Introducción a Big Data

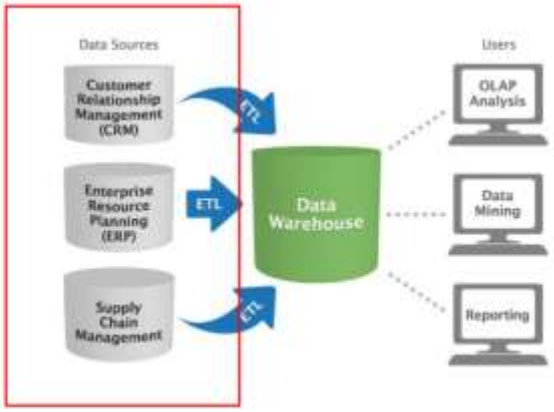
Fundación Telefónica Movistar

CURSO DE INTRODUCCIÓN A BIG DATA.

Fuentes y tipos de datos.

Data Warehouse (DW)

*"Es un repositorio de datos **integrado, no volátil, variable en el tiempo, orientado al negocio**, organizado de forma tal que facilita el análisis de grandes volúmenes de datos para la toma de decisiones".*



Fuentes de Datos

Fuentes Internas

- ERP
- CRM
- SCM
- MES
- BPM
- Sistemas Legacy
- Planillas de Cálculo
- Stream Events (Sensores, medidores)

Fuentes Externas a la Organización

- Redes Sociales
- Datos de Cámaras empresariales
- Open Data - Gobierno
- Web Crawling Data

Formatos

- Bases de Datos Relacionales
- Bases de Datos NoSQL
- Archivos (CSV, AVRO, JSON, XML, XLS, VSAM, comprimidos -zip, zlib, bzip-)

Fuentes Externas

Fuentes Internas

ERP - Enterprise Resource Planning

ERP refiere a *Enterprise Resource Planning*, que significa "sistema de planificación de recursos empresariales".

Estos programas se hacen cargo de distintas operaciones internas de una empresa, desde producción a distribución o incluso recursos humanos.

Las principales ventajas de estos sistemas son:

- Automatización de procesos de la empresa.
- Disponibilidad de la información de la empresa en una misma plataforma.
- Integración de las distintas bases de datos de una compañía en un solo programa.
- Ahorro de tiempo y costes.

Algunos de los ERP más reconocidos: SAP, JD EDwards, Sage, MIcrosoft Dynamics ERP.

Fuentes Internas

CRM - Customer Relationship Management

Las siglas CRM o Customer Relationship Management hacen referencia a un software que permite a las empresas rastrear cada interacción con los clientes, tanto actuales como futuros.

El objetivo de implementar un CRM es crear un sistema que sus empresas (por lo general, los equipos de ventas y de marketing) puedan usar para interactuar de manera más eficaz y efectiva con los clientes potenciales y actuales.

Entre los CRMs más conocidos encontramos, T3 CRM, Salesforce, Microsoft Dynamics CRM, Oracle CRM On Demand.

Fuentes Internas

SCM - Supply Chain Management

Las siglas **SCM** (gestión de la cadena de suministro, del inglés **Supply Chain Management**) se refiere a las herramientas y métodos cuyo propósito es mejorar y automatizar el suministro a través de la reducción de los stocks y los plazos de entrega.
Incluye la planificación de las actividades de **suministro, fabricación y distribución de cada producto**. Incluye la oferta y demanda dentro y fuera de la empresa.
Algunos de los SCM más reconocidos, Oracle Logistics Solution, SAP SCM, Microsoft Supply Chain Management.

Fuentes Internas

MES - Manufacturing Execution System

Es un sistema enfocado al Control de la Producción, que monitoriza y documenta la gestión de la planta.

El propósito último de un Sistema Mes es aumentar la Eficiencia de la Planta de Producción:

- Reduciendo Costes
- Mejorando la Productividad
- Aumentando la Trazabilidad y la Calidad entregada a tu cliente.

Algunos de los MES más reconocidos, SAP Manufacturing Execution, Oracle Manufacturing, Microsoft Dynamics Inventory Management.

