JUNTA DE ANDALUCIA

CENTRO DEL PROFESORADO DE MÁLAGA

Introducción al Big Data y Business Intelligence

Nombre: José Manuel García Nieto
Fecha: 6 de abril de 2017. 16:30 a 20:30

CONSEJERÍA DE EDUCACIÓN
CENTRO DEL PROFESORADO DE MÁLAGA

Presentación: http://khaos.uma.es/jmgn/

Khaos INVESTIGACIÓN

uma UNIVERSIDAD DE MÁLAGA

Esquema de Contenidos:

1. Introducción al Big Dada
2. Trabajando con los datos
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Presentación: Grupo Khaos – Universidad de Málaga

Khaos
INVESTIGACIÓN

uma
UNIVERSIDAD DE MÁLAGA

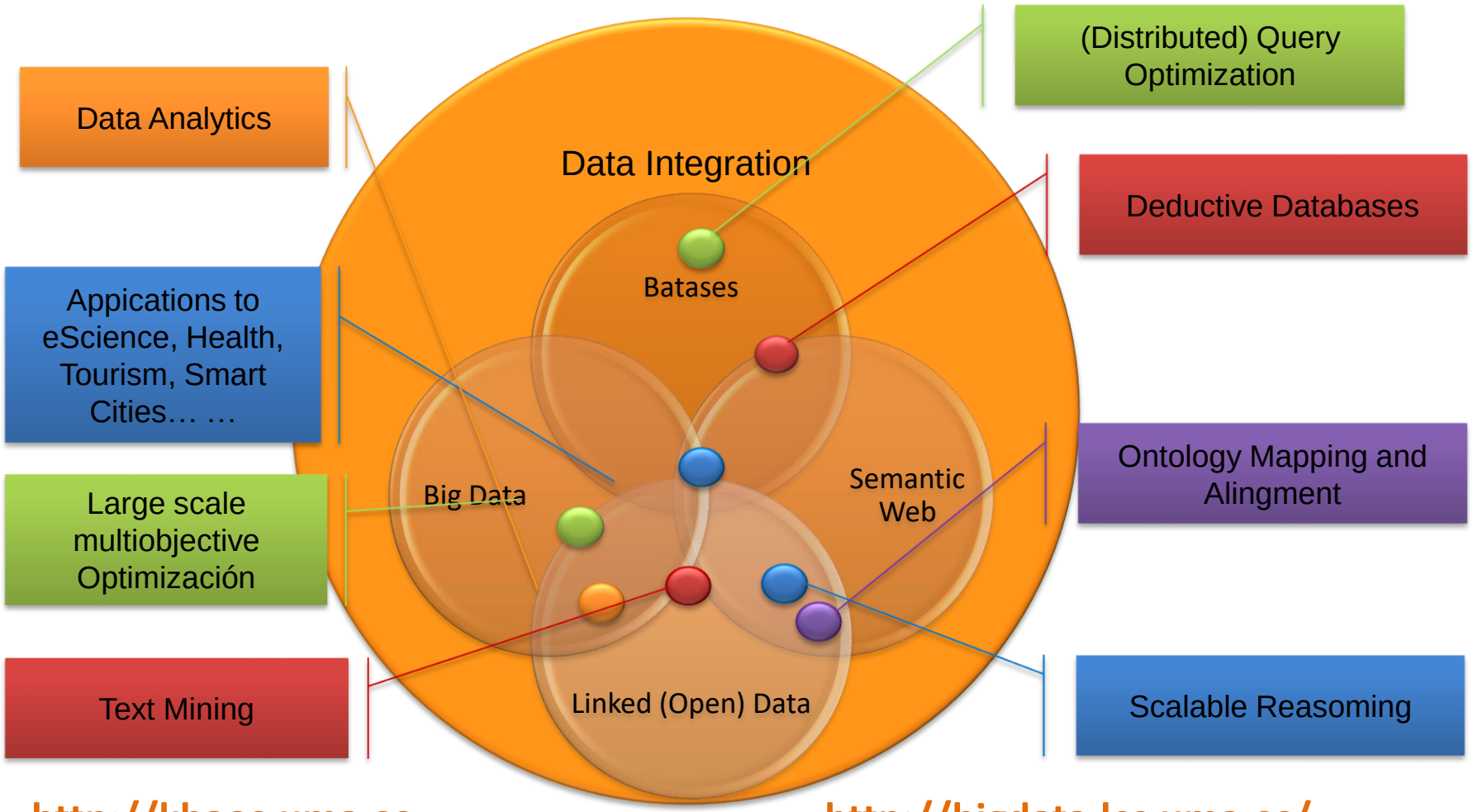
Ada Byron Research Center

http://khaos.uma.es

http://bigdata.lcc.uma.es/

Presentación: Grupo Khaos – Universidad de Málaga

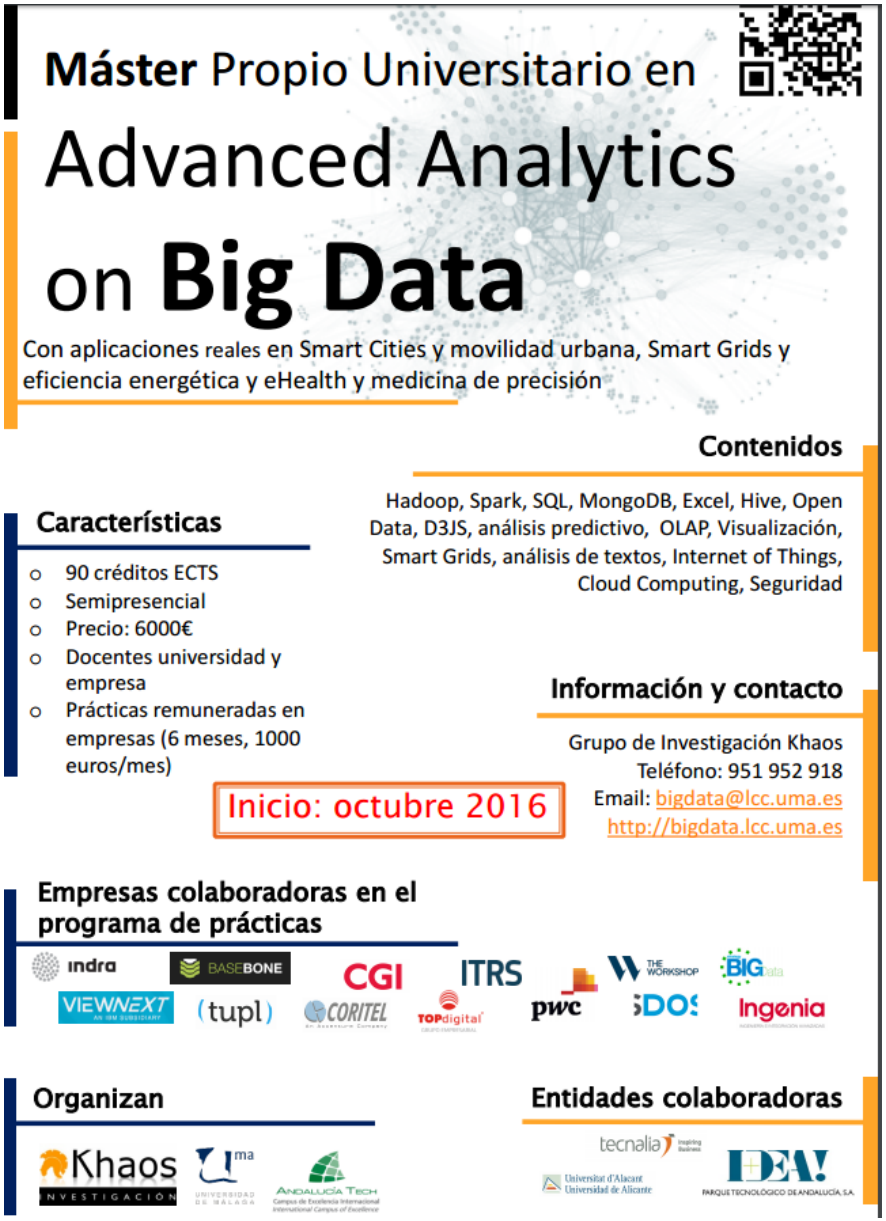
Khaos
INVESTIGACIÓN



http://khaos.uma.es

http://bigdata.lcc.uma.es/

- Líneas de trabajo:
  - **Data management and Integration** (Relational Databases, *NoSQL databases*, Linked Data, Open Data)
  - **Data Analytics**: Data minig, Text mining, (NLP&Semantics, social networks analytics, short texts,…), Information retrieval, *Big Data Analytics*. Semantic annotation and enrichment. Integrated-Data Analytics.
  - **Semantic Technologies** (Scalable Reasoning, Schema Mapping and Alingment, Semantic Assignation to Mobile Objects, Semantic base Content Recommendation, Semantics for eScience, *Smart Data for Big Data interpretation*)
  - **Metaheuristics**. Mono and multi-Objective Optimization (scalable, high throughput, Globus, *MapReduce, SPARK, Storm*). Metaheuristics and new semantic aware Machine Learning algorithms (based on *Mahout*)
- Application domains:
  - Health: **Bioinformatics, Translational Medicine, System Medicine. Also Electronic Health Records and Medical Decision Support Systems.**
  - **eCommerce and Retailing**
  - **Cultural Heritage and Tourism**
  - **Smart Cities**



http://bigdata.lcc.uma.es/

LA COLECCIÓN BIG DATA Y LOS NEGOCIOS

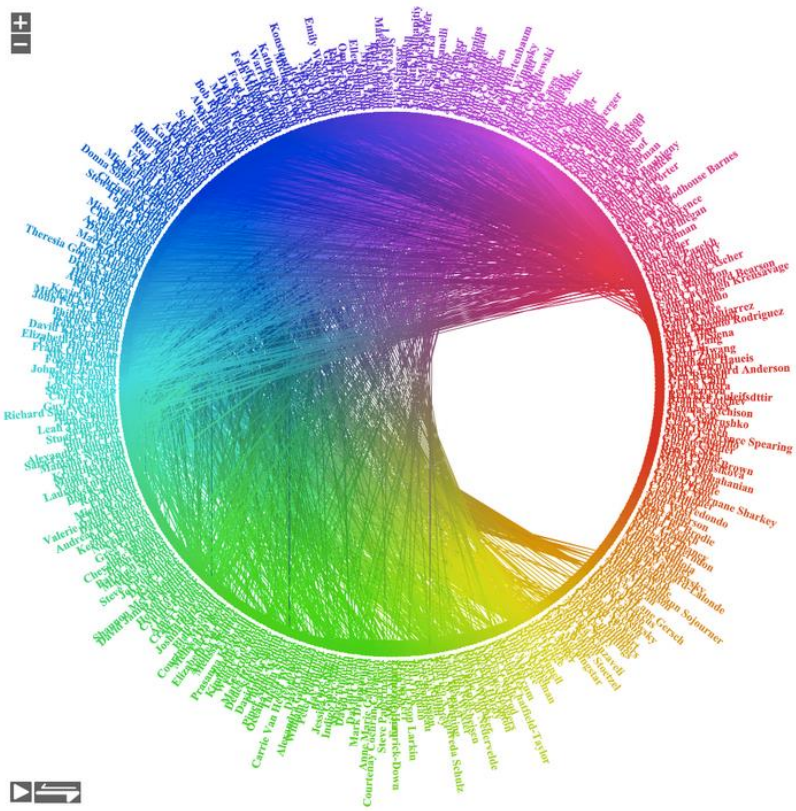
Libro 1: Introducción al Big Data
Libro 2. Big Data: gestión inteligente de los datos
Libro 3: Introducción al trabajo con datos
Libro 4: Modelado en Big Data
Libro 5: Almacenamiento de Big Data.
Libro 6: Procesamiento y Análisis Inteligente de Big Data
Libro 7: Visualización y consumo de Big Data
Libro 8: Big Dada: seguridad y gobernanza.

Introducción al Big Data

¿Por qué formarse en Big Data?

El Big Data creará 4,4 millones de empleos en todo el mundo

Big Data 18 mayo, 2014



¿Quieres un empleo de futuro? Hazte profesional del big data

Por **Pablo G. Bejerano** | 12 de enero de 2015 a las 12:00

DIGITAL

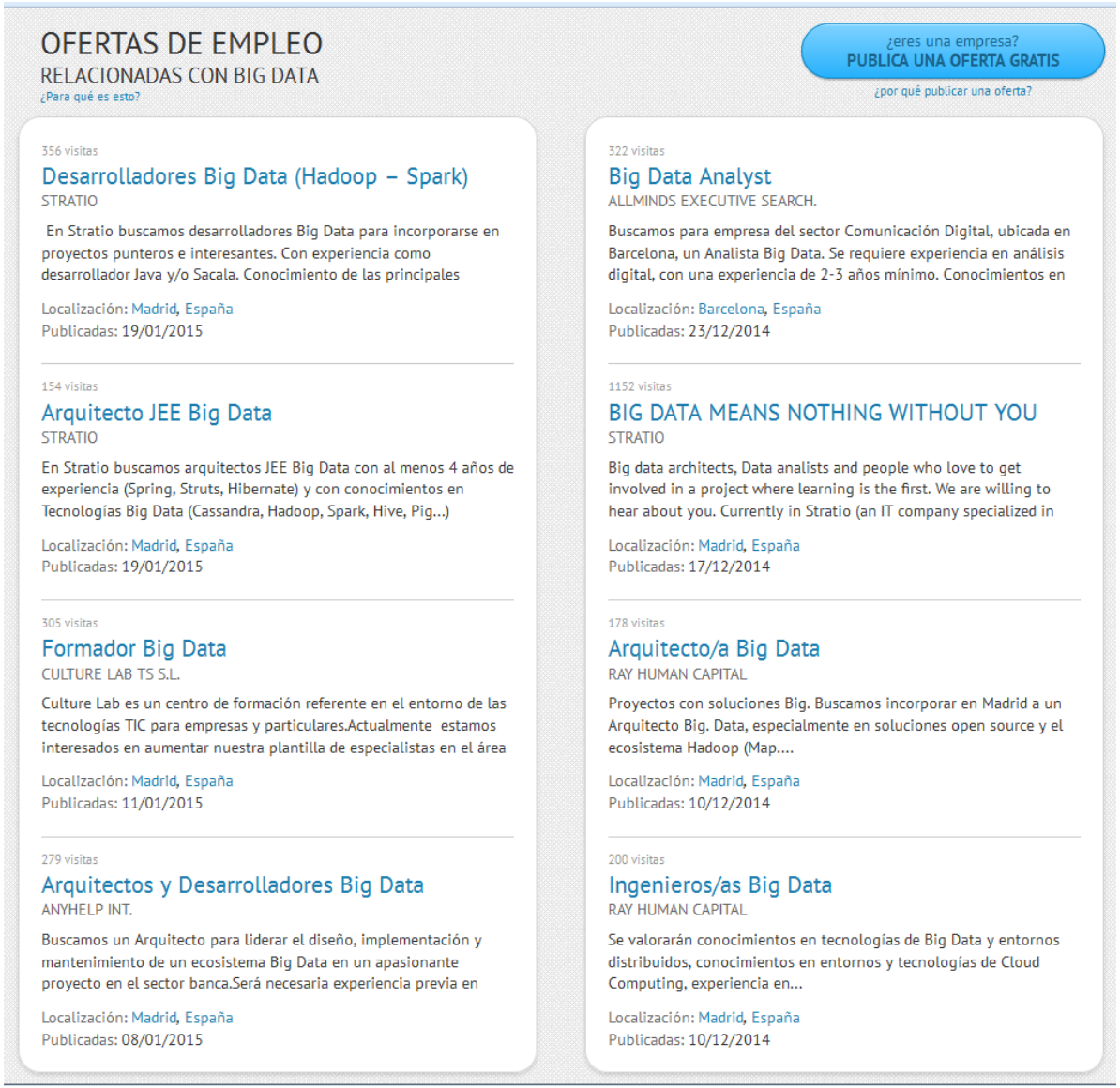
907 Me gusta 239 Twittear 31 +1

Uno de los empleos que más solicitados estarán en los próximos años será el de profesional del big data.

No hay duda de que uno de los sectores que mejor ha sorteado la crisis económica ha sido el tecnológico. De hecho para la electrónica de consumo los últimos años han sido de abundancia y en otras áreas que no son tan visibles los derroteros tampoco han ido mal. Hay un campo concreto que ha crecido de forma significativa y sus perspectivas apuntan a incrementos aún más pronunciados. Implica todo lo que tiene que ver con el procesamiento, la gestión y el análisis de datos. Por este motivo se espera que **el perfil de profesional del big data se convierta en uno de los más demandados**.

El crecimiento general de la industria tecnológica empuja inevitablemente el sector del big data. Los dispositivos móviles generan toneladas de información constantemente y a esta se sumará la que producirá Internet de las cosas y la domótica. Se trata de miles de millones de terminales equipados con sensores para recoger información y conectados a la Red, de manera que puedan enviarla. Detrás habrá fabricantes de hardware, desarrolladores de plataformas, desarrolladores de aplicaciones y otras **empresas que recibirán una gran cantidad de datos**.

¿Por qué formarse en Big Data?



OFERTAS DE EMPLEO
RELACIONADAS CON BIG DATA
¿Para qué es esto?

¿eres una empresa?
PUBLICA UNA OFERTA GRATIS
¿por qué publicar una oferta?

356 visitas
Desarrolladores Big Data (Hadoop – Spark)
STRATIO
En Stratio buscamos desarrolladores Big Data para incorporarse en proyectos punteros e interesantes. Con experiencia como desarrollador Java y/o Sacala. Conocimiento de las principales
Localización: Madrid, España
Publicadas: 19/01/2015

154 visitas
Arquitecto JEE Big Data
STRATIO
En Stratio buscamos arquitectos JEE Big Data con al menos 4 años de experiencia (Spring, Struts, Hibernate) y con conocimientos en Tecnologías Big Data (Cassandra, Hadoop, Spark, Hive, Pig...)
Localización: Madrid, España
Publicadas: 19/01/2015

305 visitas
Formador Big Data
CULTURE LAB TS S.L.
Culture Lab es un centro de formación referente en el entorno de las tecnologías TIC para empresas y particulares.Actualmente estamos interesados en aumentar nuestra plantilla de especialistas en el área
Localización: Madrid, España
Publicadas: 11/01/2015

279 visitas
Arquitectos y Desarrolladores Big Data
ANYHELP INT.
Buscamos un Arquitecto para liderar el diseño, implementación y mantenimiento de un ecosistema Big Data en un apasionante proyecto en el sector banca.Será necesaria experiencia previa en
Localización: Madrid, España
Publicadas: 08/01/2015

322 visitas
Big Data Analyst
ALLMINDS EXECUTIVE SEARCH.
Buscamos para empresa del sector Comunicación Digital, ubicada en Barcelona, un Analista Big Data. Se requiere experiencia en análisis digital, con una experiencia de 2-3 años mínimo. Conocimientos en
Localización: Barcelona, España
Publicadas: 23/12/2014

1152 visitas
BIG DATA MEANS NOTHING WITHOUT YOU
STRATIO
Big data architects, Data analists and people who love to get involved in a project where learning is the first. We are willing to hear about you. Currently in Stratio (an IT company specialized in
Localización: Madrid, España
Publicadas: 17/12/2014

178 visitas
Arquitecto/a Big Data
RAY HUMAN CAPITAL
Proyectos con soluciones Big. Buscamos incorporar en Madrid a un Arquitecto Big. Data, especialmente en soluciones open source y el ecosistema Hadoop (Map....
Localización: Madrid, España
Publicadas: 10/12/2014

200 visitas
Ingenieros/as Big Data
RAY HUMAN CAPITAL
Se valorarán conocimientos en tecnologías de Big Data y entornos distribuidos, conocimientos en entornos y tecnologías de Cloud Computing, experiencia en...
Localización: Madrid, España
Publicadas: 10/12/2014

La Explosión de Datos

La creación de datos se está produciendo a un ritmo record:

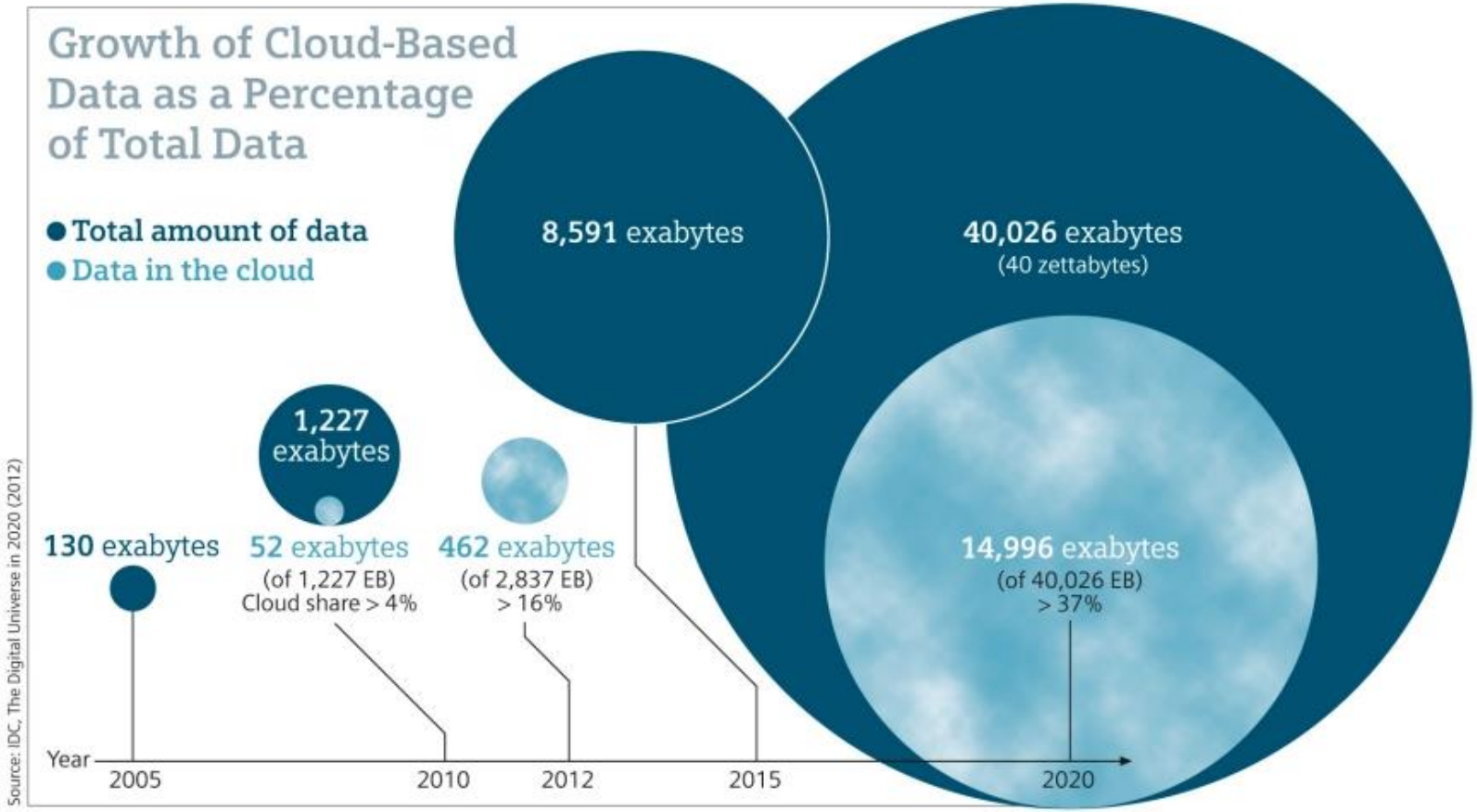
hasta el 2010 se generaron en el mundo 1ZB de datos

y

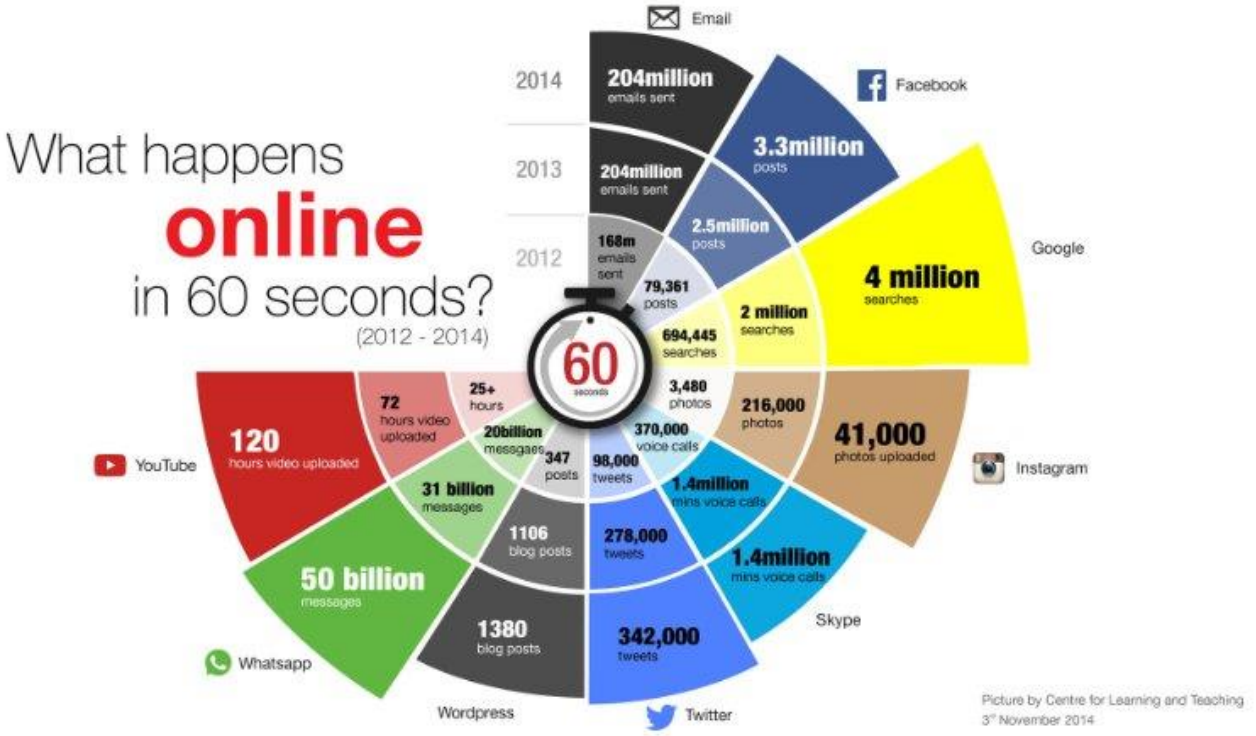
en 2014, se generaron más de 7ZB en un año

