 2004**



+15.000 implementaciones de productos

+7 delegaciones por el mundo

+1.500 clientes comerciales actuales

2004 - 2009

En el año 2004, se fundó Pentaho con apenas 5 empleados. Más tarde, en el año 2006, se construyó la plataforma y las diferentes comunidades (como Kettle, Weka o Mondrian). Para 2009, Pentaho ya contaba con más de 50 empleados y había lanzado su edición de empresa.

A partir del 2010, Pentaho siguió creciendo: el equipo ejecutivo se amplió, aumentó el número de empleados y se comenzó a trabajar con tecnologías como Big Data Analytics, convirtiéndose esta en el principal dominio de Pentaho.

2010 - 2015

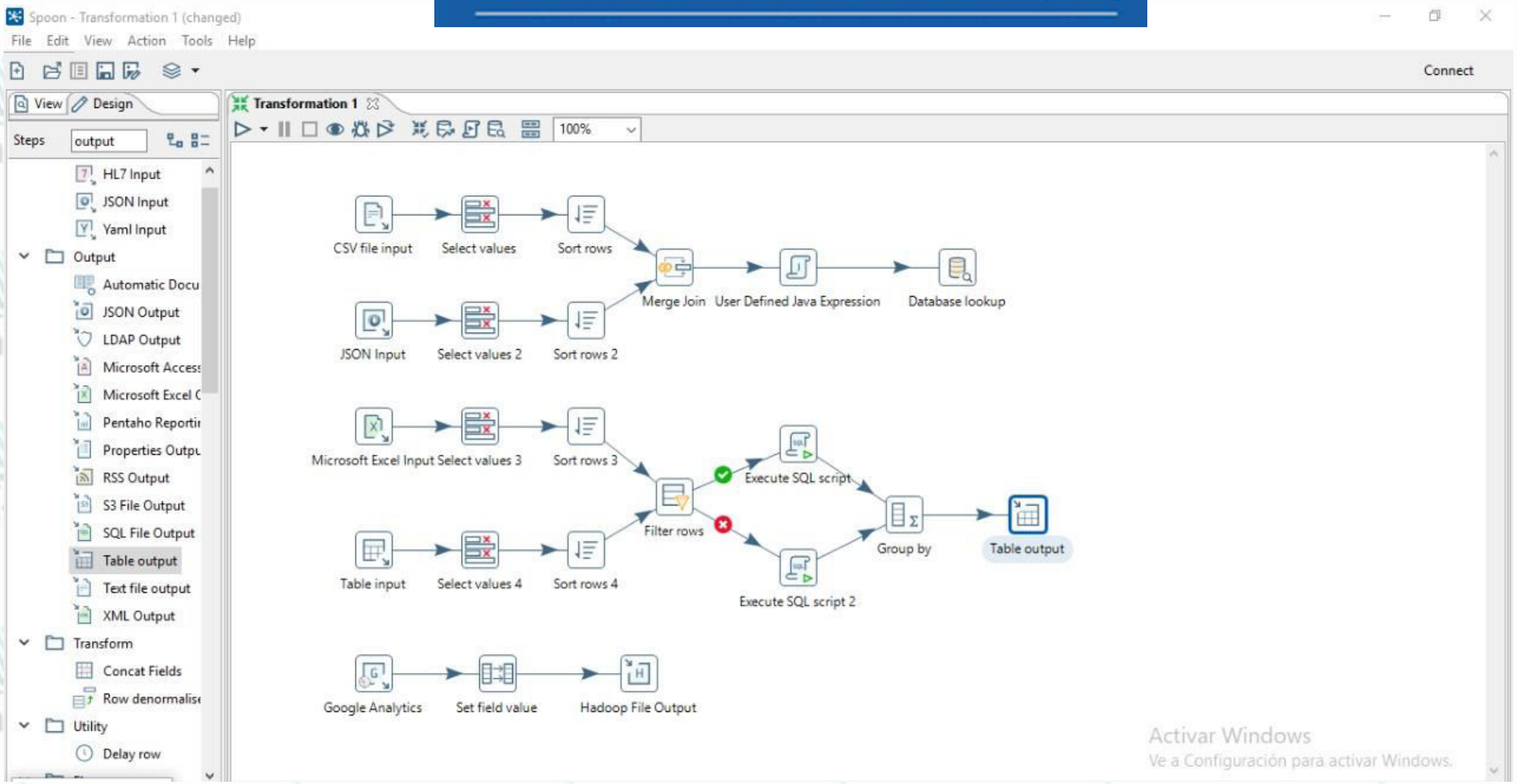
2015 - Actualidad

Desde 2015 hasta ahora, Pentaho fue adquirida por la gran y experimentada empresa japonesa Hitachi. A partir de aquí, Pentaho ha seguido creciendo, se han creado varios productos (Pentaho BA, Pentaho Data Integration, Pentaho Report Designer, CTools...) y se ha convertido en el software BI Opensource más utilizado a nivel mundial.

CePETel

Data Warehouse

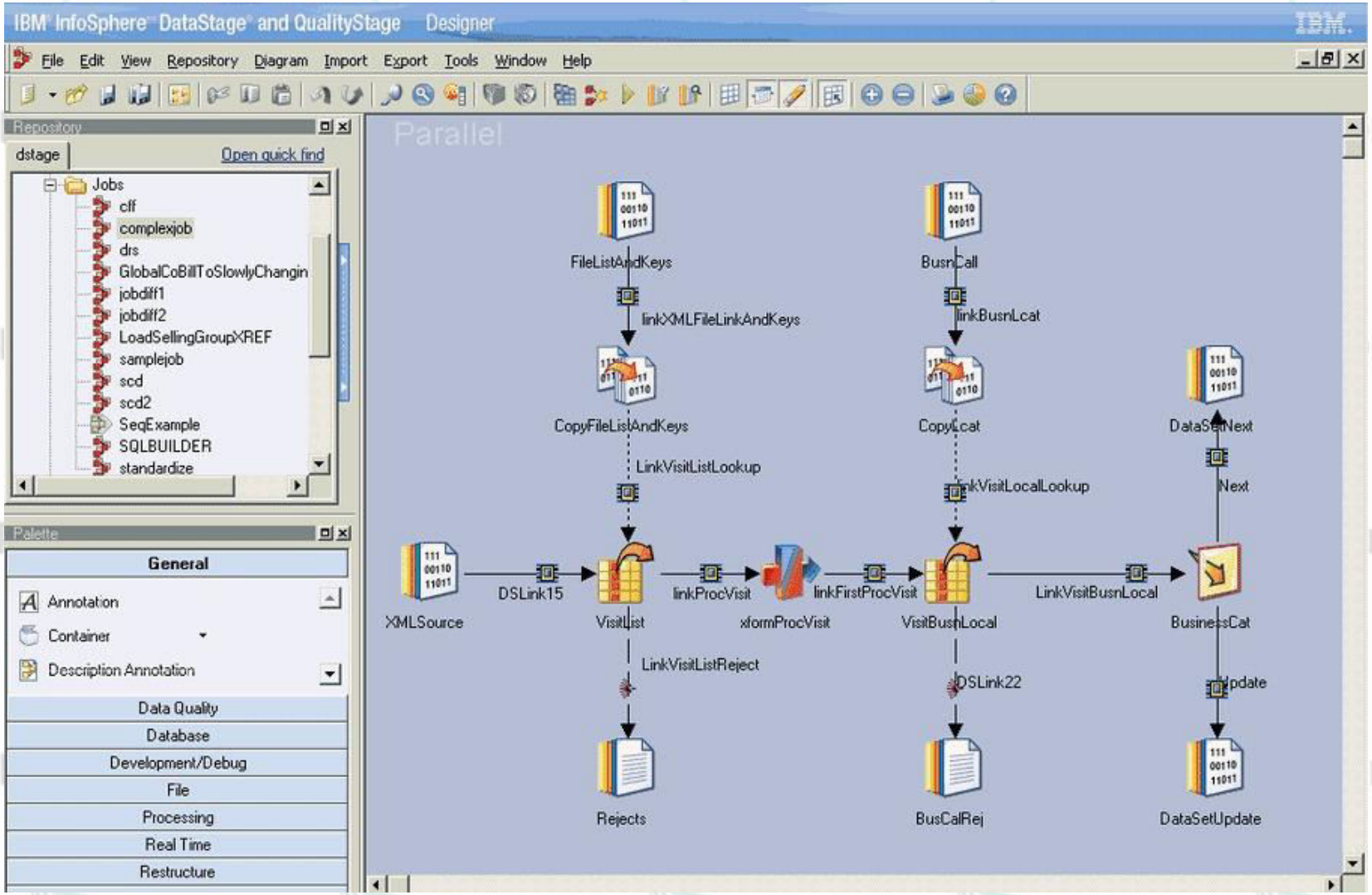
PENTAHO DATA INTEGRATION (PDI)



CePETel

Data Warehouse

Sistemas mas usados de ETL

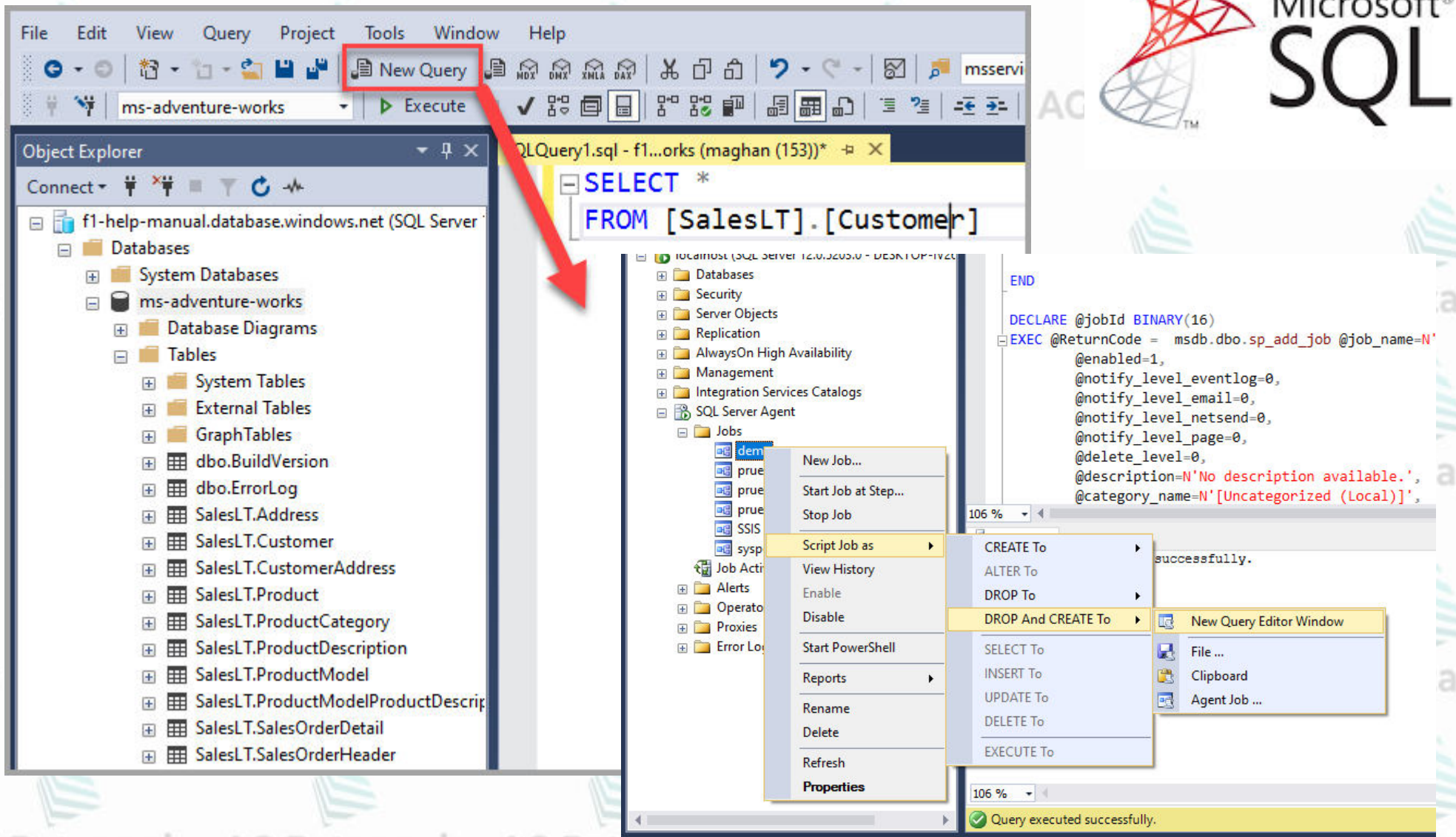


IBM DataStage®

CePETel

Data Warehouse

Sistemas mas usados de ETL



Microsoft® SQL Server®

CePETel

TP N° 3: Data Warehouse

- Investigar cuales son las características requeridas de un Data Warehouse (enumere al menos 4).

- ¿Qué es ETL? Y ¿Para qué sirve?

Armar una presentación en 1 hoja en base a lo solicitado para exponer en clase. Éxitos!!

CePETel

TP N° 4: Data Warehouse

¿Qué diferencias existen entre los dos principales enfoques de modelado de Data Warehouse (Snowflake vs. Star Schema)?.

Enumere al menos 2 ventajas y desventajas de cada uno.

Armar una presentación en 1 hoja en base a lo solicitado para exponer en clase. Éxitos!!

CePETel

TP N° 5: Data Warehouse

- Armar un modelo de ingesta de datos a un Data Warehouse. Incluir fuentes de datos internas y externas. Marcar como mínimo 5 diferencias entre procesos de ETL y ELT.

Armar una presentación en 1 hoja en base a lo solicitado para exponer en clase. Éxitos!!

CePETel

CePETel
Sindicato de los Profesionales
de las Telecomunicaciones

SECRETARÍA TÉCNICA

IPEI
Instituto Profesional de
Estudios e Investigación

BIG DATA & ANALYTICS II

TEMARIO
Módulo 1: Modelado Dimensional
Módulo 2: Data Warehouse
Módulo 3: Data Lake

Disertantes: Lic. Maria Trinidad Aquino – Ing. Raúl Alejandro Grassi

BIG DATA & ANALYTICS II

Módulo 3: Data Lake

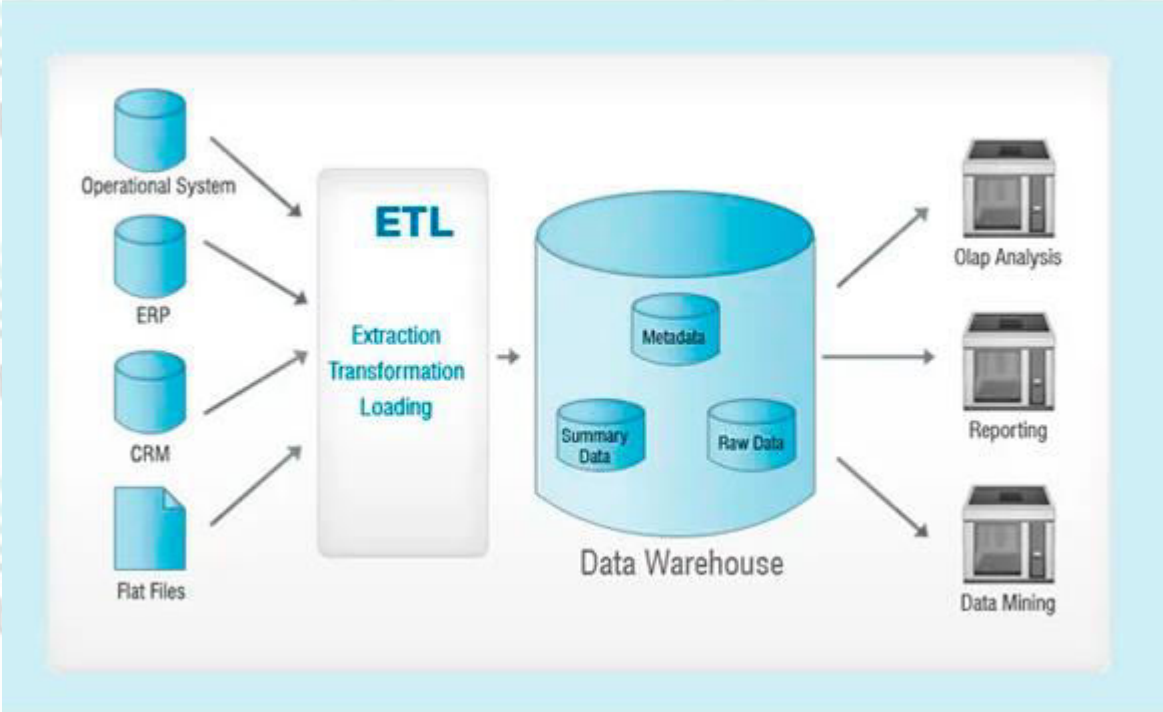
Data Warehouse

Recordando!

Es un sistema que agrega y combina información de diferentes fuentes en

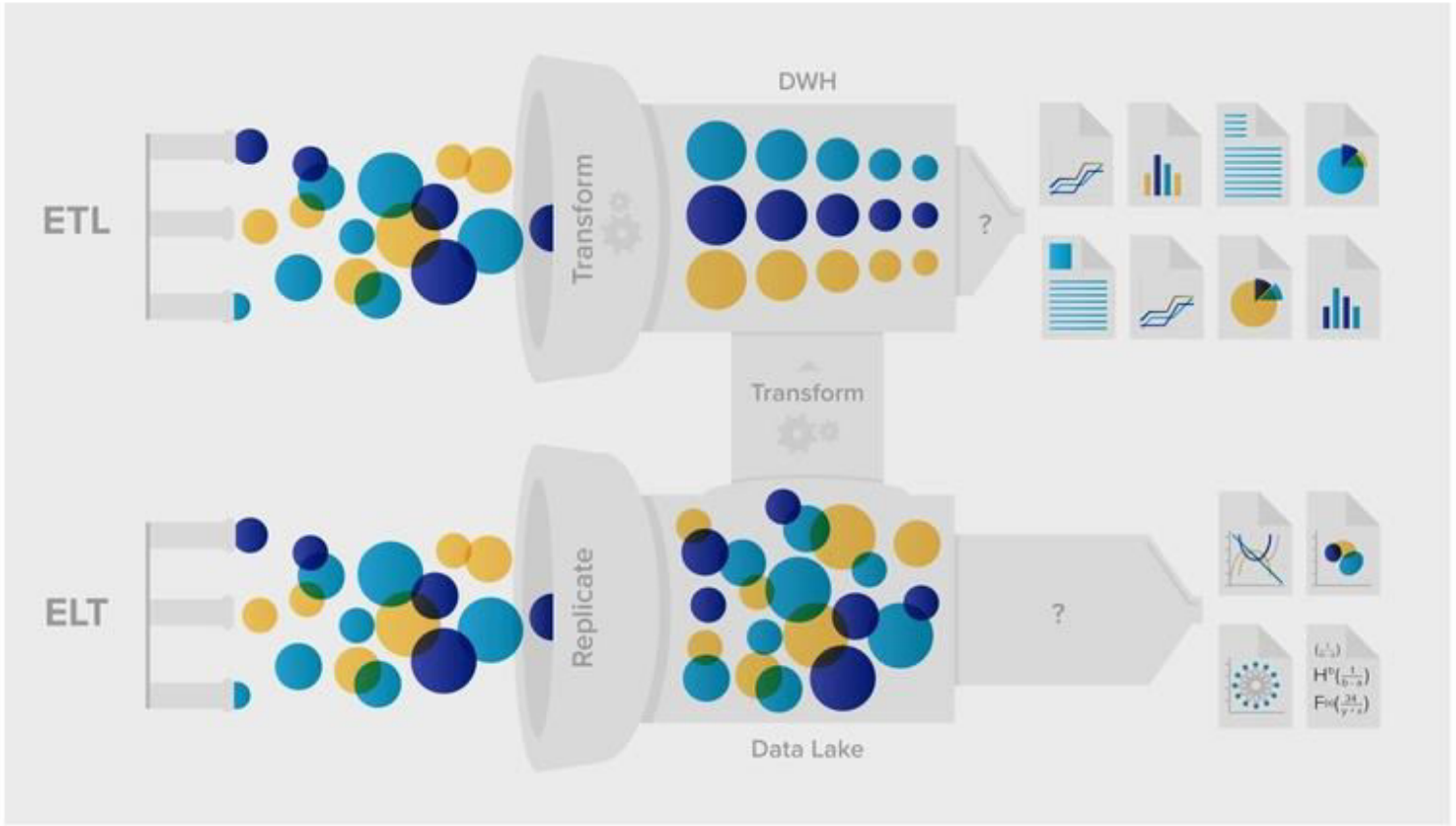
- almacén de datos único
- centralizado
- consistente

para respaldar el análisis empresarial, la minería de datos, inteligencia artificial (IA) y Machine Learning.



CePETel

Cambio de Paradigma BI: ETL a ELT



ETL

Extract Transform Load

vs.

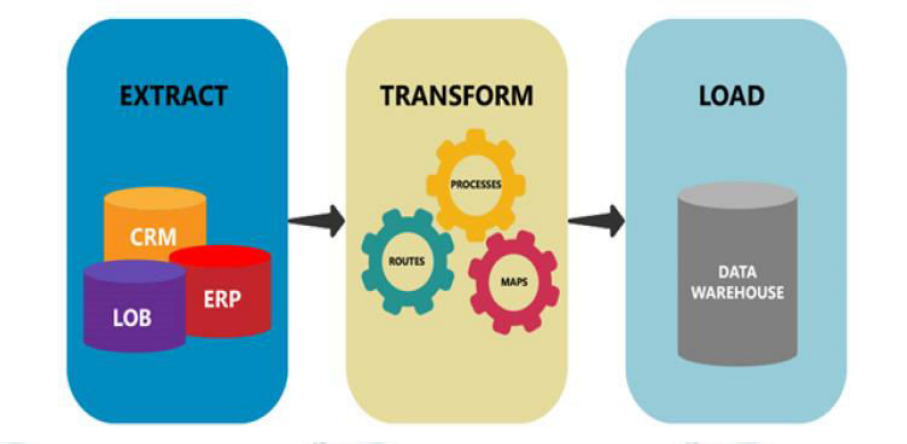
ELT

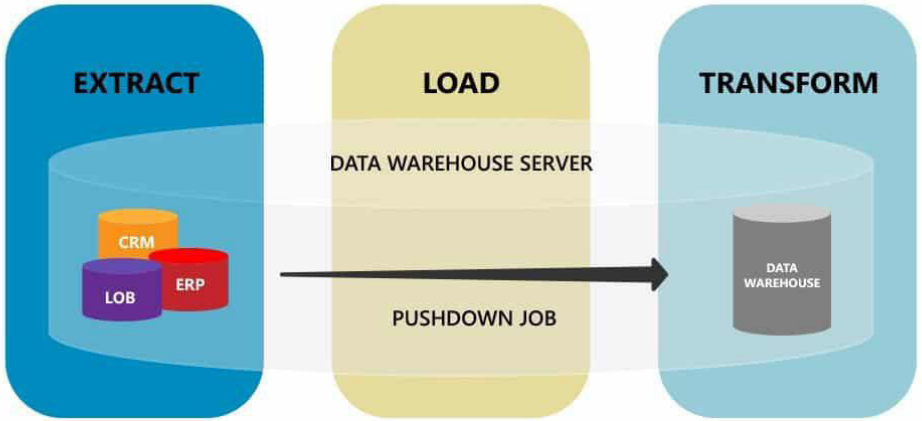
Extract Load Transform

CePETel

ELT vs ETL

ETL es un tipo de integración de datos que hace referencia a los tres pasos (extraer, transformar, cargar) que se utilizan para mezclar datos de múltiples fuentes. Se utiliza a menudo para construir un data warehouse.





Pushdown Optimization Mode

ELT (Extraer, cargar, transformar) es un método diferente de acercarse al flujo de datos, en el que los datos extraídos se cargan primero en el sistema de destino.
ELT generalmente se usa con bases de datos NOSQL como el clúster de Hadoop, un dispositivo de datos o una instalación en la nube.

CePETel

ELT vs ETL

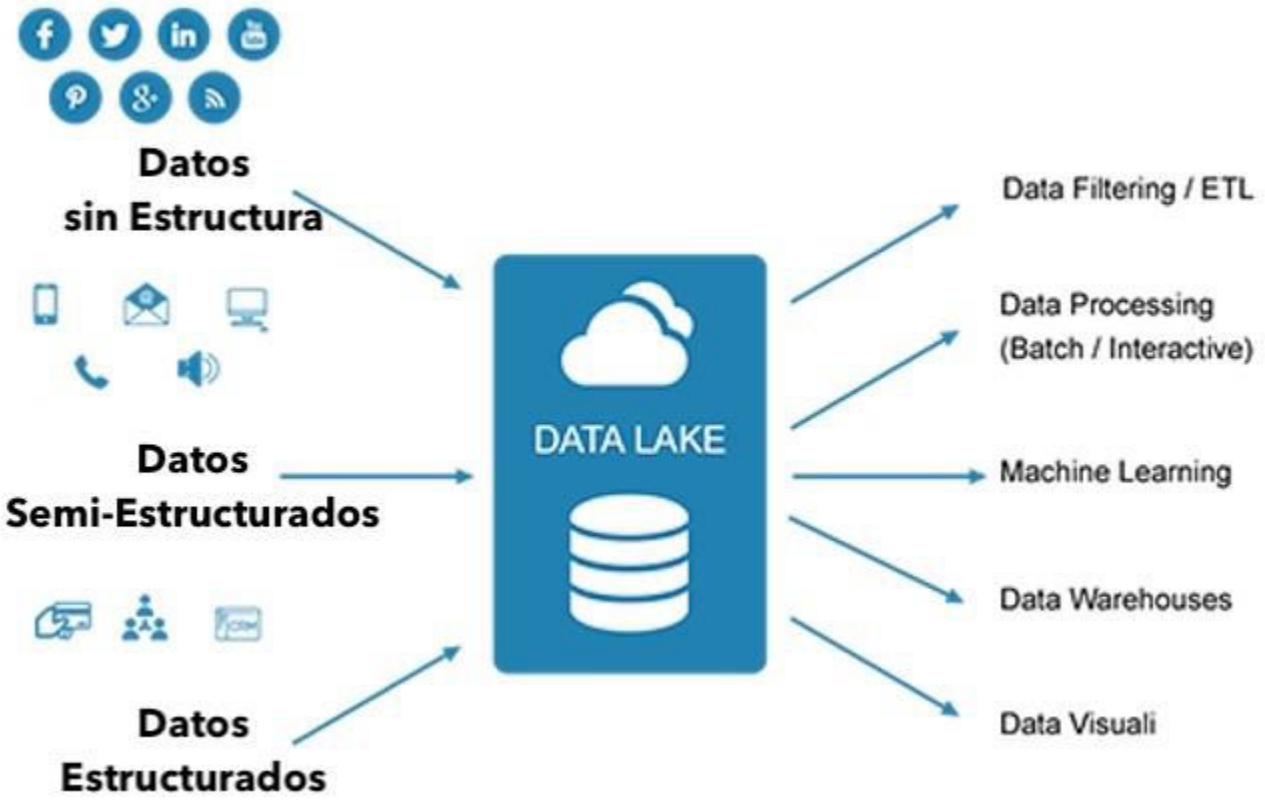
ETL	ELT
La transformación de los datos se realiza en un staging area.	La transformación de los datos se realiza en el almacén de datos de destino.
Precisa de un motor de transformación especializado e independiente.	El almacén de datos de destino debe tener capacidades de procesamiento para realizar transformaciones.
Ideal para volúmenes de datos pequeños o medios.	Ideal para grandes volúmenes de datos (Big Data).
Puntos fuertes: Facilita procesos de data quality, seguridad de datos y data compliance.	**Puntos fuertes:** Mayor flexibilidad y rapidez a la hora de cargar y transformar grandes volúmenes de datos no estructurados.