Fuentes Internas

BPM - Business Process Management

Los BPMs son un software empresarial que permite a las empresas modelizar, implementar y ejecutar conjuntos de actividades interrelacionadas –es decir, Procesos– de cualquier naturaleza, ya sea dentro de un departamento o a través de toda la organización.

Cuentan con extensiones para incluir a los clientes, proveedores y otros agentes como participantes en las tareas de los procesos.

Algunos BPMs de Mercado, JBPM, Microsoft BPM, Oracle BPM, SAP BPM, RedHat JBoss BPM.

Fuentes de Datos

Formatos

Texto / CSV

Comúnmente usados para **intercambiar** datos **Sistemas**.

Legibles y parseables.

No soportan compresión de bloques.

No almacenar header ni footer (**no metadata**). Se debe saber que es cada campo.

La estructura depende del orden de los campos. Nuevos campos deben ser agregados al final y los existentes no pueden borrarse. **Soporte limitado para evolución de esquema**.

JSON

- **JSON** - Java Script Object Notation.
- Generalmente utilizados como entradas o salidas para **API Rest**.
- Un JSON por cada línea.
- Almacena la **metadata junto con los datos, permitiendo la evolución del esquema**.
- **No soportan compresión** de bloques.
- Muy utilizado por Bases de Datos NoSQL como **MongoDB**.

Fuentes de Datos

Formatos

AVRO

Formato de almacenamiento multipropósito.

Almacena la **metadata junto con los datos**. También **permite** especificar un **esquema independiente en la lectura** del archivo. Esto lo hace el ejemplo perfecto de evolución de esquema, ya que se pueden agregar, renombrar, eliminar y cambiar el tipo de dato de los campos del archivo definiendo un nuevo esquema independiente.

Soportan compresión de bloques.

XML

- XML - **Extensible Markup Language**.
- Alcena la **metadata junto con los datos**.
- Es un **lenguaje multiplataforma** diseñado para almacenar varios tipos de datos.
- Son fácilmente modificables.
- **No soportan compresión** de bloques.
- En las Api/Rest están siendo reemplazados por archivos **JSON** cómo estándard.

Tipos de Datos

Tipo Numérico

- **Enteros**
  - **Smallint**
  - **Integer**
  - **Bigint**
- **Precisión arbitraria**
  - **Numeric(p,s)**
  - **Numeric(p)**
  - **Numeric**
- **Punto Flotante**
  - **real**
  - **double precision**
- **Serial / Autoincrementales**
  - **Smallserial**
  - **Serial**
  - **Bigserial**

Tipo Numérico

Nombre	Tamaño	Descripción	Rango
smallint	2 bytes	small-range integer	-32768 to +32767
integer	4 bytes	typical choice for integer	-2147483648 to +2147483647
bigint	8 bytes	large-range integer	-9223372036854775808 to +9223372036854775807
decimal	variable	user-specified precision, exact	up to 131072 digits before the decimal point; up to 16383 digits after the decimal point
numeric	variable	user-specified precision, exact	up to 131072 digits before the decimal point; up to 16383 digits after the decimal point
real	4 bytes	variable-precision, inexact	6 decimal digits precision
double precision	8 bytes	variable-precision, inexact	15 decimal digits precision
smallserial	2 bytes	small autoincrementing integer	1 to 32767
serial	4 bytes	autoincrementing integer	1 to 2147483647
bigserial	8 bytes	large autoincrementing integer	1 to 9223372036854775807

Tipos de Datos

Tipo Caracter

- **Longitud Fija**
  - character(n)
  - char(n)
- **Longitud Variable con límite**
  - character varying(n)
  - varchar(n)
- **Longitud variable sin límite**
  - text

Tipo Monetario

- money(n)

Nombre	Tamaño	Descripción	Rango
money	8 bytes	currency amount	-92233720368547758.08 to +92233720368547758.07

Tipo Fecha/Tiempo

- **Timestamp**
  - timestamp (sin time zone)
  - timestamp (con time zone)
- **Fecha**
  - date
- **Tiempo**
  - time (sin time zone)
  - time (con time zone)
- **Intervalo**
  - Interval