Unidades básicas de información		
Prefijos del Sistema Internacional		
Múltiplo - (Símbolo)	SI	Binario
kilobyte (kB)	10^3	2^{10}
megabyte (MB)	10^6	2^{20}
gigabyte (GB)	10^9	2^{30}
terabyte (TB)	10^{12}	2^{40}
petabyte (PB)	10^{15}	2^{50}
exabyte (EB)	10^{18}	2^{60}
zettabyte (ZB)	10^{21}	2^{70}
yottabyte (YB)	10^{24}	2^{80}

La Explosión de Datos

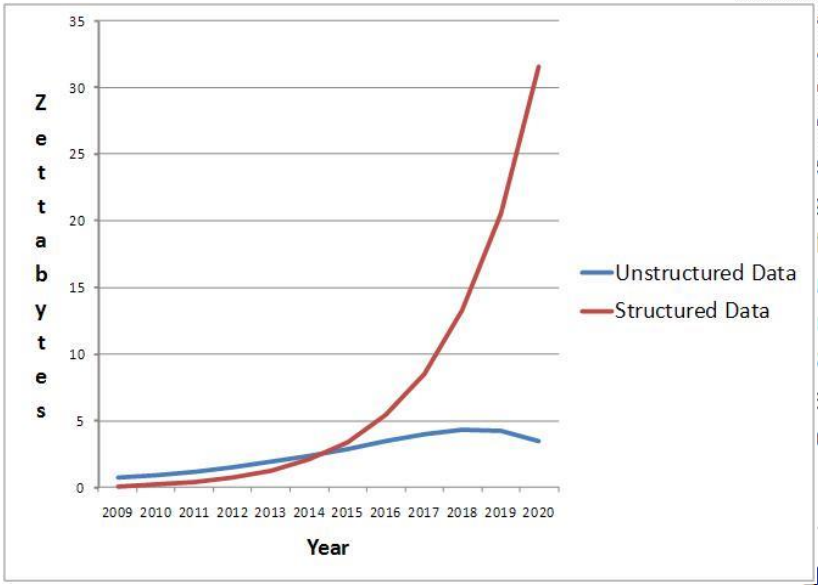


La Explosión de Datos



Introducción al Big Data

ESTAMOS EN LA ERA BIG DATA



que a los clientes les resulte fácil y rentable procesar y extraer información de volúmenes masivos de datos.

BigQuery enables businesses and developers to gain real-time business insights from massive amounts of data without any upfront hardware or software investments. Imagine a big pharmaceutical company optimizing daily marketing spend using worldwide sales and advertisement data. Or think of a small online retailer that makes product recommendations based on user clicks. Today, we are making BigQuery publicly available, an important milestone in our effort to bring Big Data analytics to all businesses via the cloud.

Key Capabilities

alyze Big Data and gain insights with familiar tools, such as Excel and SharePoint.

erage PowerPivot and Power View for Big Data using Excel.

nnect to the world's data and services with Windows Azure Marketplace.

le on demand in the private and public cloud.

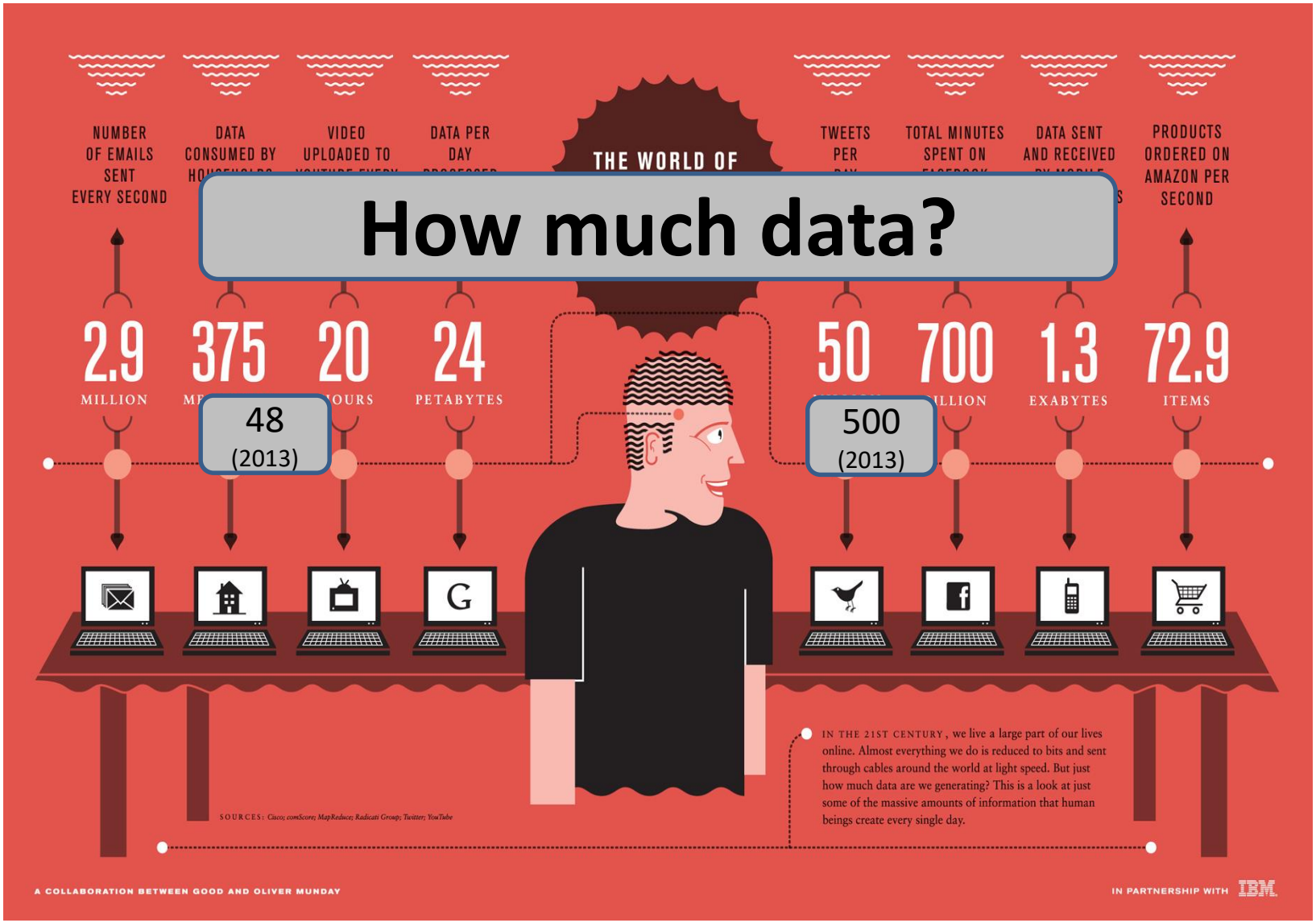
- Analysis
- Complex Event Processing
- Office Based Tools
- Predictive Analytics
- Reporting

ry day, we create 2.5 quintillion bytes of
a — so much that 90% of the data in the
ld today has been created in the last two
rs alone. This data comes from everywhere:
sors used to gather climate information,
ts to social media sites, digital pictures and
eos, purchase transaction records, and cell
ne GPS signals to name a few. This data is
data.

data spans three dimensions: Volume, Velocity and Variety.

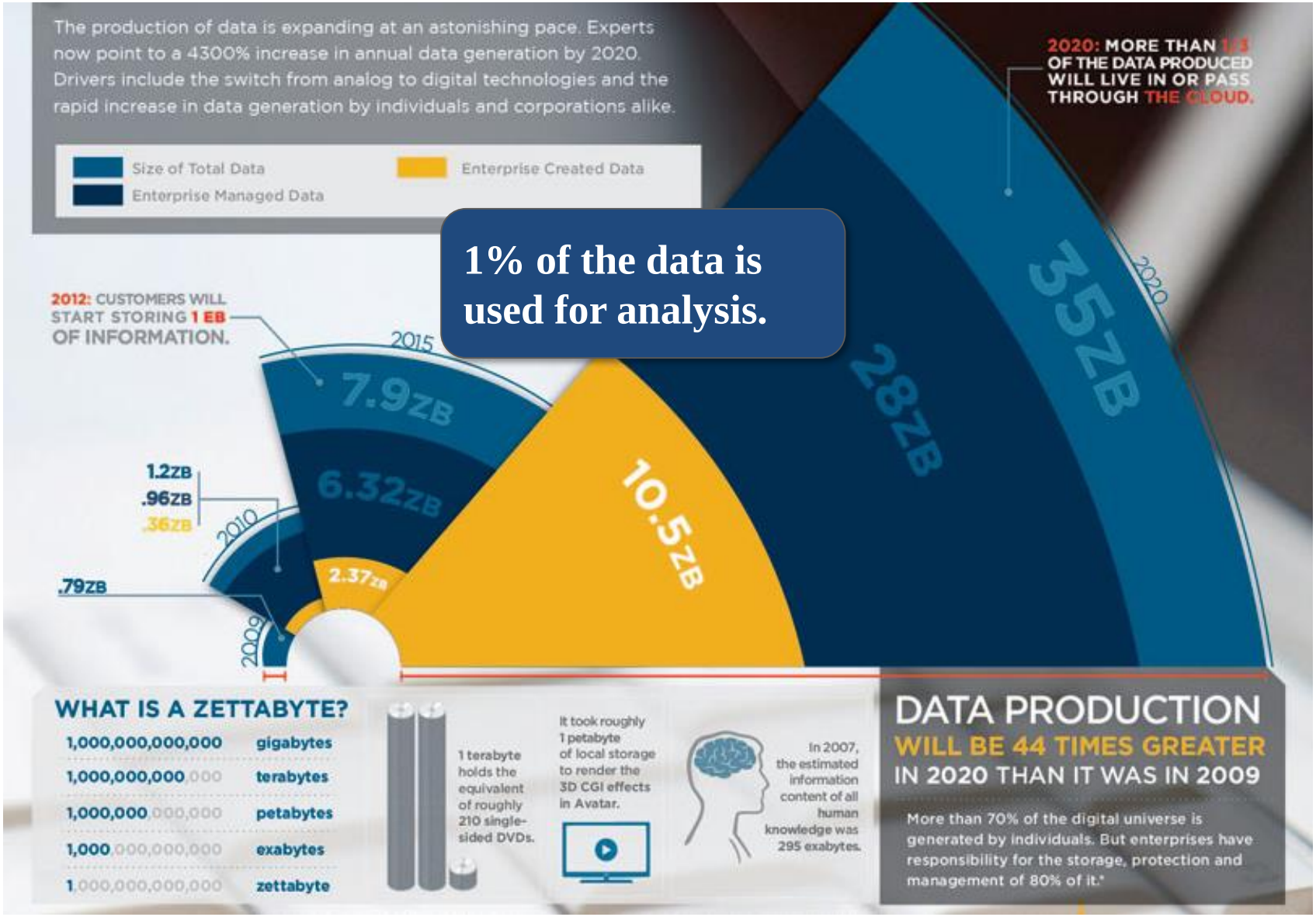
Introducción al Big Data

2011



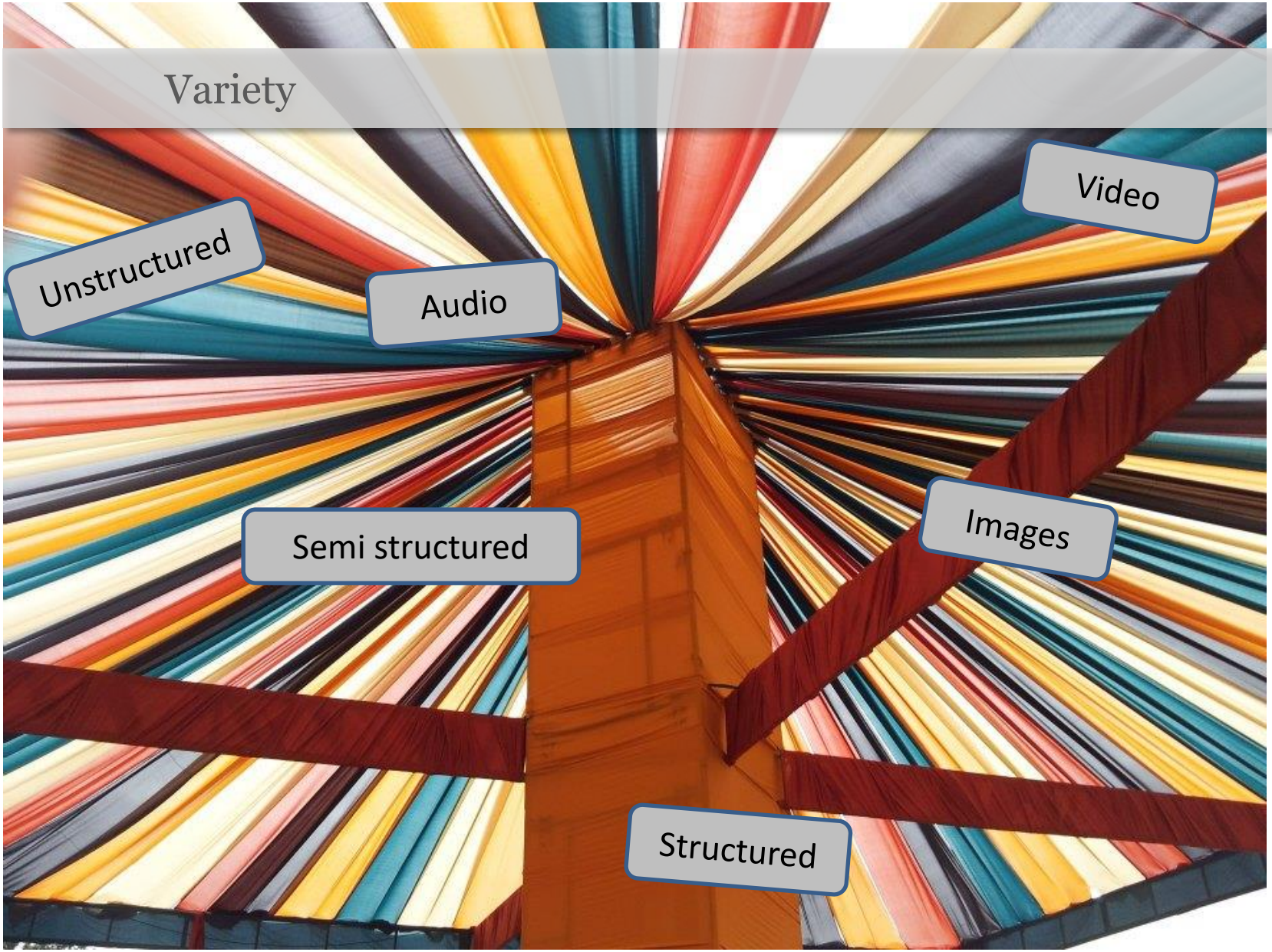
http://www.knowledgeinfusion.com/blog/2011/11/get-your-head-out-of-the-clouds-and-into-big-data/

Introducción al Big Data



http://www.csc.com/insights/flxwd/78931-big_data_growth_just_beginning_to_explode
http://www.guardian.co.uk/news/datablog/2012/dec/19/big-data-study-digital-universe-global-volume

Introducción al Big Data



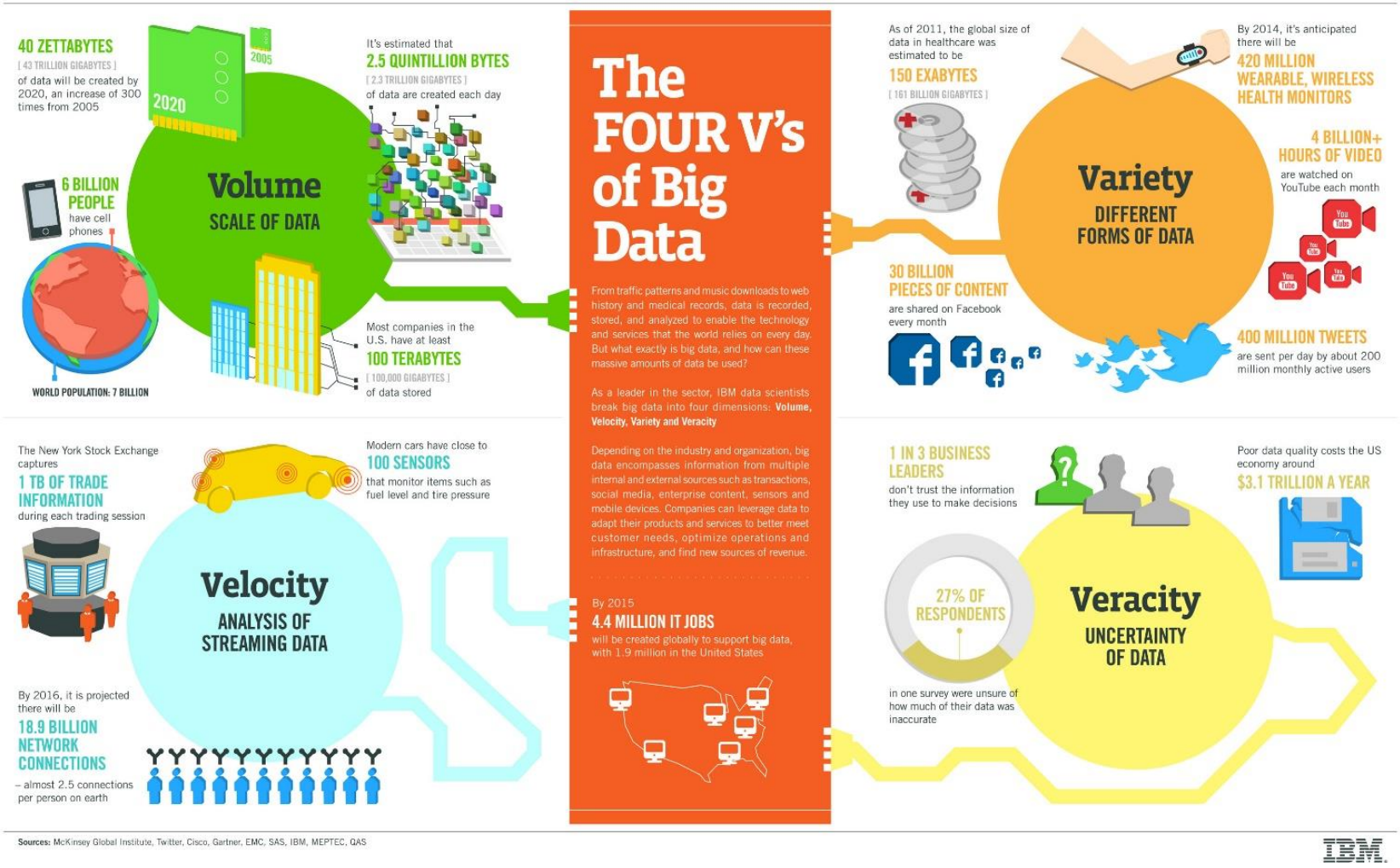
Introducción al Big Data

No sólo cantidad

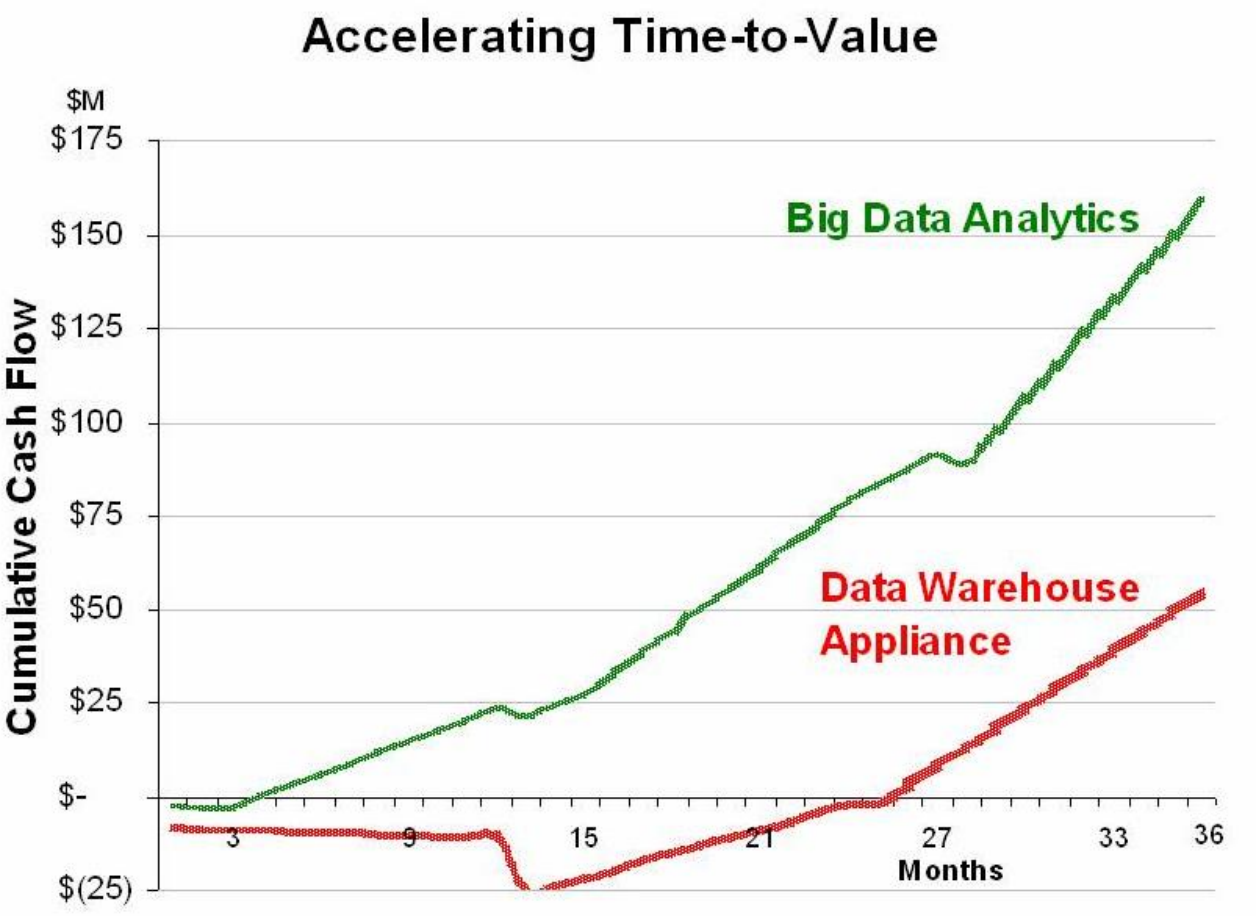
- El Big Data no es sólo una cuestión de **gestión de grandes cantidades de datos**…
  - …es también una oportunidad para alcanzar nuevos descubrimientos y conocimiento emergente a partir del **análisis de los datos**