CePETel

Data Lake - Definición

[Video: ¿Qué es y para que sirve un Data Lake?]

Un **data lake** es un repositorio de almacenamiento que contienen una gran cantidad de datos en bruto y que se mantienen allí hasta que sea necesario. A diferencia de un **data** warehouse jerárquico que almacena datos en ficheros o carpetas, un **data lake** utiliza una arquitectura plana para almacenar los datos.



CePETel

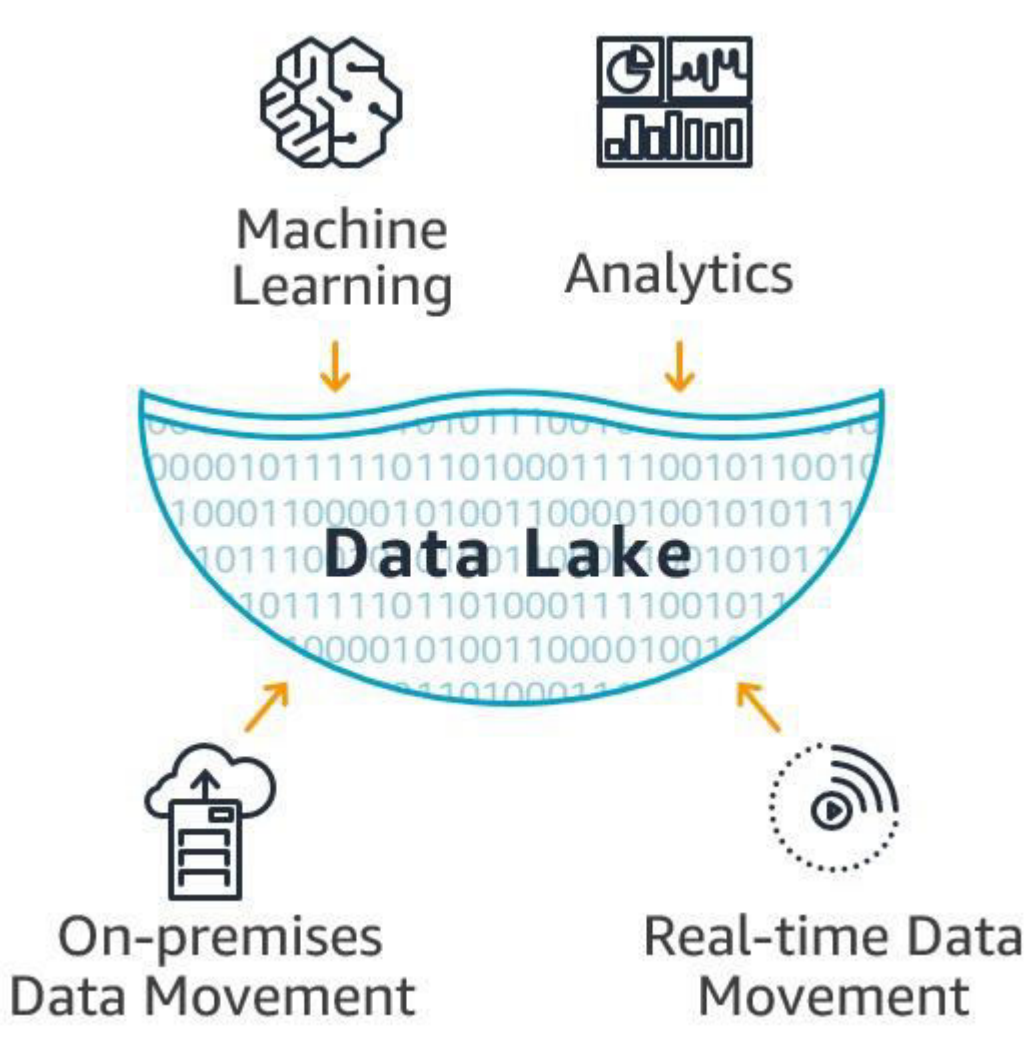
Data Lake - Definición

Un **data lake** es un **repositorio centralizado** que nos permite almacenar todos nuestros **datos estructurados y no estructurados** en cualquier escala, independientemente de su fuente o formato.

Se Implementa:

- De manera **On Premise** utilizando la Plataforma Hadoop.

- **En la nube** mediante por ej. Amazon S3 + Glue / Redshift / Amazon EMR ; Azure HD insight + Databricks



CePETel

Data Lake

VS

Data Warehouse

Data Warehouse vs. Data Lake

Video: DW vs. DL vs. DM

Un **Data Lake no es una nueva versión** 2.0 de un **Data Warehouse ni su reemplazo.**
Se pueden **complementar e integrar**, diseñando una **nueva arquitectura de datos.**

DATA WAREHOUSE	vs.	DATA LAKE
structured, processed	DATA	structured / semi-structured / unstructured, raw
schema-on-write	PROCESSING	schema-on-read
expensive for large data volumes	STORAGE	designed for low-cost storage
less agile, fixed configuration	AGILITY	highly agile, configure and reconfigure as needed
mature	SECURITY	maturing
business professionals	USERS	data scientists et. al.

CePETel

Data Warehouse vs. Data Lake

Diferencias ETL y ELT

Parámetros	ETL	ELT
Procesamiento	Los datos se transforman en el servidor de almacenamiento intermedio y luego se transfieren al Datawarehouse.	Los datos permanecen en el data lake.
Código de uso	Usado para: – Transformaciones de computación intensiva – Pequeña cantidad de datos	Cantidades grandes de datos
Transformación	Las transformaciones se realizan en el servidor ETL / área de ensayo.	Las transformaciones se realizan en el sistema de destino.
Tiempos de carga	Los datos se cargan primero en el almacenamiento intermedio y luego se mueven al sistema objetivo. Tiempo intensivo.	Los datos cargados en el sistema de destino solo una vez. Más rápido.
Tiempos de Transformación	El proceso ETL necesita esperar a que se complete la transformación. A medida que crece el tamaño de los datos, aumenta el tiempo de transformación.	En el proceso ELT, la velocidad nunca depende del tamaño de los datos.
Tiempos de mantenimineto	Necesita altos niveles de mantenimiento ya que necesita seleccionar datos para cargar y transformar.	Bajo mantenimiento ya que los datos están siempre disponibles.

CePETel

Data Warehouse vs. Data Lake

Parámetros	ETL	ELT
Complejidad de implementación	En una etapa temprana, es más fácil de implementar.	Para implementar el proceso de ELT, la organización debe tener un conocimiento profundo de las herramientas y los skills necesarios.
Soporte para Data warehouse	Modelo de ETL utilizado para datos locales, relacionales y estructurados.	Se utiliza en una infraestructura de cloud escalable que admite orígenes de datos estructurados y no estructurados.
Soporte Data lake	No soportado.	Permite usar un Data lake con datos no estructurados.
Complejidad	El proceso ETL carga solo los datos importantes, como se identificaron en el momento del diseño.	Este proceso implica el desarrollo desde la salida hacia atrás y la carga de solo datos relevantes.
Costes	Costes elevados para pequeñas y medianas empresas.	Bajos costes de entrada utilizando plataformas de Software as a Service.
Búsquedas	En el proceso de ETL, tanto los hechos como las dimensiones deben estar disponibles en el área de preparación.	Todos los datos estarán disponibles porque la extracción y la carga se producen en una sola acción.
Agregaciones	Aumento de la complejidad con la cantidad adicional de datos en el conjunto de datos.	El poder de la plataforma de destino puede procesar una cantidad significativa de datos rápidamente.

CePETel

Data Warehouse vs. Data Lake

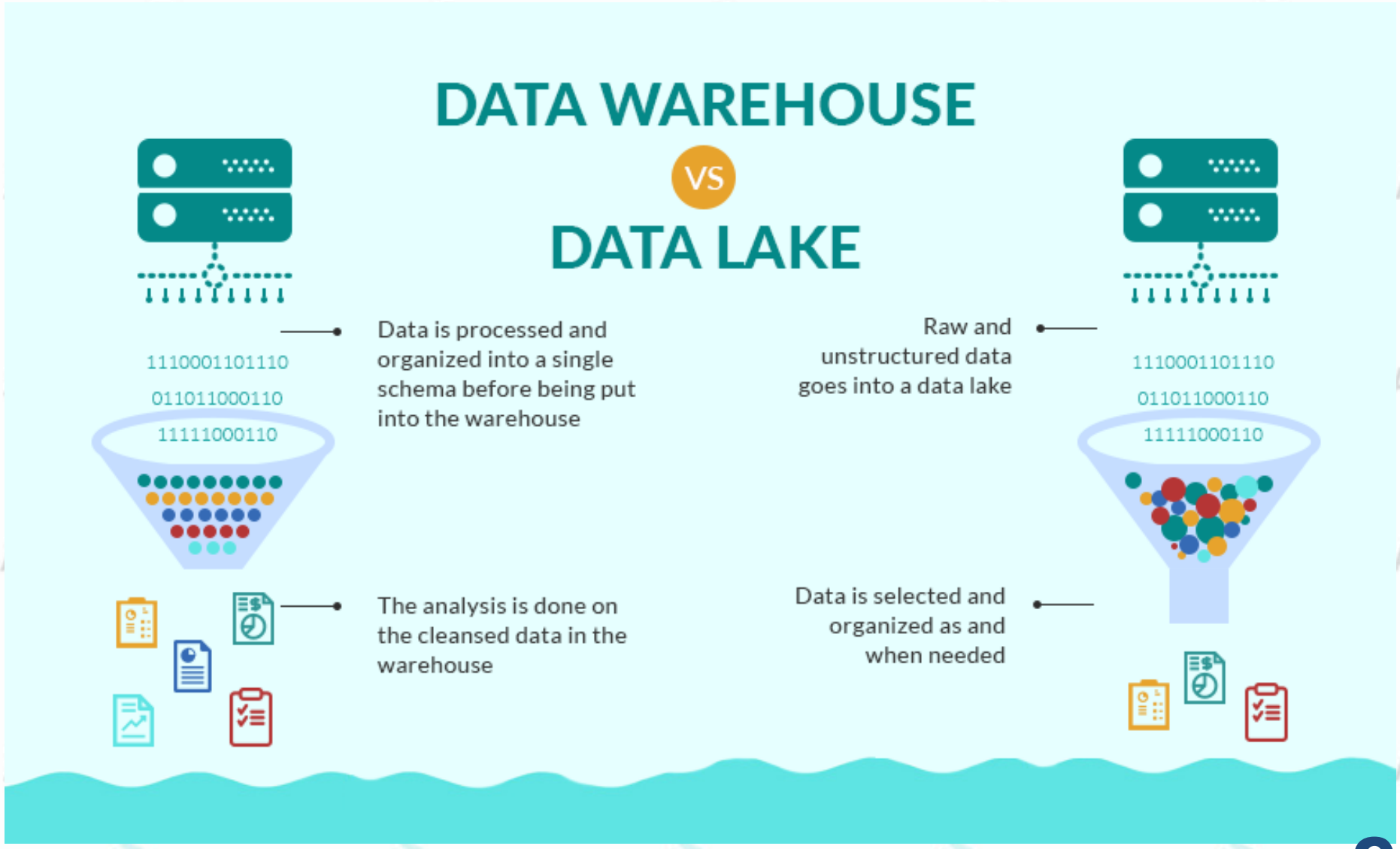
Parámetros	ETL	ELT
Cálculos	Sobrescribe la columna existente o necesidad de adjuntar el conjunto de datos y empujar a la plataforma de destino.	Permite agregar fácilmente la columna calculada a la tabla existente.
Madurez	El proceso se utiliza desde hace más de dos décadas. Está bien documentado y las mejores prácticas fácilmente disponibles.	Concepto relativamente nuevo y complejo de implementar.
Hardware	La mayoría de las herramientas tienen requisitos de hardware únicos que son caros.	En formato SaaS el coste de hardware no es un factor crucial.
Soporte para datos no estructurados	En su mayoría soporta datos relacionales	Soporte para datos no estructurados fácilmente disponibles.

Resumen

La conclusión es que el lago de datos tiene un potencial casi ilimitado, pero requiere unos conocimientos elevados para llevar a cabo la serie de transformaciones antes de lograr tener la suficiente calidad como para sacar provecho a los datos almacenados. Un datawarehouse, por el contrario, requiere una inversión significativa por adelantado, pero a cambio ofrece la capacidad de analizar todo fácilmente, y las habilidades que se requieren para consultarlo (generalmente SQL) suelen ser más fáciles de encontrar entre el equipo de analistas.

CePETel

DW vs DL

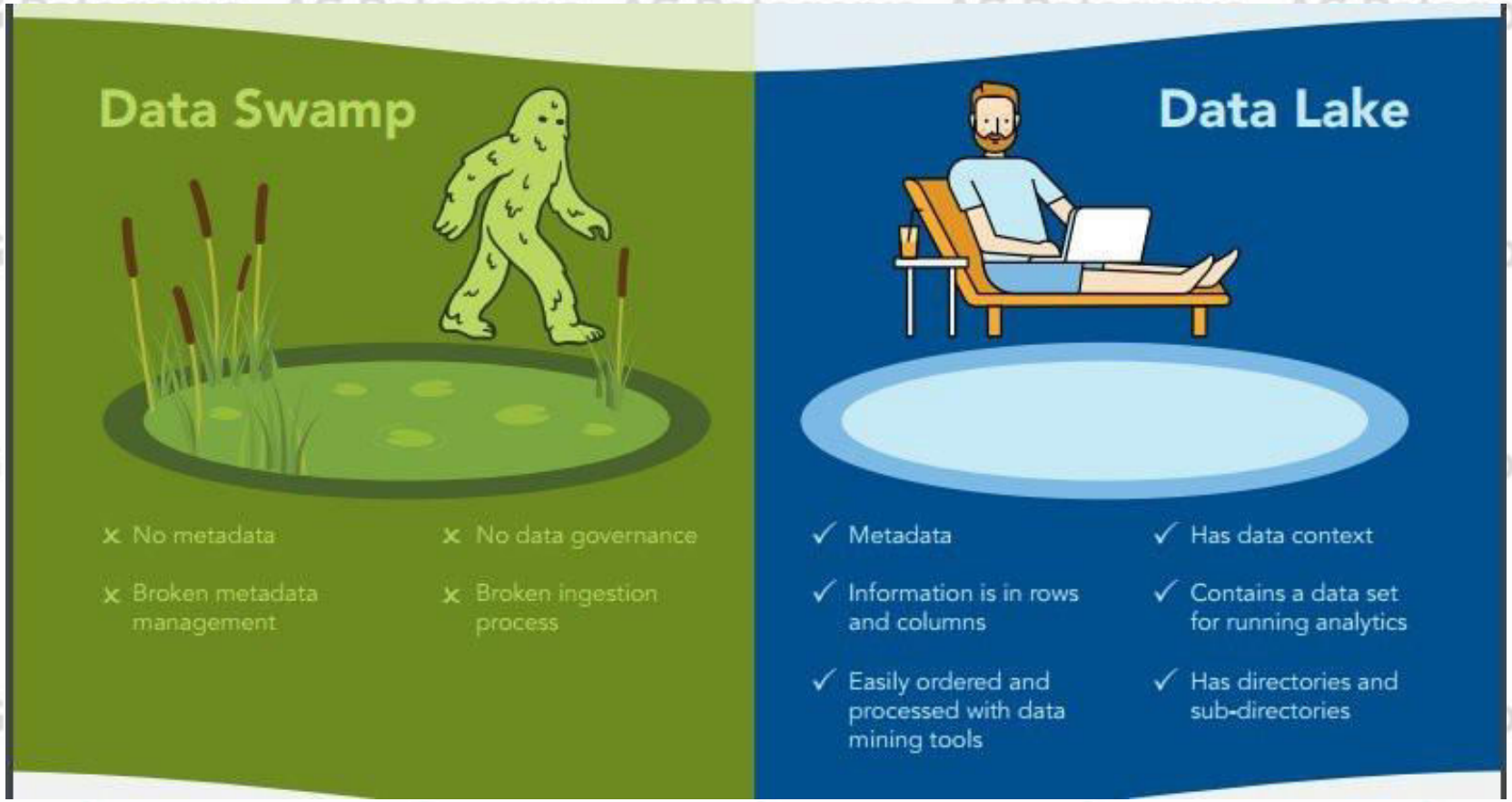


CePETel

Data Lake – Desafíos a la hora de implementarlo

Importancia de su ordenamiento

El **principal desafío** es asegurarse de que el conjunto de datos se pueda ver tal como es, en lugar de esconderse debajo de una superficie aparentemente correcta.



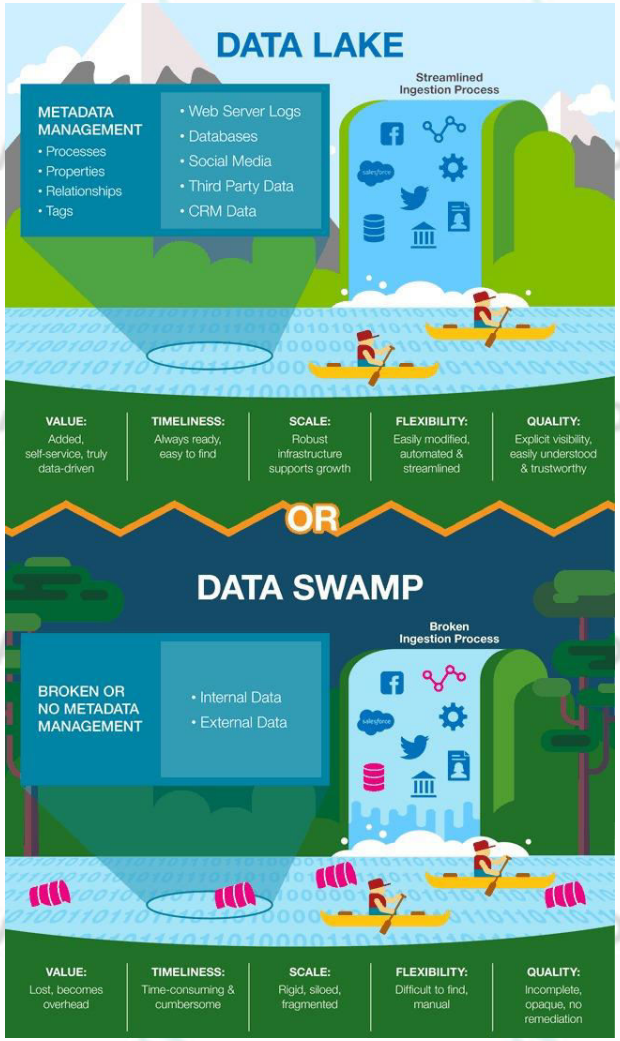
CePETel

Data Lake – Desafíos a la hora de implementarlo

Importancia de su ordenamiento

RIESGOS

- **Entorno aislado**, donde la uniformidad de la infraestructura no genera ningún intercambio de información.
- **Proliferan copias** debido al bajo costo de almacenamiento adicional.
- Sin suficiente información, es **difícil distinguir los datos** en el lago.
- Todo se ve igual y **no se puede distinguir entre los buenos y los malos.**



Se convierte en un pantano.

CePETel

Data Lake – Desafíos a la hora de implementarlo

Importancia de su ordenamiento

ORDEN:

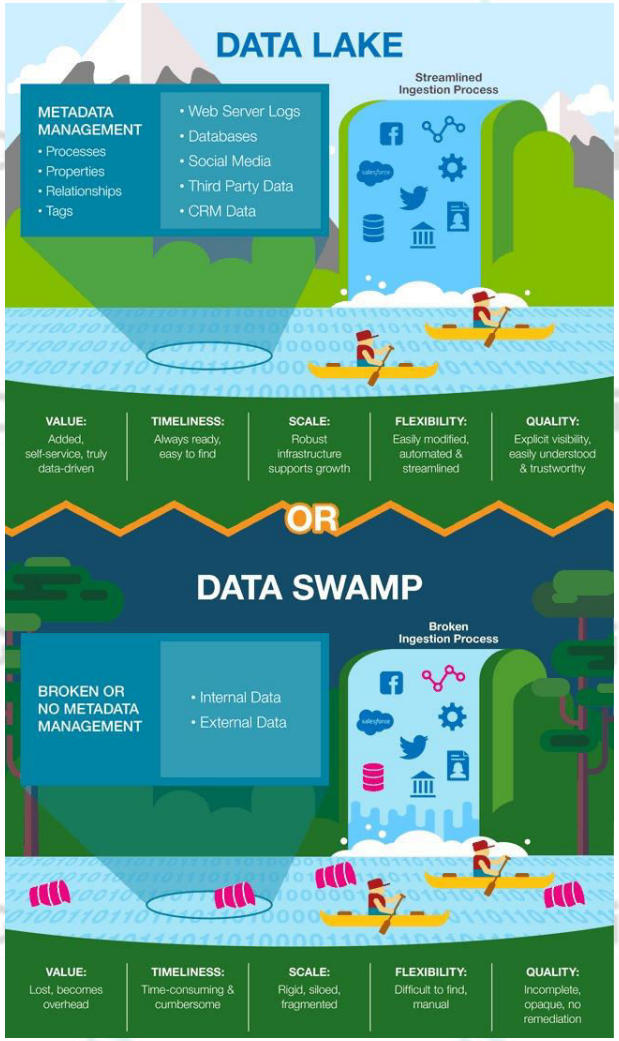
- Garantizar que tendremos en stock los datos que me van a solicitar mis clientes.
- Y que disponemos de espacio suficiente en los almacenes para cargar la nueva información.

INVENTARIO:

- Conocer el linaje del dato.
- “Glosario de términos de negocio”.

SEGURIDAD:

- Autenticar y autorizar acceso a los usuarios y grupos.
- Autorizar acceso a Información clasificada para diferentes usuarios y/o grupos.



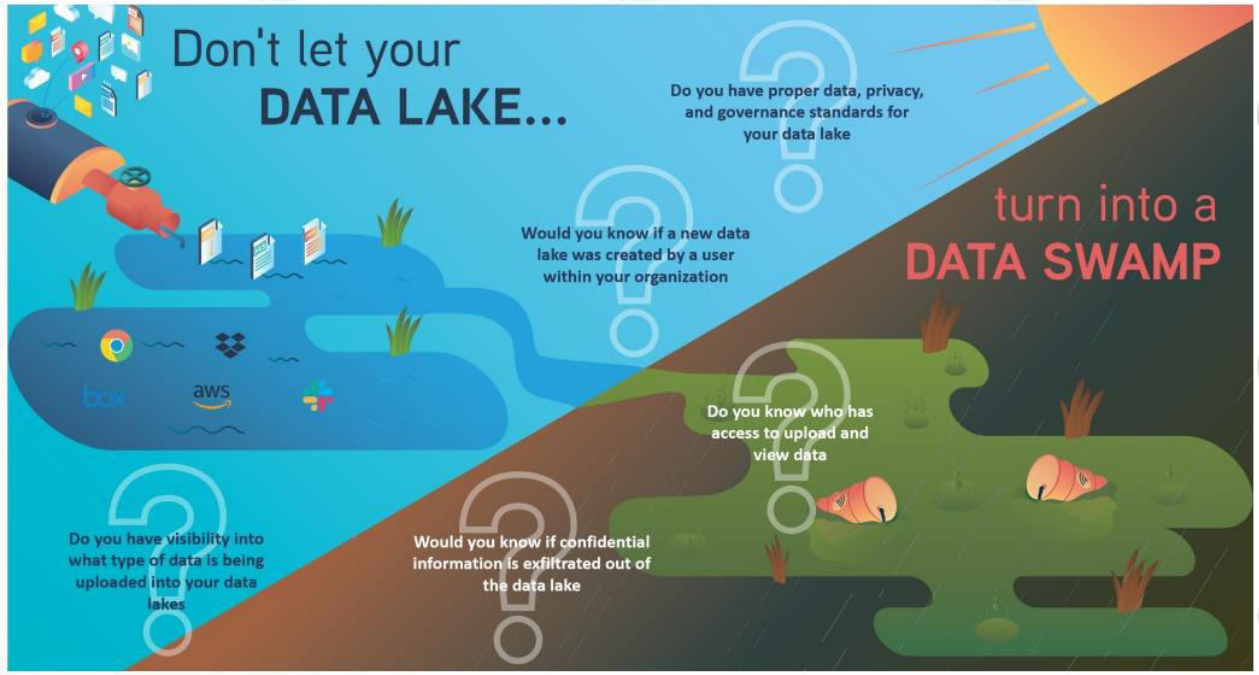
CePETel

Data Lake – Desafíos a la hora de implementarlo

Importancia de su ordenamiento

Nuevos Roles

- Responsable del Gobierno del Dato
- Data Manager
- Data Engineer
- Data Scientist



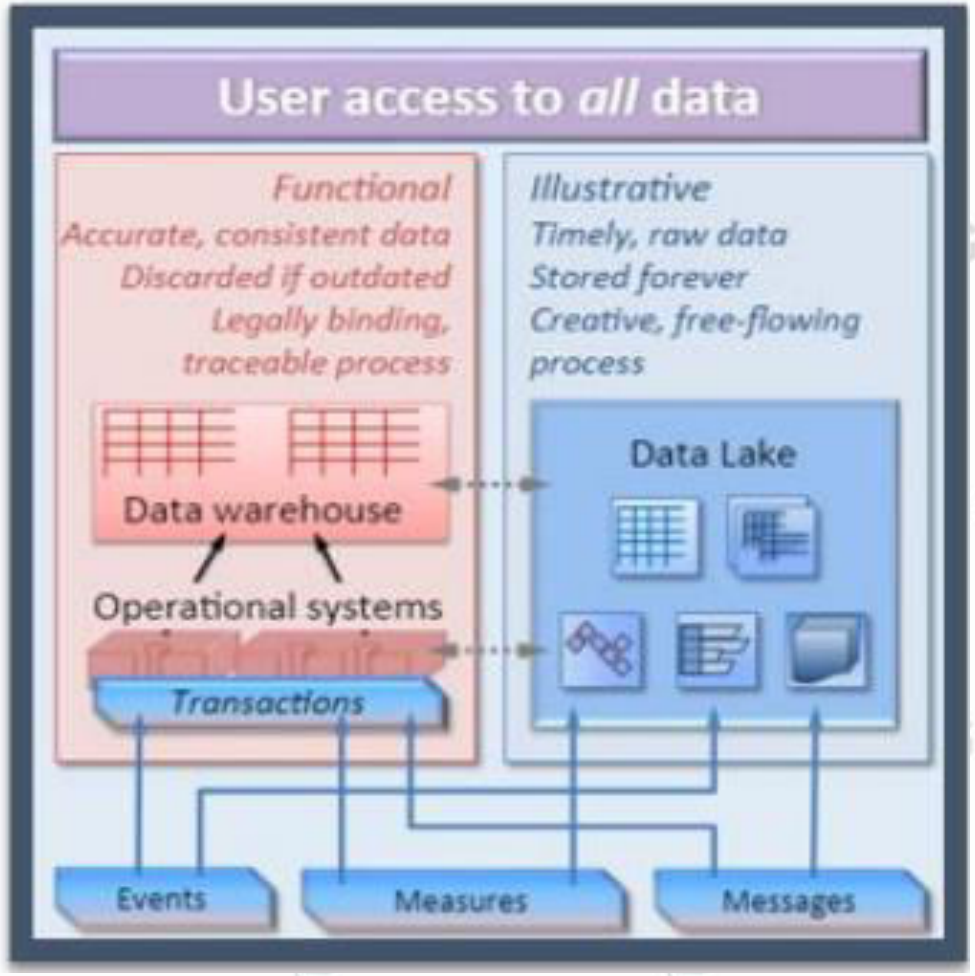
Comprender que la aplicación del Big Data Governance no debe ser responsabilidad de un departamento concreto de la organización, sino que debe ser una responsabilidad compartida de manera transversal; y de manera continua, iterativa y en permanente revisión y mejora.