Nombre	Tamaño	Descripción	Valor Min.	Valor Max.	Resolución
timestamp [(*p*)] [without time zone]	8 bytes	both date and time (no time zone)	4713 BC	294276 AD	1 microsecond / 14 digits
timestamp [(*p*)] with time zone	8 bytes	both date and time, with time zone	4713 BC	294276 AD	1 microsecond / 14 digits
date	4 bytes	date (no time of day)	4713 BC	5874897 AD	1 day
time [(*p*)] [without time zone]	8 bytes	time of day (no date)	00:00:00	24:00:00	1 microsecond / 14 digits
time [(*p*)] with time zone	12 bytes	times of day only, with time zone	00:00:00+1559	24:00:00-1559	1 microsecond / 14 digits
interval [*fields*] [(*p*)]	16 bytes	time interval	-178000000 years	178000000 years	1 microsecond / 14 digits

Tipo Booleano

- **Booleano**
  - **boolean**

Nombre	Tamaño	Descripción
boolean	1 byte	state of true or false

Tipo Binario

- **Binarios**
  - **bytea**

Nombre	Tamaño	Descripción
bytea	1 or 4 bytes plus the actual binary string	variable-length binary string

Fuentes de datos.

Definición de Base de Datos

Que es una Base de Datos?

Una BD es un conjunto de datos persistentes e interrelacionados que es utilizado por los sistemas de aplicación de una empresa, los mismos se encuentran almacenados en un conjunto independiente y sin redundancias o con redundancias mínimas.

Un poco de historia

En los últimos 30 años el mundo de la TI experimentó grandes cambios.

- Nuevas Arquitecturas de aplicaciones.
- Nuevos Paradigmas de programación.
- Nuevas herramientas para desarrollo de Software.
- Pero algo permaneció constante....

RDBMS – Relational Data Base Management Systems.

En los 70´s
Edgar Codd
Modelo Relacional
Algebra Relacional

1974
IBM
System R
1er RDBMS
SEQUEL=SQL

1977/79
Lawrence Ellison
SDL luego
Relational Sw Inc.

1980
Relational
Database
Systems

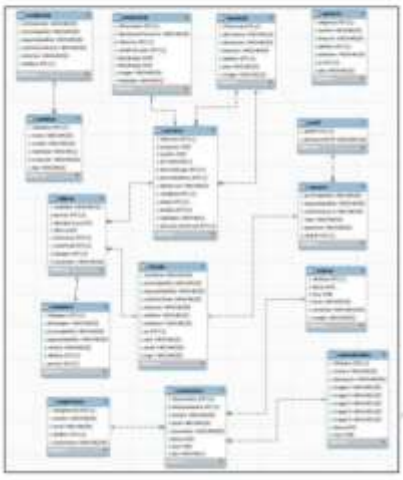
1981
IBM

RDBMS

Estándar de la Industria.

Su foco en la ejecución de **Transacciones**

- **A**tomicidad
- **C**onsistencia
- **I**solation (Aislamiento)
- **D**urabilidad



Bases de Datos OO

- A mediados de los 90 se hizo más visible el paradigma de desarrollo Orientado a Objetos.
- Era necesario una traducción de objetos a relaciones.
- Se pensó como solución en Bases de Datos específicas para resolver la complejidad.

Bases de Datos Orientadas a Objetos

Se trató de estandarizar el OQL – Object Query Language.

Definición de RDBMS

Que es un Sistema de Administración de Base de Datos?
(DBMS – Data Base Manager System)

Es un programa que permite administrar los contenidos de una/s base/s de datos almacenada en disco. También llamado Motor de Base de Datos.

El DBMS ofrece a los usuarios una percepción de la base de datos que está en cierto modo por encima del nivel del hardware y que maneja las operaciones del usuario expresadas en el nivel más alto de percepción.

El DBMS también interpreta y ejecuta todos los comandos SQL que le son enviados.

Entre los motores de Base de Datos más utilizados podemos nombrar los siguientes: Oracle, MS SqlServer, MySql, PostgreSQL, DB2, Informix, Sybase, SQLite, entre otros.