volumen, velocidad, variedad y veracidad

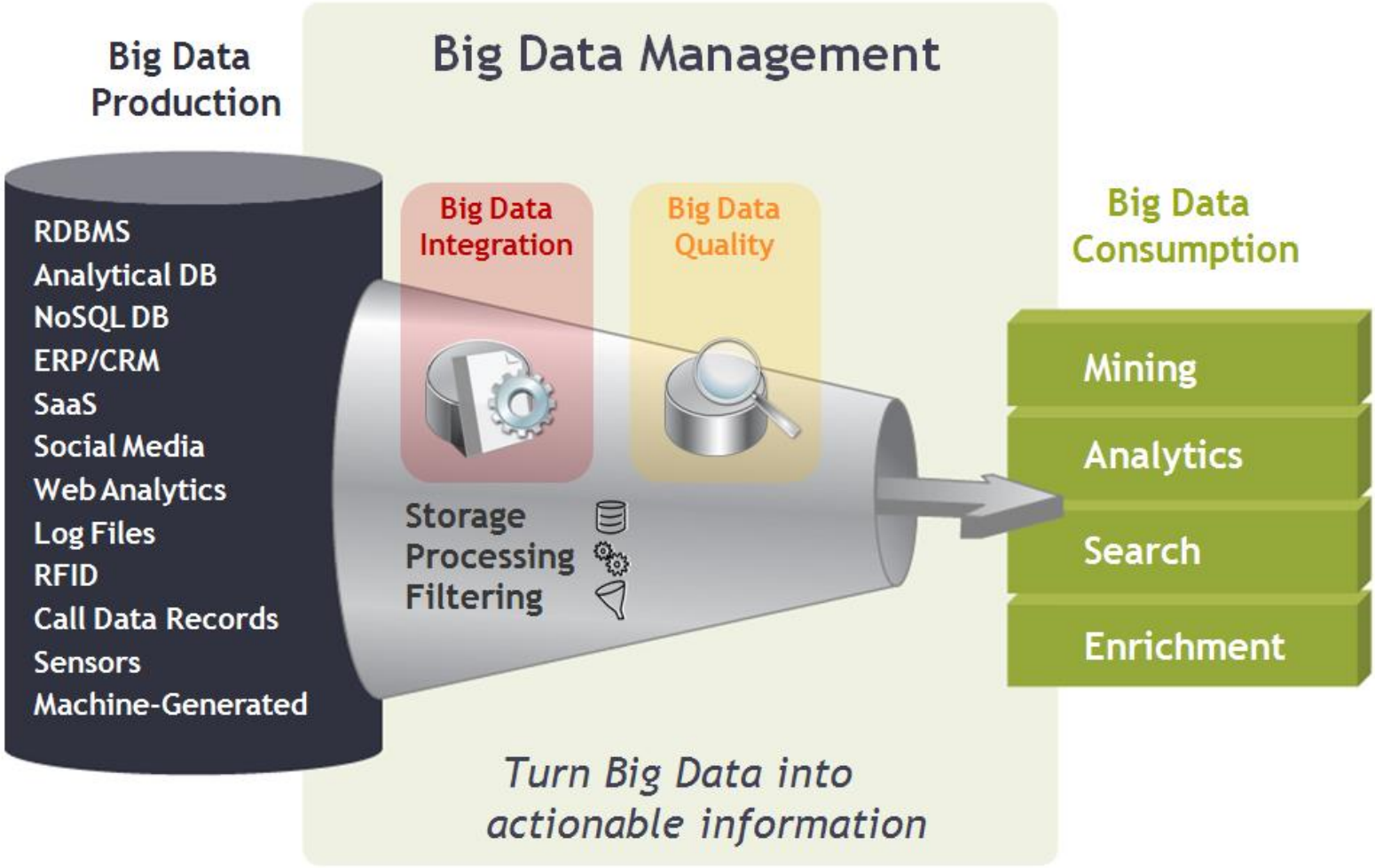
Introducción al Big Data



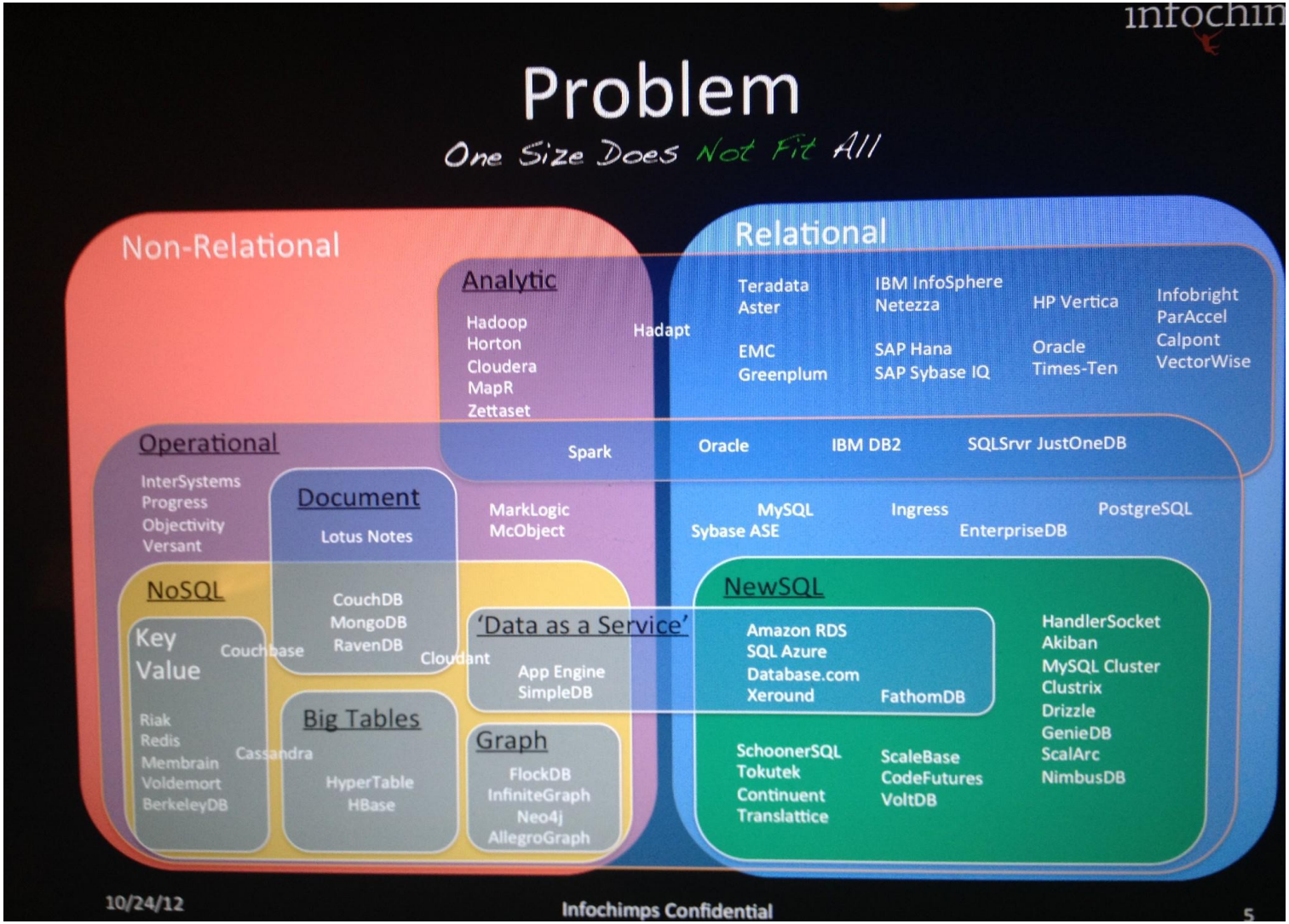
Introducción al Big Data



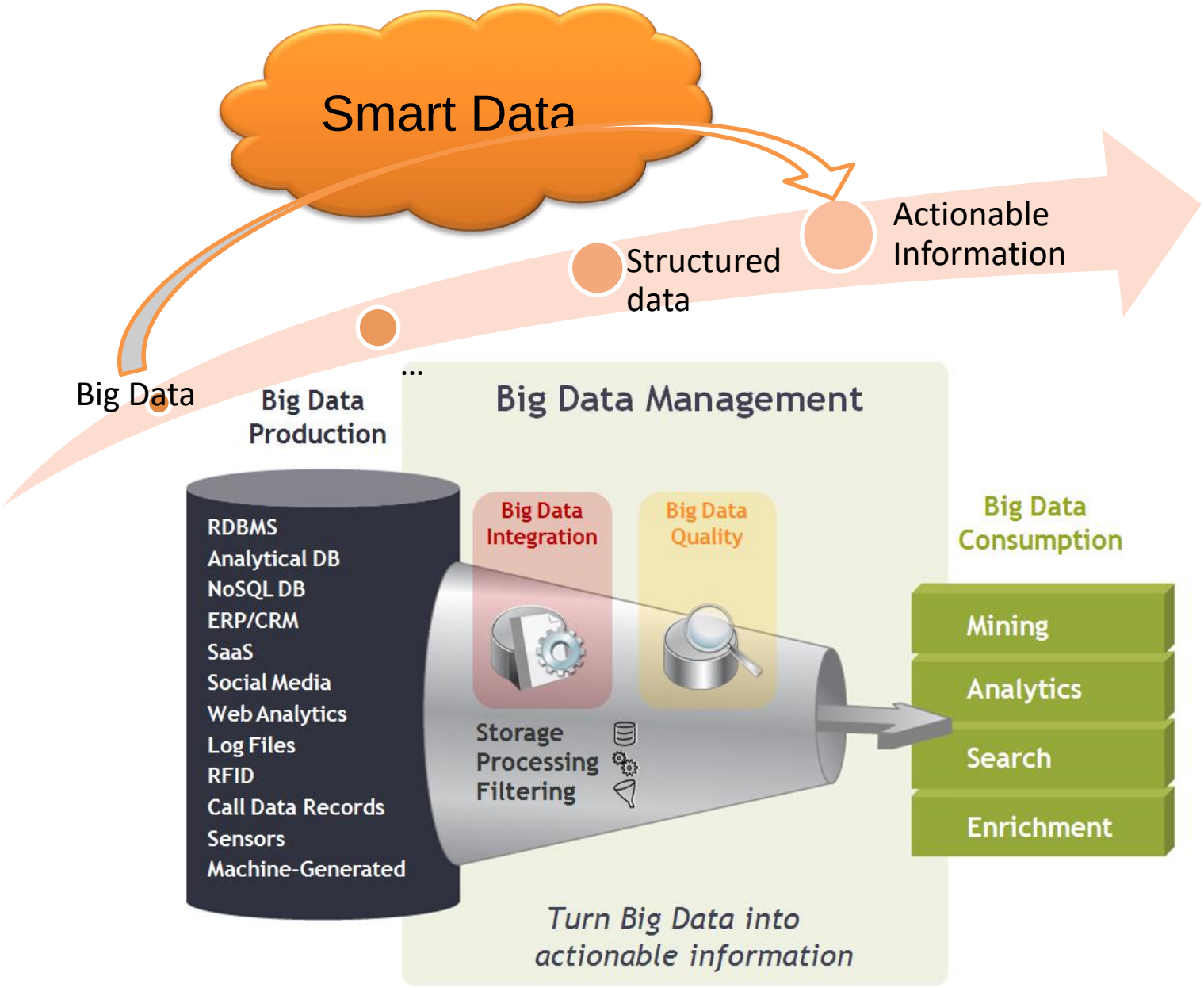
También Gestión y Análisis



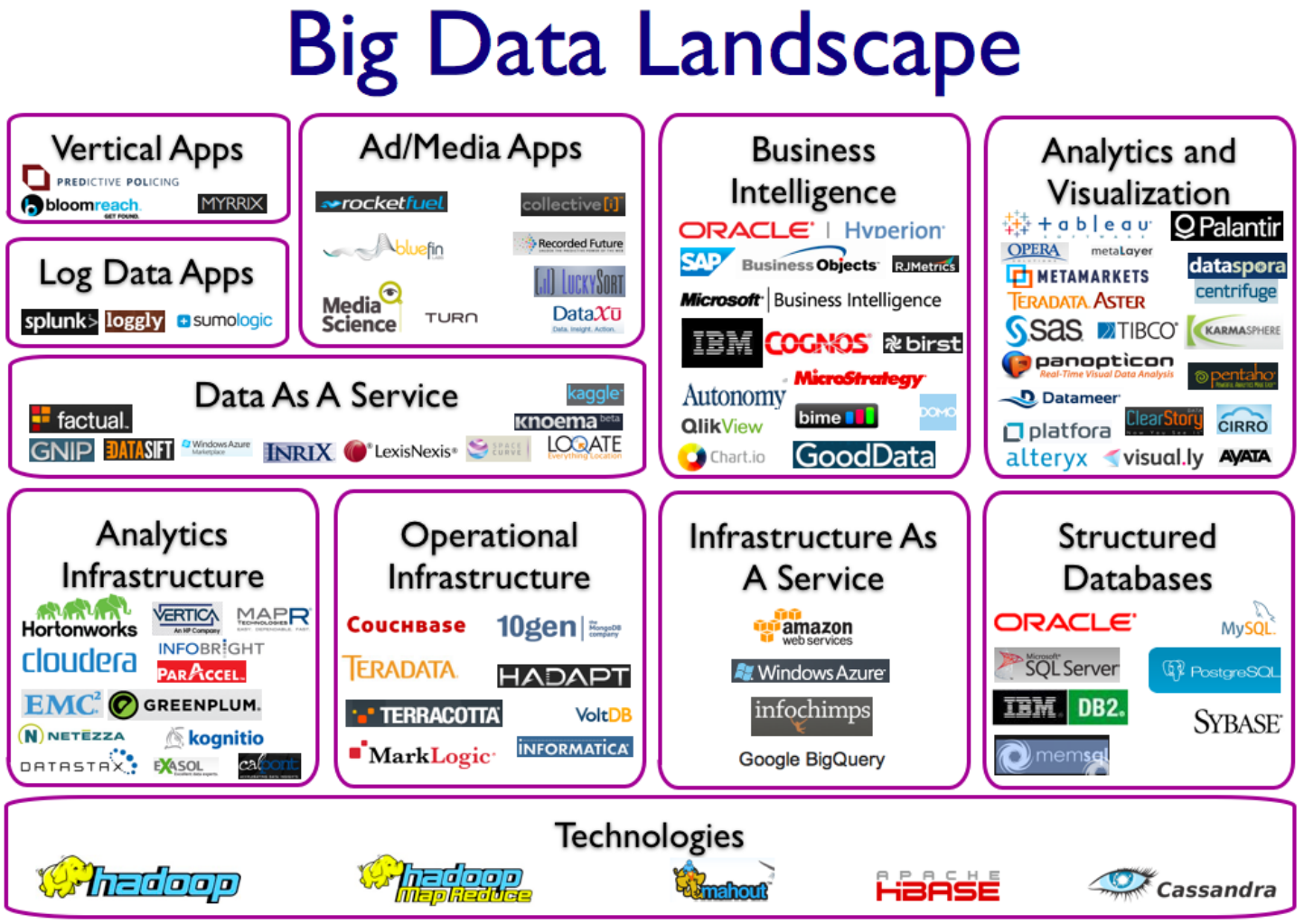
También Gestión y Análisis



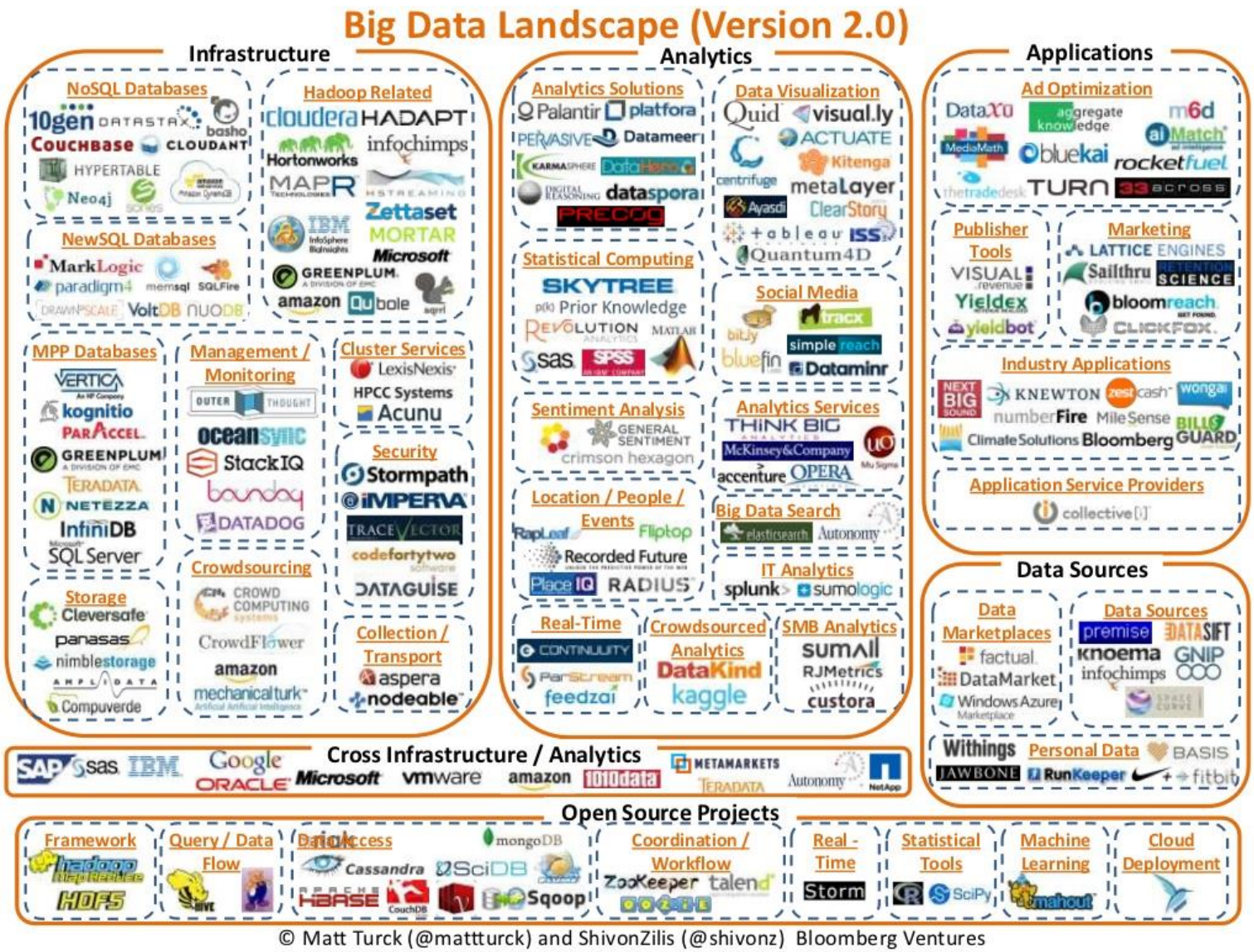
Introducción al Big Data



Introducción al Big Data

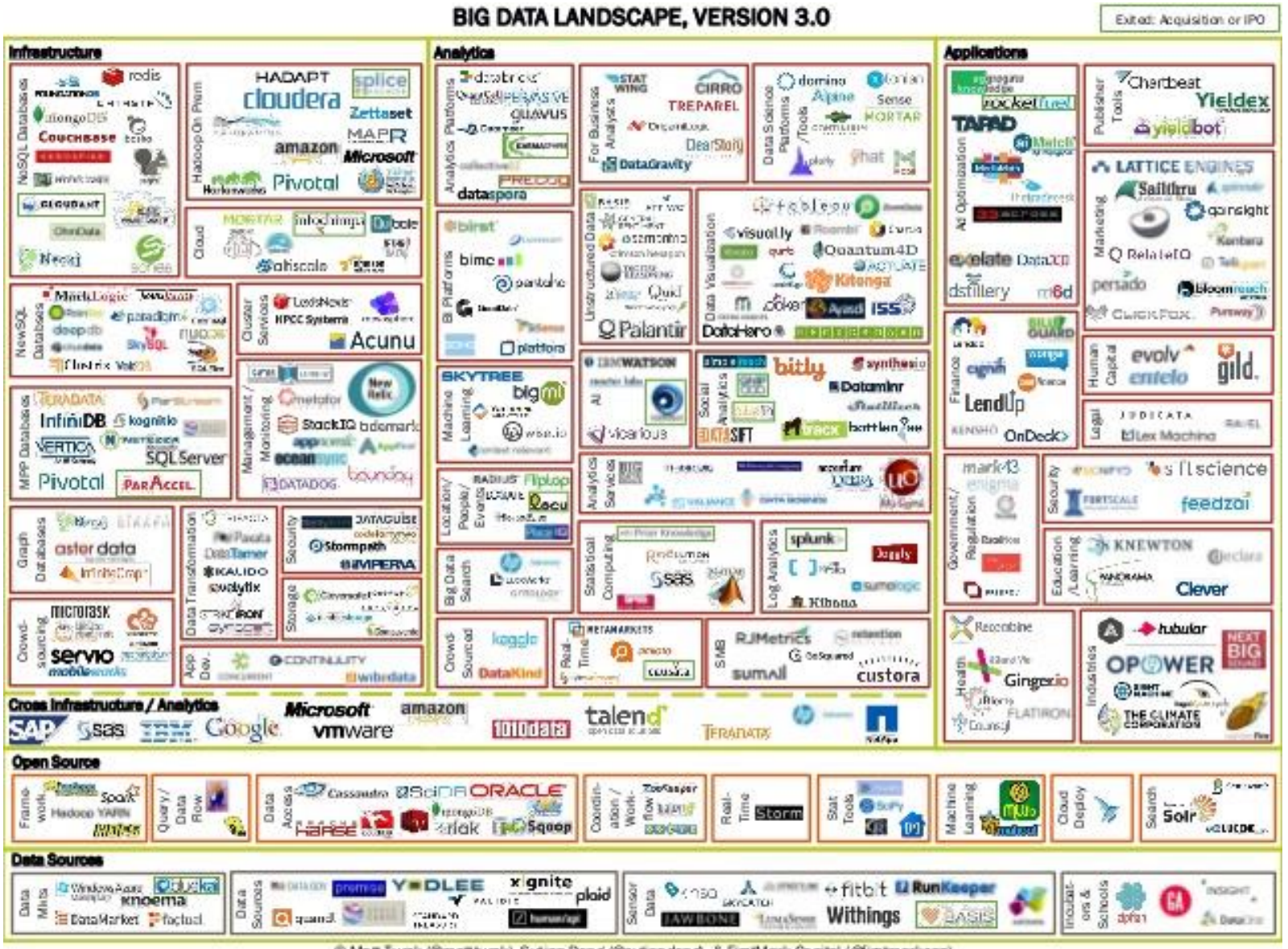


Introducción al Big Data



© Matt Turck (@mattturck) and ShivonZilis (@shivonz) Bloomberg Ventures

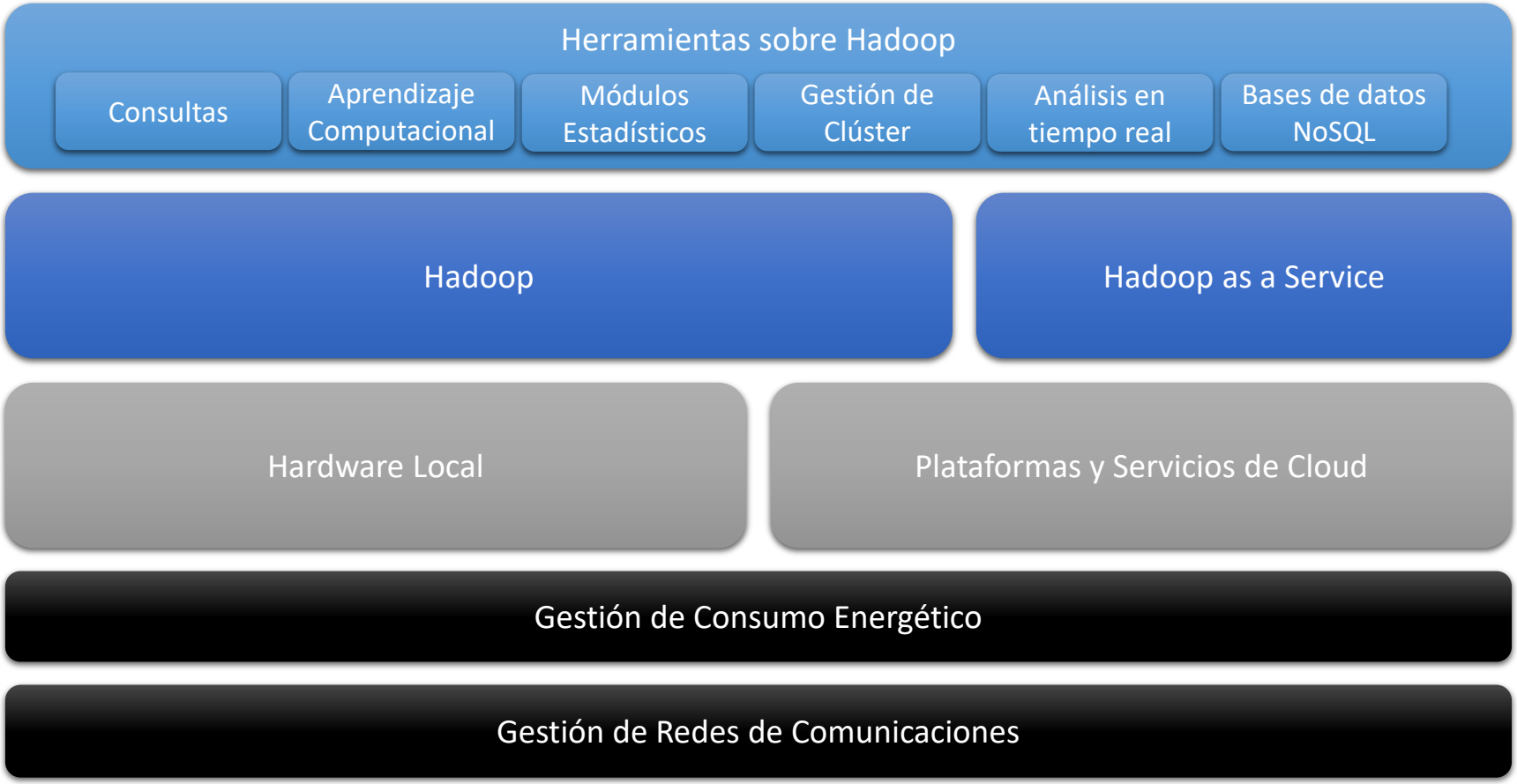
Introducción al Big Data



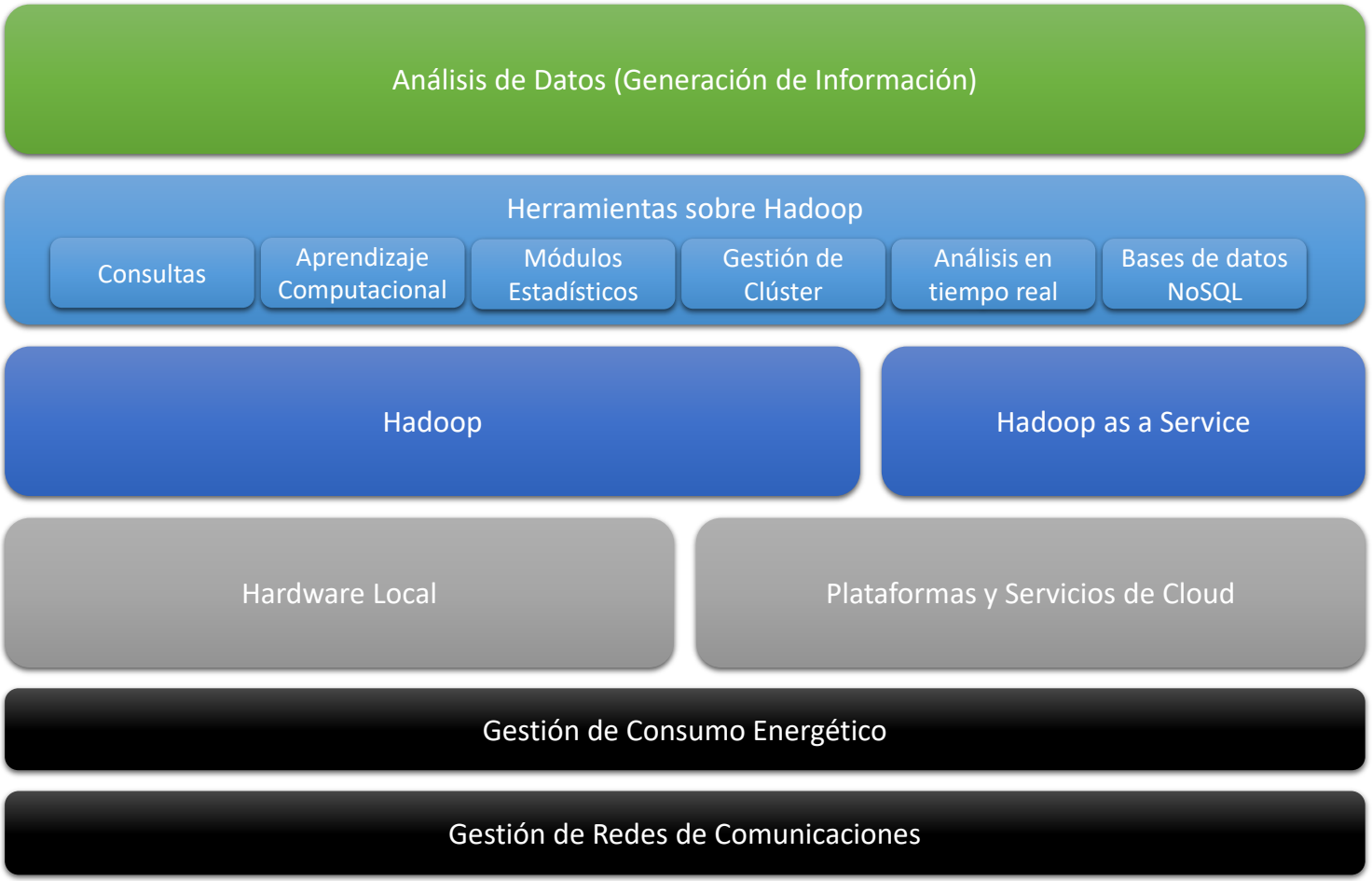
Visión Global

Visión Global

Visión Global

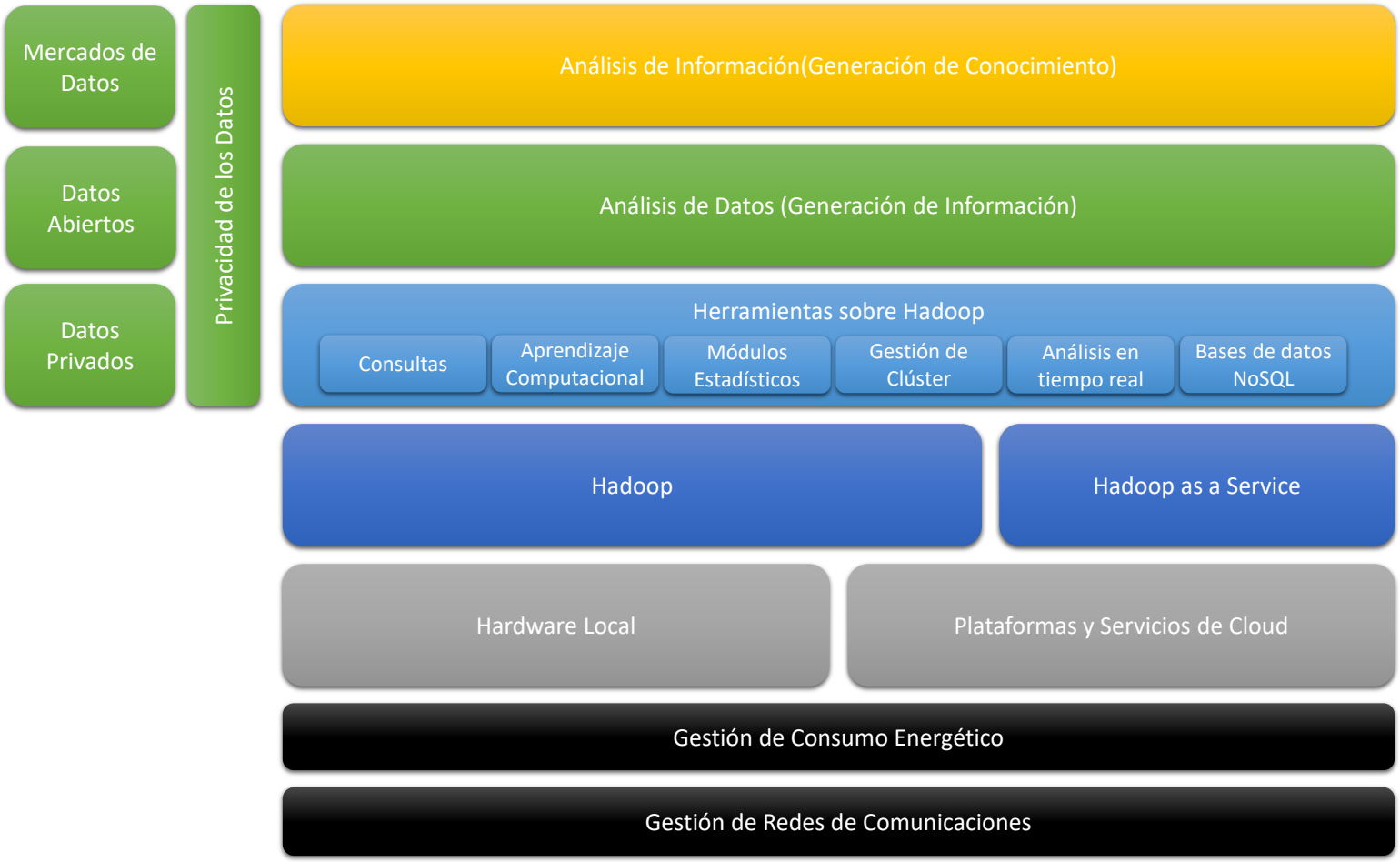


Visión Global

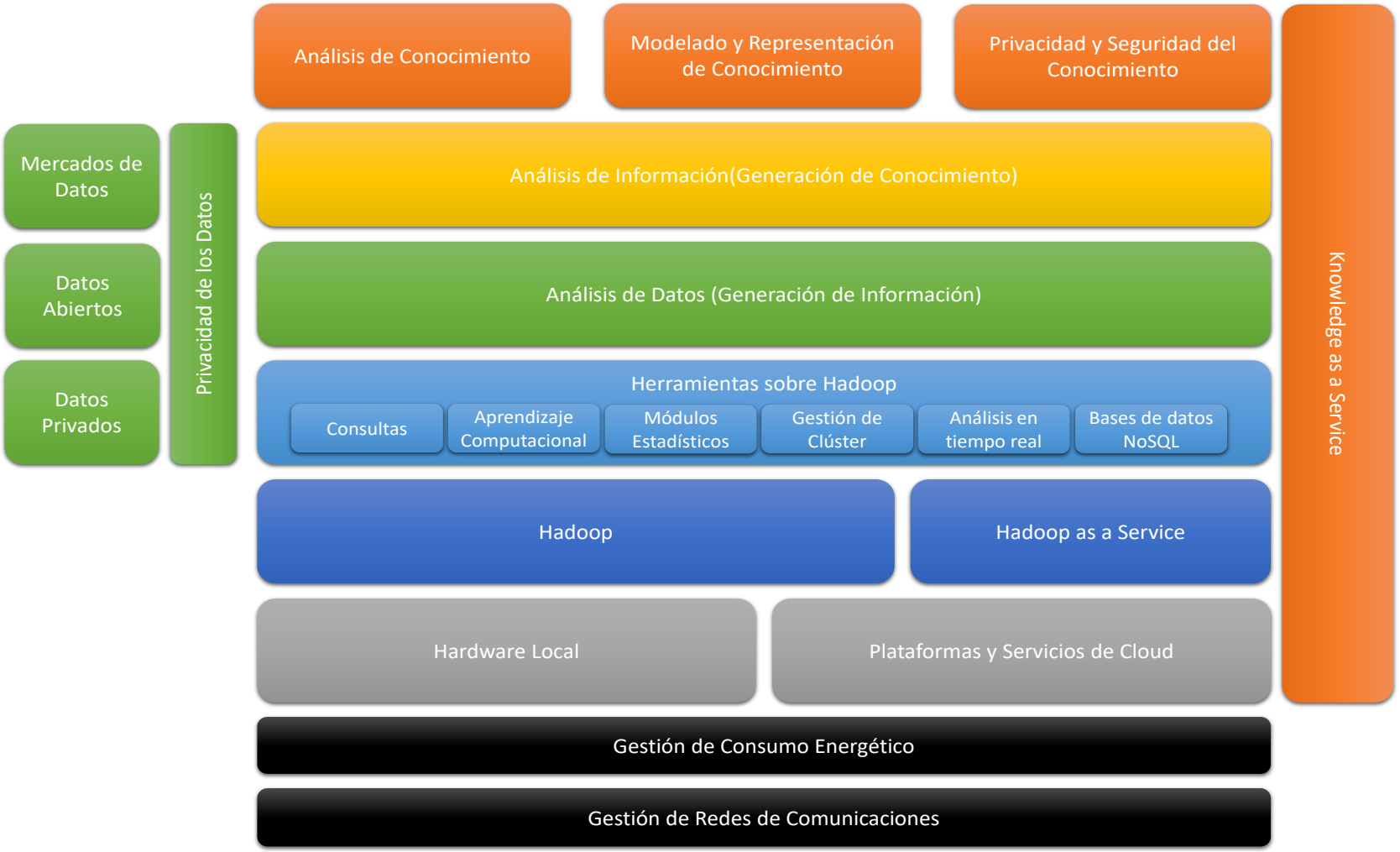


Visión Global

Visión Global



Visión Global



Conceptos alrededor del Big Data

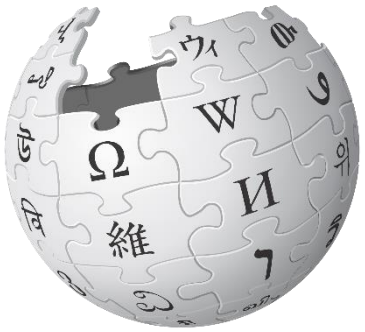
Big Data

Conceptos alrededor del Big Data