CePETel

Data Warehouse y Data Lake

SIRVEN PARA DIFERENTES PROPÓSITOS

- Data Warehouse - Ejecución y administración del Negocio Hoy
- Data lake – Predicciones e Influenciar el futuro
- Ambos requieren Optimización
- Data Warehouse: exactitud y consistencia
- Date Lake: puntualidad, volumen y datos crudos
- Enlaces Bi-direccionales entre entornos



DW + DL: Analytics

CePETel

Data Warehouse y Data Lake

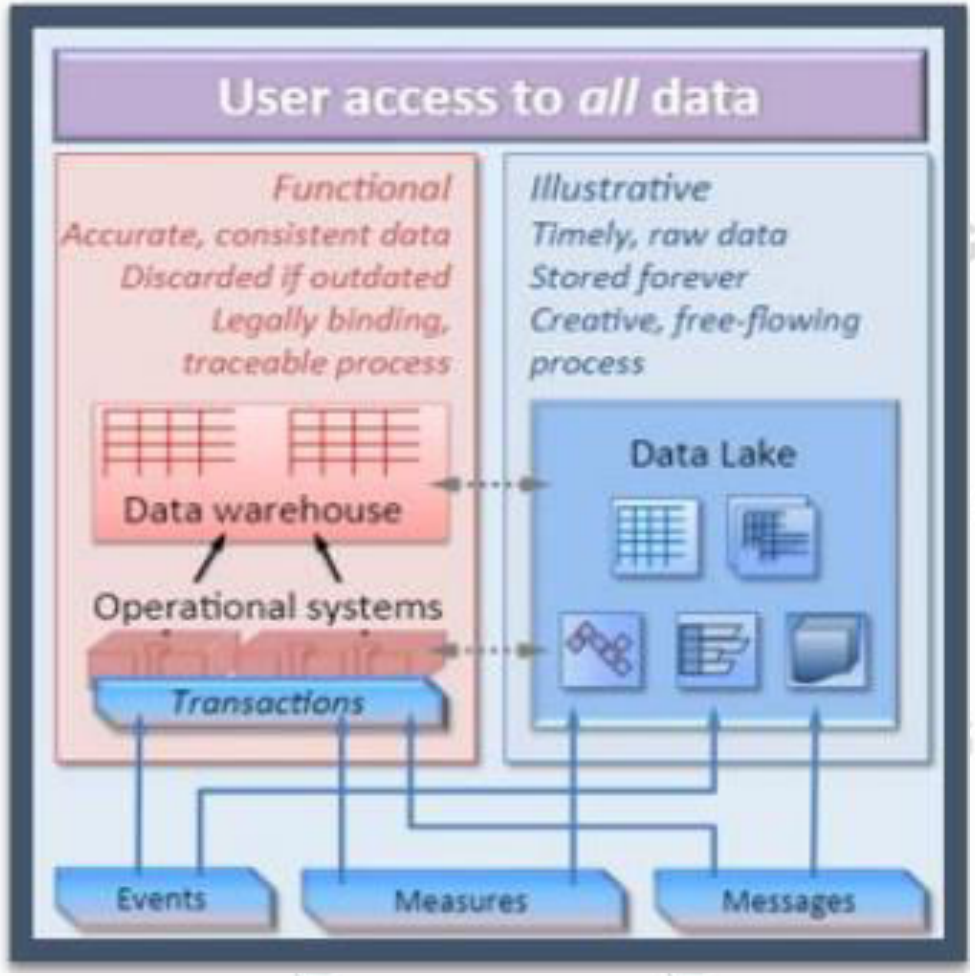
PREPARAR Y ENRIQUECER

Desafíos:

- ETL al DW son complejos y costosos
- Servidor ETL propietario con alto costo de licencia
- DW Server con impacto en tareas analíticas

Solución:

- Ingestar datos al DL
- Prepararlos y enriquecer los datos en el DL
- Transferirlos al DW
- Se reduce el costo de procesamiento
- Se reduce impacto en el procesamiento del DW



CePETel

Data Warehouse y Data Lake

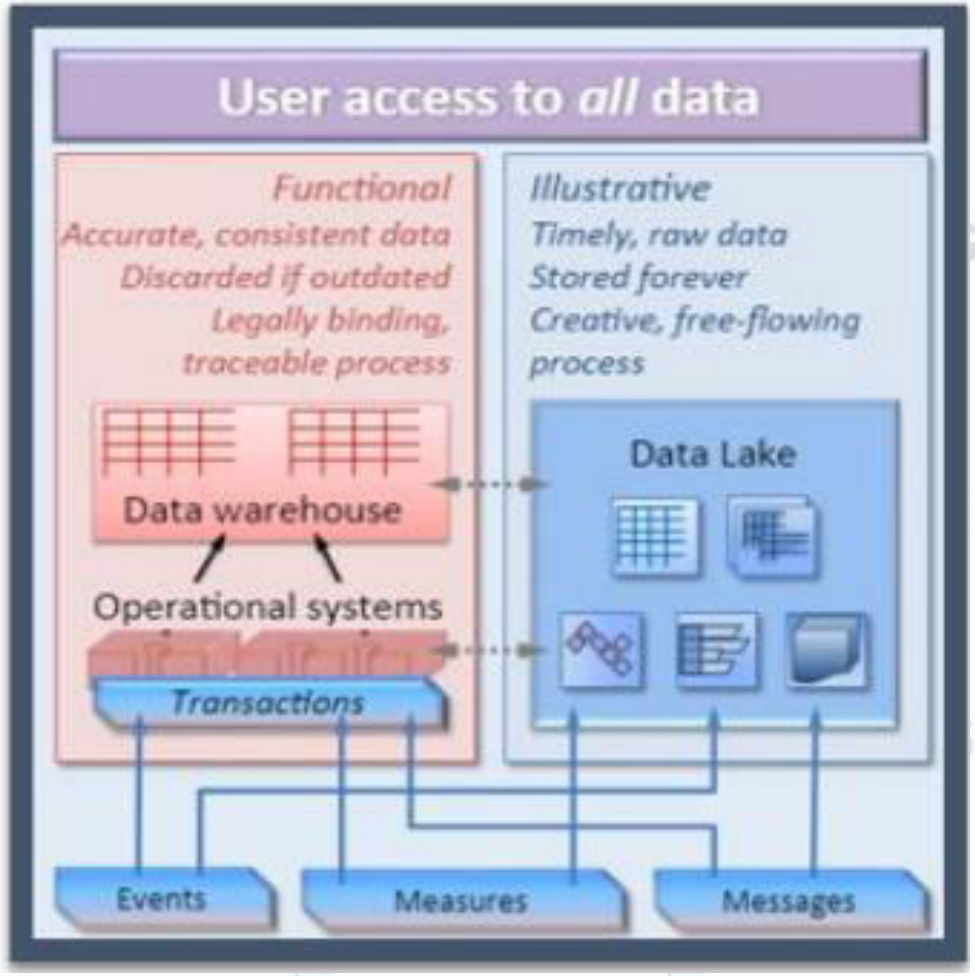
ALMACENAR

Desafíos:

- Almacenar historia en el DW es muy costoso
- Archivar en medios magnéticos es ineficiente al momento de necesitarla

Solución:

- Guardar historia en el Data Lake
- Hadoop – bajo costo
- Rápido acceso a la información cuando se requiera
- Mismas herramientas de consulta que el DW (SQL-Based)



CePETel

Recomendaciones para crear un Data Lake

Ingesta rápida y sin problemas

- No se debe crear cuellos de botella en el desarrollo de los pipelines de ingesta de datos.
- Se debe permitir que cualquier tipo de datos se cargue sin problemas de manera consistente.
- Automatizar la ingesta de datos a Hadoop deben ser rápidas, automatizadas y administrar las estructuras de destino.
- Se debería poder cargar una nueva iniciativa de Data Lake en pocos días.
- Manejar diferentes alternativas de Ingesta, Real Time, Microbatch, Batch dependiendo de la necesidad.

Flexibilidad y descubrimiento

- Una solución de Data Lake no debe encajar 100% en un patrón o comportamiento tecnológico determinado.
- El Data Lake debería permitir implementar políticas de refinamiento de datos flexibles, descubrimiento automático de datos y proporcionar un entorno de desarrollo ágil.

CePETel

Recomendaciones para crear un Data Lake

Transformación rápida de datos

- No se debe tener que sacar los datos del Data Lake para prepararse y transformarse.
- El DL debe tener la capacidad de implementar rápidamente transformaciones listas para usar.
- Debe poder realizar esas transformaciones en el entorno nativo.

Visión del Linaje de la Metadata

- Se tienen que poder implementar Herramientas de Data Governance.
- Se tiene que poder avanzar y retroceder sobre el linaje de cada Entidad y/o Atributo.
- El análisis de impacto incorporado debe estar disponible para administrar los cambios más rápidamente.

Integración con Otras Plataformas

- El DL no debe ser un repositorio estanco.
- El DL tiene que estar integrado tanto con nuestras Plataformas OLTP tanto como con nuestro DW.

CePETel

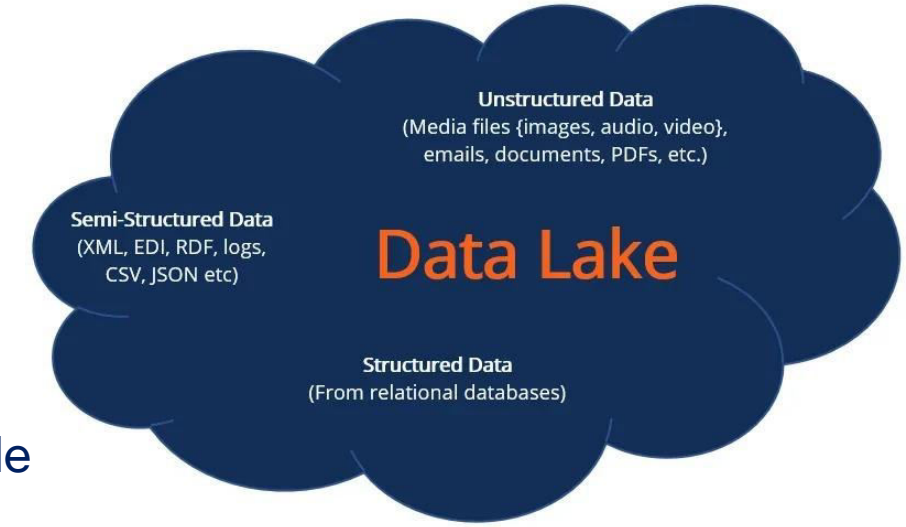
Recomendaciones para crear un Data Lake

Auditoría, balanceo y control

- Se deben poder obtener estadísticas precisas sobre los procesos realizados.
- Se deben generar datos de control para obtener una mejor comprensión de los procesos.

Visibilidad de las operaciones

- La solución de Data Lake debe poder proporcionar funciones de supervisión y depuración de operaciones en tiempo real.
- Debe contar con alertas en tiempo real sobre las nuevas llegadas de datos.



Lo más importante es que el Data Lake debe poder **utilizar las habilidades existentes en nuestros recursos** sin la necesidad de aprender nuevas tecnologías en rápida evolución.

Para **extraer el máximo valor de sus datos**, se necesita poder **adaptarse rápidamente e integrar sus datos sin problemas.**

CePETel

Recomendaciones para crear un Data Lake

Contar con un Plan de Crecimiento a Mediano Plazo (3 a 5 años)

- Tomar en cuenta cuánto planea escalar la empresa y cómo puede configurar su entorno en el futuro es importante.
- El hecho de que estar pensando en el futuro mientras se está en las etapas iniciales del proyecto nos ayudará en el futuro.

Contar con una Estrategia de Disaster Recovery

- Evaluar e implementar estrategias near real time.

CePETel

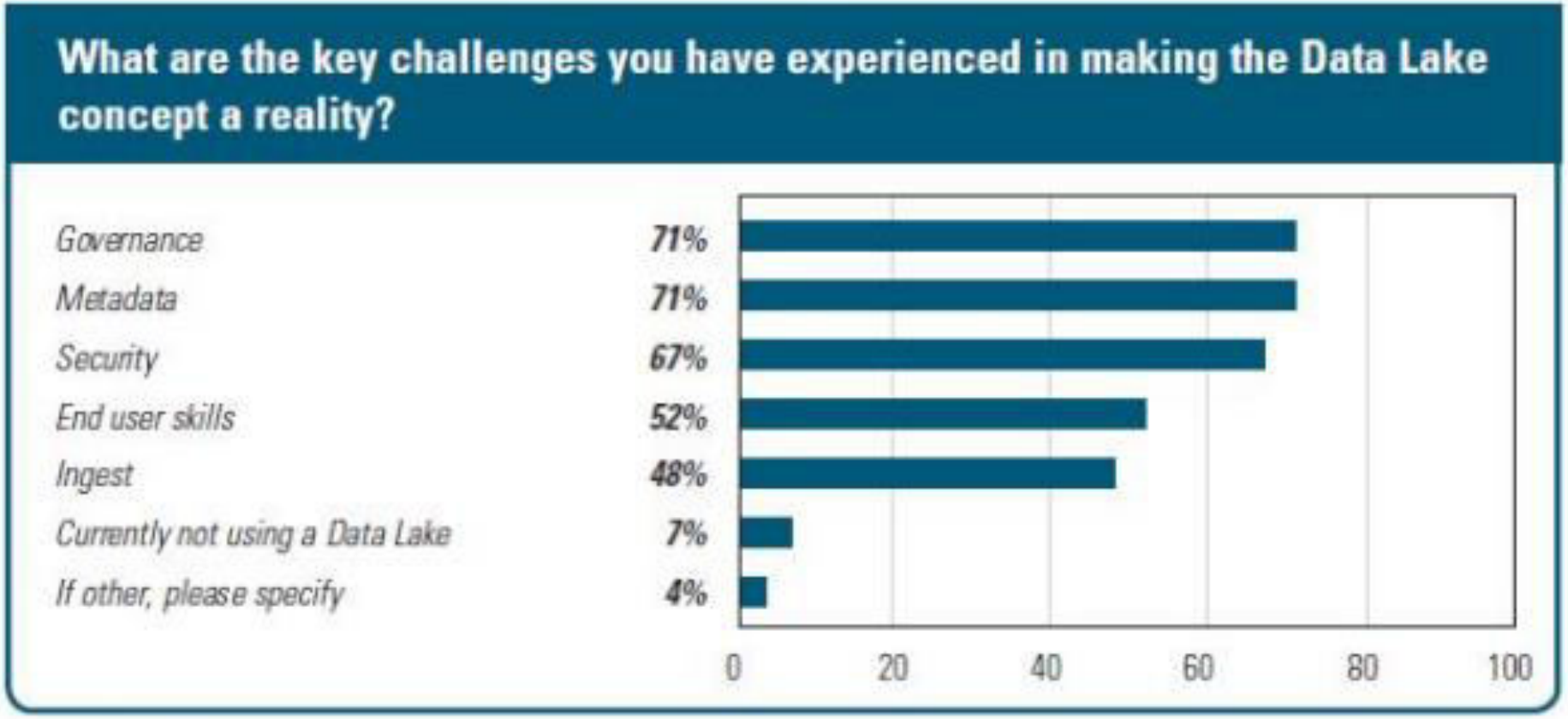
Recomendaciones para crear un Data Lake

Claves

- Contar con Datos crudos y datos modelados y refinados.
- Autenticación Centralizada con un Single Sign On.
- Autorización al acceso a datos clasificados.
- Data Governance.
- Auditoría y control.
- Integración con otras Plataformas.
- Contar con un Plan a Mediano Plazo (3-5 años).
- Tener en cuenta una estrategia para Recuperación Ante Desastres

CePETel

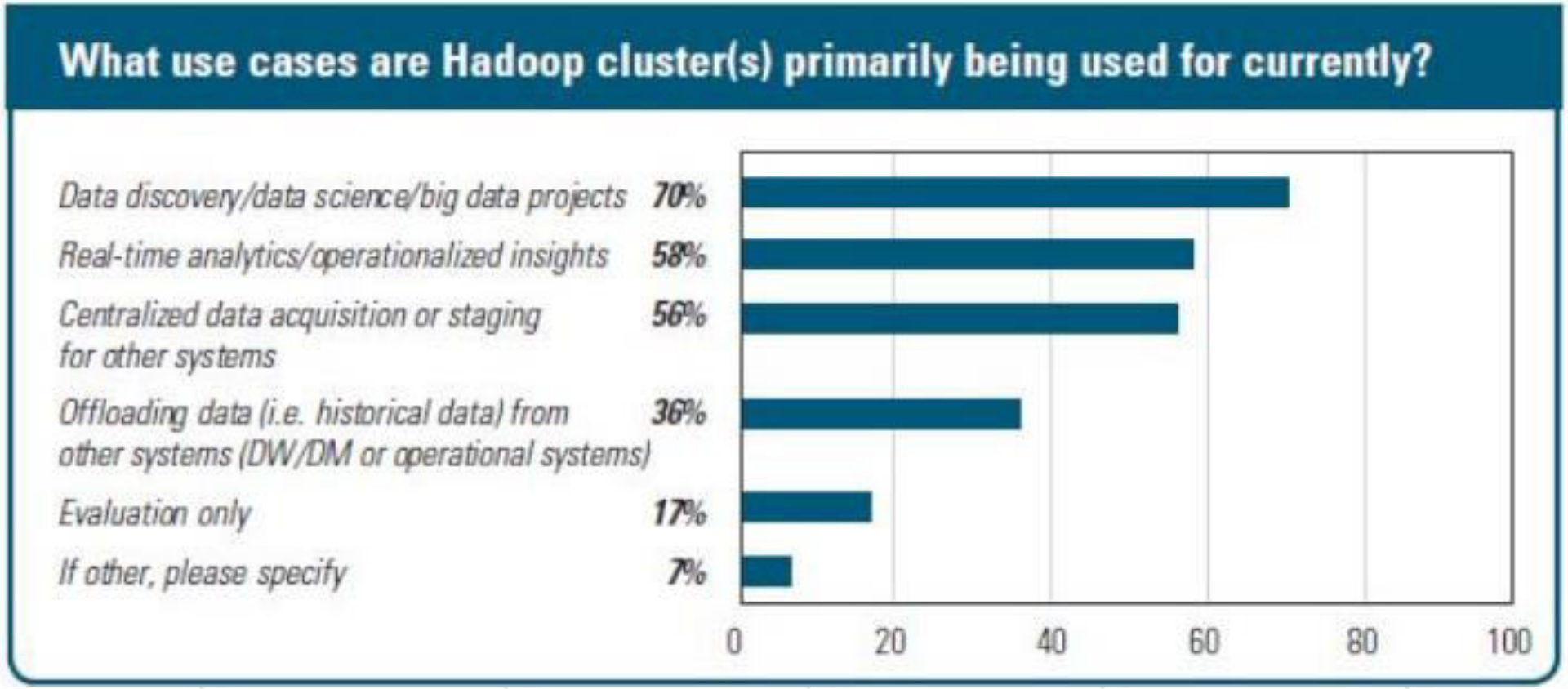
Recomendaciones para crear un Data Lake



Fuente: Radiant Advisors.

CePETel

Principales Usos de Hadoop



CePETel

Recuperación ante Desastres

Disaster Recovery

- A veces olvidamos que un Data Lake puede ser violado o que la tecnología puede fallar de alguna manera u otra.
- En caso de una emergencia, atentado, o desastre natural, se debe tener un plan de respaldo para sus datos.
- Ahora bien, algunas preguntas a realizarnos:
  - ¿Es posible o viable realizar un backup diario de un cluster de cientos de nodos con muchos Pb de información?

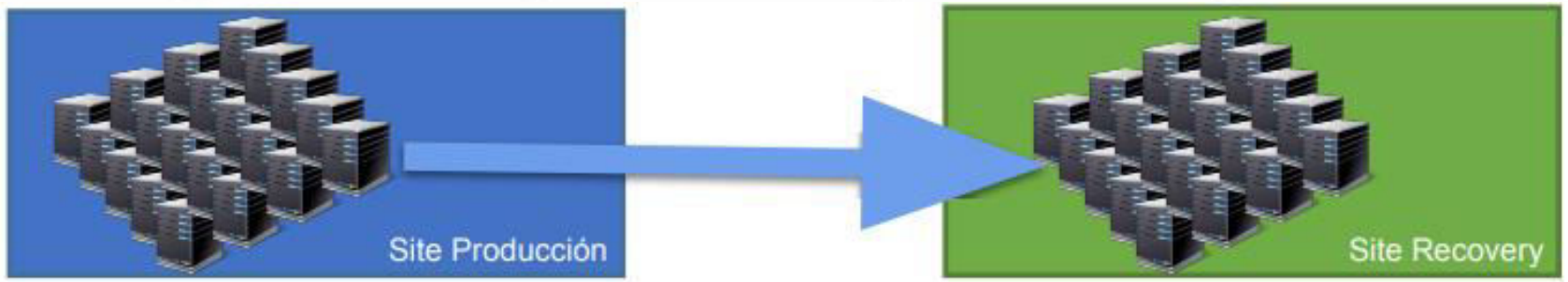
    NO!!
  - ¿Qué alternativas tenemos entonces?

CePETel

Recuperación ante Desastres

Disaster Recovery - Alternativas

- **Data Lake replicados en Real Time**



- **Data Lake principal y Data Lake reducido con datos mas recientes**

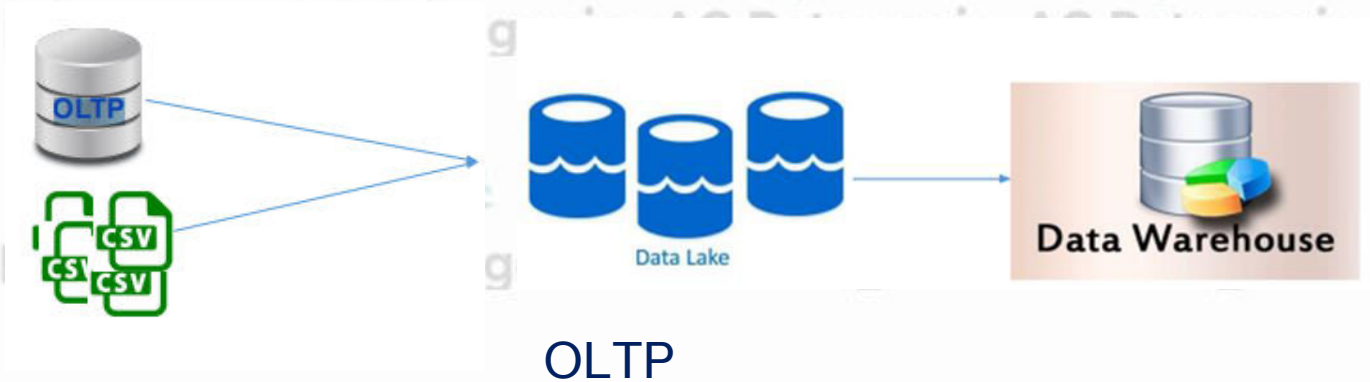
CePETel


Patrones Arquitecturales de un Data Lake

Patrones Arquitecturales

Data Lake como Staging

En esta arquitectura se observa que el DL actúa como en Área de Staging del Data Warehouse, mejorando los tiempos de transformación, agregación, limpieza y enriquecimiento de Datos.
El proceso tradicional de ETL en el Warehouse, se convierte en un proceso de ELT al Data Lake y un posterior proceso de ingesta al DW



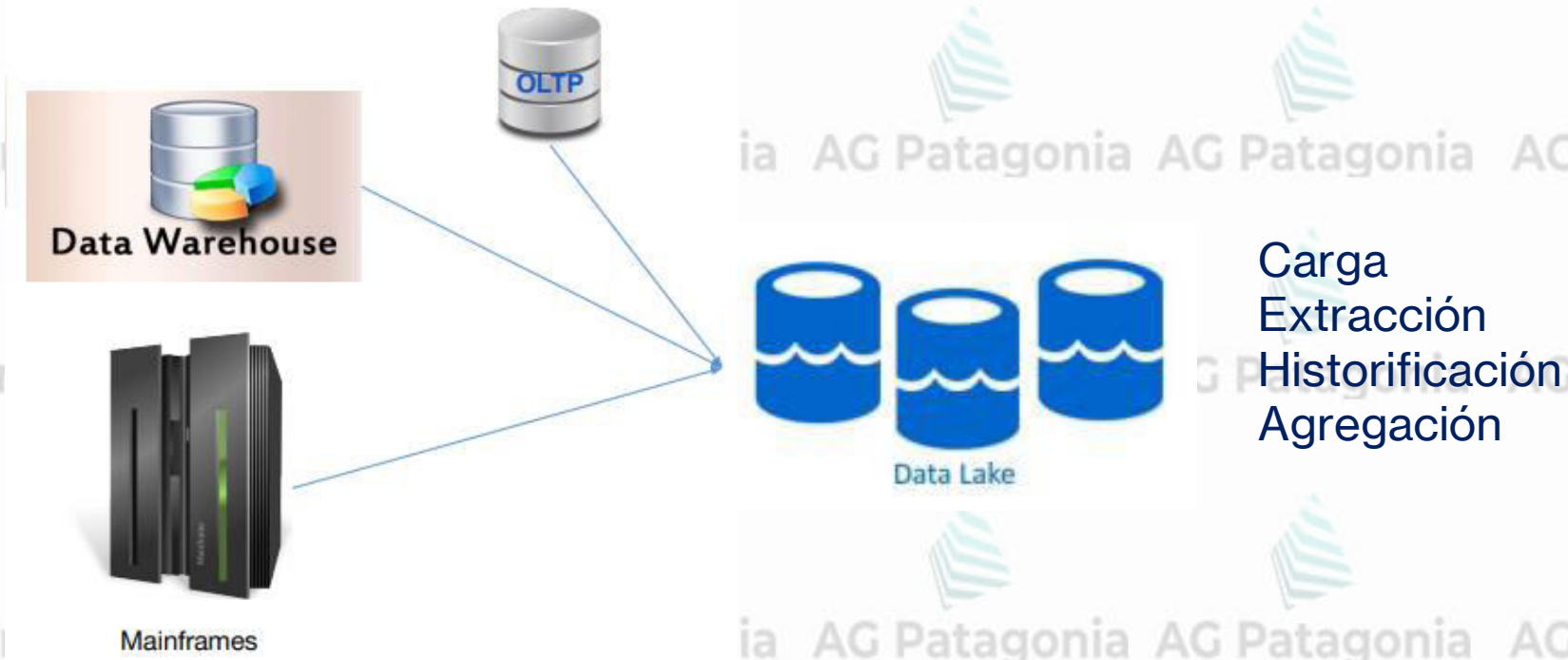
OLTP
ELT
Data cleansing
Transformación
Agregación
Enriquecimiento

CePETel

Patrones Arquitecturales

Data Lake - Historificación de Datos

En esta arquitectura se aprovecha el bajo costo del almacenamiento del Data Lake, y la capacidad para resolver consultas sobre grandes volúmenes de datos permitiendo una posterior extracción y agregación de los datos historificados.