Historia de las BD NoSQL

- Se inicia en 1966 con el surgimiento de las Bases de Datos Jerárquicas – IBM IMS para el programa espacial Apollo.
-
- En la historia más reciente, Amazon y Google se posicionan cómo líderes en buscar mecanismos de almacenamiento y recuperación para volúmenes de datos enormes.

Historia de las BD NoSQL

2000	Neo4j the graph database	Comienzo proyecto Desarrollada en JAVA. Basada en Estructuras de Grafos.
2005	CouchDB	Comienzo proyecto Inspirada en Lotus Notes. JavaScript como leng. De consulta. BD basada en documentos.
2006	Google	Proyecto BigTable Primera especificación BD en forma columnar. Escalamiento Horizontal (Pb) Google Reader, G. Maps, G. Earth, Blogger.com
2007	Amazon DynamoDB	Primera especificación BD basadas en clave-valor
	mongoDB	Similar a CouchDB Documentos basados en JSON eBay, MetLife, MTV, Telefónica...

NOT ONLY SQL

2008 Cassandra

Facebook libera Open Source

Desarrollado por autor de Amazon Dynamo

Not Only SQL

2009 redis

"RE mote DIctionary Service" Proyecto comenzado por un desarrollado italiano, Salvatore Sanfilipo para un producto de software denominado LLOG.

APACHE HBASE

Basada en especificación de BigTable

Integrado a Proyecto Apache

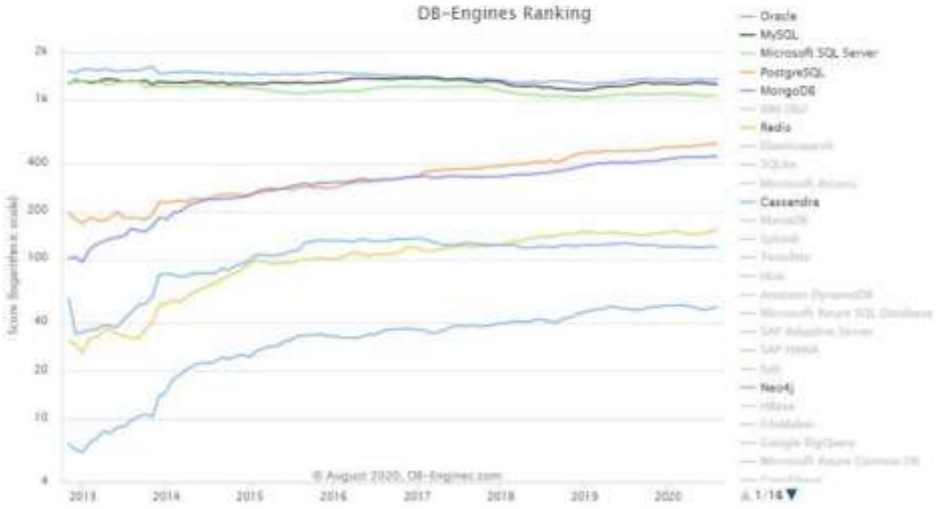
2010 redis

VMWare contrata a S. Sanfilipo y a otros desarrolladores de Redis, para continuar fulltime con el proyecto.

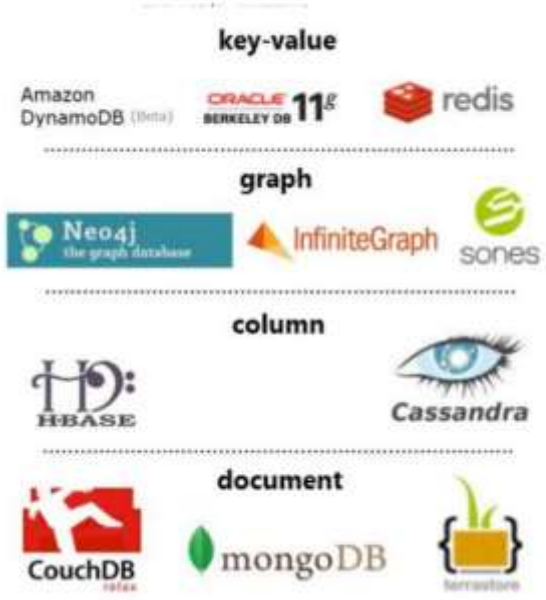
SQL – DB-ENGINES.COM

August 2020 http://db-engines.com/en/ranking

HOY



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 sólo 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