Internet de las cosas (en inglés, *Internet of things*, abreviado *IoT*) es un concepto que se refiere a la interconexión digital de objetos cotidianos con internet.

Si los objetos de la vida cotidiana tuvieran incorporadas etiquetas de radio, podrían ser identificados y gestionados por otros equipos, de la misma manera que si lo fuesen por seres humanos.

El concepto de internet de las cosas lo propuso Kevin Ashton en el Auto-ID Center del MIT en 1999,



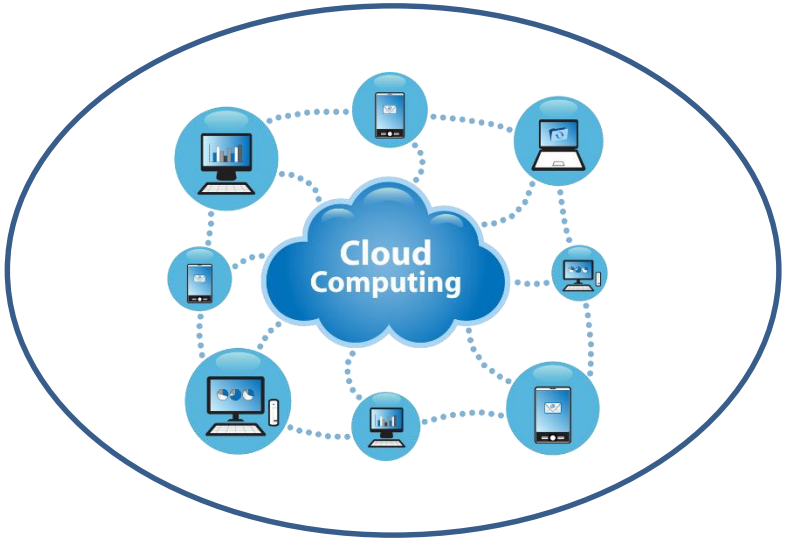
WIKIPEDIA
La enciclopedia libre

Conceptos alrededor del Big Data

Conceptos alrededor del Big Data

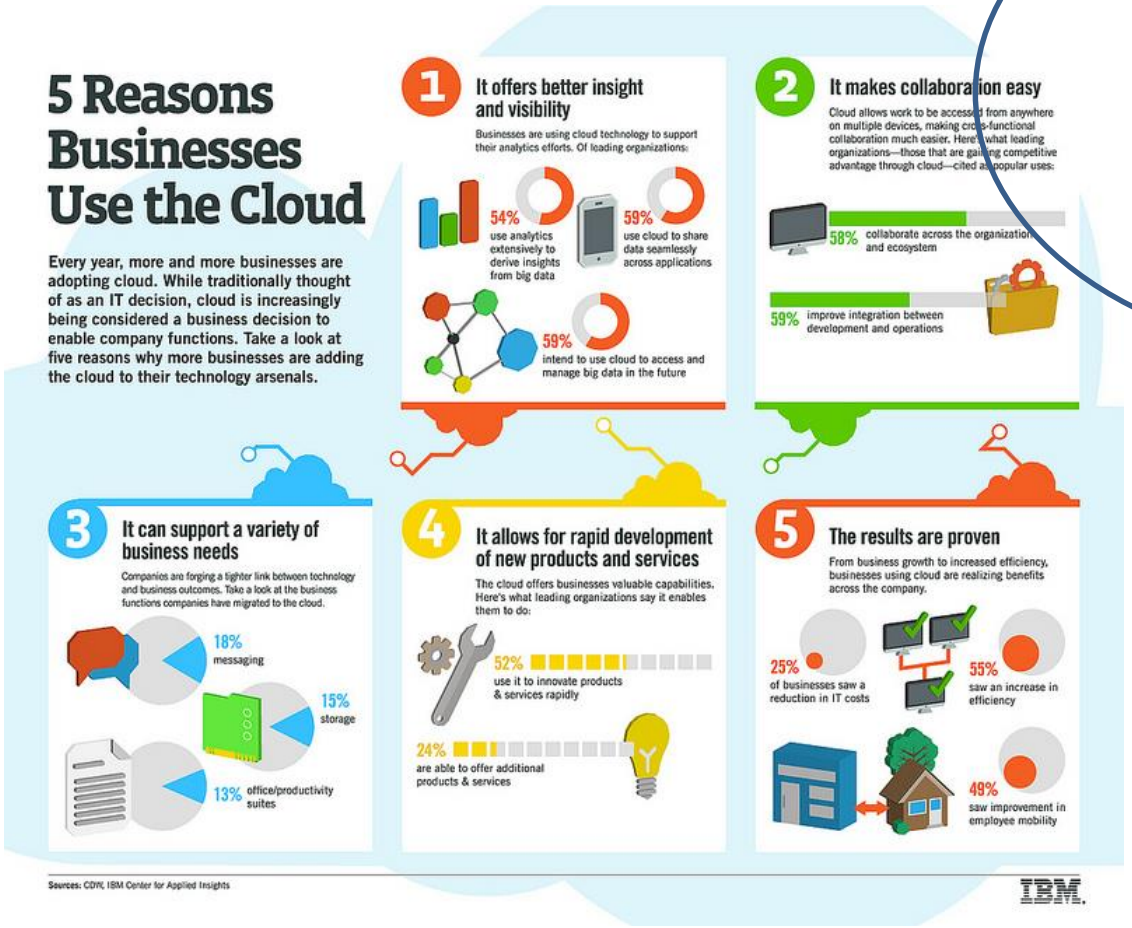
Conceptos alrededor del Big Data

La **computación en la nube**, conocida también como **servicios en la nube**, **informática en la nube**, **nube de cómputo** o **nube de conceptos** (del inglés *cloud computing*), es un paradigma que permite ofrecer servicios de computación a través de una red, que usualmente es Internet.