Carga
Extracción
Historificación
Agregación

CePETel

Patrones Arquitecturales

Data Lake - Integración con Datos Externos

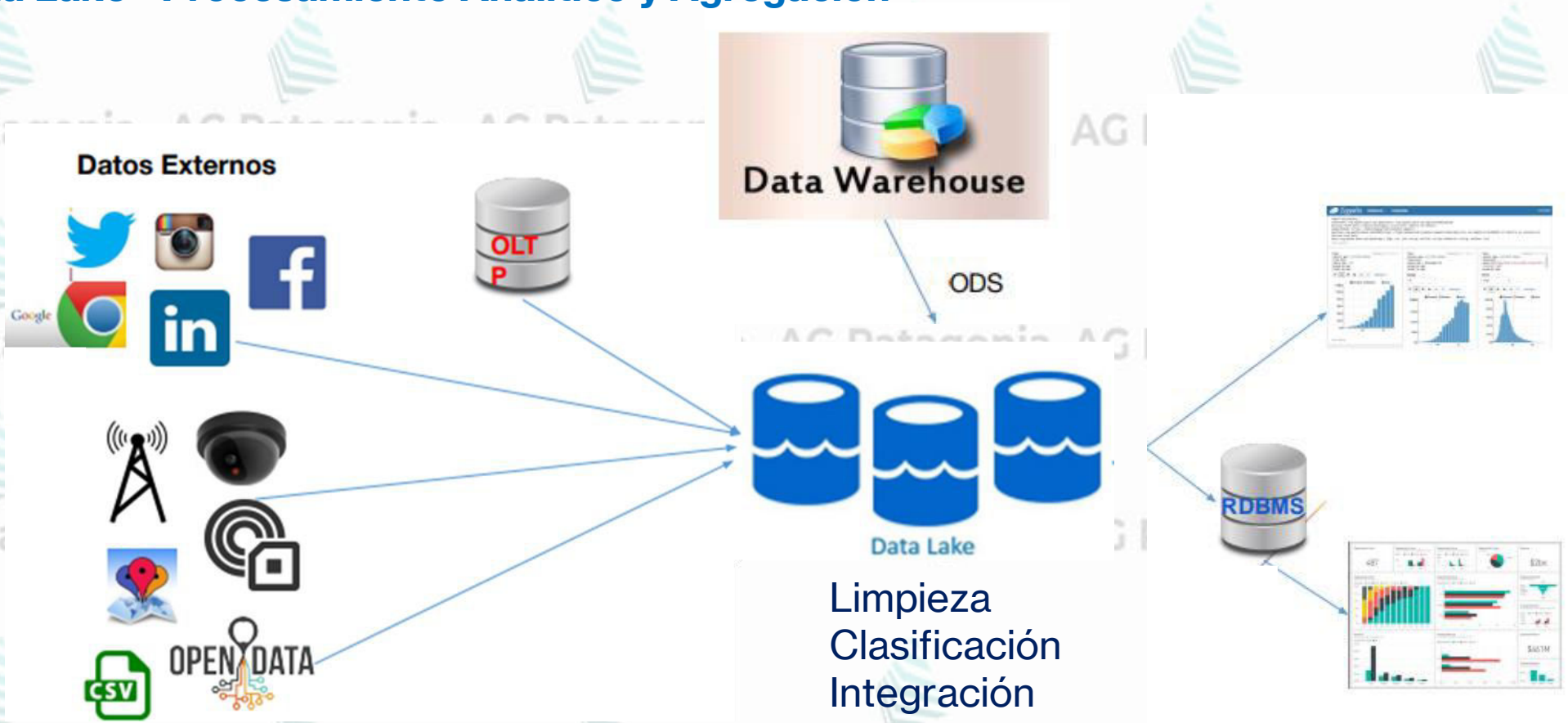
Esta arquitectura se basa en la captura e ingesta de datos externos a la organización y su posterior limpieza, transformación e integración con los datos de nuestra organización en el Data Lake, agregándolos y obteniendo subconjuntos de datos para nuestro Data Warehouse.

ELT
Integración con Datos Externos
Enriquecimiento

CePETel

Patrones Arquitecturales

Data Lake - Procesamiento Analítico y Agregación

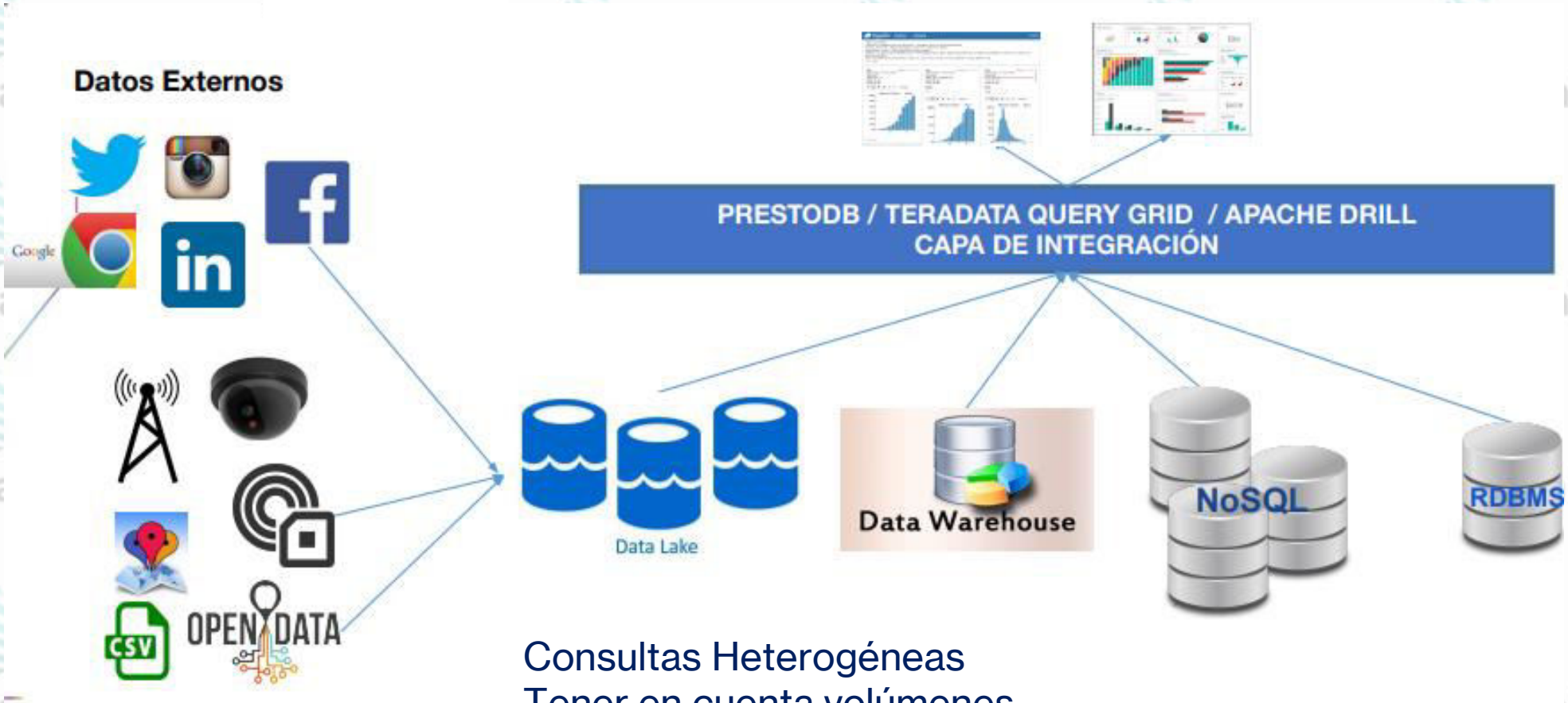


Limpieza
Clasificación
Integración
Transformación,
Agregación,
Enriquecimiento

CePETel

Patrones Arquitecturales

Herramientas de Integración – Data Hubs

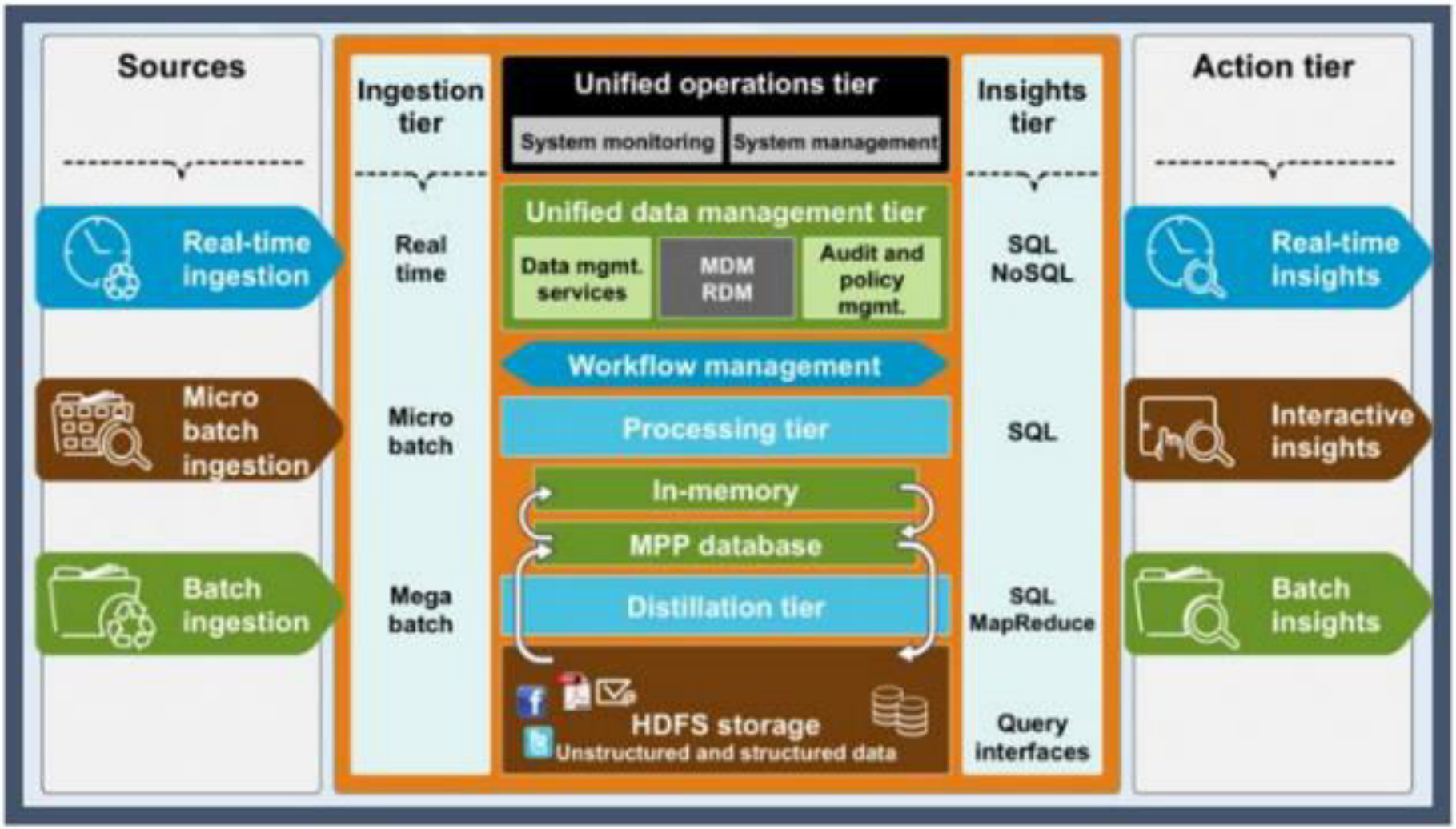


Consultas Heterogéneas
Tener en cuenta volúmenes
Tener en cuenta acceso perfomante por cada Entorno

CePETel

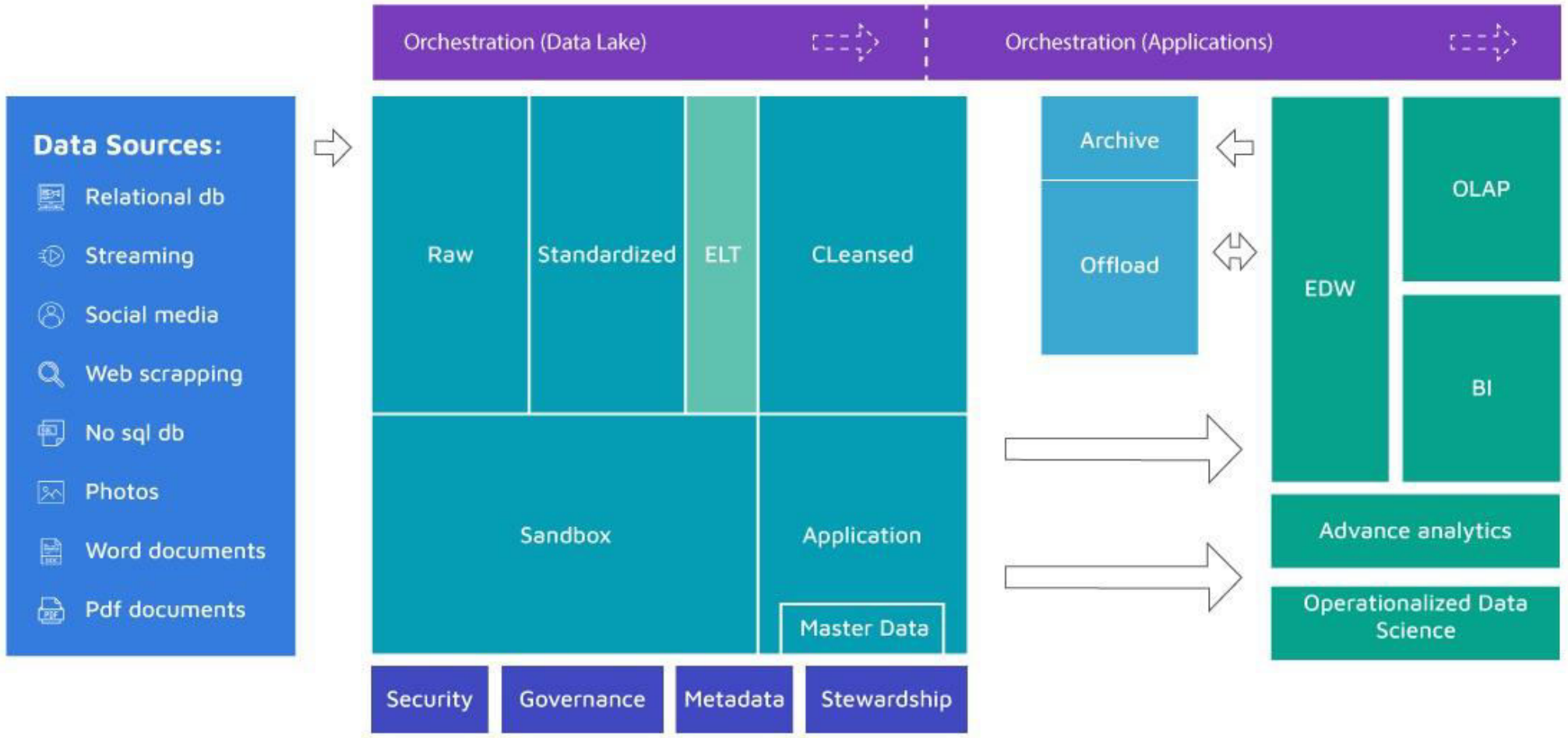
Arquitectura de un Data Lake

Arquitectura de un Data Lake



CePETel

Arquitectura de un Data Lake



CePETel

Arquitectura de un Data Lake

Un Data Lake debe presentar tres características clave:

- **Un único repositorio compartido de datos**: los data lake de Hadoop mantienen los datos en su forma original y capturan las modificaciones de los datos y la semántica contextual a lo largo del ciclo de vida de los datos. Este enfoque es especialmente beneficioso para las actividades de cumplimiento y auditoría.

- **Incluye capacidades de orquestación y programación de trabajos:** la ejecución de la carga de trabajo es un requisito previo para Hadoop empresarial. YARN permite la administración de recursos y una plataforma central para realizar operaciones consistentes, seguridad y servicios de gobierno de datos en clústeres de Hadoop, asegurando que los flujos de trabajo analíticos tengan acceso a los datos y la potencia informática que necesitan.

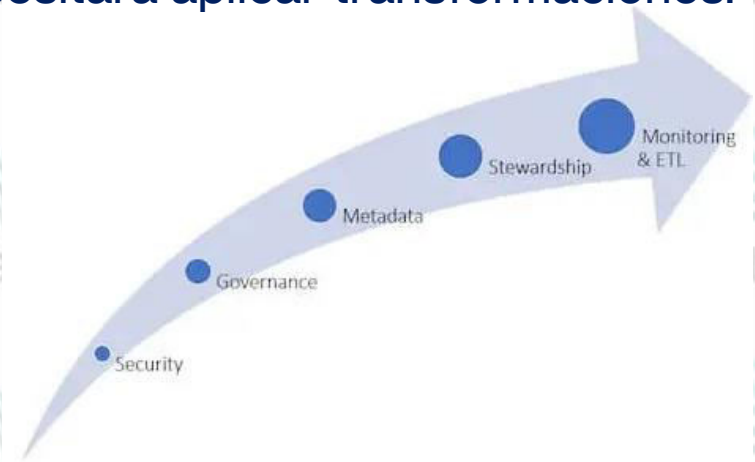
- **Tiene una colección de flujos de trabajo para ejecutar:** el fácil acceso de los usuarios es el sello distintivo de un data lake, ya que las organizaciones conservan los datos en su forma original. Los propietarios de los datos pueden fusionar datos de clientes, proveedores y operaciones, eliminando obstáculos técnicos, e incluso políticos, para compartir datos.

CePETel

Arquitectura de un Data Lake

Componentes clave:

- **Seguridad**: Es crucial pensar en este aspecto, especialmente durante la fase inicial y de arquitectura. No es como las bases de datos relacionales, con una artillería de mecanismos de seguridad.
- **Gobernanza**: el seguimiento y la supervisión de las operaciones serán vitales para medir el rendimiento y mejorar el lago de datos.
- **Metadatos**: datos que proporcionan información sobre otros datos, por lo que en su mayoría todos los esquemas, intervalos de recarga, etc.
- **Administración**: según la organización, el rol se puede asignar a un equipo separado o transferir esta responsabilidad a los propietarios (usuarios).
- **Procesos de monitoreo y ELT**: a medida que los datos van desde la capa sin procesar, necesita una herramienta para organizar el flujo a través de la capa de limpieza a la zona de pruebas y la aplicación, ya que a menudo necesitará aplicar transformaciones.



CePETel

Ingestas de Datos - Tipos

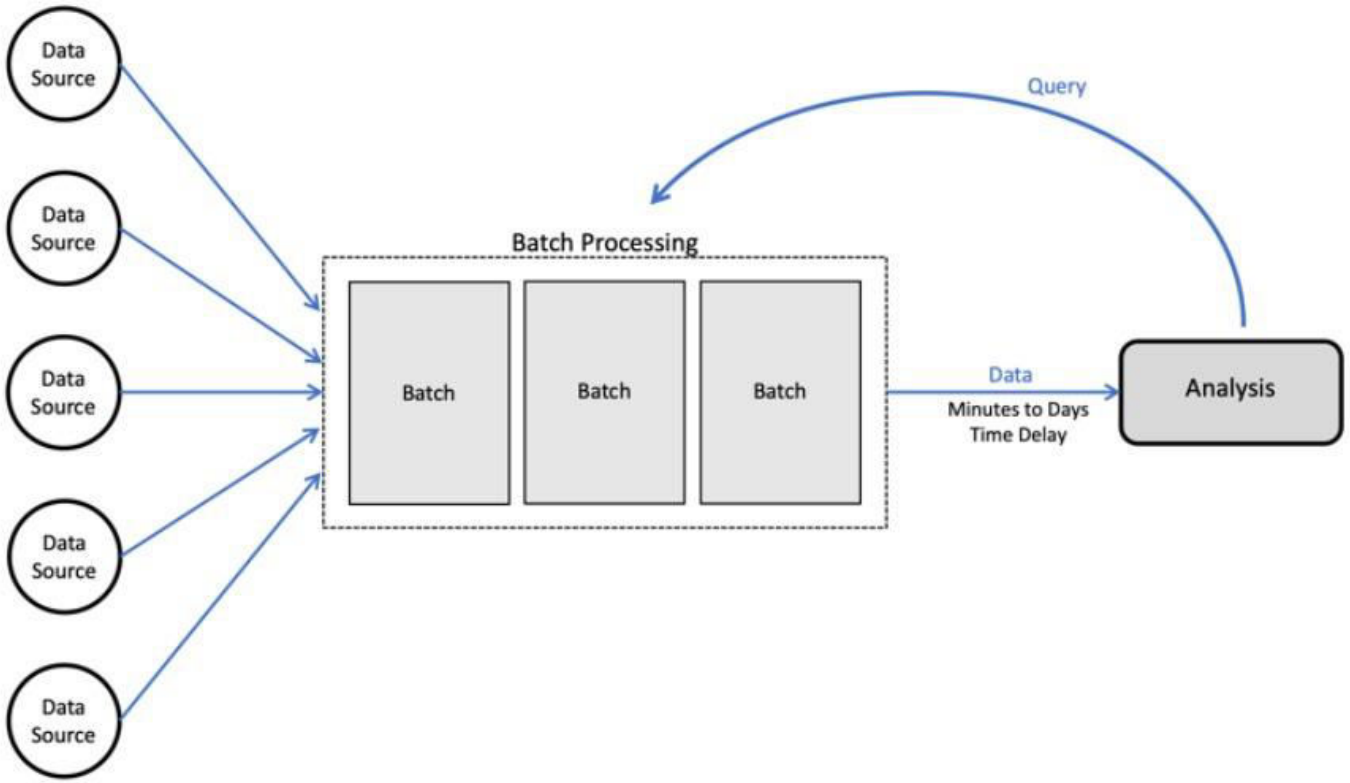
[Video: Diferencia procesamiento Batch vs. Real Time]

Ingesta Batch

En el procesamiento batch, esperamos que se "acumule" una cierta cantidad de datos sin procesar antes de ejecutar un trabajo ETL.

Por lo general, esto significa que los datos tienen entre una hora y unos pocos días antes de que estén disponibles para su análisis.

Los trabajos de ETL por batch (lotes) generalmente se ejecutarán en un horario establecido (por ejemplo, cada 24 horas) o, en algunos casos, una vez que la cantidad de datos alcance un cierto umbral.



CePETel

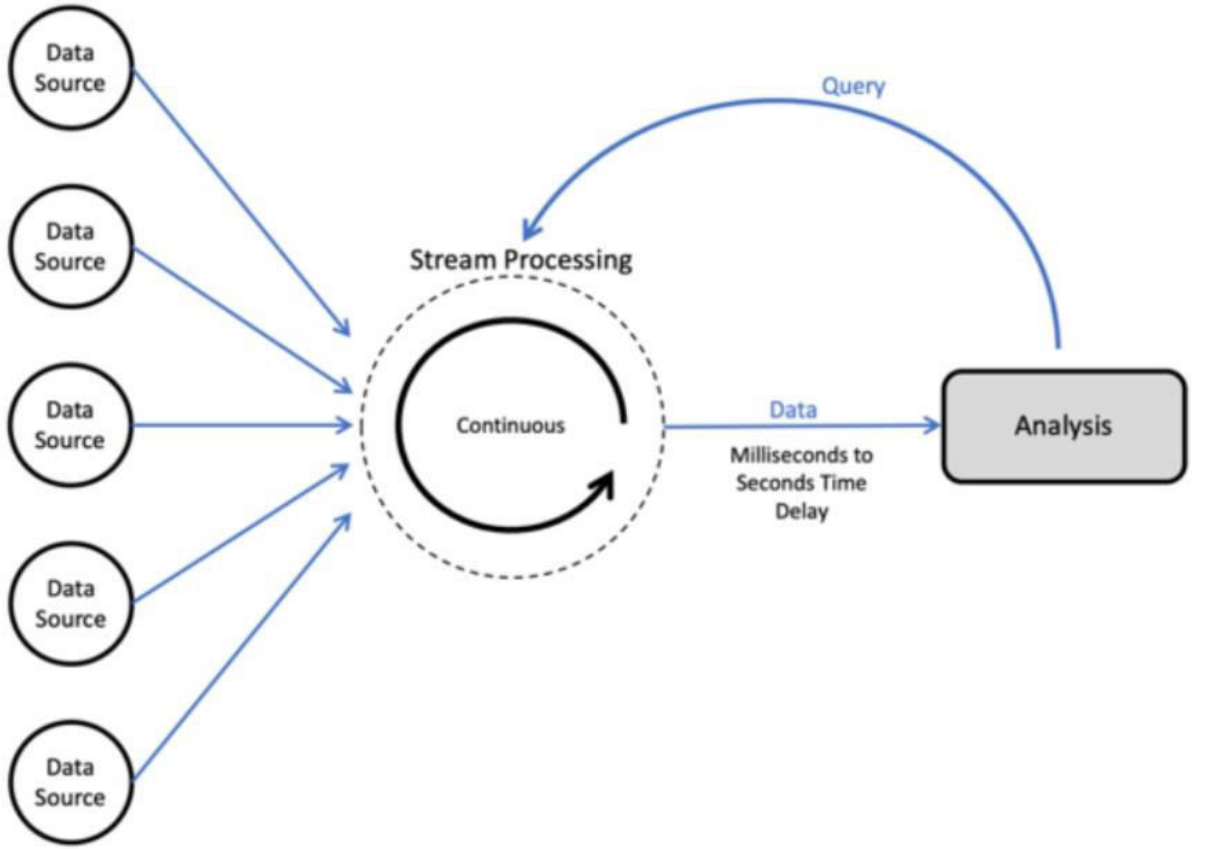
Ingestas de Datos - Tipos

Ingesta Real Time

En el procesamiento de flujo (real time processing), procesamos los datos tan pronto como llegan a la capa de almacenamiento, que a menudo también estaría muy cerca del momento en que se generaron (aunque no siempre sería así).

Esto normalmente sería en marcos de tiempo inferiores a un segundo, de modo que para el usuario final el procesamiento se realice en tiempo real.

Estas operaciones normalmente no tendrían estado, o solo podrían almacenar un estado 'pequeño', por lo que generalmente implicarían una transformación o cálculo relativamente simple.



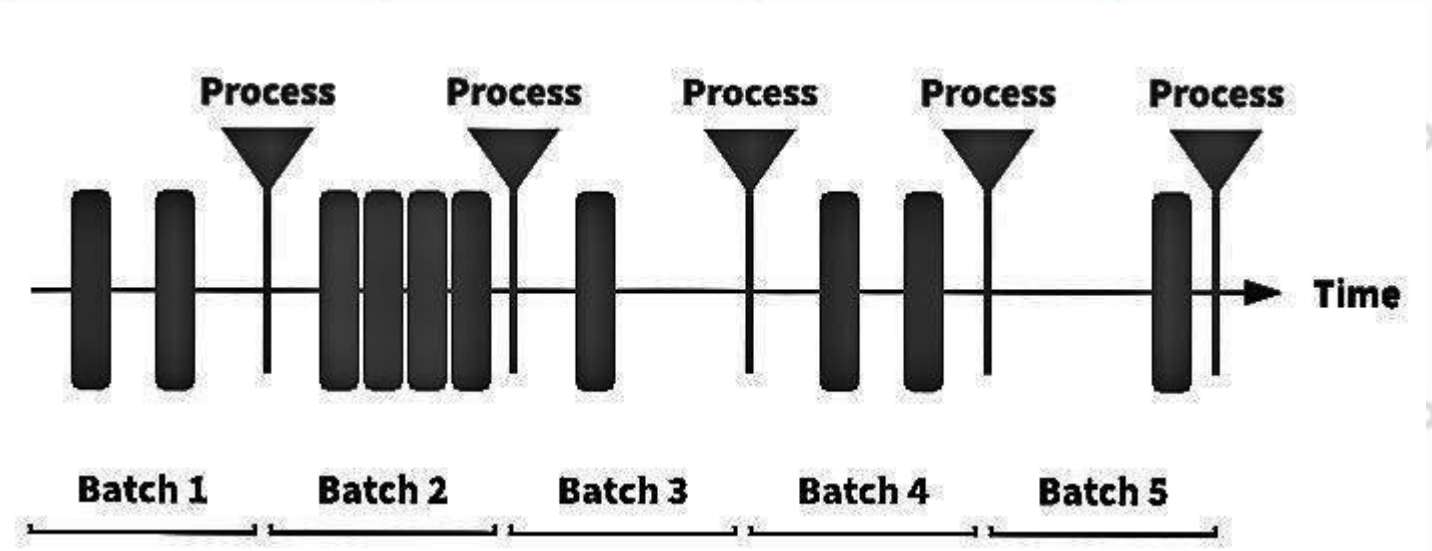
CePETel

Ingestas de Datos - Tipos

Ingesta Micro Batch

En el procesamiento de micro lotes (micro batch), ejecutamos procesos por lotes en acumulaciones de datos mucho más pequeñas, generalmente menos de un minuto de datos. Esto significa que los datos están disponibles casi en tiempo real.