NOSQL – KEY VALUE DB

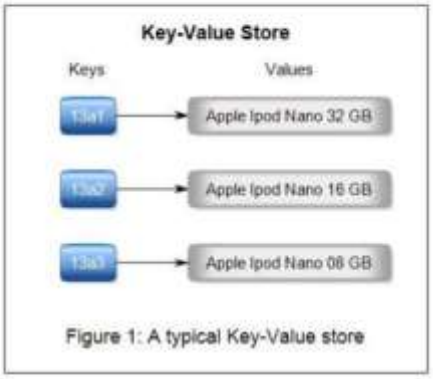


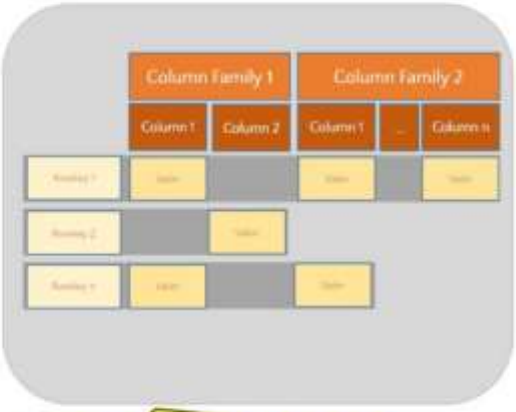
Figure 1: A typical Key-Value store

Pinterest

¿ Cuándo se Usan ?

- Almacenar información de sesiones
- Perfiles de Usuarios
- Información de carros de compras

NOSQL – COLUMNAR DB



¿ Cuándo se Usan ?

CMS, blogging

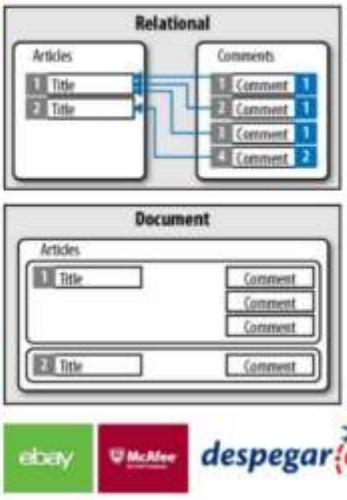
Web-analytics / Real-Time analytics

Expiring

Time series

IoT Metrics

NOSQL – DOCUMENT BASED DB



¿ Cuándo se Usan ?

- Logging de Eventos
- CMS, blogging
- Web-analytics / Real-Time analytics
- E-Commerce
- Startups/WebApps

NOSQL – GRAPH DB

¿ Cuándo se Usan ?

- **Datos interconectados**
- **Servicios de Ruteo / Despachos**
- **Motores de recomendaciones**

Persistencia Políglota

Diferentes tecnologías de bases de datos para resolver diferentes problemas desde una misma aplicación.

Búsquedas Performantes sobre Catálogo de Productos

Información Distribuida Geográficamente Profile de usuarios y Documentación de Productos con Info no estructurada

Caché de Sesiones Lockeos Distribuidos

Transacciones Económicas.

Teorema CAP

- Fue **desarrollado** más **como una conjetura** que como un teorema por el computador científico **Eric Brewer** de la **Universidad de California, Berkeley**.
- En el **año 2002**, **Seth Gilbert y Nancy Lynch del MIT** publicaron una **prueba formal** de la conjetura de Brewer, **transformándolo en un teorema**.
- **Brewer** indicaba que es **imposible en un sistema** computacional **distribuido**, **proveer simultáneamente** las **tres propiedades** antes expresadas:

  - **C***onsistency*
  - ***A**vailability*
  - **P**artitioning *tolerance*.

23