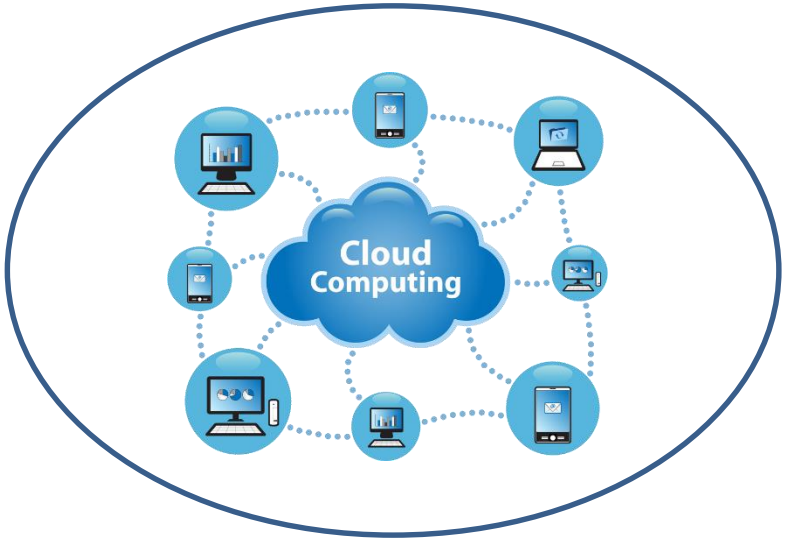


WIKIPEDIA
La enciclopedia libre

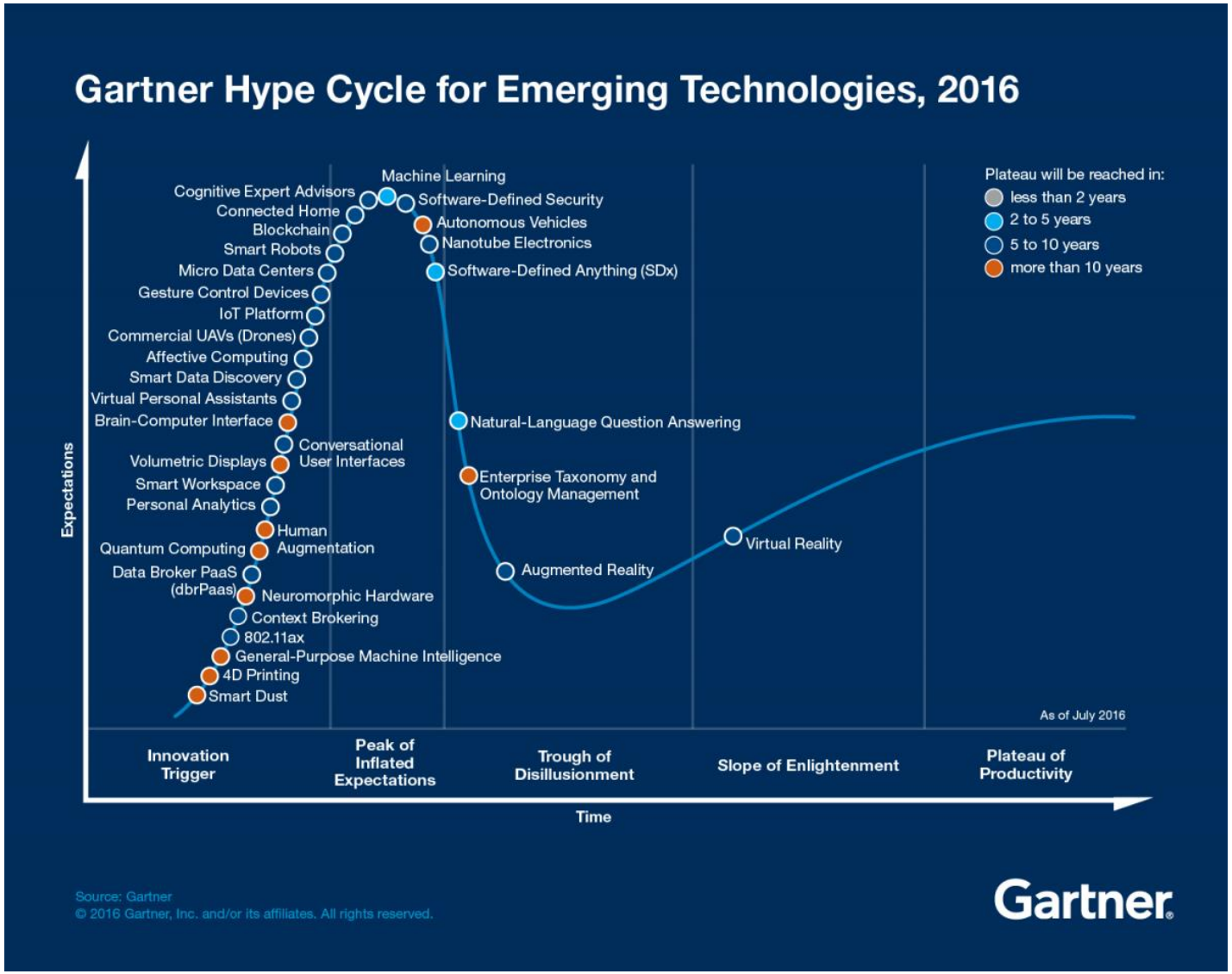
Conceptos alrededor del Big Data



Conceptos alrededor del Big Data



Conceptos alrededor del Big Data



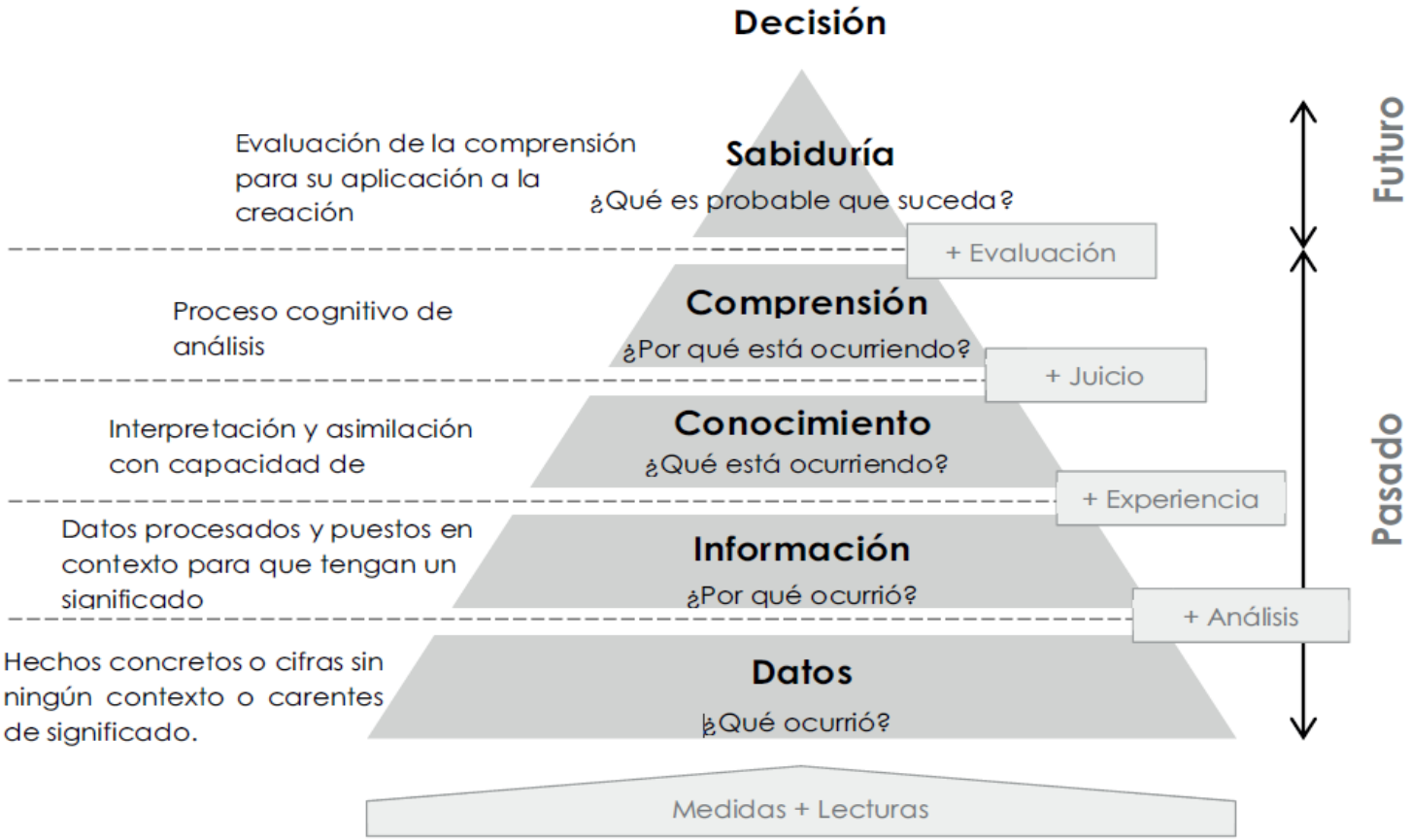
Esquema de Contenidos:

1. Introducción al Big Dada
2. Trabajando con los datos
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Trabajando con los datos

Fundamentos del Trabajo con Datos

- Descubrimiento de Conocimiento en Bases de Datos (KDD)



Trabajando con los datos

Fundamentos del Trabajo con Datos

- ¿Qué es la Inteligencia de Negocio (Business Intelligence)?
- Necesidad Creciente de Información Estratégica
  - La crisis de la información
- Sistemas Operacionales frente a la Toma de Decisiones
- Almacenes de Datos

Trabajando con los datos

El Ciclo de Vida de los Datos

- Gestión del Ciclo de Vida de los Datos

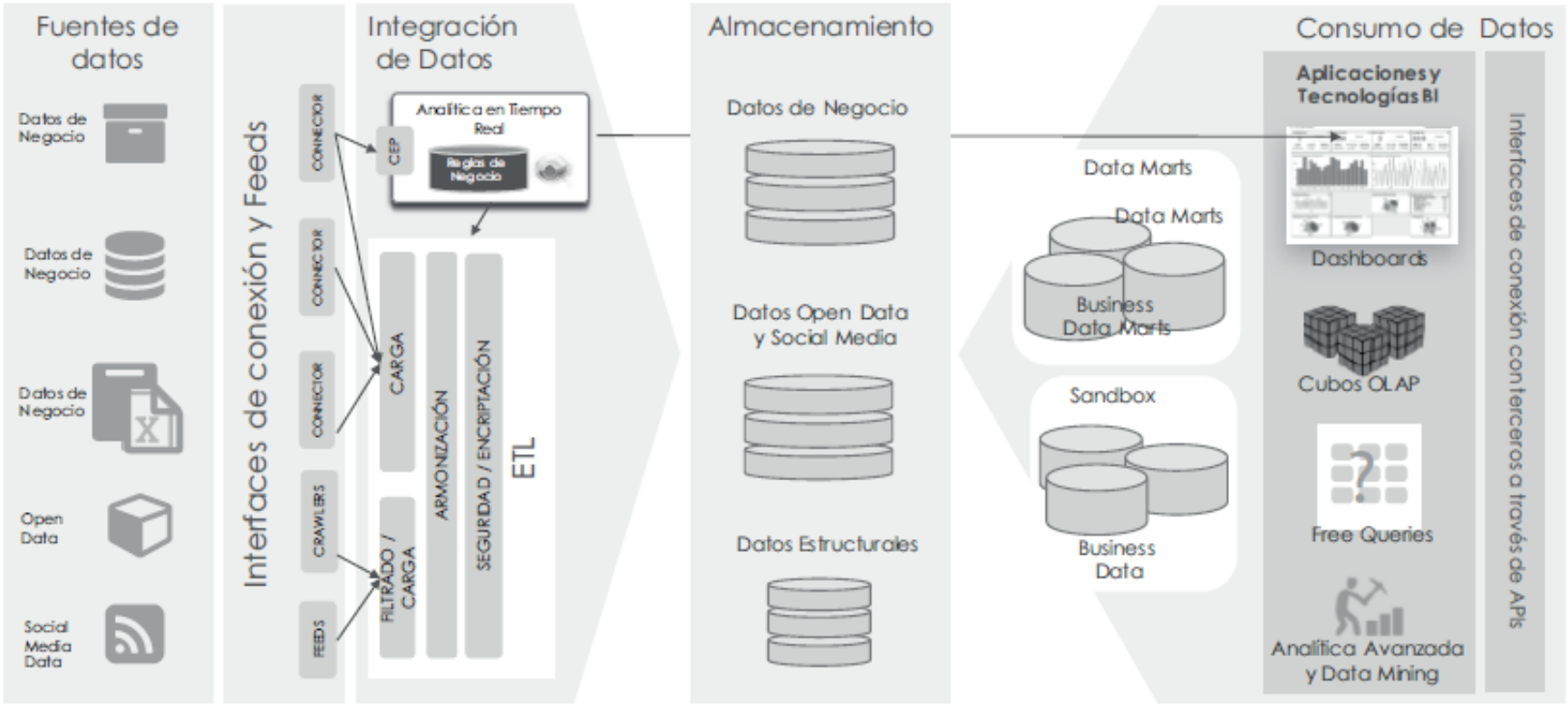
Trabajando con los datos

La Gestión de los Datos

- ¿Dispone su organización de sistemas operacionales? Identifíquelos
- ¿Qué sistemas de apoyo a la toma de decisiones utiliza su organización? ¿De dónde proceden los datos que se utilizan en estos sistemas?
- ¿Cuáles cree que pueden ser las fuentes de datos externas que aportarían valor a su empresa?
- ¿Cómo beneficiaría a su empresa el análisis de los datos procedentes de estas fuentes?

Trabajando con los datos

Componentes de un Sistema Big Data para Business Intelligence



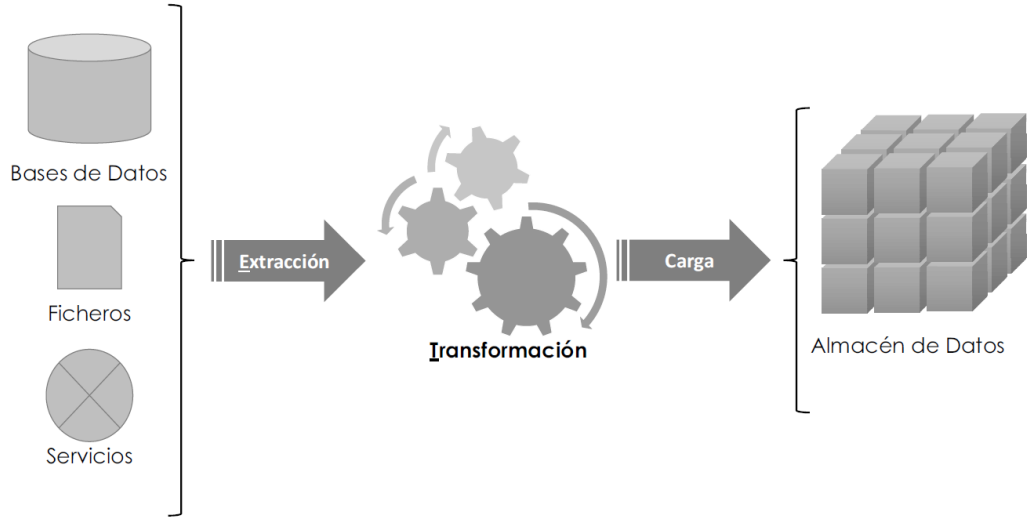
Trabajando con los datos

El Dilema del Directivo: Intuición vs. Datos

- Machine learning
- La intuición como forma de tomar decisiones.
- La gestión basada en datos (Data Driven Management).
- Proceso de transición a la toma de decisiones basada en datos.
- Ser capaz de hacer las preguntas correctas.

Trabajando con los datos

- **Extracción, transformación y carga**
  - La extracción implica "limpieza" de los datos
  - La trasformación normalmente conlleva un "cambio" de modelo
  - La carga está unida a "validación" de la calidad de los datos



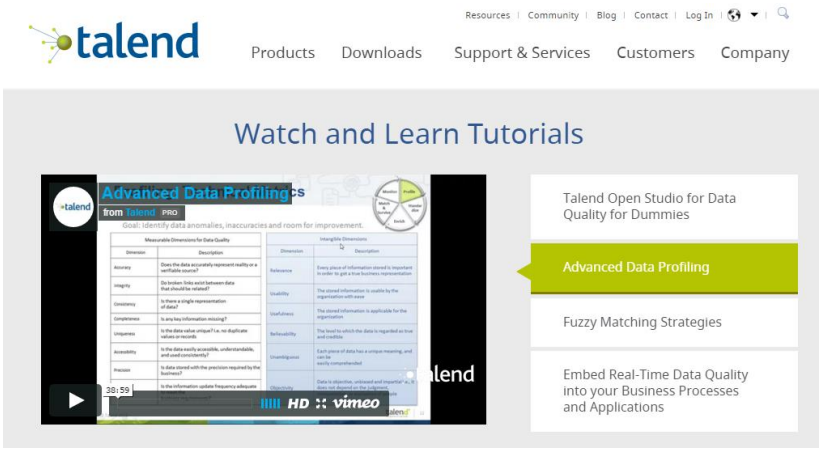
Trabajando con los datos

- **La calidad de los datos**
  - Las organizaciones actúan bajo la premisa de que la información disponible es precisa y válida. Si la información no es válida, entonces no pueden responder de las decisiones basadas en ella.

✓ El "profiling" de los datos es el análisis estadístico de los valores de los datos para evaluar su consistencia, originalidad y lógica

✓ El "sampling" (muestreo) estima la calidad de los datos mediante el análisis de un subconjunto de éstos

✓ Existen herramientas como Talend Open Studio que nos ayudan a gestionar la calidad de los datos



https://www.talend.com/products/talend-open-studio

Trabajando con los datos

- **La preparación de los datos**
  - Manipulación y transformación de los datos sin refinar, para que la información contenida en el conjunto de datos pueda ser descubierta o estar accesible de forma más fácil.
    - ✓ Operaciones: normalización, restablecimiento de incompletos, limpieza, conversión, agrupación, eliminación de redundancias, simplificación, conversión, etc.

- **Transformación de los datos**
  - Aplicación de una serie de funciones o reglas de negocio sobre los datos extraídos para convertirlos en datos que, a continuación, serán cargados en la nueva fuente.

- Ejemplo: Datos numéricos a categóricos
  - ✓ 18-30 Años: Jóvenes.
  - ✓ 31-65 Años: Adultos en edad laboral
  - ✓ 66 Años en adelante: Clientes en edad de jubilación.

Trabajando con los datos

Recolección de datos

- **¿Cómo almacenamos de manera automática la "realidad" de nuestro negocio?**

MODELO DE NEGOCIO

Construir un modelo conceptual del funcionamiento de nuestro negocio, nos permite:

- identificar las particularidades del mismo,
- recabar la información necesaria,
- utilizar los datos en beneficio del negocio

MODELO DE DATOS

MODELO CONCEPTUAL

Identifica las relaciones de más alto nivel entre las distintas entidades

Abstracto e independiente de la implementación (Entidad-Relación)

MODELO LÓGICO

Contempla entidades, atributos y relaciones de la función de negocio, para acercarse a la implementación (Modelo de datos Relacional)

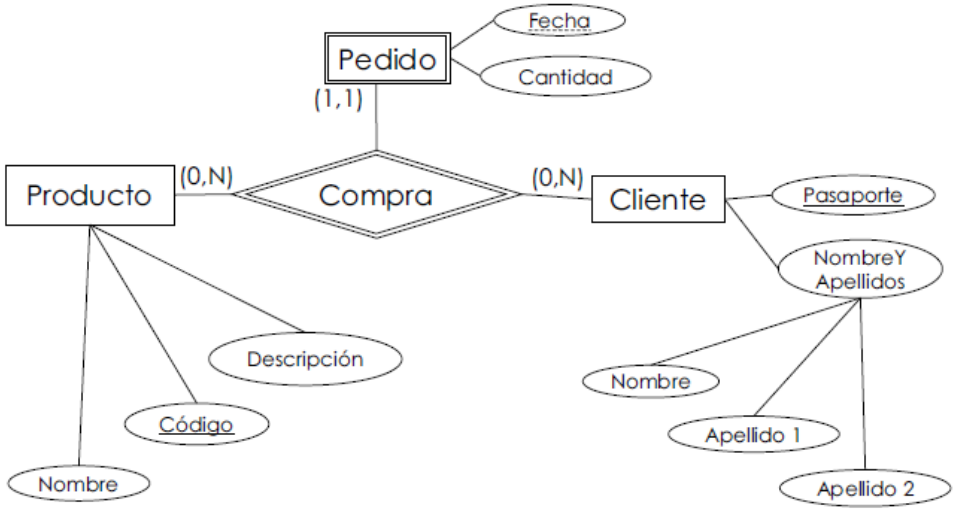
MODELO FÍSICO

Aplicación específica de un modelo de datos lógico.
(Bases de datos que se almacenan en ficheros del disco duro)

Trabajando con los datos

Recolección de datos

- **El Modelo Conceptual**
  - Trasladar el modelo de nuestro negocio –algo que utiliza un lenguaje y conceptos propios del mundo empresarial– al lenguaje y los conceptos propios del mundo de la tecnología
  - Hacer entender a los técnicos o tecnólogos nuestro negocio y sus particularidades e interrelaciones

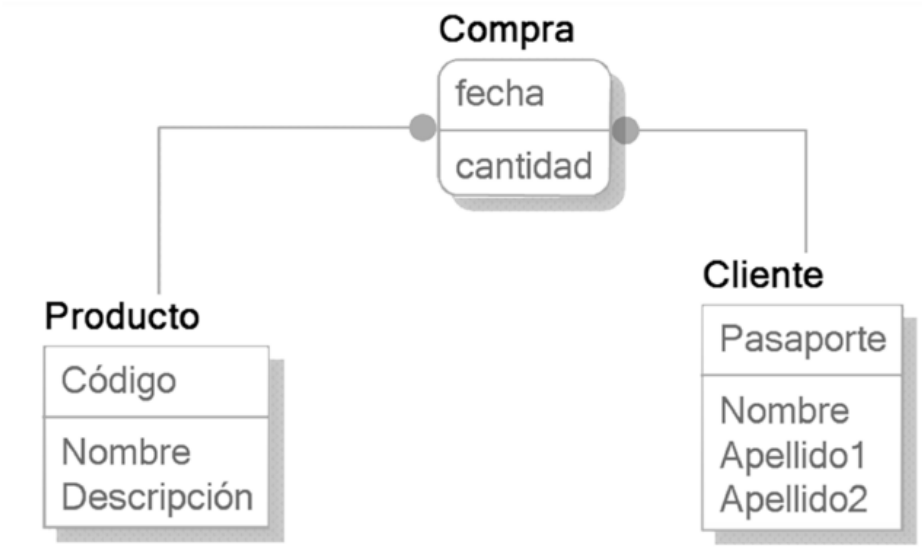


Trabajando con los datos

Recolección de datos

- **El Modelo Lógico**
  - Está comprometido con un modelo de implementación concreto. Por ejemplo, el modelo de datos relacional

- Se adecua su estructura y contenido para facilitar su implementación en un sistema gestor de bases de datos concreto

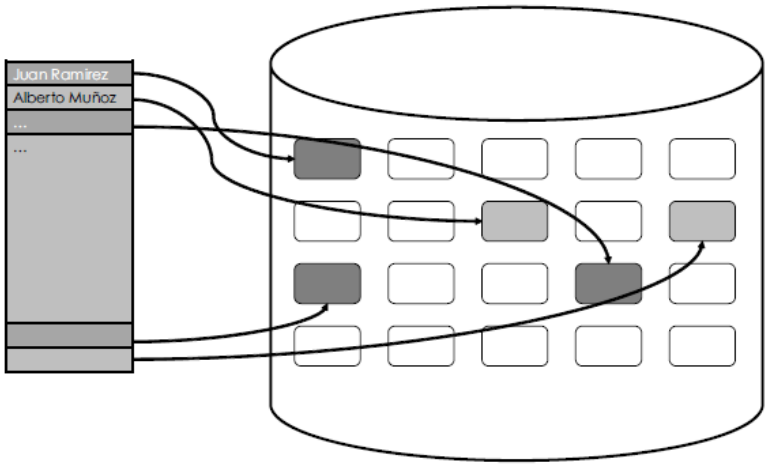


Trabajando con los datos

Recolección de datos

- **El Modelo Físico**
  - El modelo físico describe, cómo los datos que siguen el modelo lógico, son almacenados en el disco y cómo acceder eficientemente a estos datos

  - Las bases de datos en las que se apoyan la mayoría de las aplicaciones de gestión son el modelo físico que surge de la aplicación del modelo lógico definido.



  - El modelo relacional utiliza un lenguaje específico para hacer consultas a los datos almacenados conocido como lenguaje «**SQL**», acrónimo de «Structured Query Language», o lenguaje de consultas estructurado.

Esquema de Contenidos:

1. Introducción al Big Dada
2. Trabajando con los datos: Procesos ETL
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Almacenamiento y Procesamiento

Almacenamiento Masivo de Datos

- En un PC, almacenamos datos en nuestro disco duro.
- Los datos pueden ser de cualquier tipo:
  - Tablas, ficheros de texto, páginas web...
  - ...imágenes, vídeos, música...
- BIG DATA: ¿Qué pasa cuando los datos no caben en un disco duro?

Almacenamiento y Procesamiento

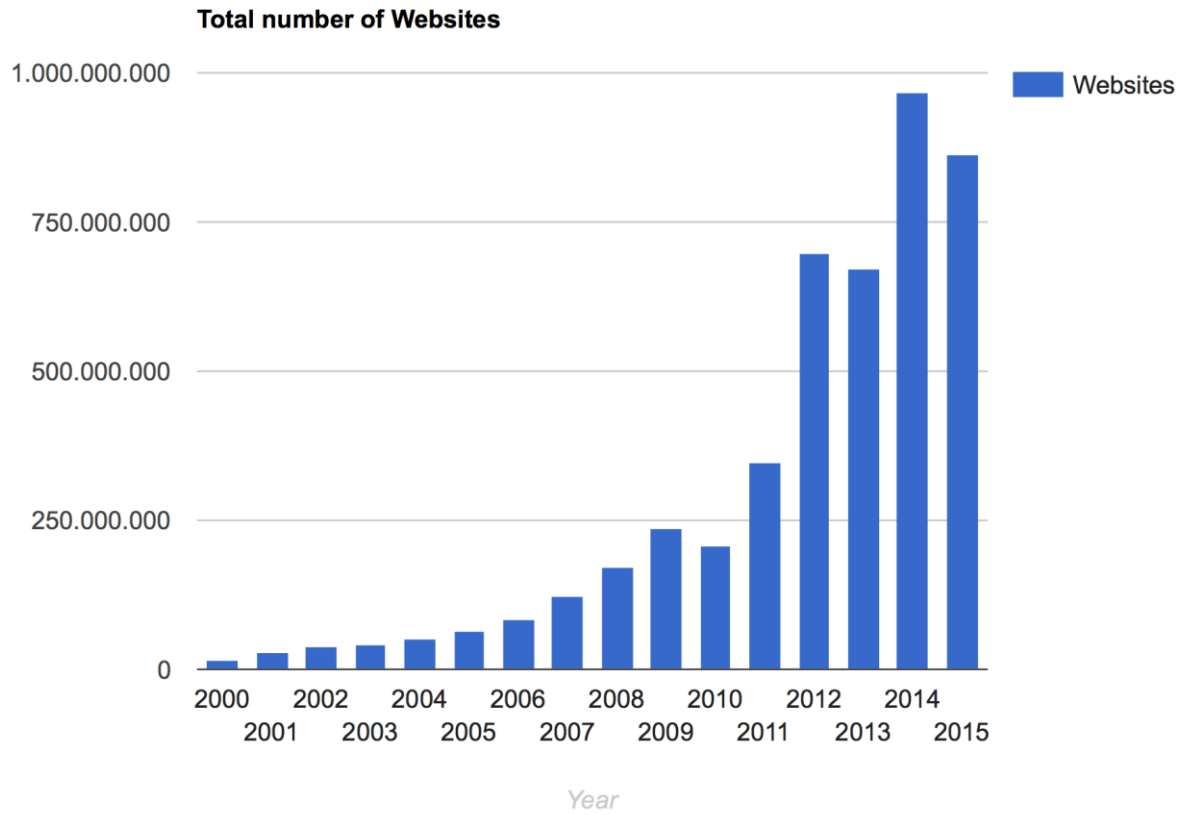
El Enfoque de Google

- Google es uno de los máximos exponentes del Big Data.
- El buscador web tiene que indexar el contenido de todas las webs posibles.
- Nace en 1998. La web ha crecido mucho desde entonces...