En la práctica, hay poca diferencia entre el microprocesamiento por lotes y el procesamiento de flujo (real time), y los términos a menudo se usarían indistintamente en las descripciones de la arquitectura de datos y las descripciones de la plataforma de software.

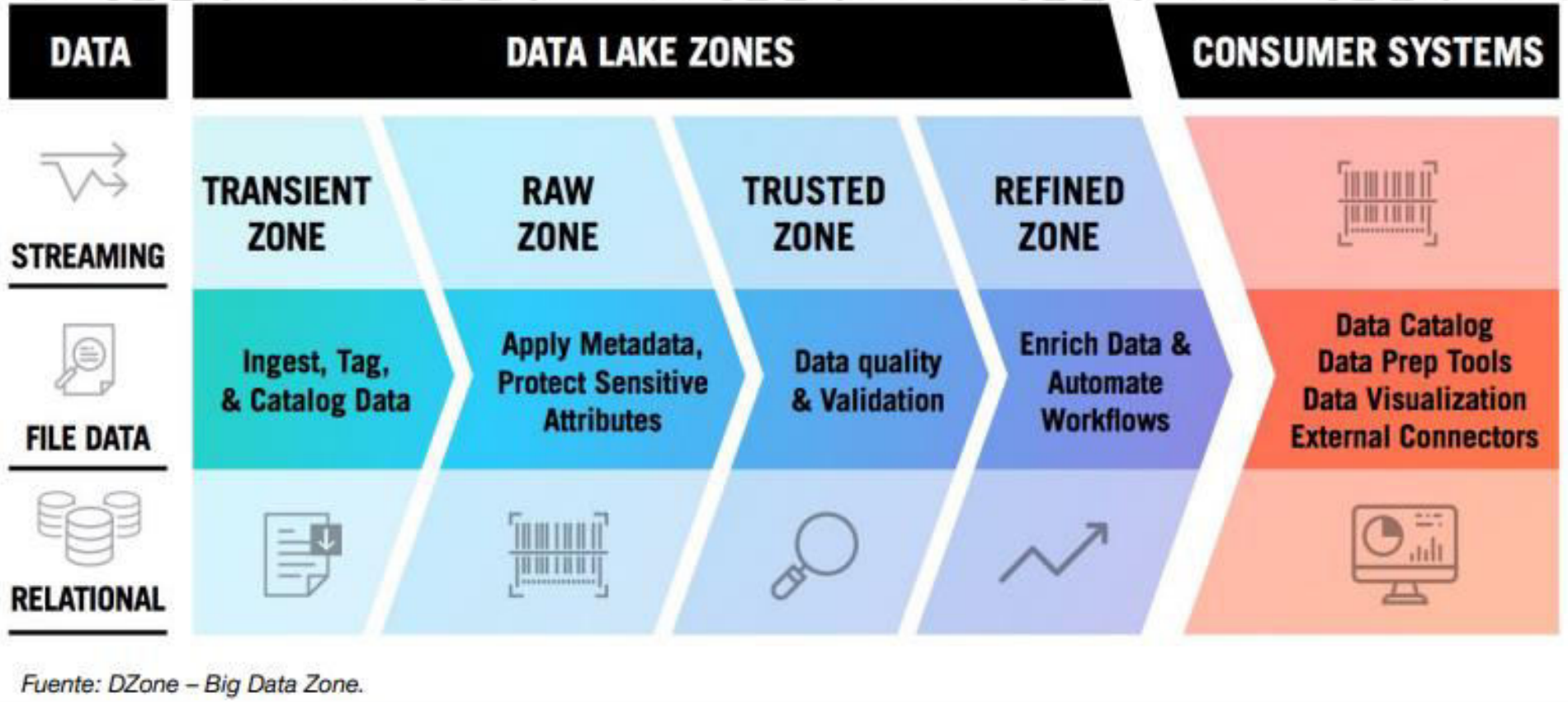


CePETel


Zonas de un Data Lake

Zonas de un Data Lake – Gobierno del Dato

Dentro de un data lake, las zonas permiten la separación lógica y/o física de datos que mantiene el entorno seguro, organizado y ágil.



Fuente: DZone – Big Data Zone.

CePETel

Zonas de un Data Lake – Gobierno del Dato

Transient Zone (Zona transitoria)

Se utiliza para almacenar datos efímeros, como copias temporales, spools de transmisión u otros datos de corta duración antes de ser ingeridos.

- Opcional/Utilizada selectivamente.
- Datos temporales.
- Mantener datos en tránsito entre zonas, utilizada por los pipelines de datos.

CePETel

Zonas de un Data Lake – Gobierno del Dato

Raw Zone (Zona sin procesar)

- Almacena datos crudos como llegan de sistemas fuentes.
- En caso de DB, se suele utilizar formatos columnares como Avro, Parquet, ORC para particionar los datos.
- No accesible excepto por procesos automáticos y cuentas de servicios/mantenimiento.
- Enmascarar y/o tokenizar datos sensibles.
- Datos se retienen indefinidamente y son inmutables.
- Nomenclatura y organización de archivos y carpetas basada en tiempo.

CePETel

Zonas de un Data Lake – Gobierno del Dato

Trusted Zone (Zona de confianza)

- Importa datos desde la Raw Zone.
- Calidad de datos: Procedimientos de validación y limpieza.
- Estandariza formato en columnas y filas (Ej: JSON -> Tabla).
- Organizada en entidades de negocio.
- Consolidación de archivos (Ej: archivos muy pequeños raw zone consolidan en único archivo)

CePETel

Zonas de un Data Lake – Gobierno del Dato

Refined Zone (Zona refinada)

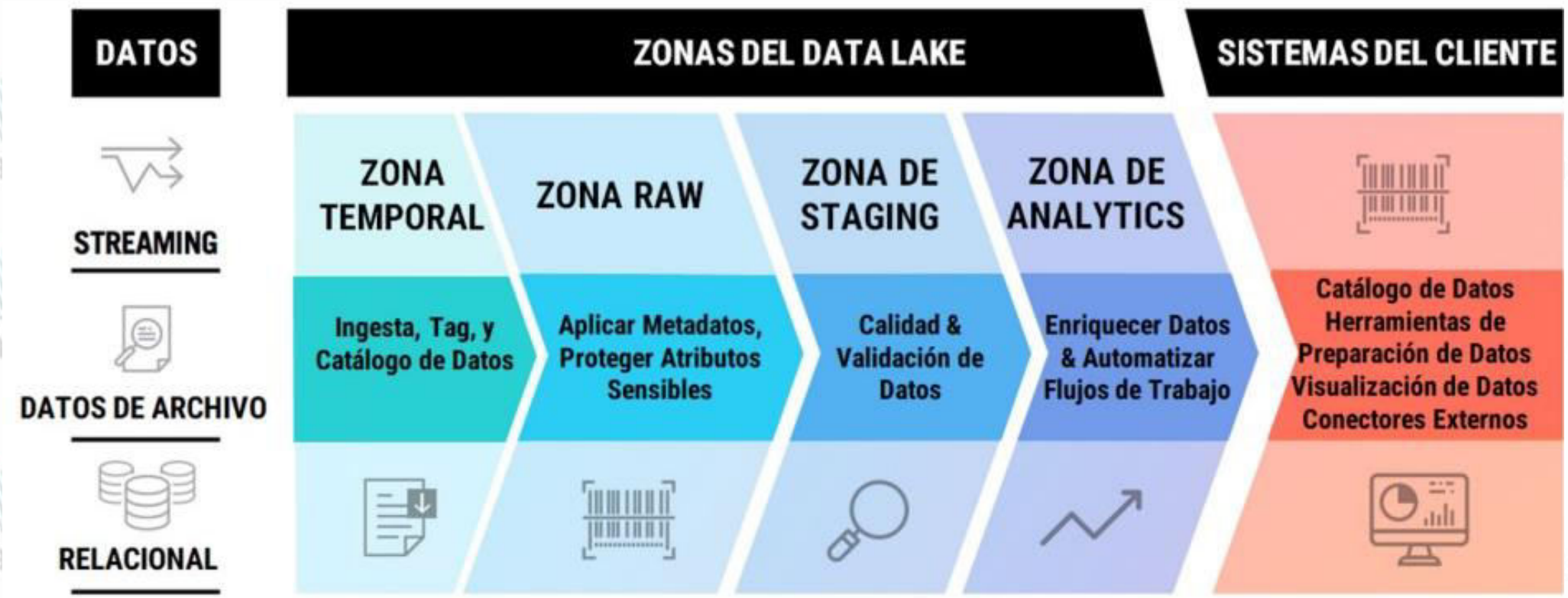
- Los datos manipulados y enriquecidos. Se pueden combinar datos de múltiples orígenes.
- Se agregan cálculos y generan datos agregados.
- Explotada por los usuarios finales con herramientas como Hive, Zeppelin, Ambari View (filosofía self-service) o otros sistemas que interactúan con el Data Lake.
- Organizados para acceso y consulta con óptima performance y más facilidad para los usuarios.

CePETel

Zonas de un Data Lake – Gobierno del Dato

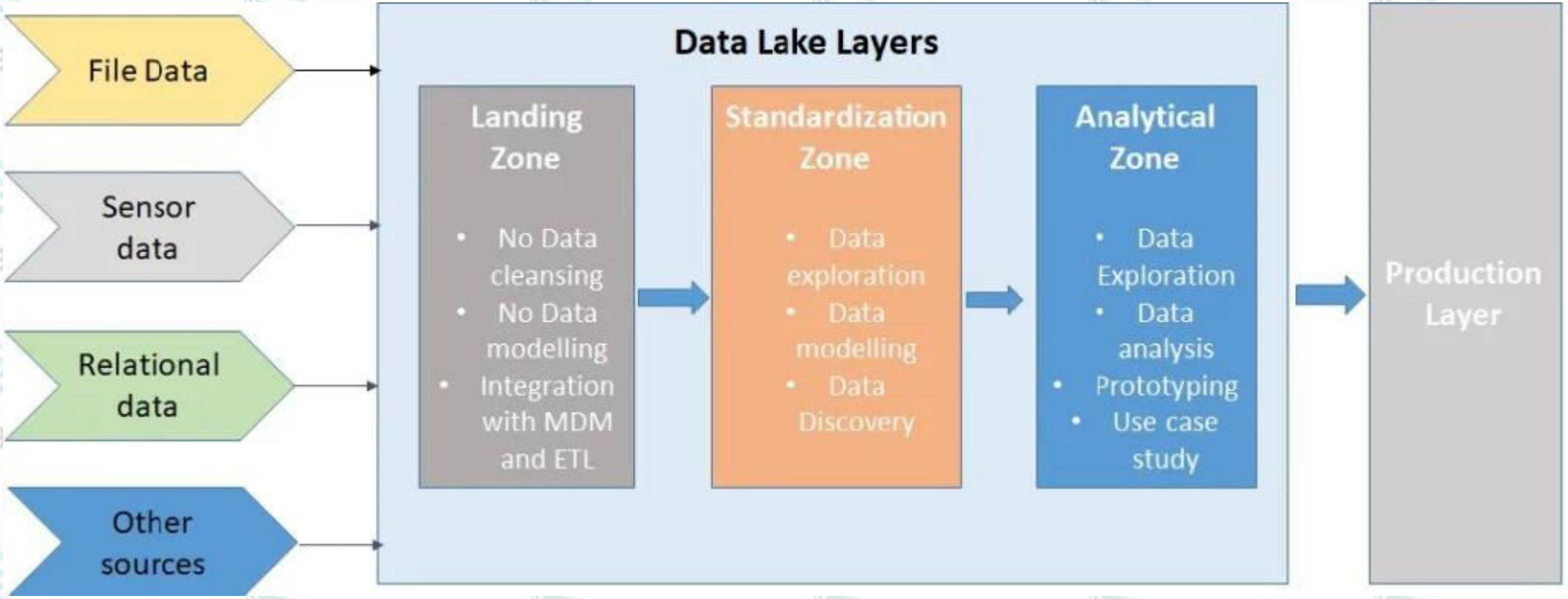
Convenciones

Las **zonas** dentro de un data lake **pueden variar proyecto a proyecto**, empresa a empresa.
En general suelen ser tres: Raw, Trusted y Refined.



CePETel

Zonas de un Data Lake



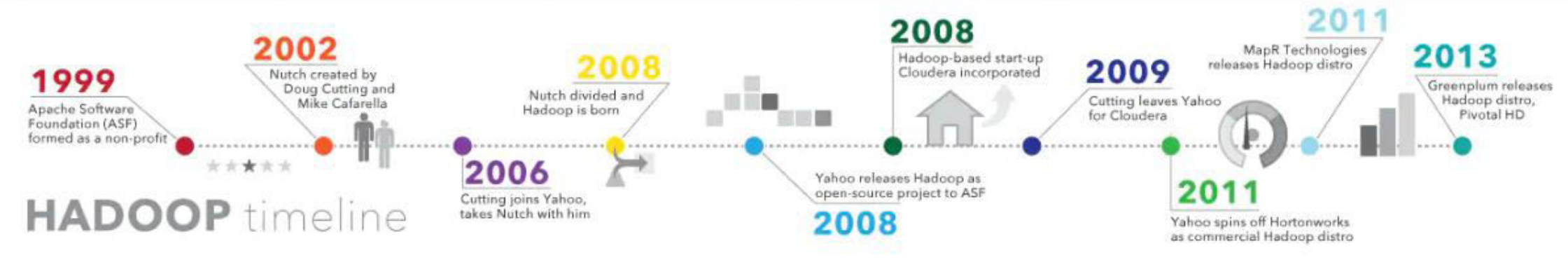
CePETel

Tecnologías para un Data Lake

Hadoop

Historia de Hadoop

A medida que la World Wide Web creció a finales de los 1900 y principios de los 2000, se crearon buscadores (o motores de búsqueda) e índices para ayudar a localizar información relevante dentro de contenido basado en texto.
En sus primeros años, los resultados de las búsquedas eran entregados por humanos. Pero a medida que la Web creció de docenas a millones de páginas, se requirió de la automatización. Se crearon los rastreadores Web, muchos como proyectos dirigidos por universidades, y entonces se iniciaron las primeras compañías de buscadores (Yahoo, AltaVista, etc.).



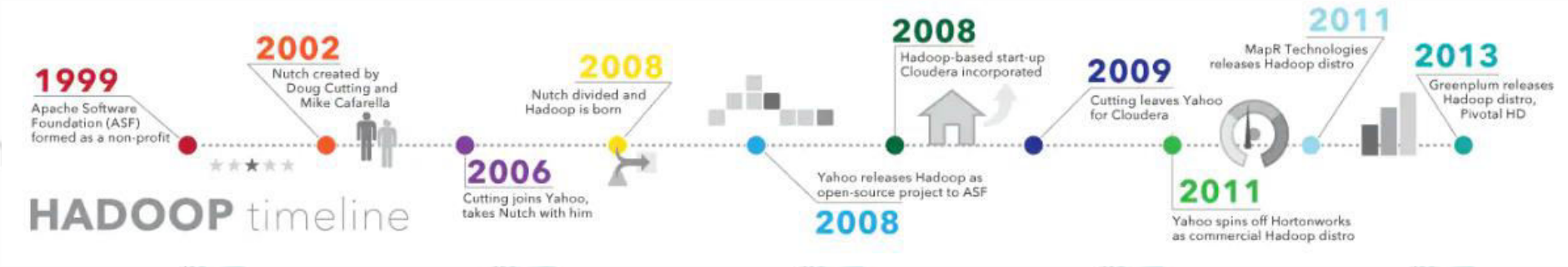
CePETel

Hadoop

Historia de Hadoop

Uno de estos proyectos fue un buscador Web de **código abierto** llamado **Nutch** – idea original de Doug Cutting y Mike Cafarella. Deseaban generar resultados de búsquedas en la Web a mayor velocidad distribuyendo datos y cálculos en diferentes computadoras de modo que se pudieran procesar múltiples tareas de manera simultánea. Durante este tiempo, estaba en progreso otro proyecto de buscador llamado **Google**. Éste se basaba en el mismo concepto – almacenar y procesar datos de manera distribuida y automatizada de modo que se pudieran generar resultados de búsquedas en la Web a mayor velocidad.

En **2006**, Cutting se unió a Yahoo y se llevó con él el proyecto Nutch, así como también ideas basadas en los trabajos iniciales de Google con la automatización del almacenaje y procesamiento de datos distribuidos. El proyecto Nutch fue dividido – la parte del rastreador Web se mantuvo como Nutch y la parte de cómputo y procesamiento distribuido se convirtió en Hadoop (en honor del elefante de juguete del hijo de Cutting). En 2008, Yahoo presentó Hadoop como proyecto de código abierto. Hoy día, la estructura y el ecosistema de tecnologías de Hadoop son gestionados y mantenidos por la Apache Software Foundation (ASF) sin fines de lucro, que es una comunidad global de programadores de software y otros contribuyentes.

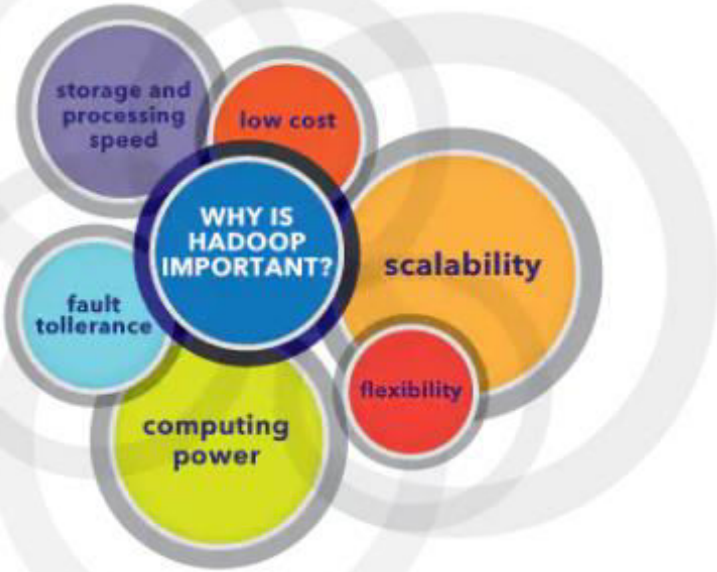


CePETel

Hadoop

¿Por qué es importante Hadoop?

- **Capacidad de almacenar y procesar enormes cantidades de cualquier tipo de datos, al instante.** Con el incremento constante de los volúmenes y variedades de datos, en especial provenientes de medios sociales y la Internet de las Cosas (IoT), ésa es una consideración importante.
- **Poder de cómputo**. El modelo de cómputo distribuido de Hadoop procesa big data a gran velocidad. Cuantos más nodos de cómputo utiliza usted, mayor poder de procesamiento tiene.
- **Tolerancia a fallos**. El procesamiento de datos y aplicaciones está protegido contra fallos del hardware. Si falla un nodo, los trabajos son redirigidos automáticamente a otros modos para asegurarse de que no falle el procesamiento distribuido. Se almacenan múltiples copias de todos los datos de manera automática.

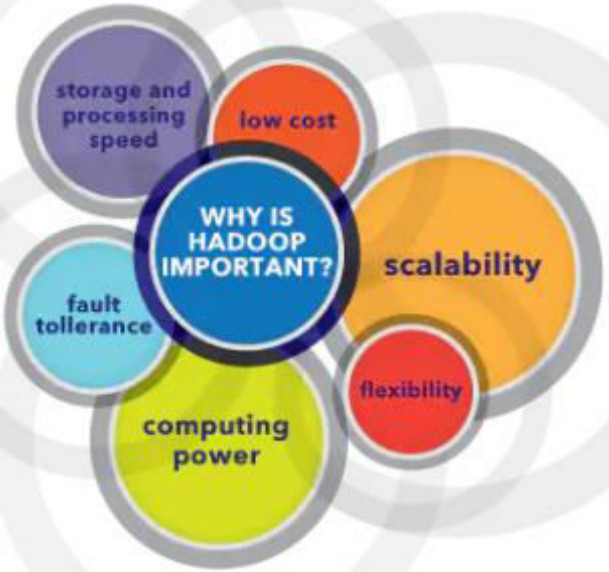


CePETel

Hadoop

¿Por qué es importante Hadoop?

- **Flexibilidad**. A diferencia de las bases de datos relacionales, no tiene que procesar previamente los datos antes de almacenarlos. Puede almacenar tantos datos como desee y decidir cómo utilizarlos más tarde. Eso incluye datos no estructurados como texto, imágenes y videos.
- **Bajo costo.** La estructura de código abierto es gratuita y emplea hardware comercial para almacenar grandes cantidades de datos.
- **Escalabilidad**. Puede hacer crecer fácilmente su sistema para que procese más datos son sólo agregar nodos. Se requiere poca administración.



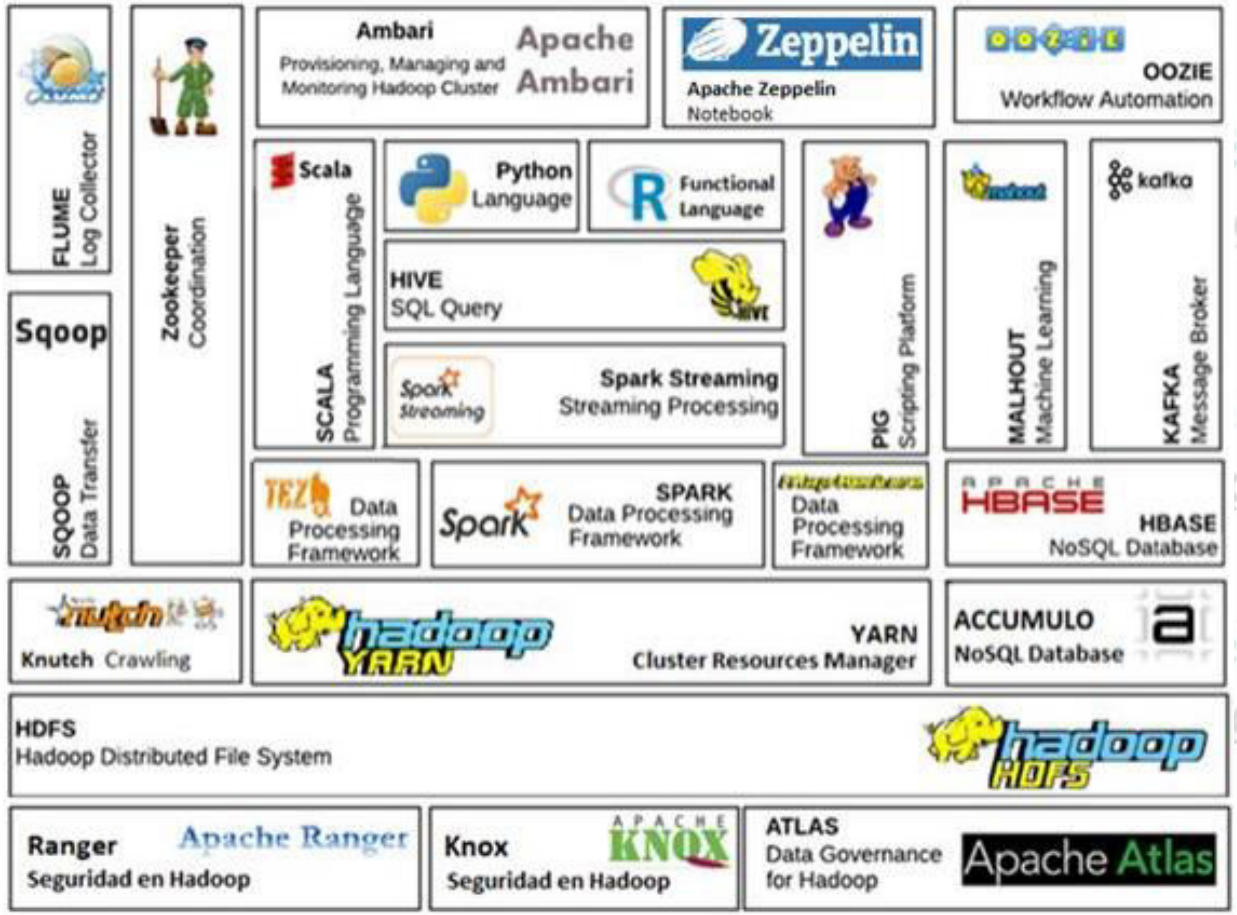
CePETel

Hadoop - Definición

¿Qué es y por qué es tan importante?

Hadoop es una estructura de software de **código abierto** para almacenar datos y ejecutar aplicaciones en clústeres de hardware comercial. Proporciona almacenamiento masivo para cualquier tipo de datos, enorme poder de procesamiento y la capacidad de procesar tareas o trabajos concurrentes virtualmente ilimitados.

Hadoop es un sistema de código abierto que se utiliza para almacenar, procesar en paralelo y analizar grandes volúmenes de datos.



CePETel

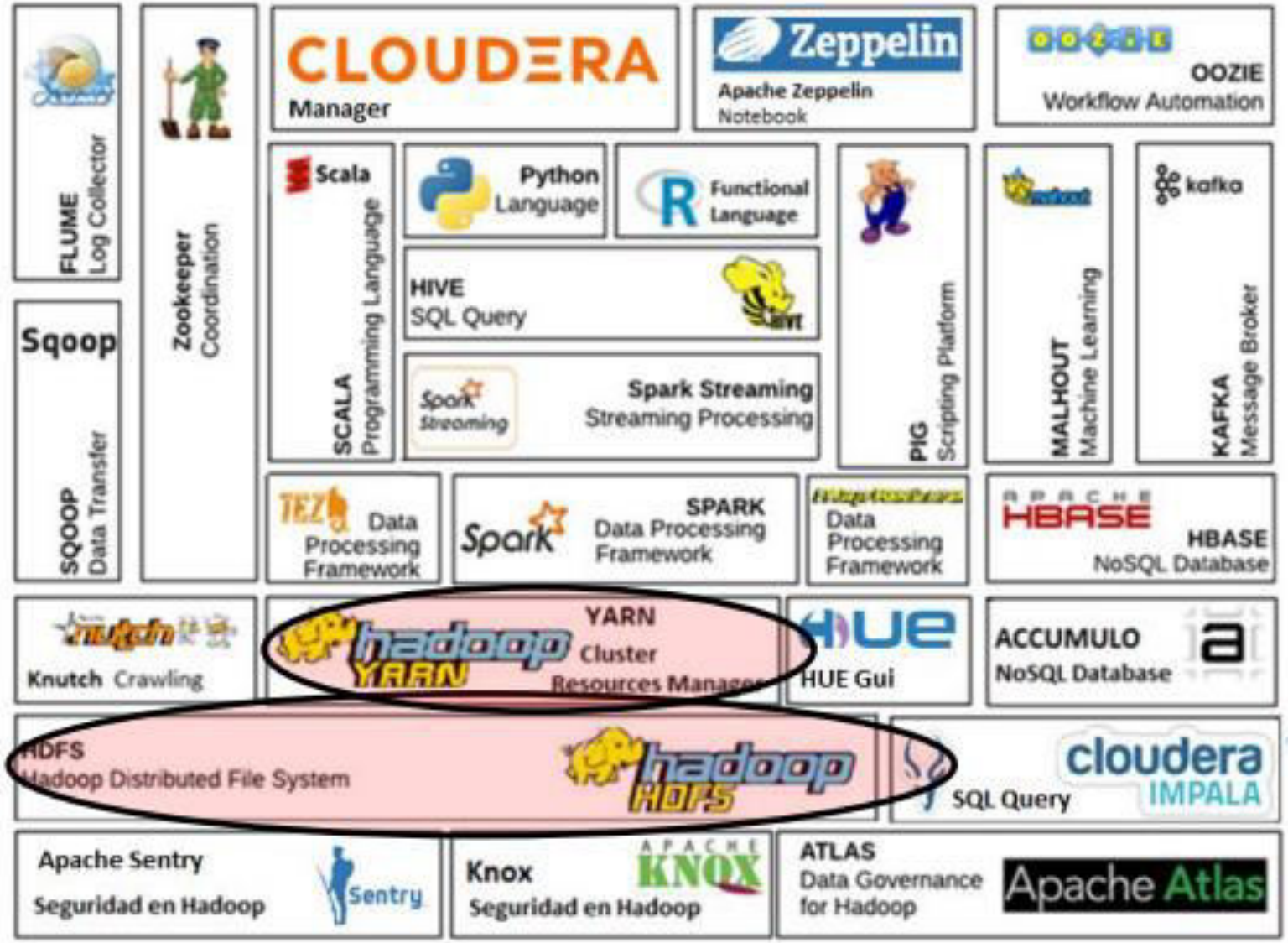
Hadoop – Características principales

- **Aísla a** los desarrolladores de todas las **dificultades** presentes en la **programación paralela.**
- Cuenta con un ecosistema que sirve de gran ayuda al usuario, ya que permite **distribuir los archivos en nodos**, que no son otra cosa que servidores con commodity-hardware.
- Es capaz de **ejecutar procesos en paralelo** en todo momento.
- Dispone de **módulos de control** para la **monitorización de los datos.**
- **Cuenta con distintas herramientas para realizar consultas de los datos.**
- También **potencia la aparición de distintos add-ons**, que facilitan el trabajo, **manipulación y seguimiento de toda la información** que en él se **almacena**.

CePETel

Ecosistema Hadoop – HDFS - Yarn

[Video: Hadoop]



CePETel

HDFS - ¿Qué es?

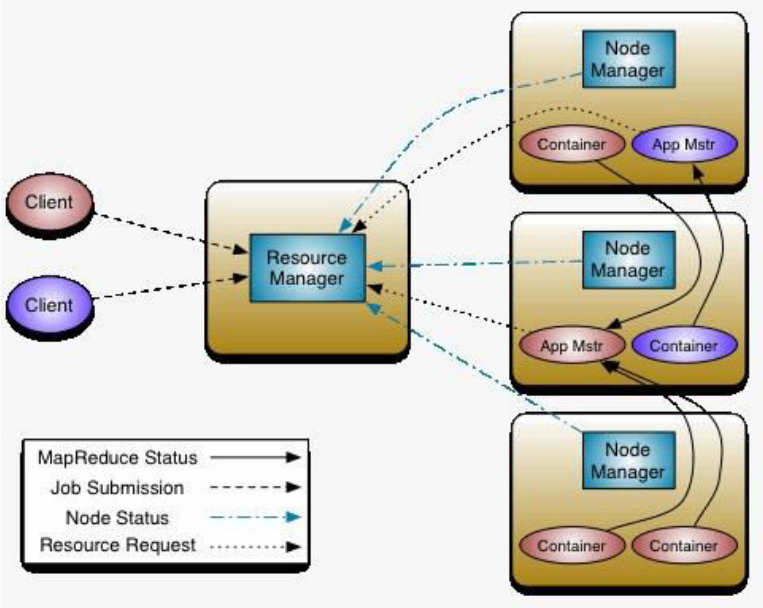
- Hadoop Distributed File System
    - File System escrito en java, basado en GFS (Google File System).
    - Puede montarse sobre casi cualquier file system (ext3, ext4, ntfs, fat32).

Pensado para:

- Archivos grandes y volumen de datos (órdenes de Terabytes).
- Trabajar con un esquema de Write once / read many times.
- Los contenidos de los archivos no pueden modificarse. Solo agregar datos al final de un archivo.
- La latencia para recuperar un archivo entero es mucho más importante que para obtener una línea de un archivo.

CePETel

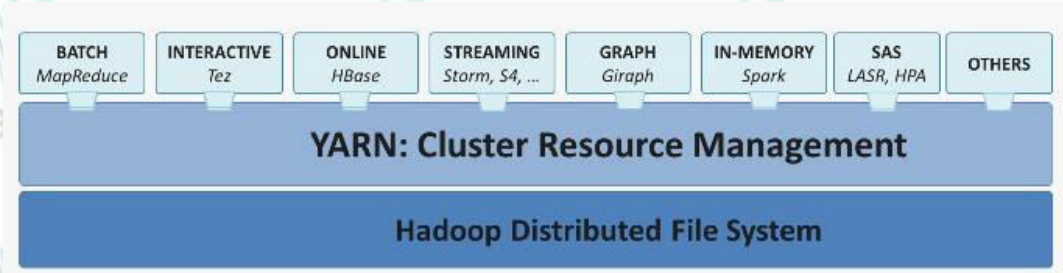
Yarn - ¿Qué es?



- Hadoop ahora tiene un **entorno de gestión de recursos y aplicaciones distribuidas** donde se pueden implementar múltiples aplicaciones de procesamiento de datos totalmente personalizadas y específicas para realizar una tarea en cuestión
  - **"Yet Another Resource Negotiator"**

- Surge por las siguientes necesidades:
  - Escalabilidad
  - Disponibilidad
  - Utilización
  - Multitenancy

- Presenta una separación entre la administración de recursos y la ejecución y monitoreo de procesos.



CePETel

HDFS - Conceptos

• **HDFS** está diseñado para trabajar con un número modesto de **archivos extremadamente grandes**.

• Implementa un **modelo de acceso a datos**: Write Once, Read Often.

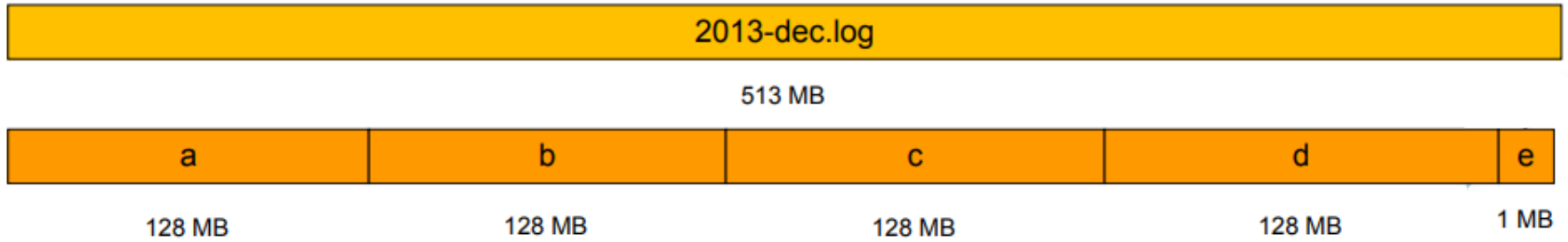
• Los **contenidos** de los archivos **no pueden modificarse** más allá de **agregar** datos al **final del archivo**.

• Las **operaciones más comunes** son:
  • Agregar contenido al final del archivo
  • Borrar un archivo
  • Renombrar un archivo
  • Modificar los atributos del archivo, como “owner”

CePETel

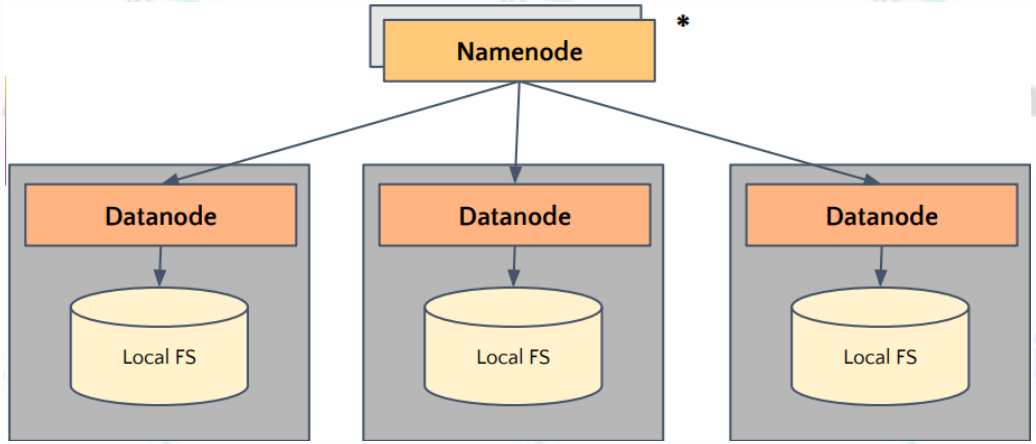
HDFS – Almacenamiento de datos

- Un tamaño de **bloque** típico en Linux es de 4KB, mientras que en Hadoop es de **128 MB**.
- Esto es porque Hadoop está diseñado para almacenar datos en el **orden de los Petabytes**.
- Para cada bloque existe **metadata** que es **trackeada** por un server central.
- Si el tamaño de bloque fuese más chico, habría un overhead de **metadata** enorme.
- Una instancia HDFS puede consistir de miles de servidores, **cada uno almacenando parte de los datos** del sistema de archivos.
- Cuando se almacena un archivo en HDFS **lo parte en un conjunto de bloques individuales** y almacena esos bloques en nodos esclavos.
- HDFS **no sabe que está almacenado dentro del archivo**, por lo que lo parte por el tamaño del bloque, sin ninguna regla que sea entendible por humanos.
- Existe un tamaño de bloque predefinido para la instancia Hadoop aunque **se puede definir un tamaño de bloque custom por archivo.**



HDFS – Arquitectura

Un cluster de HDFS está compuesto por **dos tipos de nodos**:



Namenode

Nodo maestro de la operación con HDFS.
Es el encargado de almacenar toda la metadata para los nodos del cluster: Qué bloques componen qué archivos y cual es la estructura de directorios del Filesystem son las preguntas que este nodo debe responder.

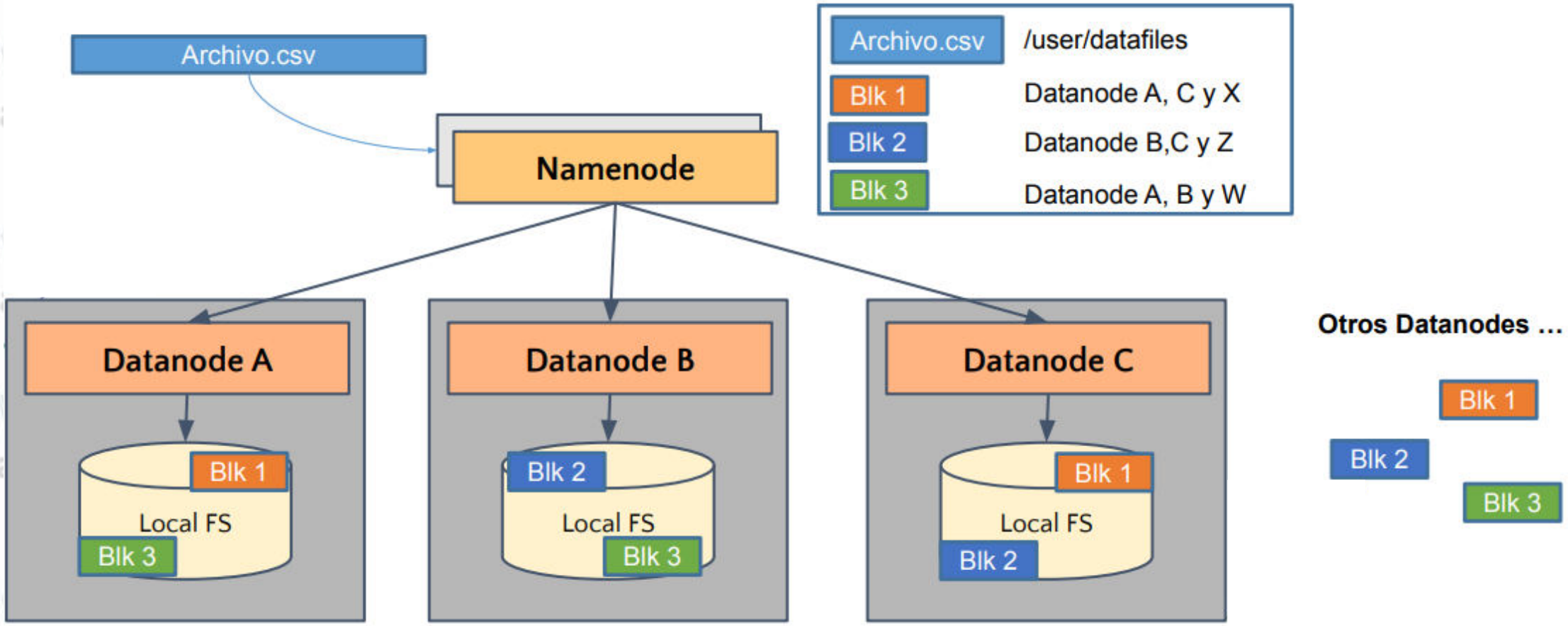
Datanode

Nodos workers del cluster.
Los bloques de datos son almacenados y servidos por estos nodos. Periódicamente deben reportar al namenode la lista de los bloques que contienen.

No es posible utilizar el File System **sin el namenode**. Si el Namenode explotara, sería imposible reconstruir archivos desde los datanodes.

CePETel

HDFS – Arquitectura

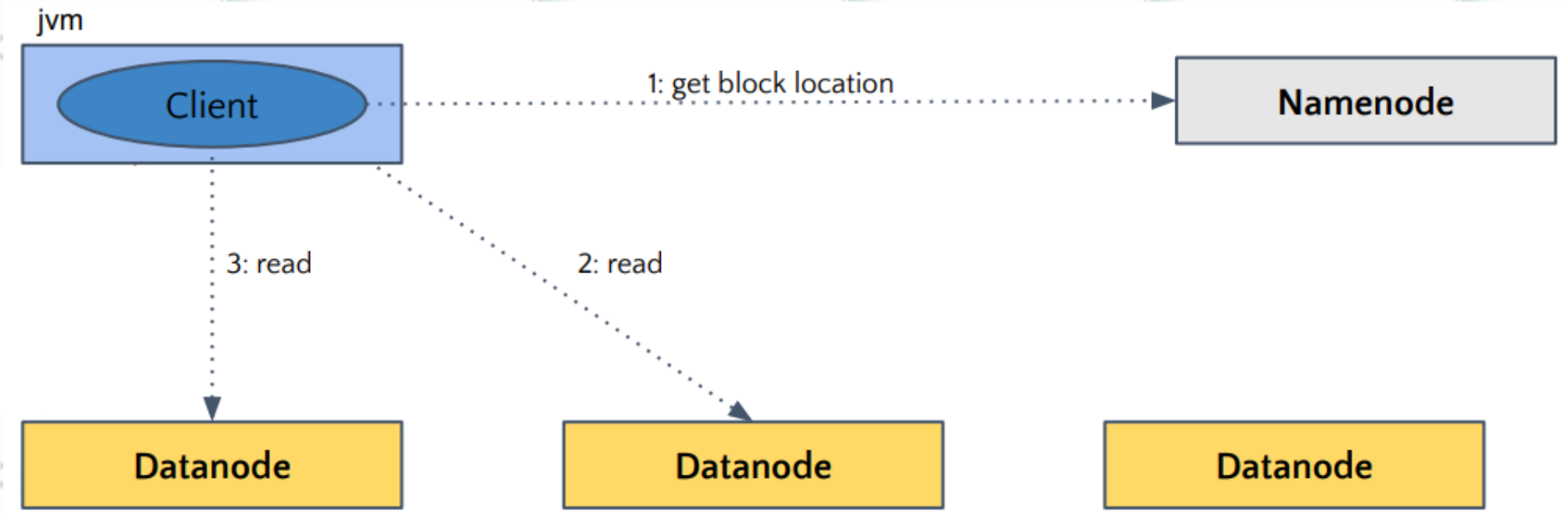


CePETel

HDFS – Lectura de Datos

• El NameNode devuelve una lista de Datanodes que contienen los bloques ordenados por proximidad.

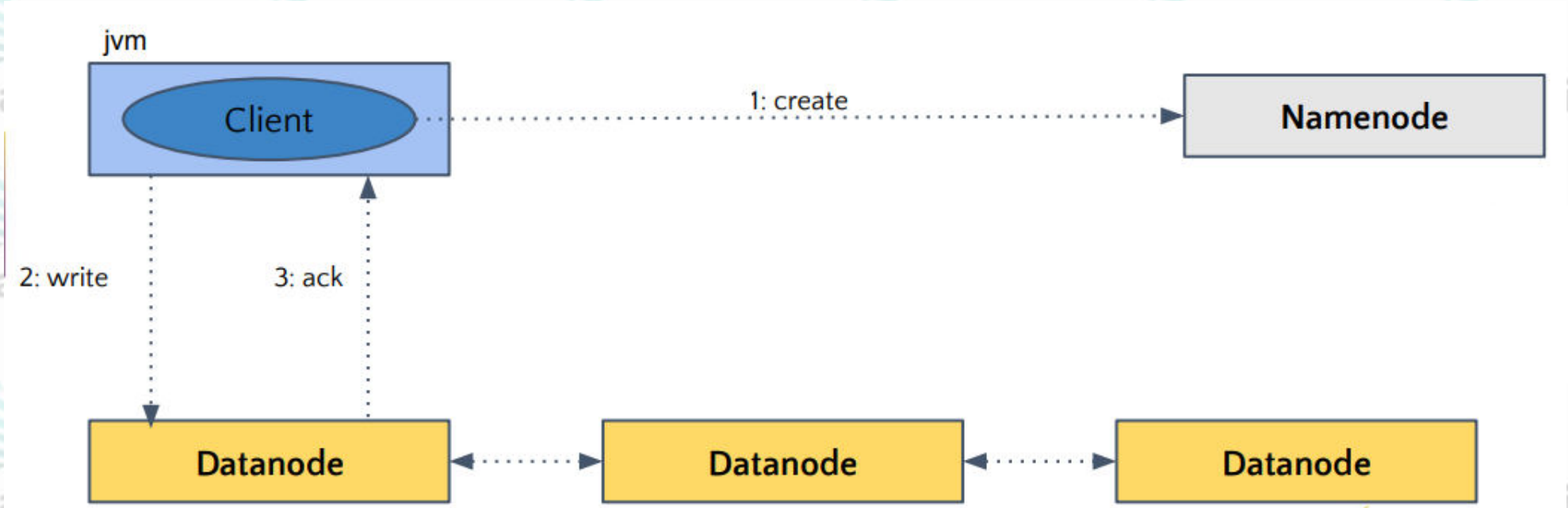
• Un aspecto importante del diseño: El Namenode solo sirve las direcciones y guía al cliente, no participando en la transferencia de datos, permitiendo escalar de mejor manera minimizando su tráfico de red.



CePETel

HDFS – Escritura de Datos

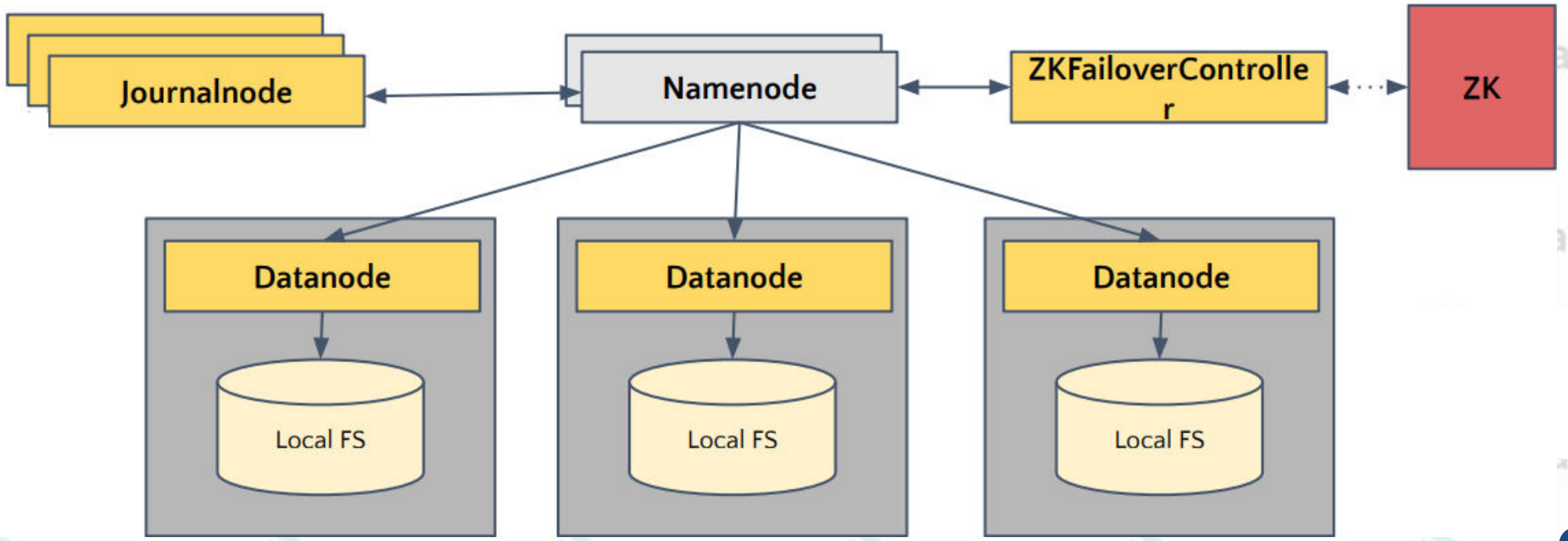
•El Namenode selecciona una lista de Datanodes para cada Bloque en base a su criterio para escribir.
• Los Datanodes arman un pipe y se escriben unos a otros sincrónicamente hasta dsf.namenode.replication.min y asincrónicamente hasta dfs.replication



CePETel

HDFS – Tolerancia a Fallos

- La **falla es la norma** y no la excepción
- El hardware es común. Hadoop está pensado para usar **hardware commodity**.
- Dado que hay un gran número de **componentes** y que cada componente tiene una **probabilidad relativamente alta de fallar**, esto significa que siempre existe la posibilidad de que algún componente no esté funcionando.
- La **detección de fallas** y la **recuperación automática** y rápida es un objetivo clave de la arquitectura de HDFS.



CePETel

HDFS – Tolerancia a Fallos

- En Hadoop 2 se introduce el concepto de **alta disponibilidad**, evitando que exista un único punto de fallo en el sistema.
- Un cluster de Hadoop 2 con HA permite **configurar un máximo de 2 namenodes**. Entre ellos, uno será el **activo** (se encarga de recibir request) y **otro** estará en **Standby** hasta la eventual caída del primario tomando su lugar.

Para configurarlo es necesario:

- Contar con un Storage consistente distribuido de los metadatos del cluster
- Contar con un Controlador de Failover que observe el estado de ambos Namenodes.
- Configurar Datanodes para que notifiquen sus bloques a ambos Namenodes
- Configurar a los clientes para que sepan encontrar al Namenode primario

Existen varias soluciones para esto. Hadoop utiliza **QJM** (Quorum Journal Managers) como solución de Storage de Edits y metadatos y **Zookeeper** como Controlador de Failover.

HDFS – Tolerancia a Fallos

Quorum Journal Manager

Un QJM es un Storage altamente disponible. Todas las modificaciones en el cluster son notificados a los Journal nodes.

El Standby Namenode se mantiene actualizado gracias a ellos.

- Diseñado para proveer edit logs disponibles
- Todas las escrituras deben realizarse en la mayoría de los nodos (W > N/2).
- Contiene una implementación dedicada de HDFS

Frente a la caída del Namenode Activo, el Standby puede tomar su lugar en pocos segundos ya que mantiene una versión casi actualizada de los metadatos y los mapeos de los bloques en memoria

ZKFailover

Este failover controller es la opción por default de Hadoop.

Utiliza Zookeeper para manejar la coordinación de los estados entre los Namenodes, lanzando un Failover (notificando al Standby Namenode) si fuese necesario.

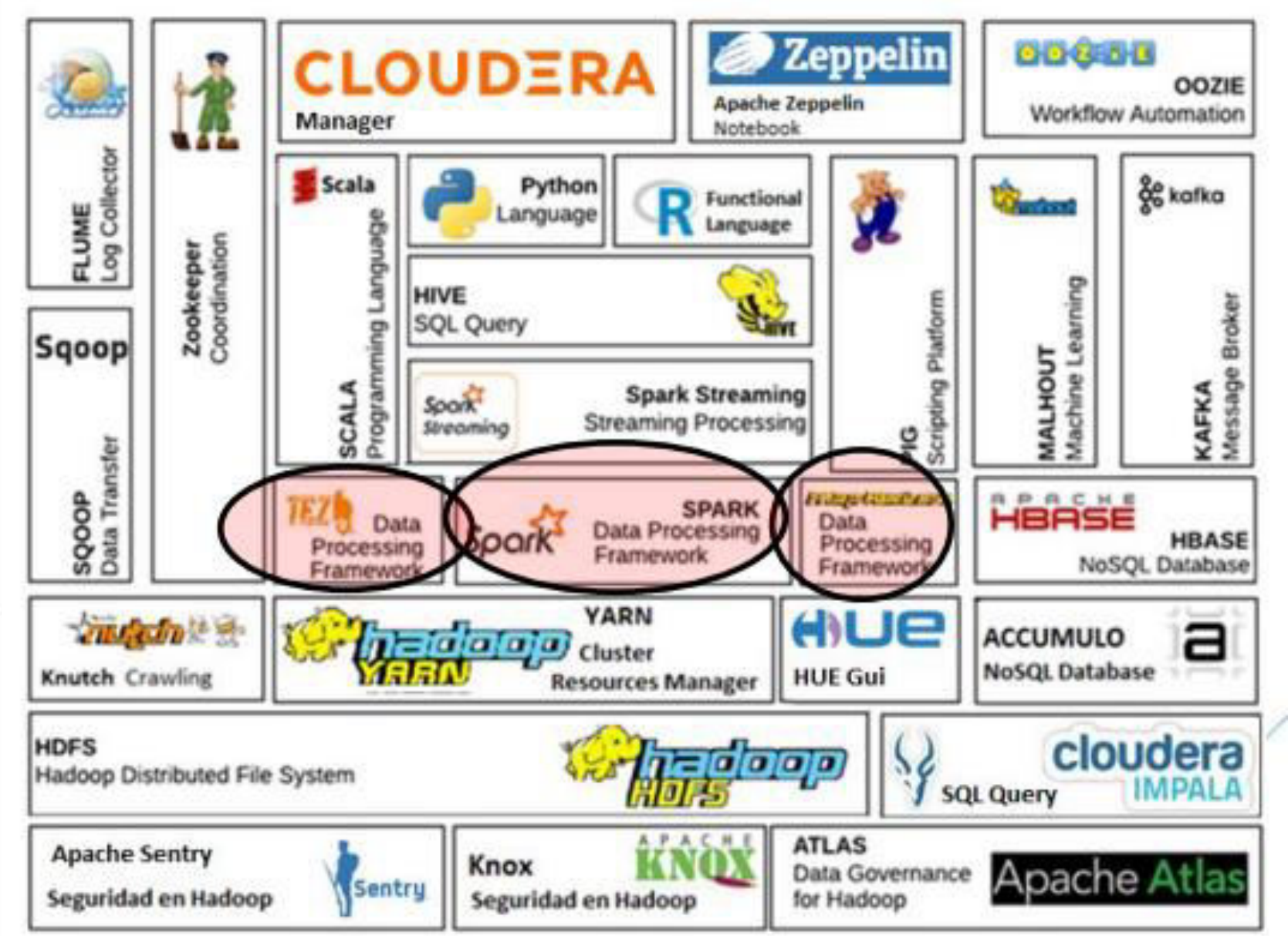
En Zookeeper se mantiene el estado de los Namenodes asegurando que sólo uno puede ser activo en cualquier momento.

Un Failover puede lanzarse de manera manual para realizar tareas de mantenimiento en los Namenodes de manera alternada.