Almacenamiento y Procesamiento

El Enfoque de Google

- Evolución del número de sitios web (no de páginas web)



Fuente: www.internetlivestats.com/total-number-of-websites/

Almacenamiento y Procesamiento

El Enfoque de Google

- Google no es solo un motor de búsqueda.
- En los primeros 5 años de vida (1998-2003), consigue lanzar estos productos (y más)

Google books 

Google Search for Work

Google AdSense

Google Developers

Google AdWords

Blogger

Google Shopping

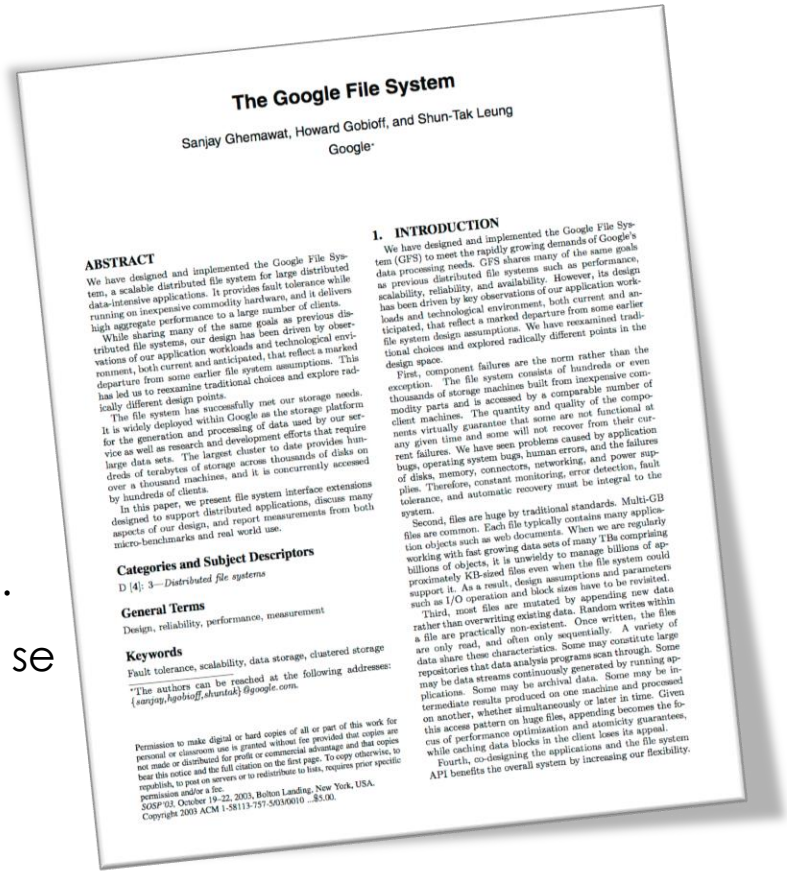
Google News

Almacenamiento y Procesamiento

Google File System

- En 2003, Google anuncia Google File System (GFS)

- Es un sistema de ficheros distribuido. Permite:
  - Almacenar grandes cantidades de datos.
  - Distribuir (repartir) los datos entre varios ordenadores.
  - Permitir lecturas de ficheros más eficientes.
  - Evitar la pérdida de datos si un ordenador se estropea.



The Google File System

Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung

Google*

ABSTRACT

We have designed and implemented the Google File System, a scalable distributed file system for large distributed data-intensive applications. It provides fault tolerance while running on inexpensive commodity hardware, and it delivers high aggregate performance to a large number of clients.

While sharing many of the same goals as previous distributed file systems, our design has been driven by observations of our application workloads and technological environment, both current and anticipated, that reflect a marked departure from some earlier file system assumptions. This has led us to reexamine traditional choices and explore radically different design points.

The file system has successfully met our storage needs. It is widely deployed within Google as the storage platform for the generation and processing of data used by our service as well as research and development efforts that require large data sets. The largest cluster to date provides hundreds of terabytes of storage across thousands of disks on over a thousand machines, and it is concurrently accessed by hundreds of clients.

In this paper, we present file system interface extensions designed to support distributed applications, discuss many aspects of our design, and report measurements from both micro-benchmarks and real world use.

Categories and Subject Descriptors

D [4]: 3—*Distributed file systems*

General Terms

Design, reliability, performance, measurement

Keywords

Fault tolerance, scalability, data storage, clustered storage

*The authors can be reached at the following addresses: {*sanjay,hgobioff,shuntak*} @google.com.*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SOSP'03, October 19–22, 2003, Bolton Landing, New York, USA.

Copyright 2003 ACM 1-58113-757-5/03/0010 ...$5.00.

1. INTRODUCTION

We have designed and implemented the Google File System (GFS) to meet the rapidly growing demands of Google's data processing needs. GFS shares many of the same goals as previous distributed file systems such as performance, scalability, reliability, and availability. However, its design has been driven by key observations of our application workloads and technological environment, both current and anticipated, that reflect a marked departure from some earlier file system design assumptions. We have reexamined traditional choices and explored radically different points in the design space.

First, component failures are the norm rather than the exception. The file system consists of hundreds or even thousands of storage machines built from inexpensive commodity parts and is accessed by a comparable number of client machines. The quantity and quality of the components virtually guarantee that some are not functional at any given time and some will not recover from their current failures. We have seen problems caused by application bugs, operating system bugs, human errors, and the failures of disks, memory, connectors, networking, and power supplies. Therefore, constant monitoring, error detection, fault tolerance, and automatic recovery must be integral to the system.

Second, files are huge by traditional standards. Multi-GB files are common. Each file typically contains many application objects such as web documents. When we are regularly working with fast growing data sets of many TBs comprising billions of objects, it is unwieldy to manage billions of approximately KB-sized files even when the file system could support it. As a result, design assumptions and parameters such as I/O operation and block sizes have to be revisited.

Third, most files are mutated by appending new data rather than overwriting existing data. Random writes within a file are practically non-existent. Once written, the files are only read, and often only sequentially. A variety of data share these characteristics. Some may constitute large repositories that data analysis programs scan through. Some may be data streams continuously generated by running applications. Some may be archival data. Some may be intermediate results produced on one machine and processed on another, whether simultaneously or later in time. Given this access pattern on huge files, appending becomes the focus of performance optimization and atomicity guarantees, while caching data blocks in the client loses its appeal.

Fourth, co-designing the applications and the file system API benefits the overall system by increasing our flexibility.

Almacenamiento y Procesamiento

Hadoop Distributed File System

- En 2004, D. Cutting y M. Cafarella liberan una versión de código abierto de GFS.
- En 2005, este sistema se denomina **HDFS** y nace el proyecto HADOOP.

Almacenamiento y Procesamiento

HDFS – Distribución

- Cada fichero se divide en bloques de 64 o 128 MB.
- Los bloques se almacenan divididos entre varios ordenadores.
  - Cada ordenador se denomina **nodo**.
  - El conjunto de ordenadores se denomina **cluster**.
- VENTAJA 1: Si se duplica el número de nodos, caben el doble de datos.
- VENTAJA 2: Se pueden almacenar ficheros que no quepan en un ordenador.
- VENTAJA 3: Las lecturas son más rápidas:

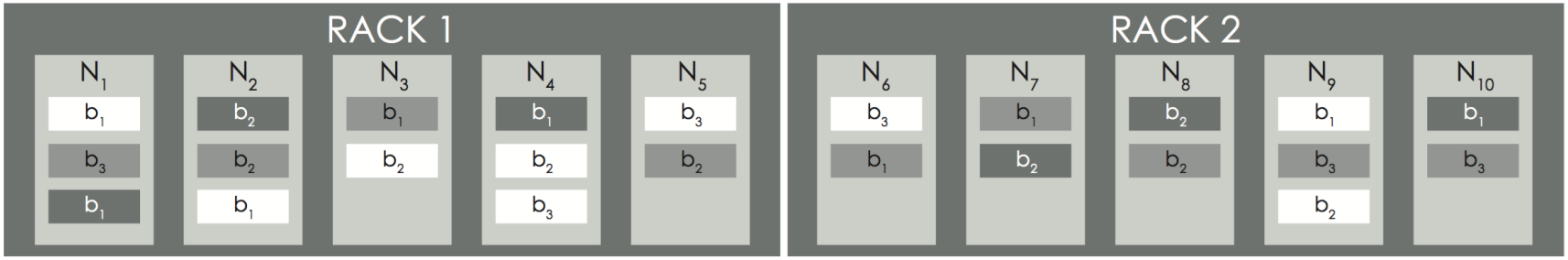
se puede leer un fichero de varios

ordenadores a la vez.

Almacenamiento y Procesamiento

HDFS – Replicación

- PROBLEMA: Si hay muchos nodos, la probabilidad de que uno falle es alta.
- SOLUCIÓN: Cada bloque se almacena en varios nodos, por defecto 3.
- Cada bloque almacenado en un nodo se denomina **réplica**.
- Las réplicas tienen en cuenta la topología: mejor almacenarlas en *racks** o centros de datos diferentes.



* Un rack o armario es un contenedor donde se almacenan varios nodos que comparten sistema de alimentación y de red.

Almacenamiento y Procesamiento

Limitaciones HDFS

- HDFS es un sistema de ficheros
- Permite almacenar datos sin estructura (cualquier tipo de ficheros).
- Los datos no están indexados.
- Es ideal cuando queremos recuperar y procesar datos de forma masiva.
- INCONVENIENTE: No es útil si queremos recuperar un conjunto pequeño de los datos.

Almacenamiento y Procesamiento

SOLUCIÓN: Bases de Datos

- Una base de datos almacena datos con una cierta estructura.
  - Esta estructura se denomina **modelo de datos**.
- Normalmente se pueden definir **índices**, que permiten recuperar datos más eficientemente.
- Operaciones habituales (CRUD)
  - Recuperación.
  - Inserción / Actualización.
  - Borrado.

Almacenamiento y Procesamiento

Bases de Datos NoSQL

- Desde hace años se han utilizado sobre todo **bases de datos relacionales**.
  - También denominadas **SQL**, por el lenguaje de consulta que implementan.
- En los últimos años se han extendido las BBDD **NoSQL**.
- Características principales (no todas aplican siempre)
  - Almacenan datos semiestructurados.
  - Son escalables (permiten almacenar Big Data).
- Cada tipo de base de datos NoSQL tiene sus propias ventajas e inconvenientes.

Almacenamiento y Procesamiento

NoSQL: BBDD Clave-Valor

- Son las bases de datos con el modelo de datos más simple.
  - Almacenan **valores** (de cualquier tipo) asociados a una **clave**.
- ESCALABILIDAD: Son muy escalables, basta con distribuir los datos.
- RENDIMIENTO: Son muy rápidas, porque las claves están indexadas.
- OPERACIONES: Su operativa es muy limitada:
  - Recuperar un valor a partir de su clave.
  - Insertar un dato (par clave-valor).
  - Borrar un valor dada su clave.

Almacenamiento y Procesamiento

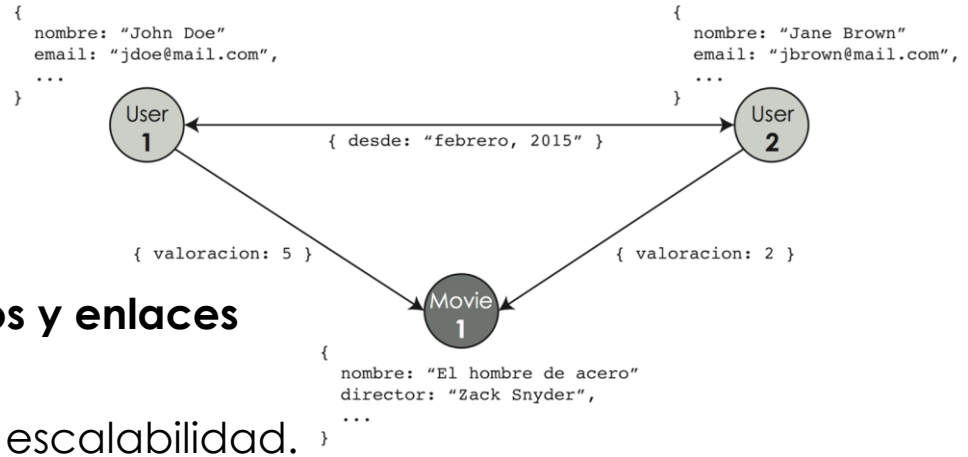
NoSQL: BBDD Documentales

- Almacenan documentos **semiestructurados**. Cada documento puede tener campos distintos.
- ESCALABILIDAD: Soportan ***sharding***, es decir, distribución de los documentos.
- RENDIMIENTO: Son rápidas, se pueden definir índices en cualquier campo.
- OPERACIONES: Permiten una amplia operativa, pero no soportan JOIN.
  - Recuperar documentos por cualquier campo, con consultas avanzadas.
  - Insertar nuevos documentos y actualizar los existentes.
  - Borrar documentos.
  - Realizar operaciones de agregación de datos.
- La más extendida es **MongoDB**.

Almacenamiento y Procesamiento

NoSQL: BBDD Orientadas a Grafos

Figura 37 – *Ejemplo de grafo representando una red social*



- Almacenan datos en forma de **nodos y enlaces**
- ESCALABILIDAD: No destacan por su escalabilidad.
- RENDIMIENTO: Son rápidas, se pueden definir índices en cualquier campo.
- OPERACIONES: Permiten una amplia operativa, soportando JOIN.
  - Recuperar documentos por cualquier campo, pudiendo recuperar sus enlaces.
  - Insertar nuevos nodos o enlaces, y actualizar los existentes.
  - Borrar nodos o enlaces.
  - Realizar operaciones de agregación de datos.
- La más extendida es **Neo4j**.

Almacenamiento y Procesamiento

NoSQL: BBDD Orientadas a Columnas

- Almacenan datos en tablas, donde cada fila es un registro y existen familias de columnas.
- Cada registro tiene una **clave** (*rowkey*), y cada familia de columnas almacena pares clave-valor.
- ESCALABILIDAD: Son muy escalables, distribuyen los datos en varios nodos.
- RENDIMIENTO: Son eficientes para recuperar datos por su *rowkey*, que está indexada.
- OPERACIONES: Soportan operaciones básicas, pero no soportan JOIN.
  - Recuperar registros, pudiendo obtener solo algunas familias de columnas o valores.
  - Insertar nuevos registros, y actualizar los existentes.
  - Borrar registros.

Almacenamiento y Procesamiento

Comparación de BBDD SQL y NoSQL

Figura 38 – *Comparación entre tecnologías de bases de datos*

	Relacionales *MySQL*	**Clave-Valor** *Voldemort*	**Documentos** *MongoDB*	**Grafos** *Neo4j*	**Columnas** *HBase*
Modelo de Datos	Tablas y relaciones	Pares clave-valor	Documentos BSON/JSON	Nodos y enlaces	Tablas con columnas no fijas
Relaciones (JOIN)	✔	✖	✖	✔	✖
Transacciones ACID	✔	✖	✖	✔	✖
Índices	✔	Solo en la clave	✔	✔	Solo en la clave de fila (*rowkey*)
Datos semiestructurados	✖	✔	✔	✔	✔
Escalabilidad horizontal	✖	✔	✔	✖	✔
Replicación de datos	✔	✔	✔	✔	✔

Almacenamiento y Procesamiento

Almacenamiento de Big Data en la Nube

- Los sistemas de Cloud Storage nos permiten almacenar grandes cantidades de datos:
  - Ficheros
  - Datos estructurados en BBDD relacionales
  - Datos semiestructurados en BBDD no relacionales
- No tenemos que adquirir equipos: ahorramos en discos duros, ordenadores, etc.
- No tenemos que preocuparnos de gestionar *backups*, la empresa lo gestiona por nosotros.
- Pagamos por la cantidad de datos que almacenamos, o bien por las veces que accedemos a ellos.

Almacenamiento y Procesamiento

Almacenamiento de Ficheros

- Existe el concepto de "*bucket*" (cubo).
  - Un cubo es un espacio donde almacenamos ficheros y directorios.
- Dentro de cada *bucket*, encontramos una jerarquía de directorios y ficheros, como es habitual.
- El servicio de *cloud* gestiona por nosotros la disponibilidad, *backups*, etc.
- Principales competidores:
  - Amazon S3
  - Google Cloud Storage

Almacenamiento y Procesamiento

Bases de Datos NoSQL

- Principal solución: Amazon DynamoDB.
- El modelo de datos está compuesto por tablas.
- Cada tabla contiene objetos, que a su vez contienen atributos.
- Los objetos deben tener una clave primaria. El resto de atributos no está definido a priori.
- Escala horizontalmente para almacenar Big Data.

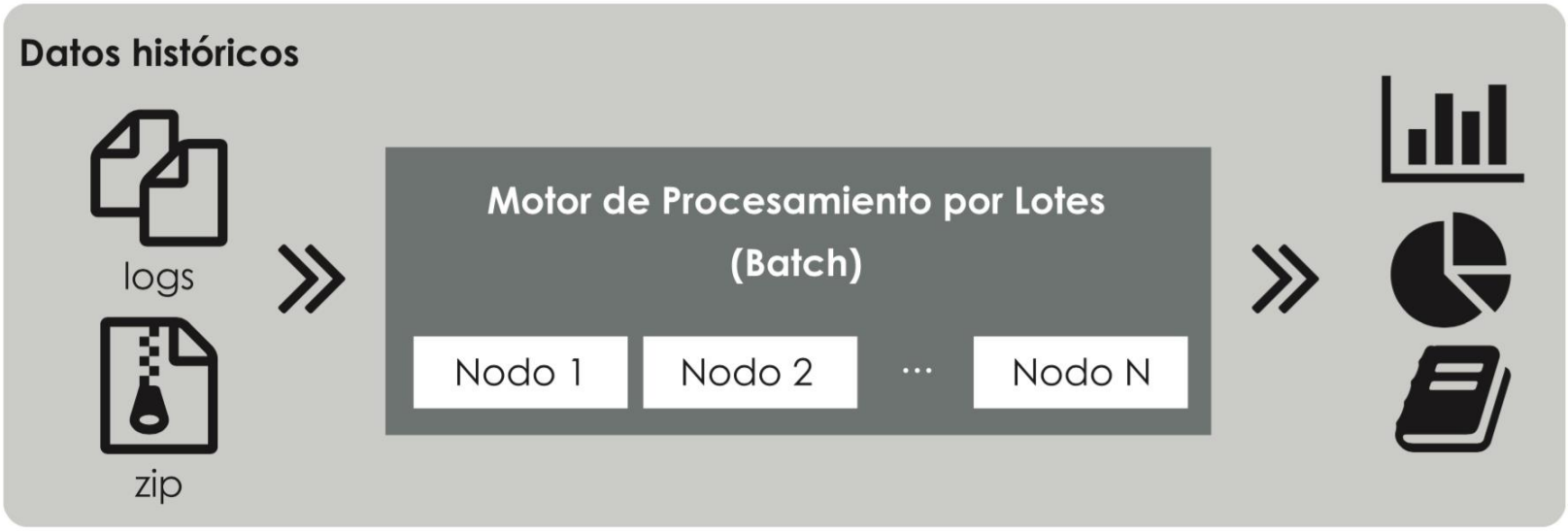
Almacenamiento y Procesamiento

Procesamiento de Big Data

- Dos paradigmas principales:
  - Por lotes.
  - En tiempo real.
- Procesamiento por lotes:
  - Se analizan grandes volúmenes de datos históricos.
  - El procesamiento puede llevar minutos, horas, días...
- Procesamiento en tiempo real:
  - Se analizan grandes volúmenes de datos que llegan a gran velocidad.
  - El procesamiento debe ser inmediato.

Almacenamiento y Procesamiento

Procesamiento de Big Data por Lotes (Batch)



Almacenamiento y Procesamiento

El (Nuevo) Enfoque de Google

- En 2004, Google sigue lanzando nuevos productos...

Picasa
Google Shopping
Google+ Local
orkut
Gmail by Google
Google
Google books
Google AdSense
Google Developers
Google AdWords
Blogger
Google News
Google Search for Work

Almacenamiento y Procesamiento

La Solución de Google

- En 2004, Google anuncia MapReduce
- Es un *framework* para el procesamiento de datos. Permite:
  - Procesar datos almacenados sobre GFS.
  - Hacerlo de manera distribuida y paralela.
  - Evitar la pérdida de datos si un ordenador se estropea.

MapReduce: Simplified Data Processing on Large Clusters

Jeffrey Dean and Sanjay Ghemawat
jeff@google.com, sanjay@google.com
Google, Inc.

Abstract

MapReduce is a programming model and an associated implementation for processing and generating large data sets. Users specify a *map* function that processes a key/value pair to generate a set of intermediate key/value pairs, and a *reduce* function that merges all intermediate values associated with the same intermediate key. Many real world tasks are expressible in this model, as shown in the paper.

Programs written in this functional style are automatically parallelized and executed on a large cluster of commodity machines. The run-time system takes care of the details of partitioning the input data, scheduling the program's execution across a set of machines, handling machine failures, and managing the required inter-machine communication. This allows programmers without any experience with parallel and distributed systems to easily utilize the resources of a large distributed system.

Our implementation of MapReduce runs on a large cluster of commodity machines and is highly scalable: a typical MapReduce computation processes many terabytes of data on thousands of machines. Programmers find the system easy to use: hundreds of MapReduce programs have been implemented and upwards of one thousand MapReduce jobs are executed on Google's clusters every day.

1 Introduction

Over the past five years, the authors and many others at Google have implemented hundreds of special-purpose computations that process large amounts of raw data, such as crawled documents, web request logs, etc., to compute various kinds of derived data, such as inverted indices, various representations of the graph structure of web documents, summaries of the number of pages crawled per host, the set of most frequent queries in a given day, etc. Most such computations are conceptually straightforward. However, the input data is usually large and the computations have to be distributed across hundreds or thousands of machines in order to finish in a reasonable amount of time. The issues of how to parallelize the computation, distribute the data, and handle failures conspire to obscure the original simple computation with large amounts of complex code to deal with these issues.

As a reaction to this complexity, we designed a new abstraction that allows us to express the simple computations we were trying to perform but hides the messy details of parallelization, fault-tolerance, data distribution and load balancing in a library. Our abstraction is inspired by the *map* and *reduce* primitives present in Lisp and many other functional languages. We realized that most of our computations involved applying a *map* operation to each logical "record" in our input in order to compute a set of intermediate key/value pairs, and then applying a *reduce* operation to all the values that shared the same key, in order to combine the derived data appropriately. Our use of a functional model with user-specified map and reduce operations allows us to parallelize large computations easily and to use re-execution as the primary mechanism for fault tolerance.

The major contributions of this work are a simple and powerful interface that enables automatic parallelization and distribution of large-scale computations, combined with an implementation of this interface that achieves high performance on large clusters of commodity PCs.

Section 2 describes the basic programming model and gives several examples. Section 3 describes an implementation of the MapReduce interface tailored towards our cluster-based computing environment. Section 4 describes several refinements of the programming model that we have found useful. Section 5 has performance measurements of our implementation for a variety of tasks. Section 6 explores the use of MapReduce within Google including our experiences in using it as the basis

To appear in OSDI 2004

Almacenamiento y Procesamiento

Hadoop MapReduce

- En 2005, una versión de código abierto de **MapReduce** se incluye en HADOOP.

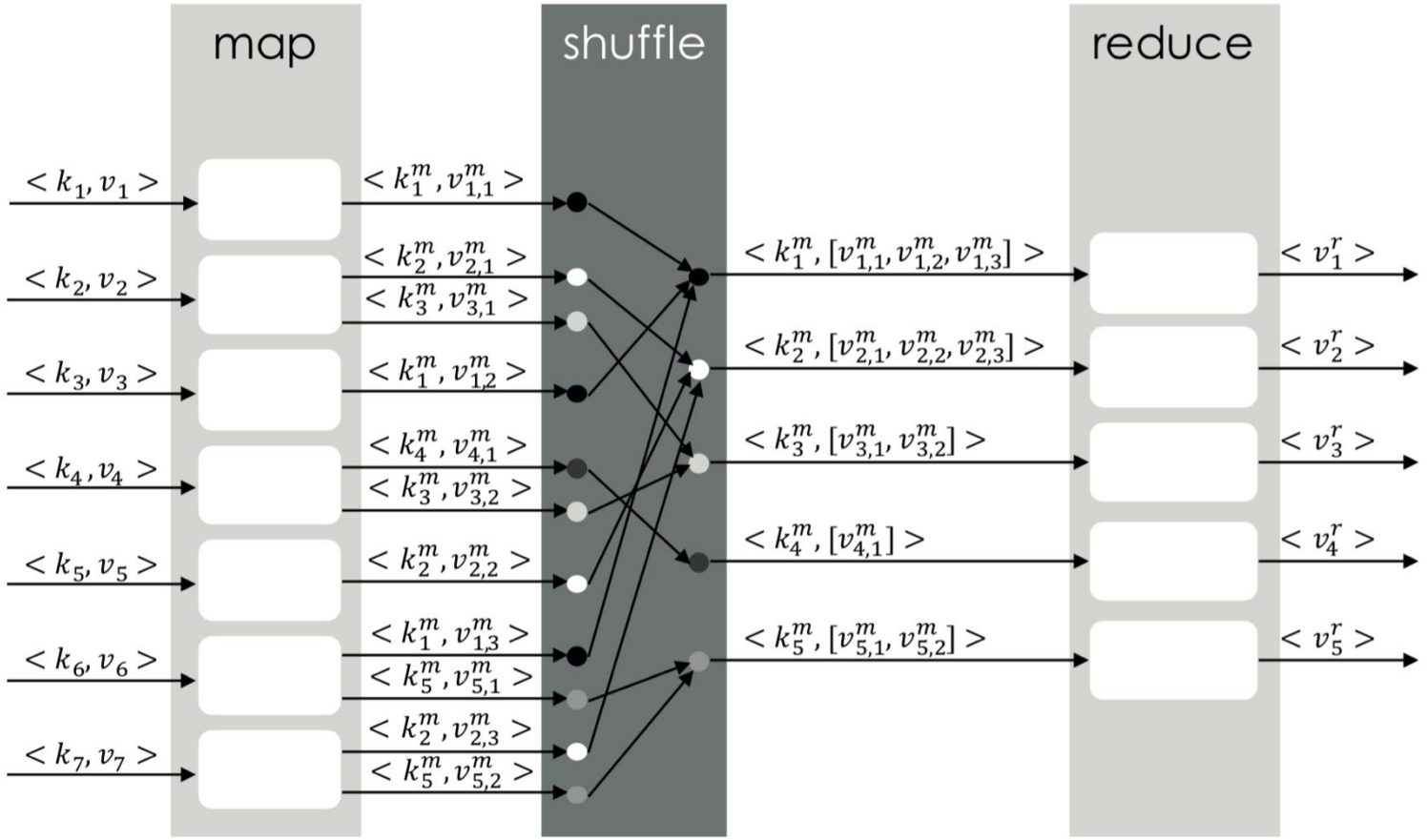
Almacenamiento y Procesamiento

El Paradigma MapReduce

- Consta de dos rutinas principales.
- La rutina **map** realiza una transformación de los datos.
  - ENTRADA: una tupla clave-valor.
  - SALIDA: una o varias tuplas clave-valor.
- Una fase intermedia denominada **shuffle** se encarga de agrupar la salida del map por claves.
- La rutina **reduce** realiza una agrupación o agregación de los datos:
  - ENTRADA: una clave, junto con todos sus valores asociados.
  - SALIDA: un valor.

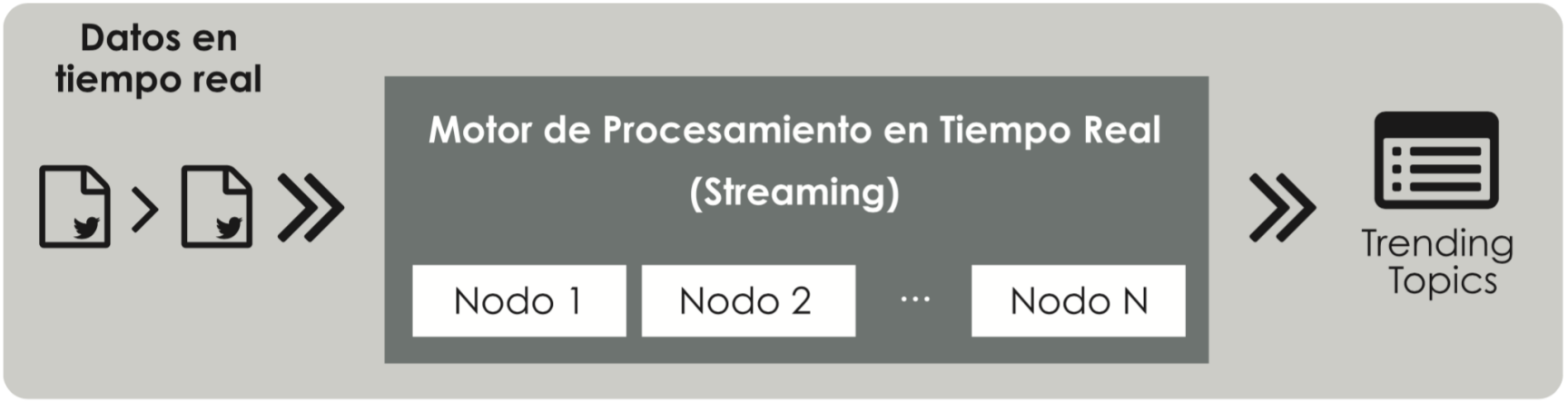
Almacenamiento y Procesamiento

El Paradigma MapReduce



Almacenamiento y Procesamiento

Procesamiento de Big Data en Tiempo Real (Streaming)



Almacenamiento y Procesamiento

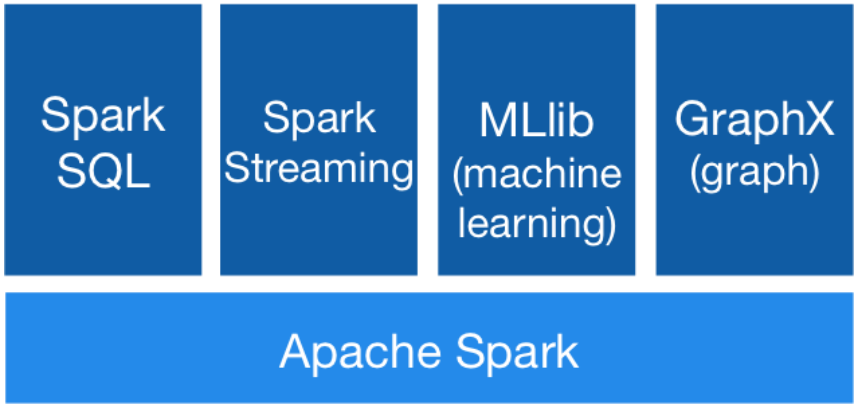
La Revolución del Procesamiento de Datos: Apache Spark

- Apache Spark nace en 2010, y es creado por Matei Zaharia.
- Se le considera el sucesor de Hadoop, y es muy utilizado por la comunidad del Big Data.
- En 2014 Spark bate un record: ordena 100 TB de datos en solo 23 minutos.
- Su clave es el concepto de "RDD": *Resilient Distributed Dataset*.
  - Es una abstracción de un conjunto de datos.
  - El programador trabaja con él como lo haría normalmente.
  - Pero por detrás, el conjunto de datos está distribuido.

Almacenamiento y Procesamiento

Spark: una Solución Polivalente

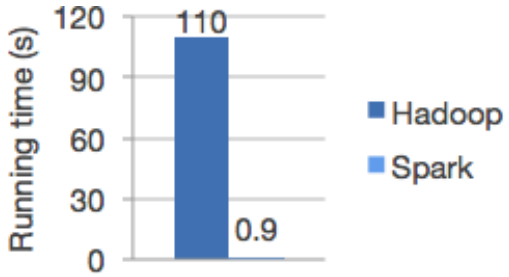
- Spark está compuesto por varios módulos que permiten abordar muchos problemas:
  - **Spark Core**: procesamiento por lotes de datos, incluyendo operaciones MapReduce.
  - **Spark Streaming**: procesamiento en tiempo real de Big Data.
  - **Spark MLlib**: aprendizaje automático sobre Big Data.
  - **Spark SQL**: consultas SQL sobre Big Data.
  - **Spark GraphX**: procesamiento y análisis de grandes grafos.



Almacenamiento y Procesamiento

Spark vs. Hadoop

- Spark se anuncia como el competidor de Hadoop, al ser más rápido.
  - Esto se debe a que realiza optimizaciones en memoria.
- Spark es cómodo para los desarrolladores.
  - Se integra con Java, Scala y Python.
- Spark y Hadoop se complementan:
  - Spark puede instalarse en un cluster de Hadoop.
  - Puede leer datos de HDFS-




```
text_file = spark.textFile("hdfs://...")

text_file.flatMap(lambda line: line.split())
    .map(lambda word: (word, 1))
    .reduceByKey(lambda a, b: a+b)
```

Almacenamiento y Procesamiento

Procesamiento por Lotes en la Nube

- Los servicios de *Cloud Processing* nos permiten:
  - Procesar grandes cantidades de datos.
  - Sin necesidad de invertir en comprar máquinas potentes.
  - Apagar las máquinas (interrumpiendo el gasto) cuando haya finalizado el procesamiento.
  - Escalabilidad: crear más máquinas cuando haya más datos que procesar.
  - Elasticidad: mejorar las capacidades de nuestras máquinas cuando sea necesario.
- Se denomina ***instancia*** a una máquina virtual que procesa datos en la nube.

Almacenamiento y Procesamiento

Amazon Elastic MapReduce

- Amazon Elastic MapReduce permite iniciar un *cluster* de Hadoop en minutos.
- Podemos definir el número de máquinas y su rendimiento.
- Además, podemos escoger las herramientas que queremos instalar.k
- Nos permite un procesamiento por lotes rápido y económico

Almacenamiento y Procesamiento

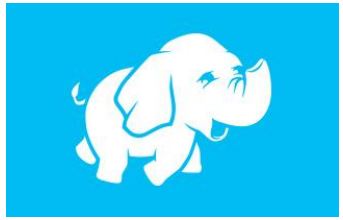
Google BigQuery

- Permite realizar consultas sobre terabytes de datos en pocos minutos.
- Se emplea el lenguaje SQL, lo que simplifica la consulta sobre los datos.
- Es menos flexible que Hadoop MapReduce: los datos deben ser estructurados.
- Permite un procesamiento por lotes cómodo, sencillo y económico.
- Google cobra por el volumen de datos leídos.

Almacenamiento y Procesamiento

Azure Spark

- Azure es el grupo de servicios en la nube de Microsoft.
- Azure HDInsights proporciona servicios de Big Data: Hadoop, Storm, Spark...
- Azure Spark permite desplegar un *cluster* de Spark en cuestión de minutos.
- Una vez creado, podemos utilizar un *notebook* para desarrollar aplicaciones con Spark.



Esquema de Contenidos:

1. Introducción al Big Dada
2. Trabajando con los datos
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Análisis mediante la Minería de Datos

Análisis de Big Data

- Analizar datos supone sacar un valor de los mismos:
  - Algo que no se ve a simple vista en los datos.
  - Algo que proporciona un valor de negocio.
  - Algo que nos ayuda a verificar o desmentir una hipótesis.
- Dos tipos de análisis importantes son el **análisis predictivo** y el **análisis de patrones**.
- Se enmarcan dentro del ámbito del **aprendizaje automático** (*machine learning*).

Análisis mediante la Minería de Datos

Ejemplos de Problemas de Aprendizaje Automático

- Detectar si un correo electrónico es *spam* o no lo es.
- Predecir si un gasto con tarjeta de crédito es legítimo o fraudulento.
- Predecir si un cliente va a tener problemas para pagar un crédito en función de su historial bancario.
- Estimar el gasto que va a realizar un cliente en nuestro comercio en función de su perfil demográfico.
- Segmentar a nuestros usuarios o clientes en función de su perfil, para personalizar nuestras ofertas.
- Recomendar un producto a un usuario en función de sus potenciales intereses.

Análisis mediante la Minería de Datos

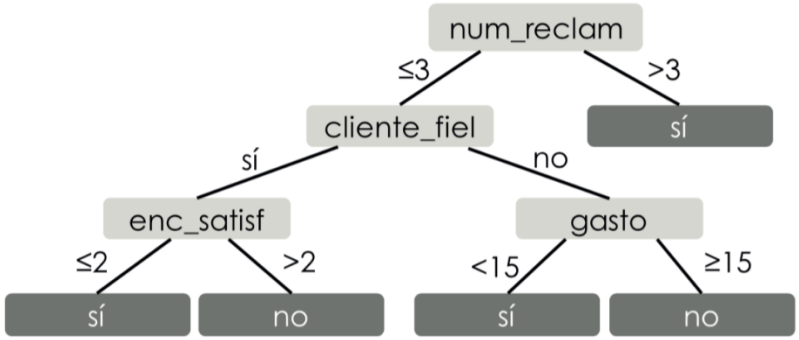
Análisis Predictivo

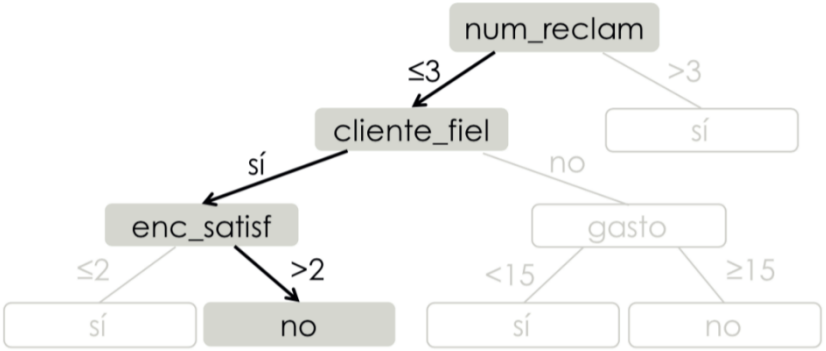
- El propósito es aprender un modelo a partir de los datos existentes.
- Este modelo servirá para realizar predicciones con nuevos datos.
- Algo de vocabulario:
  - Una **instancia** es un dato.
  - Un **atributo** es una característica sobre nuestros datos.
  - La **clase** (o **salida**) es el valor que queremos predecir o estimar.
- Dos tipos de problemas similares, pero con técnicas distintas:
  - **Clasificación**: la salida es categórica, es decir, una etiqueta de entre varias posibles.
  - **Regresión**: la salida es numérica.

Análisis mediante la Minería de Datos

Análisis Predictivo: Árboles de Decisión / Regresión

- Un árbol de decisión permite ir "haciendo preguntas" a los datos para ir bajando por ramas del árbol.
- Cada nodo del árbol es una pregunta a los datos.
- Cuando llegamos a una hoja (no hay más ramas), entonces tenemos nuestra salida.

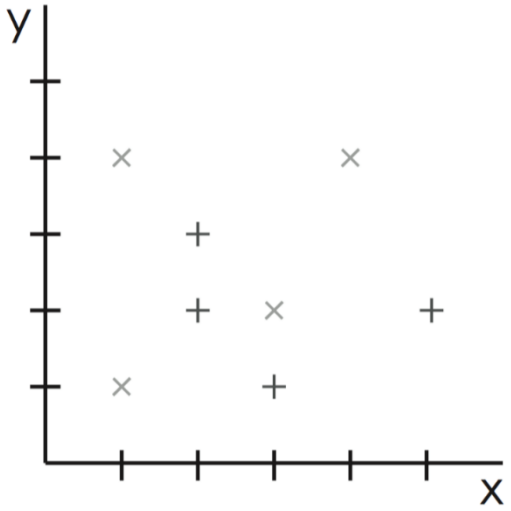


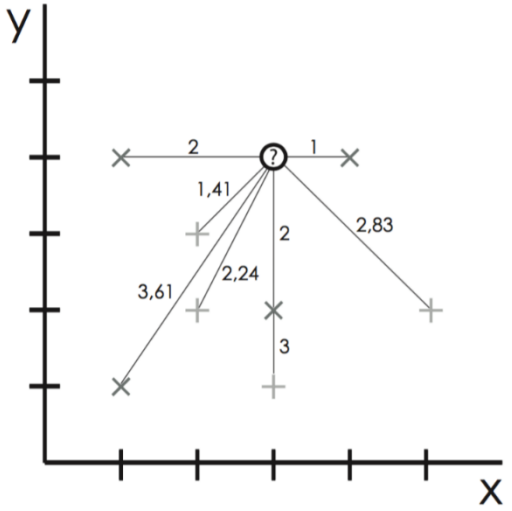


Análisis mediante la Minería de Datos

Análisis Predictivo: Modelos Geométricos

- Se basan en funciones de distancia para estimar la clase o salida a predecir.
- En resumen: ¿cuáles son los datos más parecidos a este que quiero predecir?
- La técnica más empleada es *k-Nearest Neighbors* (k-NN).

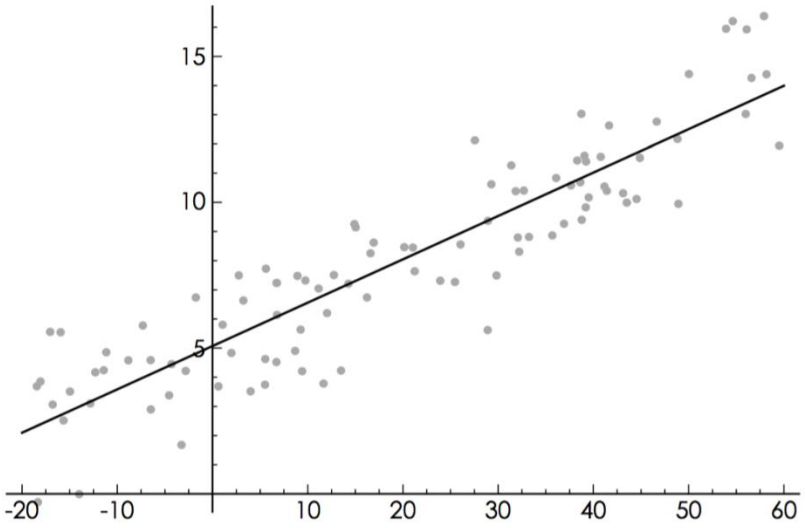




Análisis mediante la Minería de Datos

Análisis Predictivo: Regresión Lineal

- Dado un conjunto de puntos en el espacio, queremos aprender un plano (o recta) que los ajuste.
- El objetivo es minimizar el error, es decir, la distancia entre el plano y los puntos.
- La ecuación de este plano (o recta) nos servirá para estimar la salida de los nuevos datos.



Análisis mediante la Minería de Datos

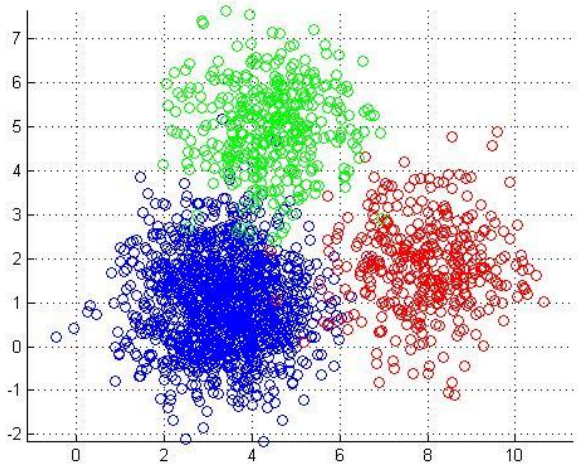
Análisis de Patrones

- No se busca predecir una clase o salida.
- En su lugar, se busca agrupar datos que son parecidos.
- Ejemplo: ¿qué usuarios son parecidos entre sí?
- También se denomina ***clustering***, agrupación o segmentación.

Análisis mediante la Minería de Datos

Análisis de Patrones

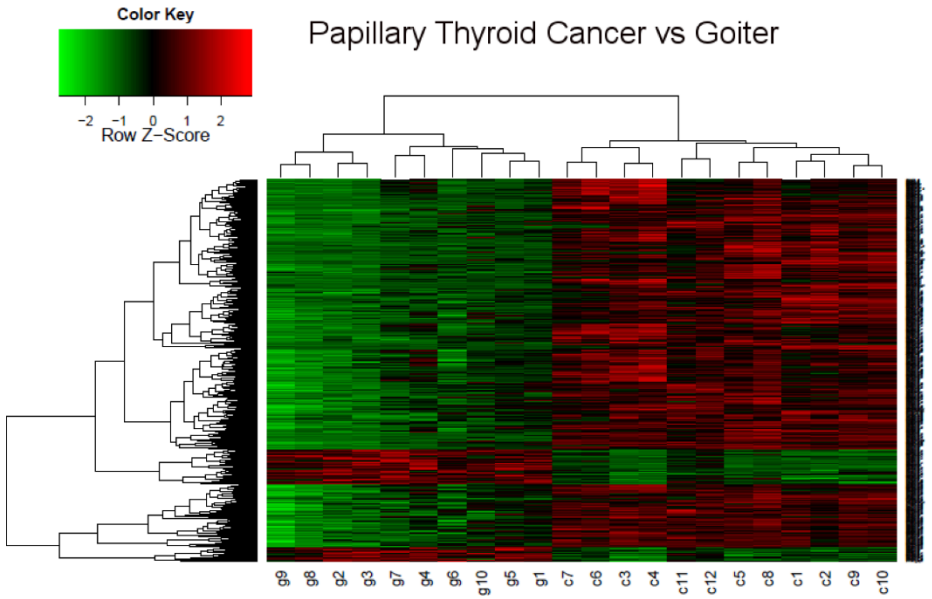
- No se busca predecir una clase o salida.
- En su lugar, se busca agrupar datos que son parecidos.
- Ejemplo: ¿qué usuarios son parecidos entre sí?
- También se denomina ***clustering***, agrupación o segmentación.
- Se tiene que definir una medida de distancia (métrica) entre los datos.



Análisis mediante la Minería de Datos

Análisis de Patrones: técnicas jerárquicas

- Pueden ser **aglomerativas** o **divisivas**.
- En las aglomerativas, se van uniendo los datos más cercanos de dos en dos en cada paso.
- En las divisivas, todos los datos comienzan en un único ***cluster*** y se van separando en cada paso.