CePETel

Componentes de Hadoop

Ecosistema Hadoop - Motores de Procesamiento Distribuido



CePETel

Ecosistema Hadoop - Motores de Procesamiento Distribuido

MapReduce

MapReduce es un modelo de programación y su implementación asociada para procesar y generar grandes sets de datos.
Framework de procesamiento distribuido (en clusters) que divide los problemas de procesamiento de grandes volúmenes de datos en subproblemas (Map) y luego recopila las mini-respuestas (Reduce) para generar conclusiones

Spark

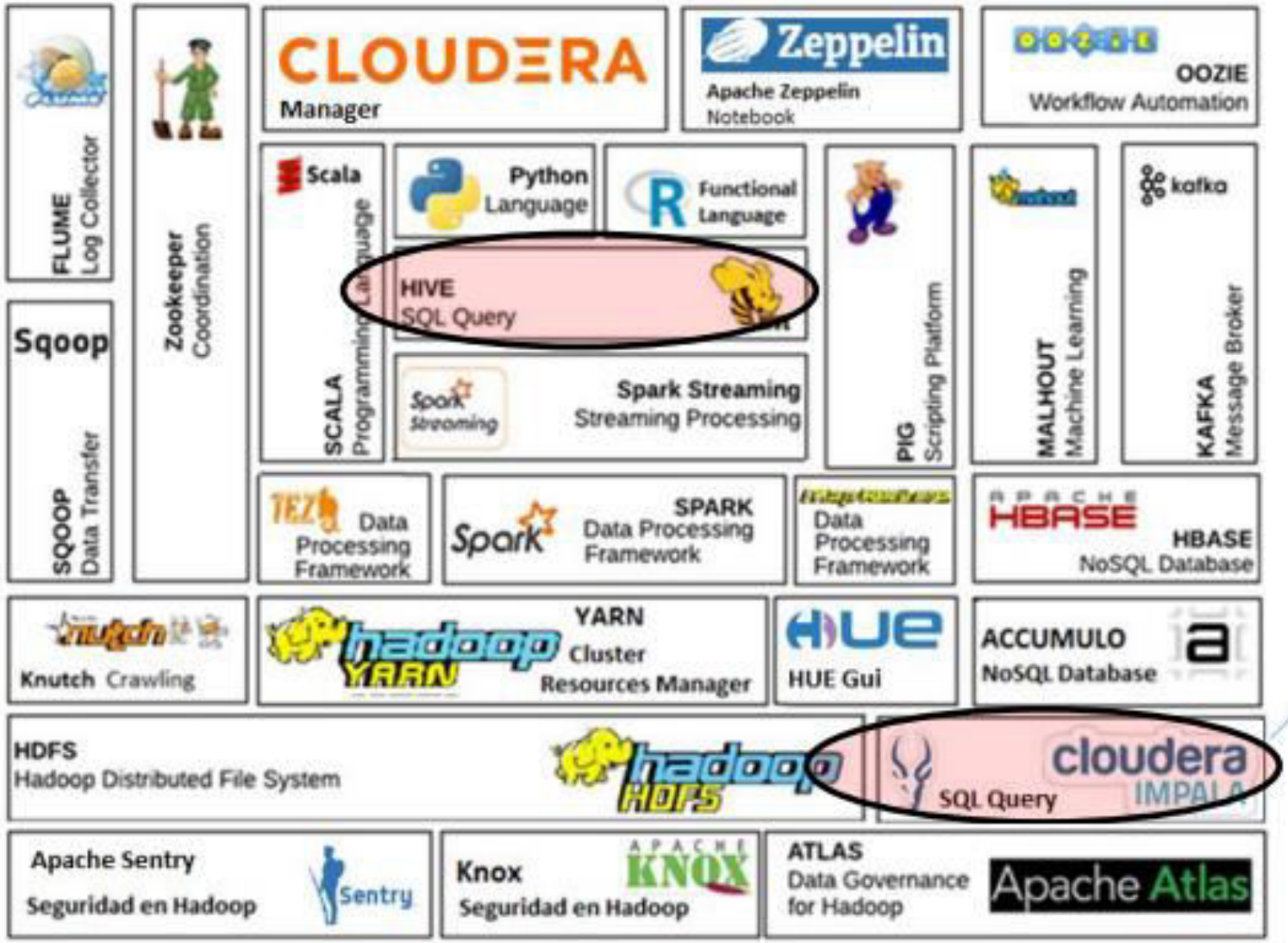
Es el framework de computación distribuido más popular para desarrollar aplicaciones paralelas y tolerantes a fallos. Se basa en el modelo MapReduce y lo extiende con capacidades de streaming y de consultas interactivas. Incluye librerías de machine learning como MLlib y de streaming y soporta los lenguajes de programación Java, Scala, Python y R.

Tez

Framework de procesamiento distribuido (en clusters) que modela su procesamiento en función a grafos de flujo de datos (DAG – directed acyclic graph).
Arquitectura Customizable diseñada para extensibilidad y optimizaciones de performance definidos por los usuarios.

CePETel

Ecosistema Hadoop - Motores de SQL



CePETel

Ecosistema Hadoop - Motores de SQL

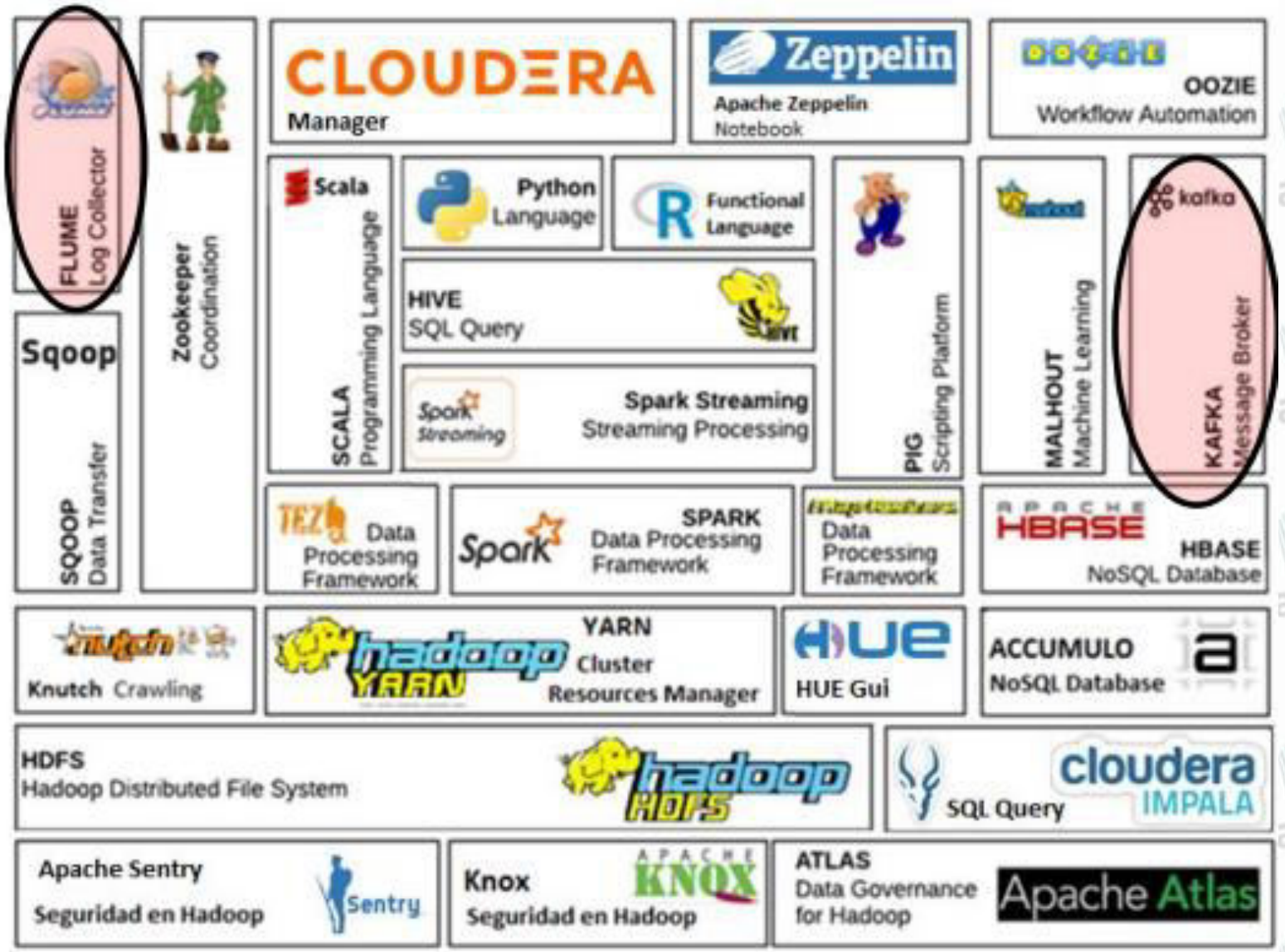
Apache Hive

Permite realizar consultas sobre los datos almacenados en HDFS mediante el lenguaje HQL (Hive Query Language), muy similar a SQL. Compone la base de un Data Warehouse con gran escalabilidad.

Apache Impala

Es un motor de consultas SQL para Hadoop inicialmente desarrollado por Cloudera. Permite realizar consultas interactivas de baja latencia sobre datos almacenados en HDFS sin la necesidad de movimiento de datos. Es muy usada en consultas analíticas y Business Intelligence.

Ecosistema Hadoop - Near Real Time Ingest



CePETel

Ecosistema Hadoop - Near Real Time Ingest

Apache Kafka

Kafka es un sistema de intermediación de mensajes basado en el modelo publicador/subscriptor en el que varios productores y subscriptores pueden leer y escribir. Se ha convertido en una plataforma de streaming de eventos distribuida y eje central de muchas arquitecturas Big Data.

Apache Flume

Es un servicio de recolección de datos distribuido ● Escalable ● Configurable ● Extensible ● Administrable ● Open Source

Es una solución para recolectar datos en todos los formatos.

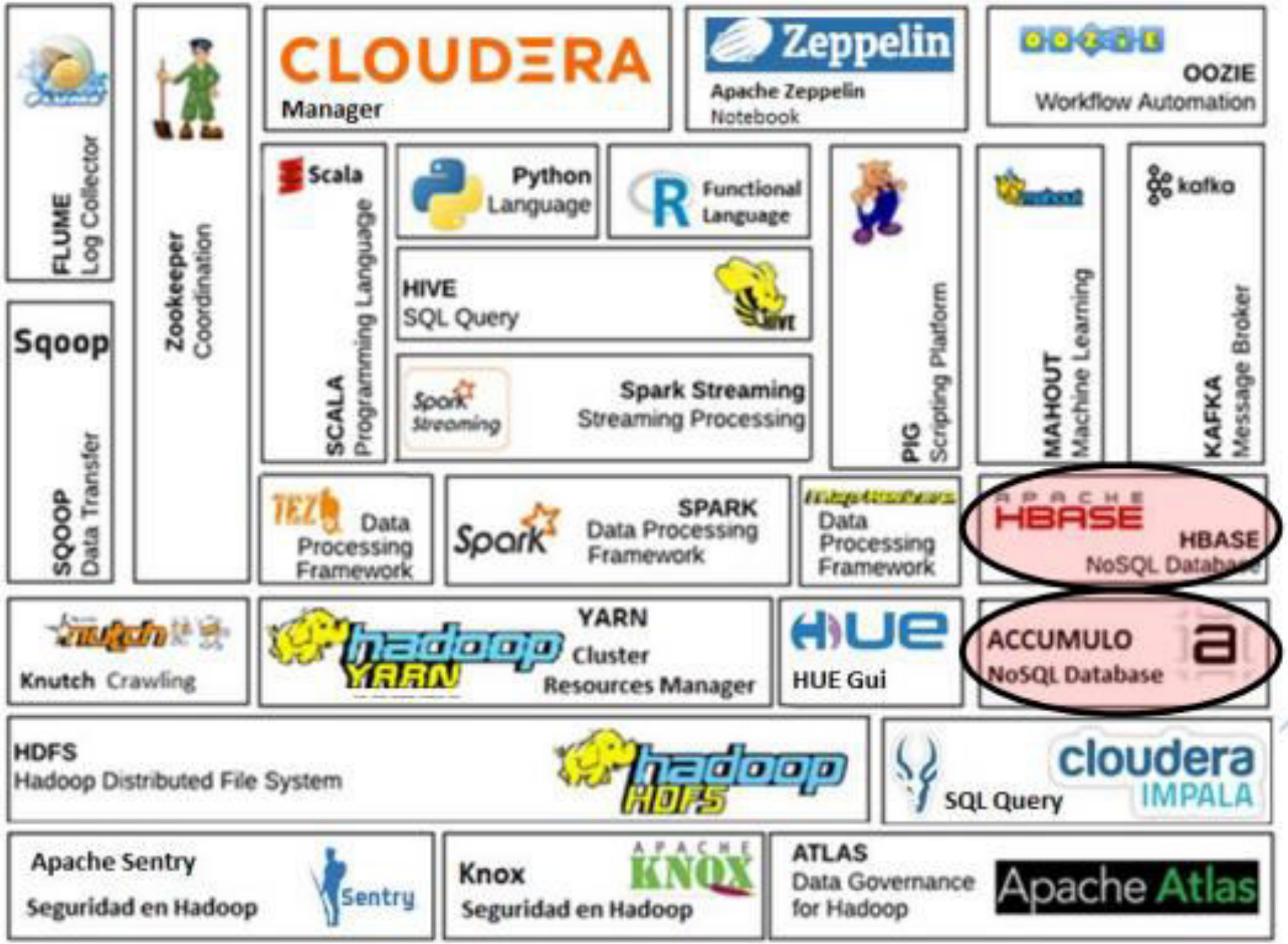
Apache NiFi

Es una plataforma de logística de datos integrados en tiempo real y de procesamiento de eventos sencillos, que automatiza el movimiento de datos entre fuentes y sistemas de datos diversos, ingestando datos de forma rápida, sencilla y segura. ”

No forma parte del ecosistema Hadoop

CePETel

Ecosistema Hadoop – No SQL Databases



CePETel

Ecosistema Hadoop – NoSQL Databases

Apache HBase

Es una base de datos no relacional, columnar y distribuida creada sobre el sistema de ficheros de Hadoop (HDFS) que puede escalar horizontalmente.

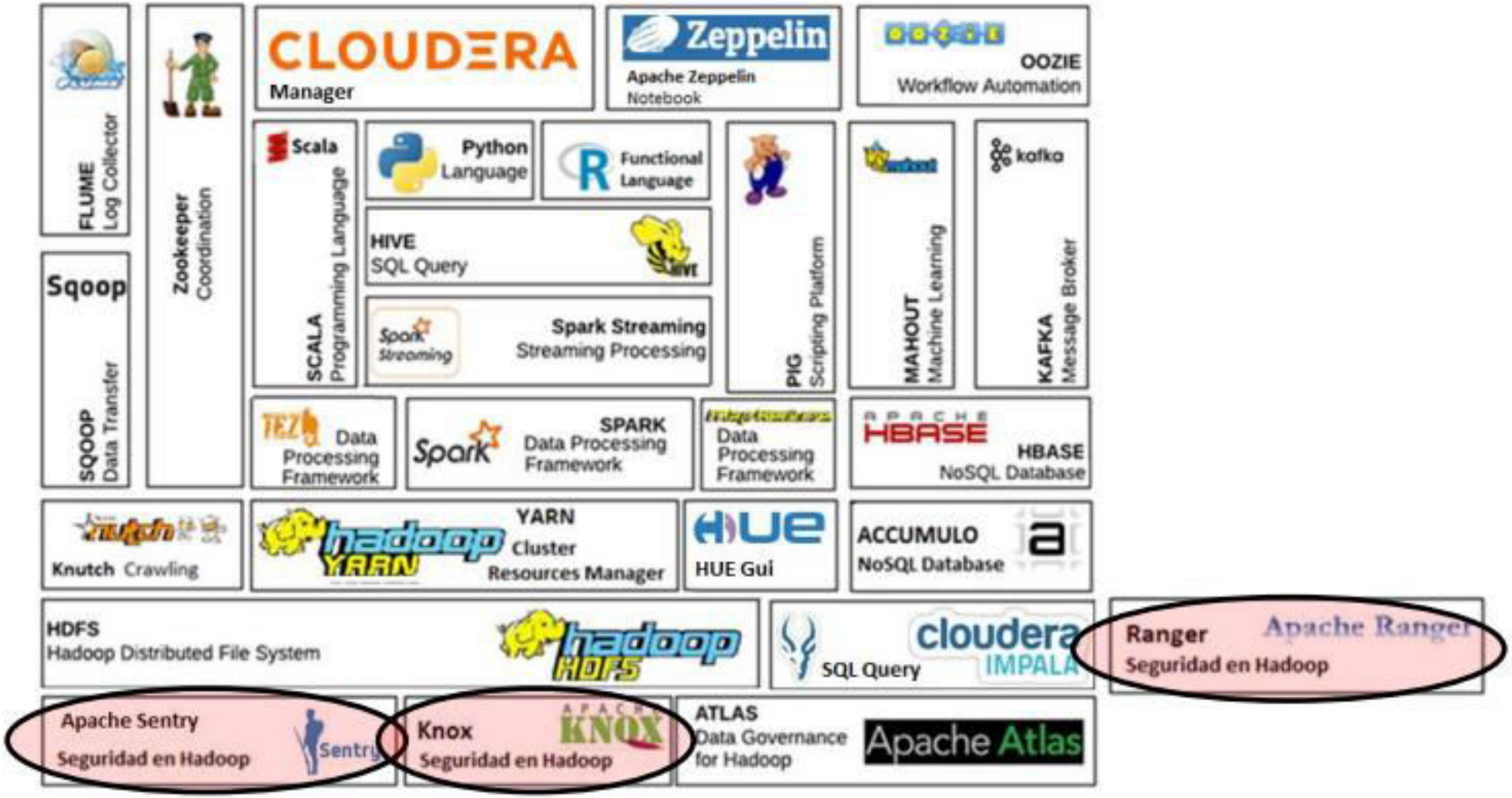
HBase utiliza un modelo de datos muy similar al de Google Big Table diseñado para proporcionar acceso aleatorio a una gran cantidad de datos estructurados. Tiene un modelo tolerante a fallos para almacenar columnas dispersas, muy comunes en big data. Está escrito en Java.

Apache Accumulo

Base de datos

- Del tipo Clave/Valor.
- Distribuida, escalable y de alto rendimiento.
- Basada sobre la especificación de Big Table.
- Diseñada para operar sobre el file system distribuido de Hadoop (HDFS) aprovechando sus características de escalabilidad, tolerancia a fallas, y alta disponibilidad.
- Construida sobre Hadoop y Zookeeper.
- Escrita en Java

Ecosistema Hadoop – Seguridad



CePETel

Ecosistema Hadoop – Seguridad

Apache Sentry

Es la pieza del ecosistema que se encarga de aplicar las políticas de autorización sobre los componentes del clúster y sobre los datos y metadatos de Hadoop. Permite controlar los privilegios de cada usuario y aplicación del sistema que usan los componentes de Hadoop de forma modular.

Apache Ranger

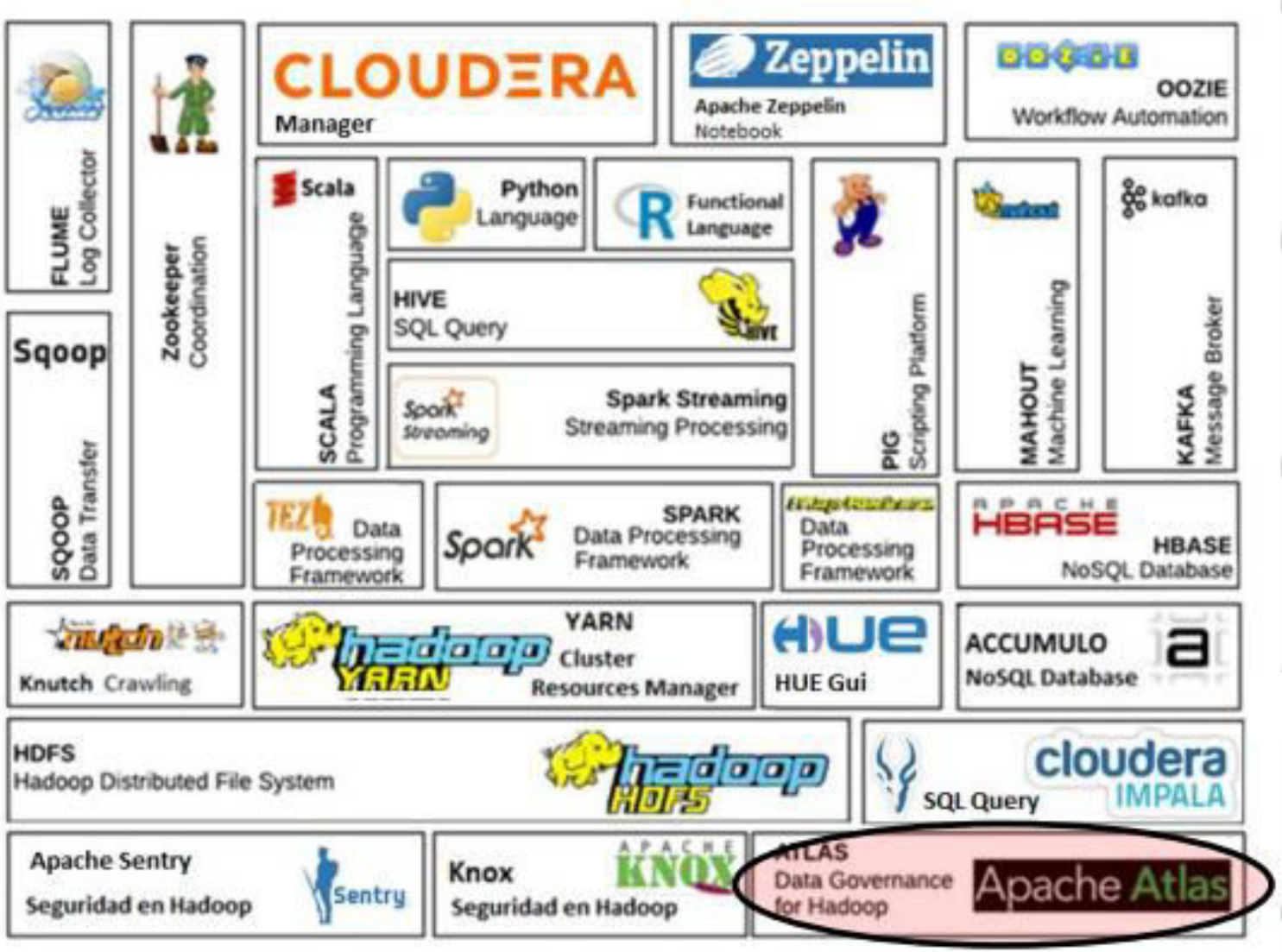
Proporciona una interfaz para gestionar y monitorizar la seguridad de los datos y de los servicios y componentes de la plataforma.

Apache Knox

Actúa como puerta de enlace de las aplicaciones. Proporciona un único punto de autenticación y acceso para las peticiones REST y HTTP hacia los servicios del clúster.

CePETel

Ecosistema Hadoop – Data Governance



CePETel

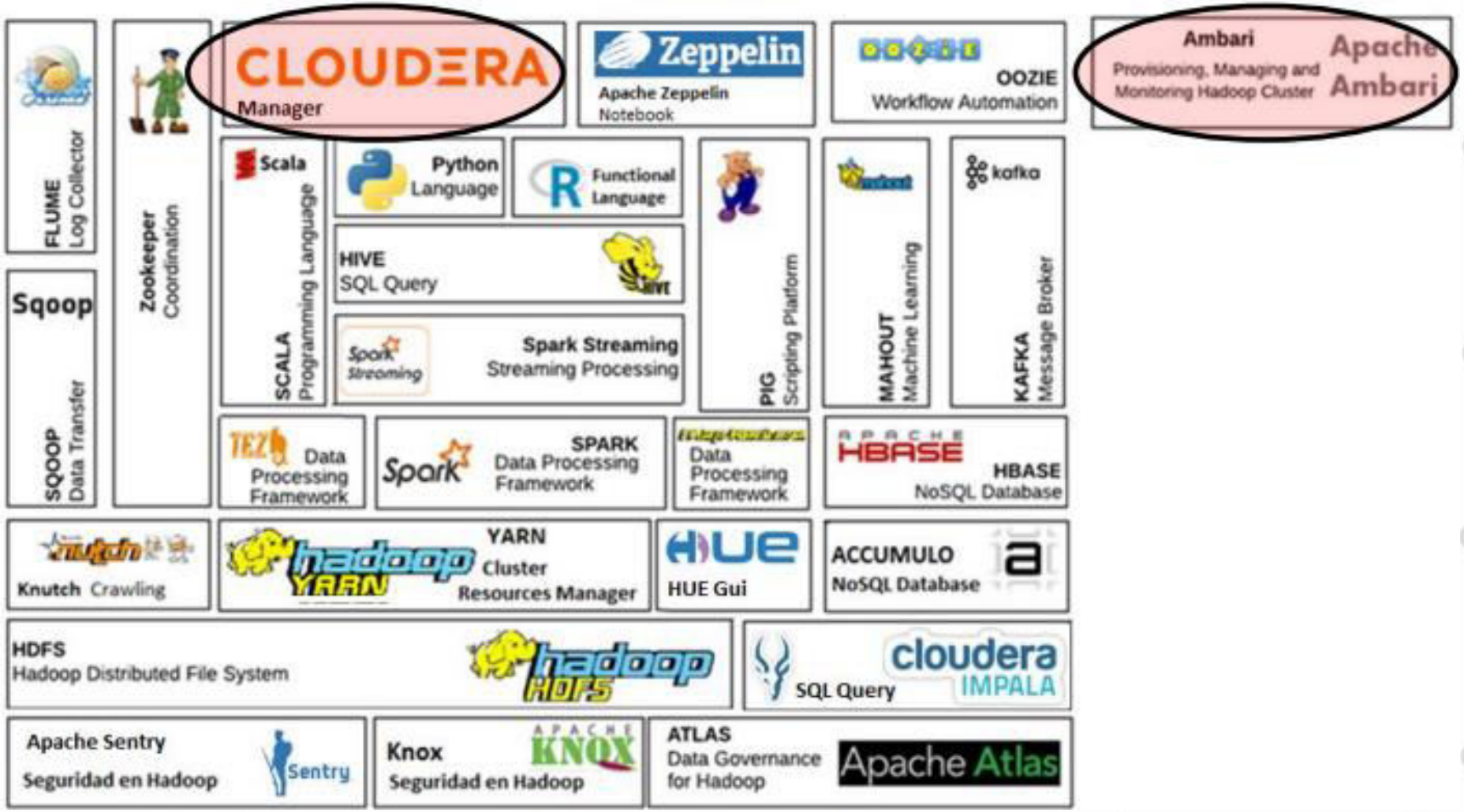
Ecosistema Hadoop – Data Governance

Apache Atlas

- Proporciona las capacidades de gobierno del del dato para cumplir los requisitos regulatorios en el data lake. Ofrece gestión de metadatos, clasificación y catálogos para los elementos.
- Provee capacidades de Data Governance para Hadoop.
- Permite intercambiar metadata con otras herramientas o procesos.
- Control de acceso a los datos y metadata.
- Intercambio de Metadata con otras herramientas que gestionan Metadata.
- Mantiene una historia del origen de los datos, su linaje.
- Es escalable y extensible.
- Permite la clasificación de datos.
- Auditoría Centralizada

CePETel

Ecosistema Hadoop – Management and Monitoring



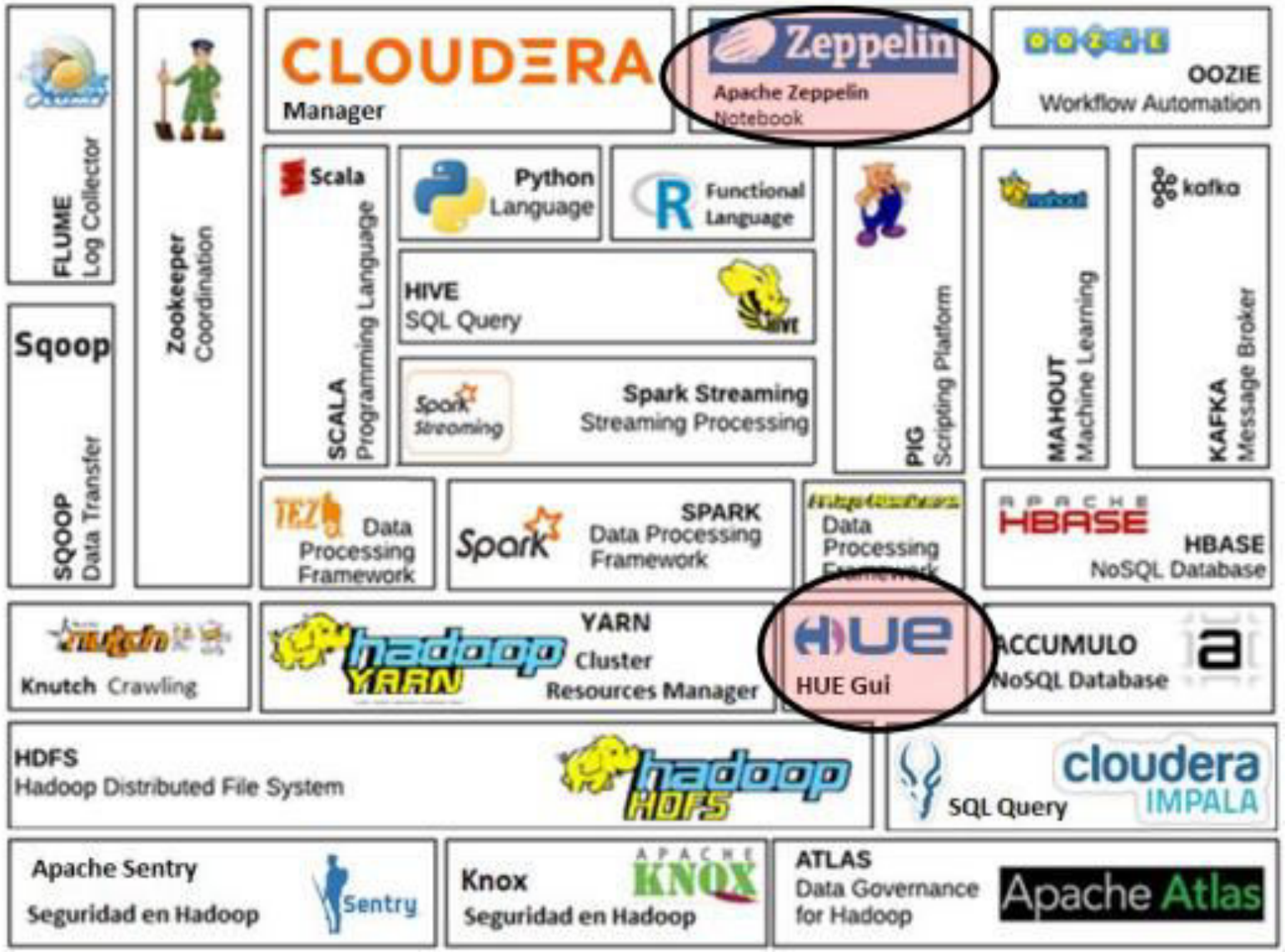
CePETel

Ecosistema Hadoop – Management and Monitoring

Ofrece una interfaz web intuitiva y fácil de usar para la gestión de Hadoop y además proporciona una API REST.

- Administrar y provisionar el cluster Hadoop.
- Un asistente paso a paso para la instalación de servicios de Hadoop a través de múltiples equipos.
- Proporciona la forma de gestionar gestión central para iniciar, detener y volver a configurar los servicios de Hadoop en todo el clúster.
- Monitoriza el clúster Hadoop.
- Ofrece un panel de control para vigilancia de la salud y el estado del cluster Hadoop.
- Se encarga de la instalación de los paquetes de Hadoop en el clúster.
- Ambari aprovecha Nagios para el sistema de alerta y enviará mensajes de correo electrónico cuando se requiere su atención.

Ecosistema Hadoop – Graphic User Interfaces (GUI)



CePETel

Ecosistema Hadoop – Graphic User Interfaces (GUI)

He (Hadoop User Experience)

Hue es la interfaz web para la gestión de Hadoop. Permite crear tablas en Hive, realizar consultas, navegar el sistema de ficheros, cambiar permisos y propietarios. También puede diseñar jobs de MapReduce y conocer su estado. Además permite realizar la gestión por parte de los administradores de las cuentas de usuario.

Apache Zeppelin

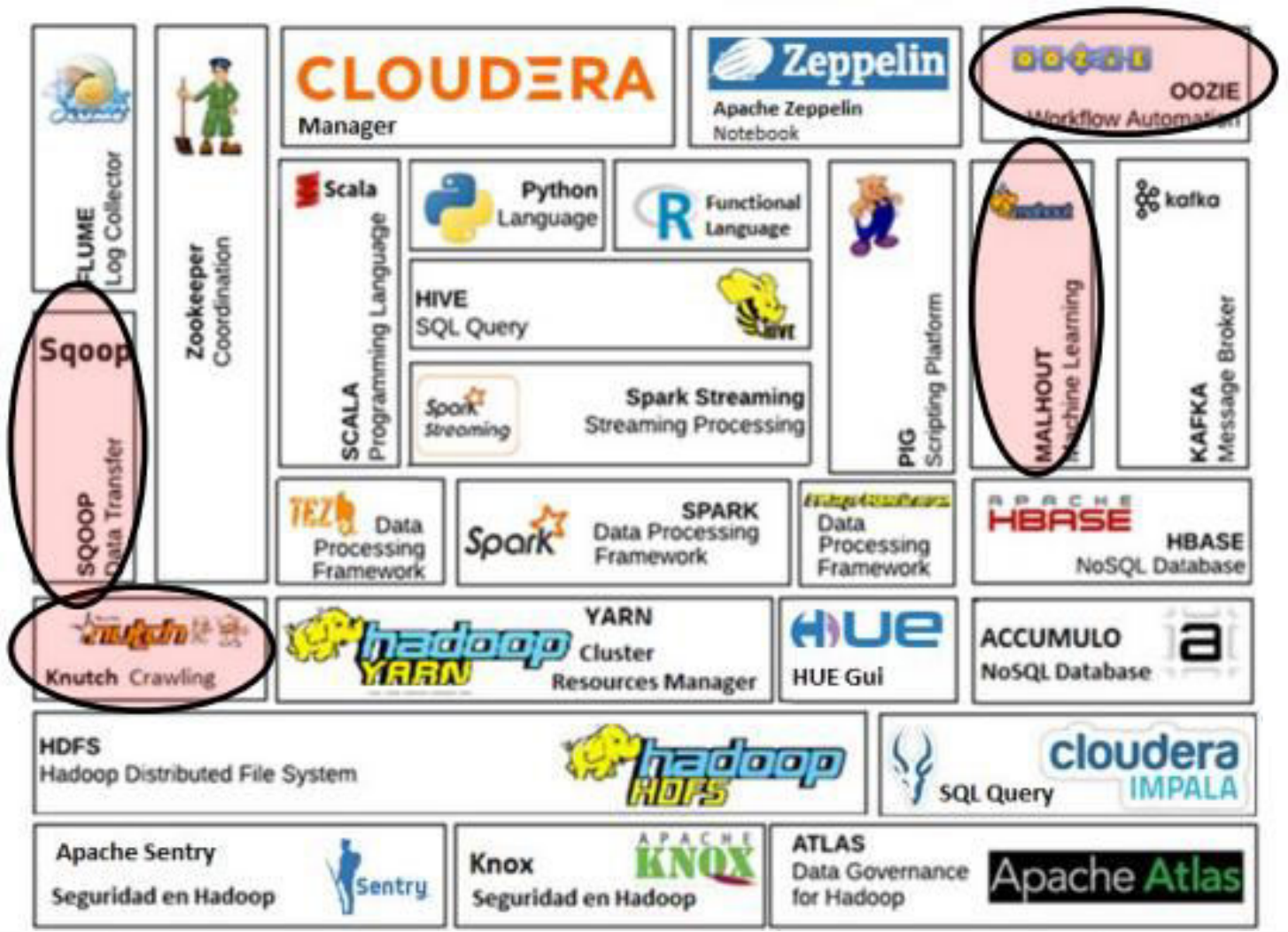
Es una notebook multipropósito que puede ser utilizado en procesos de:

● Data Ingest, ● Data Disvovery, ● Data Analytics, ● Data Visualization

A través de Zeppelin se podrá acceder a diferentes aplicaciones simulando línea de comando, ideal para desarrollar scripts con ejecuciones intermedias antes de su paso a producción.
Soporta la ejecución de scripts y bloques anónimos de código gracias a distintos intérpretes que tiene implementados. Entre los intérpretes más comunes podemos encontrar: Scala sobre Apache Spark, Python sobre Apache Spark, Spark SQL, R, JDBC, Markdown, Shell, hdfs, mongoDB, cassandra.

CePETel

Ecosistema Hadoop – Otras Funcionalidades



CePETel

Ecosistema Hadoop – Otras Funcionalidades

Apache Sqoop (SQL to Hadoop)

Es una herramienta diseñada para transferir datos entre Hadoop y bases de datos relacionales. Al igual que Flume, es una herramienta de ingesta de datos para Hadoop, aunque Sqoop se caracteriza por poder importar y exportar datos estructurados.
Permite importar tablas individuales o bases de datos enteras a HDFS de una manera sencilla y eficiente. Con Sqoop, también es posible importar datos desde bases de datos relacionales directamente a tablas Hive. Cuando ejecutamos un comando en Sqoop, la tarea se divide en subtareas, que realizan la operación Map de forma distribuida y paralela.

Apache Oozie

Oozie es el planificador de workflows para administrar trabajos de Hadoop. Gestiona los trabajos y permite tratarlos como una sola unidad lógica. Permite agregar dos tipos de trabajos: workflow y coordinator. Los trabajos de tipo workflow se componen de una secuencia de acciones que deben ser ejecutadas en serie. Los trabajos Oozie coordinator son ejecutados cuando se cumple la condición de que los datos necesarios para la tarea estén disponibles.

Ecosistema Hadoop – Otras Funcionalidades

Apache Nutch

Es un motor de búsquedas web open source

- Funcionalidades:
  - Internet e Intranet crawling
  - Parsea diferentes formatos de documentos (PDF, HTML, XML, JS, DOC, PPT, etc)
  - Tiene una interfaz web para consultar el índice
  - Tiene gestión de recrawls

Apache Mahout

Mahout proporciona el entorno para crear aplicaciones escalables de Machine Learning. Se compone de librerías específicas escritas en Java y optimizadas para funcionar sobre Hadoop. Entre sus funcionalidades, se incluyen el filtrado colaborativo, clustering y clasificación.

Ecosistema Hadoop - Otras Funcionalidades

Apache Pig

Pig es la plataforma de scripting para Hadoop, originalmente desarrollada en Yahoo. Proporciona la base para implementar flujos de datos, ETLs y procesamiento distribuido. Tiene dos componentes: Pig Latin y Pig Runtime, el entorno de ejecución.

Provee de un lenguaje de alto nivel llamado Pig Latin para crear flujos de datos que permite escribir programas MapReduce de forma simple y en pocas líneas de código, con una sintaxis similar a SQL. El compilador interno se encarga de convertir Pig Latin en una secuencia de programas MapReduce.

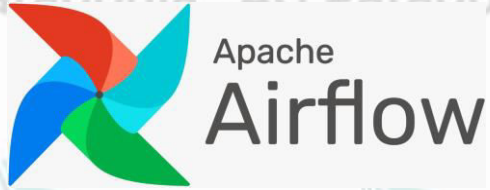
Apache Zookeeper

Servicio centralizado que permite mantener conexiones estables entre servidores con distintas tecnologías. Actúa de coordinador de servicios big data y trabajos Hadoop. Zookeeper se usa principalmente para mantener aplicaciones distribuidas funcionando de forma correcta. También gestiona algunas configuraciones y permite el consenso en los sistemas.

Provee una interfaz simple para mantener la consistencia de datos.

Otras Funcionalidades de Apache

Apache Airflow



Es una herramienta basada en Python de tipo workflow manager usada como orquestador de servicios: gestionar, monitorizar y planificar flujos de trabajo.

Airflow se usa para automatizar trabajos programáticamente dividiéndolos en subtareas. Permite su planificación y monitorización desde una herramienta centralizada. Los casos de uso más comunes son la automatización de ingestas de datos, acciones de mantenimiento periódicas y tareas de administración. Para ello, permite planificar trabajos como un cron y también ejecutarlos bajo demanda.

En Airflow, se trabaja con **DAGs** (Directed Acyclic Graphs). Son colecciones de tareas o de trabajos a ejecutar conectados mediante relaciones y dependencias. Son la representación de los workflows.

Kudu

Tecnología de almacenamiento columnar distribuido desarrollado para Hadoop. Está enfocado a almacenar datos estructurados con acceso aleatorio de baja latencia. Es un motor orientado a conectar HDFS y HBase como base de datos NoSQL.

CePETel

Anexo

Recomendaciones para crear un Data Lake

	Most Important Use Group & Use-Cases	Time-to-Market Questions & Solutions	Cost Implementation & Ownership	Users (# & Types)	Data Growth Volume & Variety
Data Lake	Predictive & Advanced Analytics	Weeks - Months	$$$$$		
Data Warehouse	Multi-Purpose Enabler of Operational & Performance Analytics	Hours - Days	$$$$$		
Data Mart	Line of Business Specific Reporting & Analytics	Minutes - Hours	$$$$$		

CePETel

Flujo de los datos en un proceso de Data Lake

CePETel

TP N° 6: Data Lake

- Identificar si las siguientes características corresponden a Data Warehouse o Data Lake (ver siguiente slide)

Éxitos!!

CePETel

Característica	Data Lake	Data Warehouse
Datos procesados y tabulares		
Schema al momento de leer los datos		
Barato para grandes volúmenes de datos		
Seguridad completa y robusta		
Desarrollo ágil		
Procesos ELT		
Guardar historia completa al máximo nivel de detalle		
Usado por analista de negocios y Data Scientist		
Procesos ETL complejos y costosos, en general con herramientas propietarias.		
Los datos se enriquecen, validan y transforman una vez dentro del repositorio		
Consultas mas interactivas y veloces con menores tiempos de respuesta		
Procesamiento de varios millones de datos a bajo costo		
Consultable mediante herramientas de explotación y/o SQL		

TP N° 7: Data Lake

1. Verdadero/Falso: Una característica importante de YARN es la posibilidad de distribuir el procesamiento de tareas de los procesos en cada uno de los nodos.
2. Seleccione las características de HDFS
    a. Recomendable para almacenar grandes archivos del orden de gigabytes de datos.
    b. Recomendable para almacenar pequeños y cientos de archivos de datos.
    c. Distribuir datos en múltiples nodos.
    d. Replicar archivos en múltiples nodos.
    e. Priorizar performance para traer archivos enteros antes que para traer sólo una línea de un archivo.
3. Verdadero/Falso: Apache Ambari y HUE son interfaces gráficas de usuario que cumplen el mismo propósito.

CePETel

TP N° 7: Data Lake (continuación)

4. Seleccione los componentes orientados a la seguridad dentro de Hadoop
    a. Apache Zeppelin
    b. Apache Ranger
    c. Apache Nutch
    d. Apache Knox
    e. Apache Sentry
5. Apache Sqoop permite importar datos desde
    a. Bases de datos NoSQL
    b. Archivos planos / csv en múltiples nodos.
    c. Bases de datos transaccionales
    d. Enterprise Data Warehouse
    e. Archivos comprimidos .zip/.rar

CePETel

CePETel

CePETel
Sindicato de los Profesionales
de las Telecomunicaciones

SECRETARÍA TÉCNICA

IPEI
Instituto Profesional de Estudios e Investigación

BIG DATA & ANALYTICS II

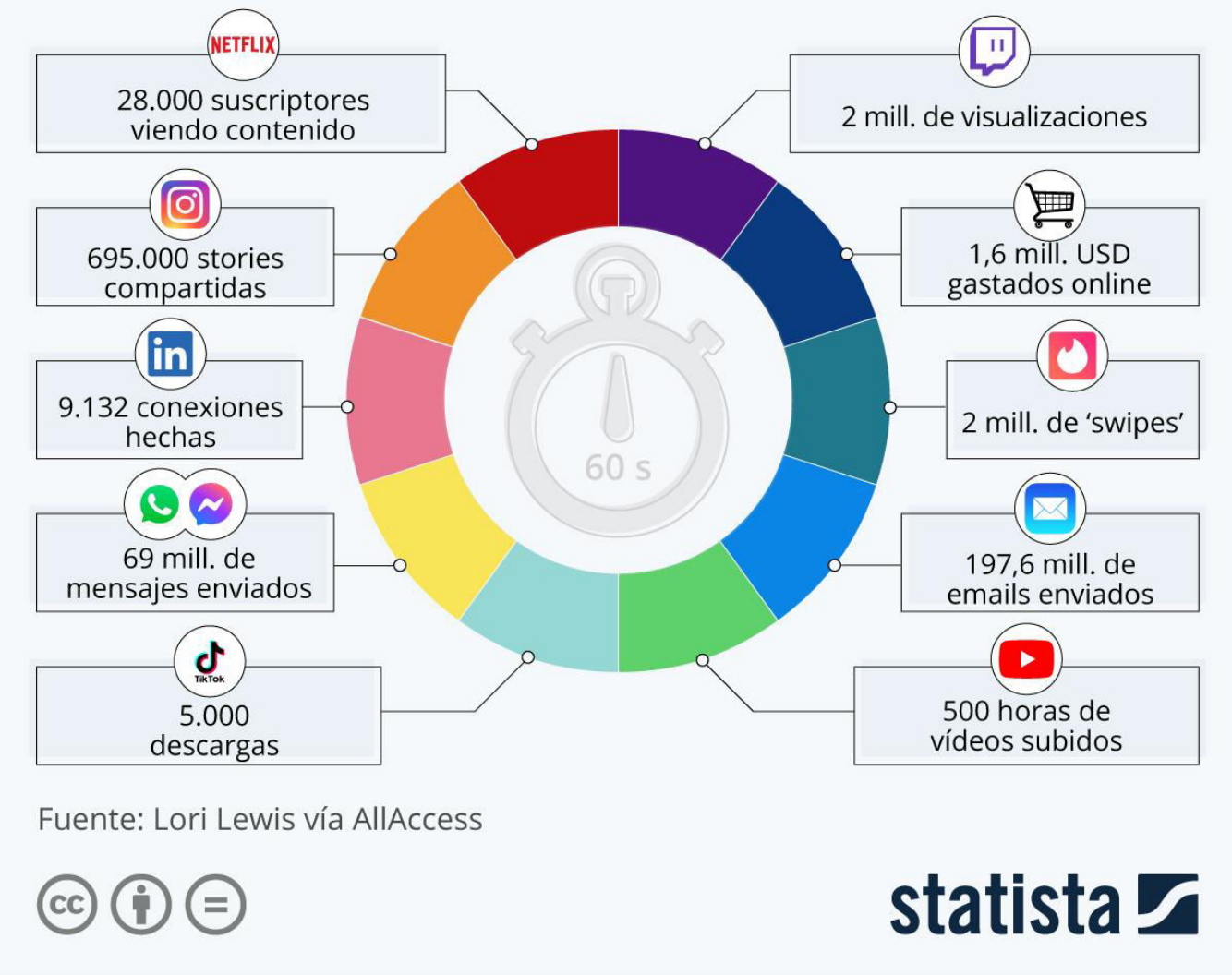
TEMARIO
Resumen Curso Big Data & Analytics I

Disertantes: Lic. Maria Trinidad Aquino – Ing. Raúl Alejandro Grassi

Los datos nunca duermen!

El mundo pasa una media de [6 horas y 54 minutos al día en Internet](), pero, en sólo un minuto, ¿cuánta actividad digital se genera? 60 segundos pasan rápidamente, pero en este lapso de tiempo suceden un sinfín de cosas en el mundo online.

Esto sucede en internet en 1 minuto



CRECIMIENTO EXPLOSIVO DE DATOS VS. COSTO DE ALMACENAMIENTO

Big Data

Una Primer Definición

"***Volumen masivo de datos***, tanto ***estructurados como no-estructurados***, los cuales son ***demasiado grandes y difíciles de procesar*** con las bases de datos y el software ***tradicionales***."
(ONU, 2012)

Big Data no es sólo un repositorio de datos: además de almacenar los datos, se aplican procesos para tratarlos y analizarlos.

Big Data no es únicamente una herramienta de analítica: es capaz de correlacionar datos procedentes de distintas fuentes y en diferentes formatos.

Big Data es más que una plataforma de gestión de datos: es capaz de trabajar con datos en tiempo real.

Big Data

Es el sector de IT que hace referencia a la gestión y análisis de ***grandes conjuntos de datos*** que por la ***velocidad*** a la que se generan, la capacidad para tratarlos y los ***múltiples formatos y fuentes***, es necesario procesarlos con mecanismos distintos a los tradicionales.

CADA DÍA CREAMOS 2,5 QUINTILLONES DE BYTES DE DATOS. (2,5 Exabytes)

EL 90% DE LOS DATOS DEL MUNDO DE HOY SE GENERARON EN LOS ÚLTIMOS 2 AÑOS

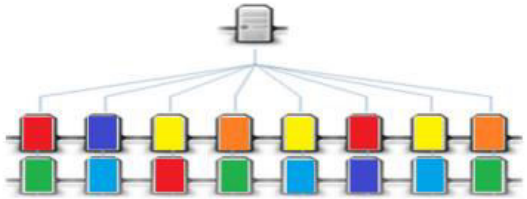
Implementando BIG DATA

Condiciones fundamentales

ESCALAMIENTO HORIZONTAL

ALTA DISPONIBILIDAD

DISTRIBUCIÓN DE DATOS



PROCESAMIENTO DISTRIBUIDO

SQL vs. NoSQL

Bases de datos relacional o SQL

¿Qué es un sistema de base de datos relacional?

El modelo relacional es una forma intuitiva y directa de representar datos. Las bases de datos relacionales **son el modelo más utilizado actualmente**. Su foco está en la ejecución de **transacciones.**

Una base de datos relacional es, en esencia, un conjunto de tablas (o relaciones) formadas por filas (registros) y columnas (campos); así, cada registro (cada fila) tiene una ID única, denominada clave y las columnas de la tabla contienen los atributos de los datos.

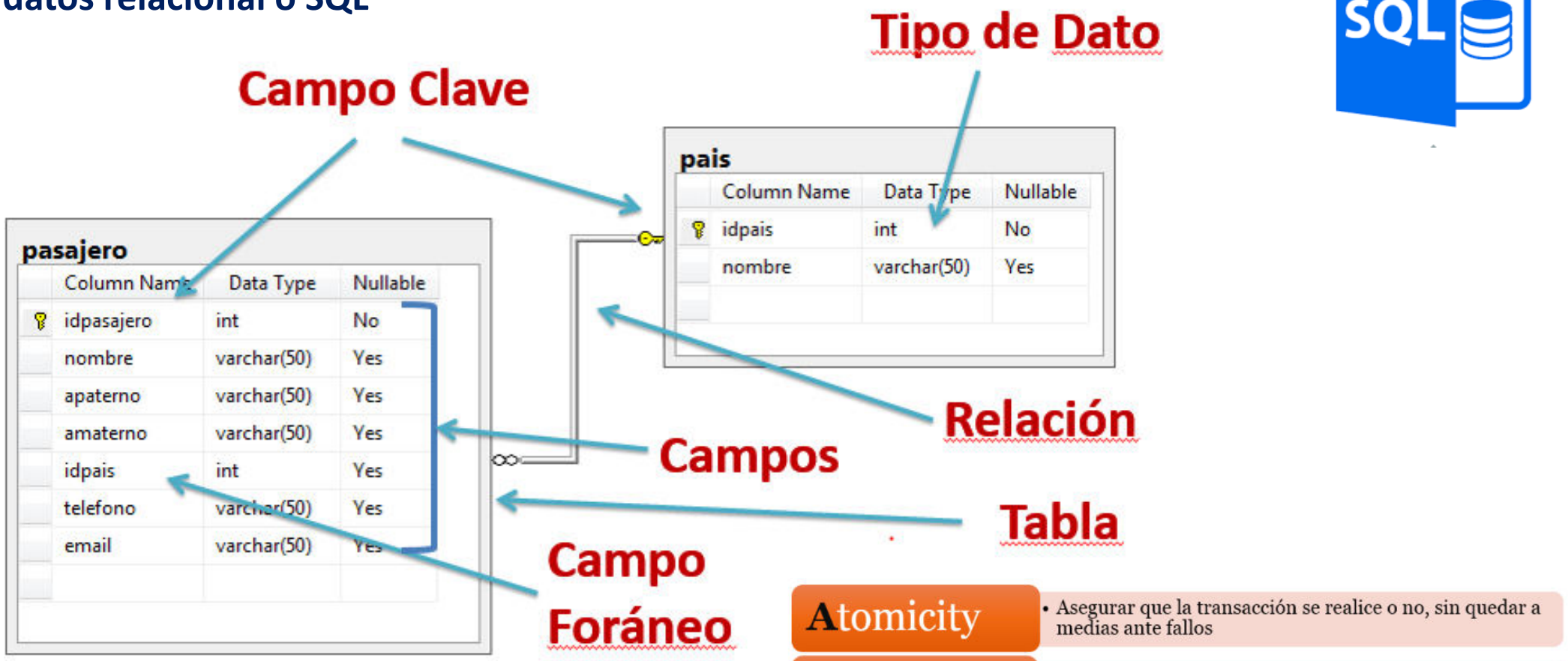
Una de las principales características de la base de datos relacional es evitar la duplicidad de registros y a su vez garantizar la integridad referencial, es decir, que si se elimina uno de los registros, la integridad de los registros restantes no será afectada. Además, gracias a las claves se puede acceder de forma sencilla a la información y recuperarla en cualquier momento.

Así, una tabla con los datos de los empleados de una empresa podría verse así:

Empleados							
ID_e	1º Apellido	2º Apellido	Nombre	Nº SS	Calle	CP	Municipio
1							
2							
3							
4							

SQL vs. NoSQL

Bases de datos relacional o SQL



Atomicity	• Asegurar que la transacción se realice o no, sin quedar a medias ante fallos
Consistency	• Asegurar el estado de validez de los datos en todo momento
Isolation	• Asegurar independencia entre transacciones
Durability	• Asegurar la persitencia de la transacción ante cualquier fallo

SQL vs. NoSQL

Bases de datos relacional o SQL

Definición de RDBMS

Para poder almacenar, administrar, consultar y recuperar los datos guardado en la base de datos relacional es necesario emplear un software específico, **denominado sistema de gestión de bases de datos relacionales (RDBMS) o Motor de Base de Datos**. Este software proporciona una interfaz entre los usuarios y/o las aplicaciones y la base de datos, además de contar con funciones administrativas para gestionar el acceso, almacenamiento y rendimiento.

El DBMS ofrece a los usuarios una **percepción de la base de datos** que está **por encima del nivel del hardware** y que **interpreta y ejecuta todos los comandos SQL** que le son enviados.

Entre los **motores de Bases de Datos más utilizados** podemos nombrar los siguientes: **Oracle, MS SqlServer, MySQL, PostreSQL, DB2, SQL Lite**, entre otros.

Implementando BIG DATA

BASES DE DATOS NoSQL

¿Qué es NoSQL?

Sistemas de gestión de bases de datos que difieren del modelo clásico de bases de datos relacionales: no usan SQL como lenguaje de consulta, los datos almacenados no requieren estructuras fijas como tablas, no garantizan consistencia plena y escalan horizontalmente.

Not Only SQL

Surgieron para complementar a las bases de datos tradicionales, no para reemplazarlas.

Implementando BIG DATA

BASES DE DATOS NoSQL

Ventajas y Desventajas

Feature	NoSQL Databases	Relational Databases
Performance	High	Low
Reliability	Poor	Good
Availability	Good	Good
Consistency	Poor	Good
Data Storage	Optimized for huge data	Medium sized to large
Scalability	High	High (but more expensive)