Análisis mediante la Minería de Datos

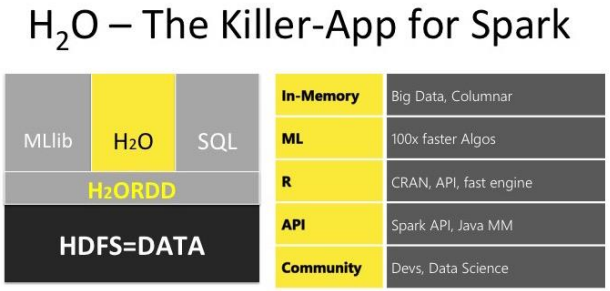
Recomendación

- La recomendación combina análisis de patrones y análisis predictivo.
- Por un lado, queremos segmentar usuarios o clientes parecidos entre sí.
- Por otro lado, queremos segmentar productos parecidos entre sí.
- A continuación, realizamos predicción: ¿qué producto le gustará a un usuario?
- Para ello, miramos los usuarios parecidos a él, y los productos que han adquirido.
- Existen dos enfoques: **colaborativo** o **basado en contenidos**.

Análisis mediante la Minería de Datos

Análisis Escalable de Big Data

- El análisis de Big Data introduce nuevos desafíos.
- El software clásico de aprendizaje automático puede dar problemas cuando:
  - Los conjuntos de datos tienen muchas instancias.
  - Los datos tienen muchos atributos.
- Existe software específico para realizar aprendizaje automático con Big Data:
  - Apache Mahout
  - Apache Spark Mllib
  - Apache Flink
  - H2O



Análisis mediante la Minería de Datos

Análisis de Big Data en la Nube

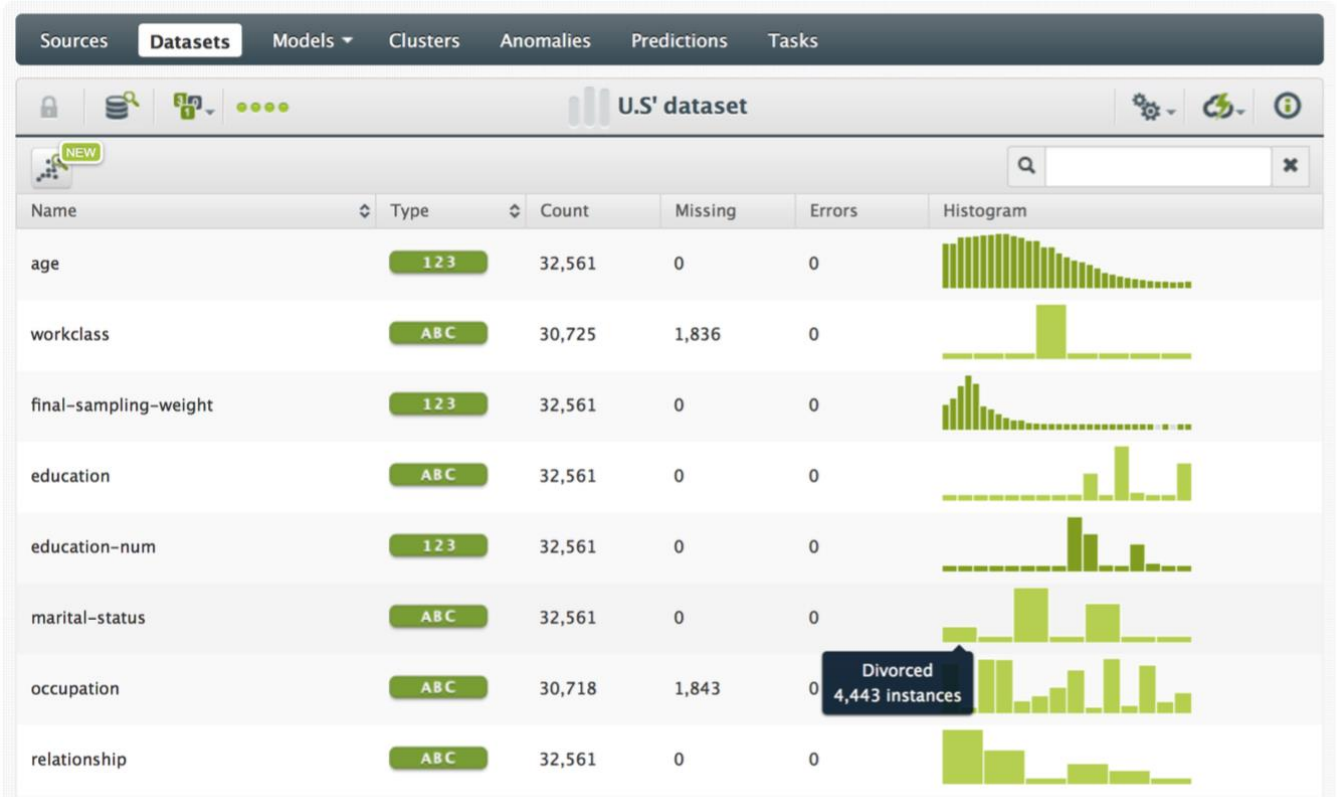
- Otra opción es emplear herramientas para realizar aprendizaje automático en la nube.
- Los grandes competidores ofrecen herramientas de este tipo:
  - Amazon ML.
  - Google Cloud Prediction.
  - Microsoft Azure ML.
- Una opción más intuitiva es BigML.

https://bigml.com/

Análisis mediante la Minería de Datos

BigML: Gestión de Datos

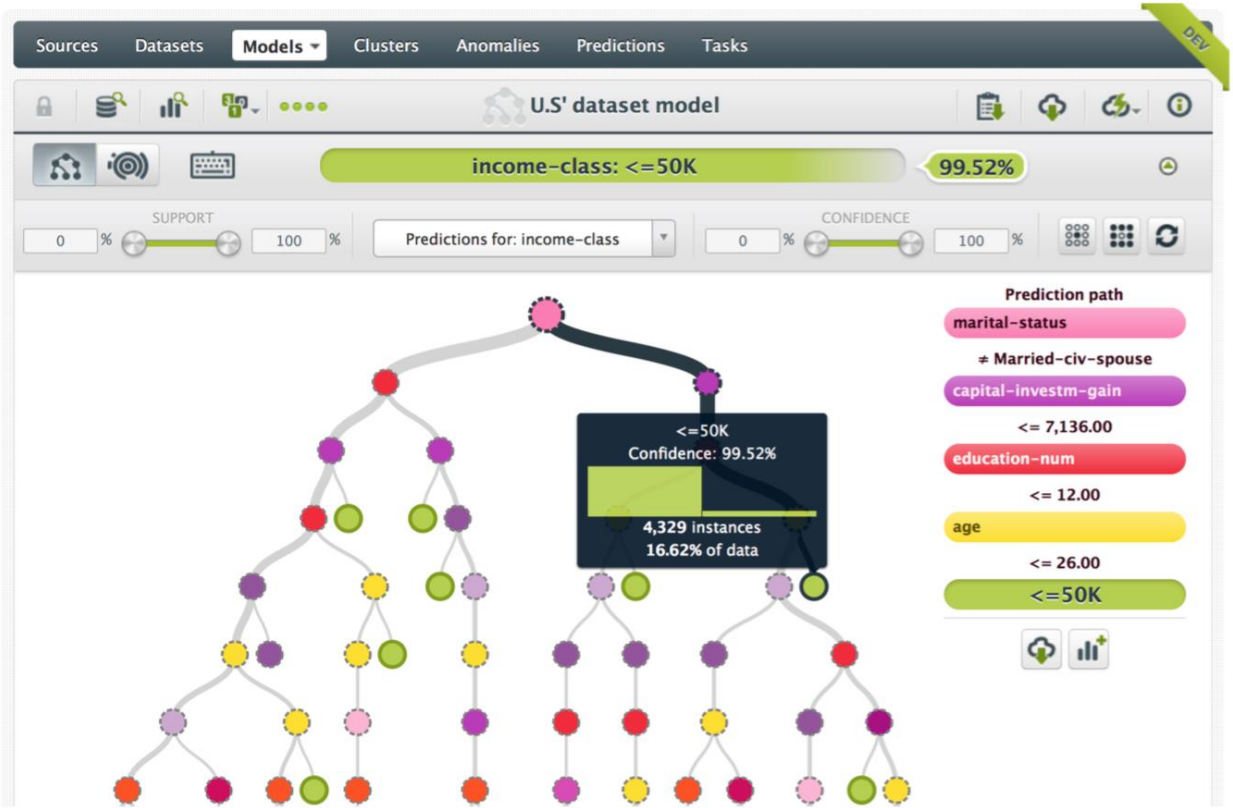
- BigML permite gestionar los datos, mostrando las distribuciones de valores de cada atributo.



Análisis mediante la Minería de Datos

BigML: Análisis Predictivo

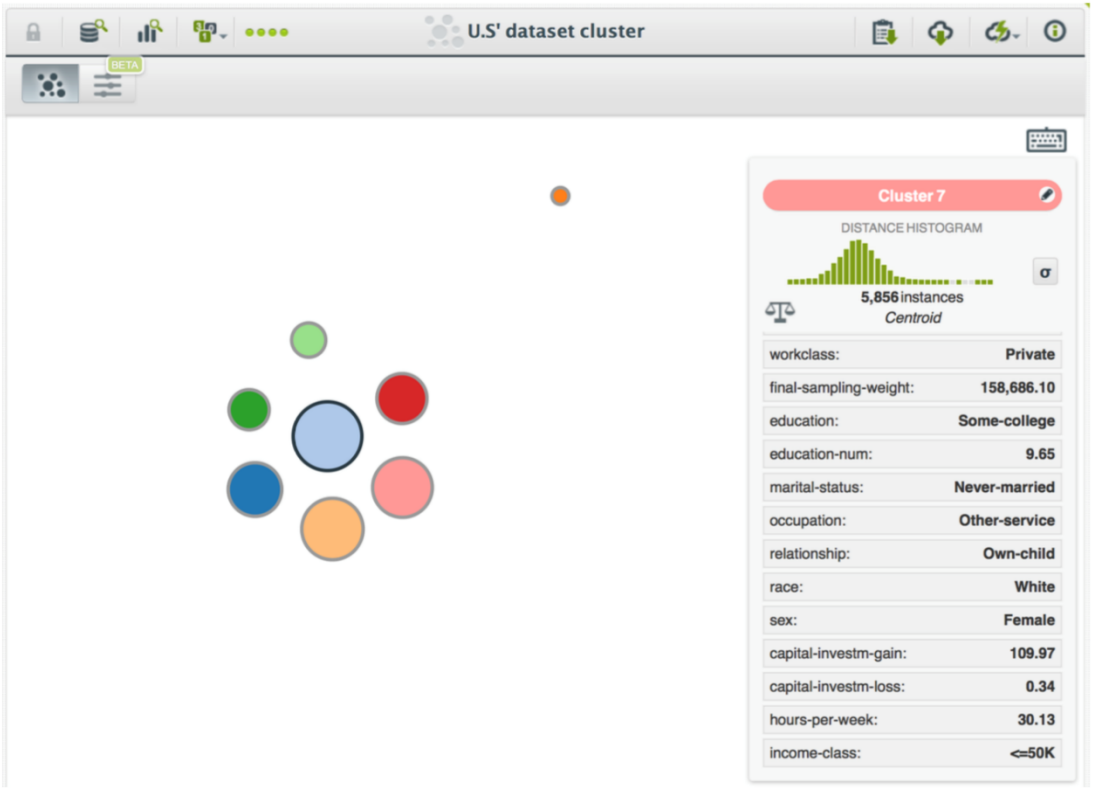
- BigML permite aprender automáticamente árboles de decisión a partir de datos y visualizarlos.



Análisis mediante la Minería de Datos

BigML: Análisis Predictivo

- BigML permite realizar tareas de agrupación de datos para encontrar *clusters* similares.



Esquema de Contenidos:

1. Introducción al Big Dada
2. Trabajando con los datos
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Visualización y Consumo de Datos

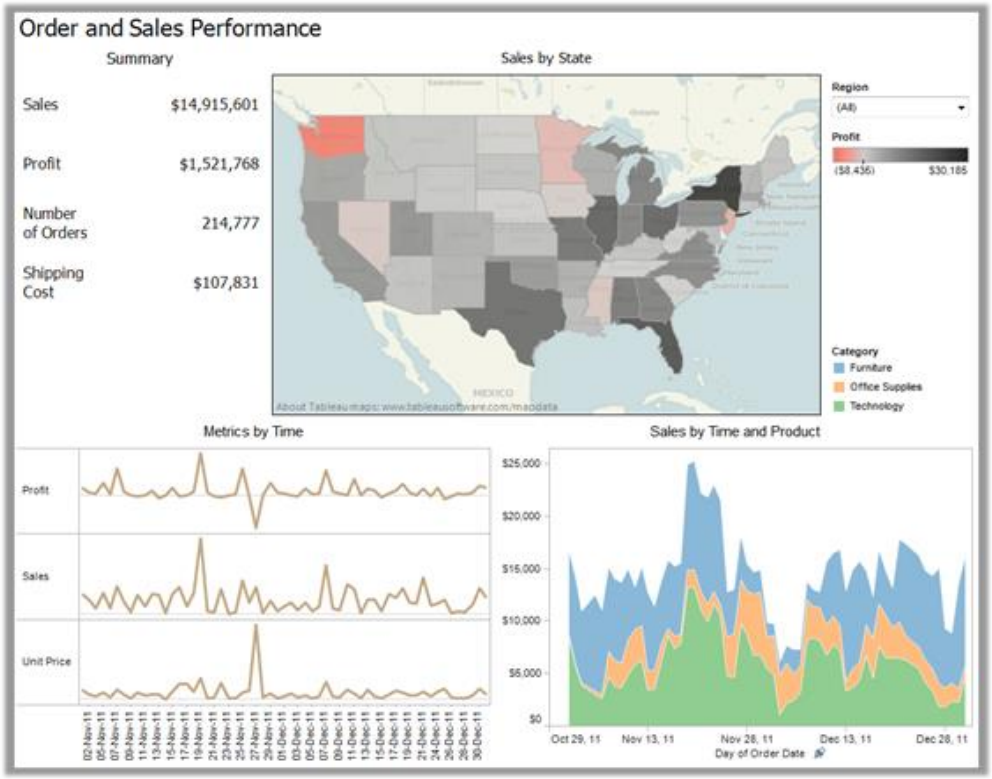
Objetivos de la visualización de datos

- Aprovechar la habilidad humana de extraer patrones a partir de imágenes.
- Ayudar al usuario a comprender más rápidamente patrones descubiertos automáticamente por un sistema automático de extracción del conocimiento o KDD (Knowledge Discovery in Databases)

Visualización y Consumo de Datos

- Ejemplo de panel de monitorización de ventas mediante la herramienta Tableau.

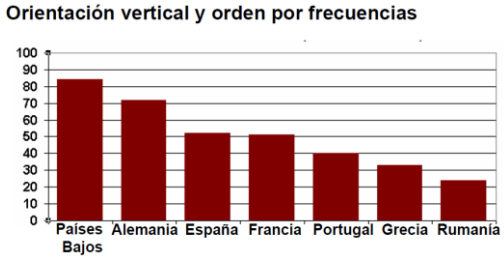
- Se muestran diferentes medidas (ventas, beneficios, pedidos y costes de envío) sobre un mapa, medidas (beneficios, ventas, precio unitario) a lo largo del tiempo y medida de ventas a lo largo de tiempo por categoría de productos

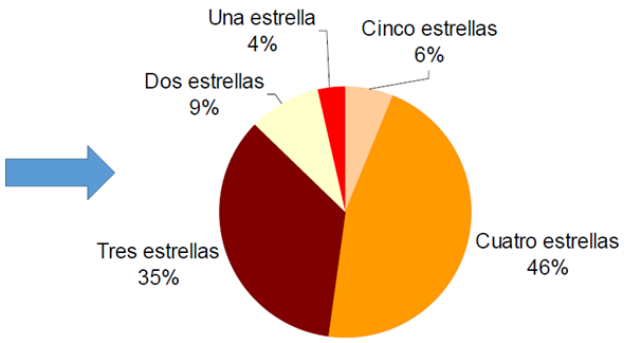


Visualización y Consumo de Datos

Tipos Fundamentales de Gráficos Estadísticos

- Gráfico de Barras
- Gráfico de Líneas
- Gráfico de Sectores
- Pictograma
- Gráfico de Dispersión
- Cartograma





Visualización y Consumo de Datos

Nuevo paradigma

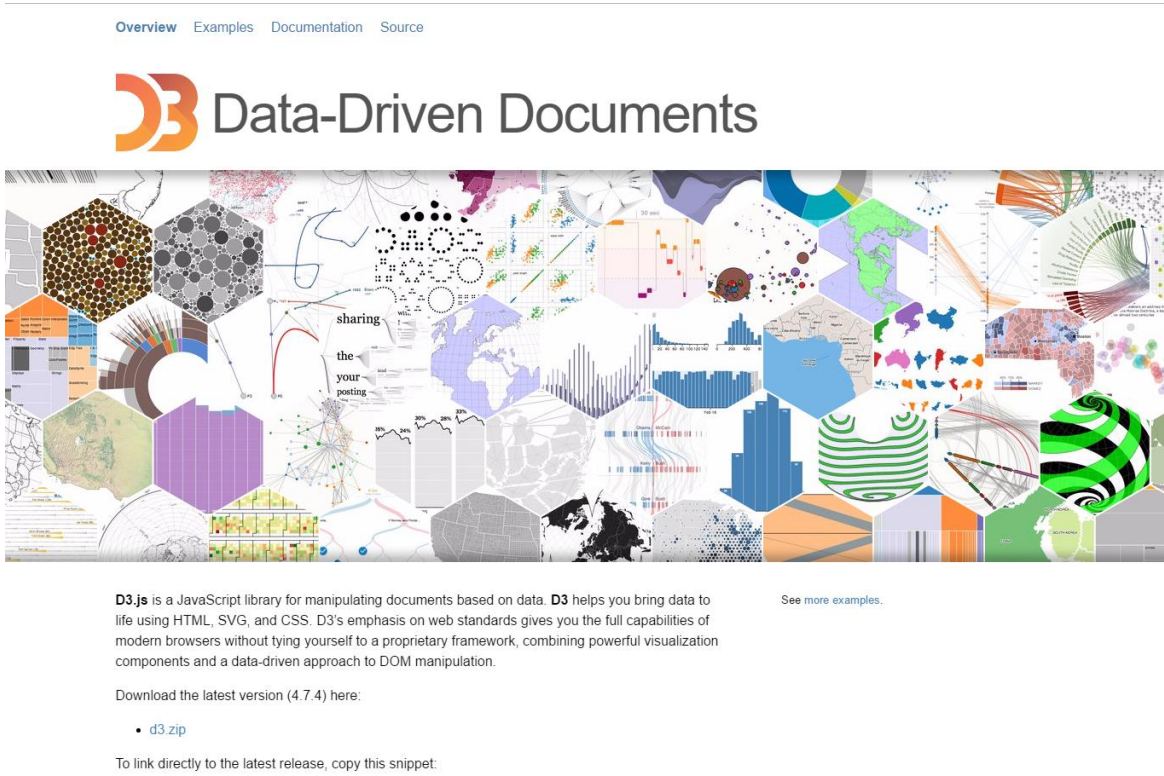
- El uso de los nuevos métodos de visualización se amplía, concediendo a la interfaz gráfica un rol protagonista en ámbitos aparentemente tan alejados de los negocios como el del arte y la cultura.

- Tanto la interfaz como las nuevas herramientas de visualización constituyen ya mucho más que meros elementos (aunque con funciones destacadas) en el ámbito cultural, conformando un nuevo paradigma.

Visualización y Consumo de Datos

Librerías y APIs para la Visualización de Datos

- Google Chart Tools (http://chart.apis.google.com)
- JavaScript InfoVis Toolkit
- D3.js (http://d3js.org/)



Visualización y Consumo de Datos

Webmapping: Visualización de Datos en Mapas Web

- Cada vez son más las aplicaciones que consumen grandes volúmenes de datos referentes a la "geolocalización". Herramientas
  - OpenLayers (http://openlayers.org/)
  - Leaflet
  - CartoBD

Esquema de Contenidos:

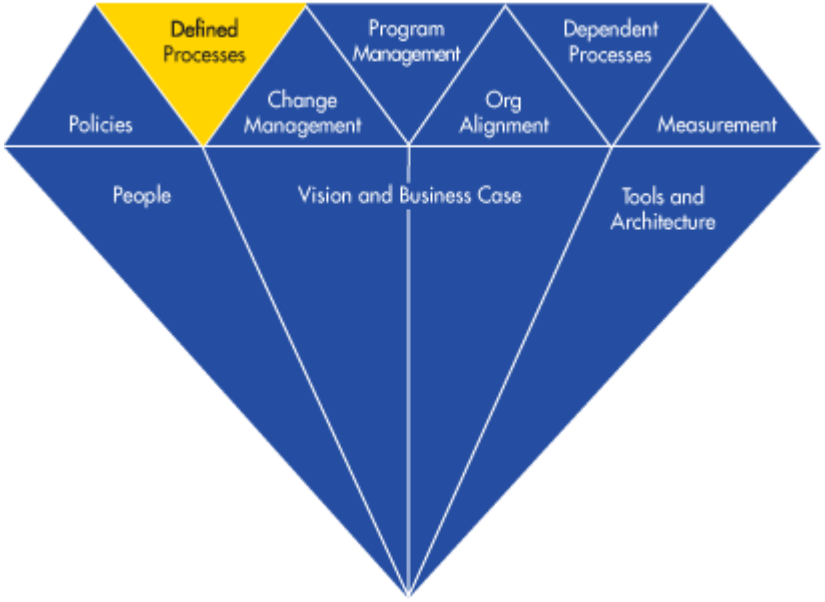
1. Introducción al Big Dada
2. Trabajando con los datos
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Seguridad y Gobernanza

Según el estudio 'Transforming the Data Center', elaborado por Cloudera e Intel, la seguridad y la gobernanza de los datos se han convertido en los grandes desafíos de Big Data. Así lo piensan el 60% de los participantes en esta investigación.

Gobernanza de datos integral

- La gobernanza de datos integral significa gestionar los datos por completo en todos los silos organizativos, arquitectónicos y políticos de la empresa. Requiere adaptar las personas, el proceso, **las políticas y la tecnología para garantizar la entrega de datos fiables y seguros**, de modo que se pueda **cumplir las normativas del sector, reducir el coste** que supone hacer negocios y expandir el negocio.



https://www.informatica.com/es/solutions/data-governance.html

Esquema de Contenidos:

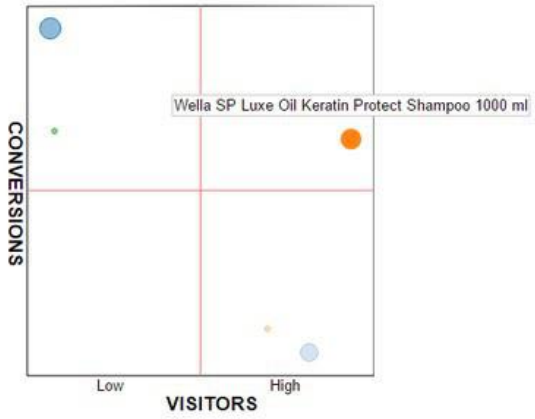
1. Introducción al Big Dada
2. Trabajando con los datos
3. Almacenamiento y Procesamiento: Técnicas, Herramientas y Plataformas
4. Análisis mediante la Minería de Datos
5. Visualización y Consumo de Datos
6. Seguridad y Gobernanza
7. Aplicaciones Reales de Negocio: Casos de Éxito

Use Case 1: Clickstream

Analizando la Huella Digital en Comercio Electrónico

26 marzo, 2016 José Manuel García Nieto

Dentro del contexto del comercio electrónico, cuando hablamos de la "huella digital" nos referimos a la secuencia de pasos, en forma de "clicks", y acciones que realiza un cliente a lo largo de su visita a una tienda virtual. Imagine que entra en su sitio de comercio electrónico favorito con la intención de comprar un determinado producto de su catálogo. Por ejemplo, estamos interesados en com
ordenador portátil.



https://elblogdelainnovaciondigital.wordpress.com/2016/03/26/analizando-la-huella-digital-en-comercio-electronico/

Use Case 1: Clickstream

HORTONWORKS

Products Solutions Clientes Services & Support Sobre nosotros GET STARTED

Descargue sandbox

TUTORIAL SERIES

REAL WORLD EXAMPLES

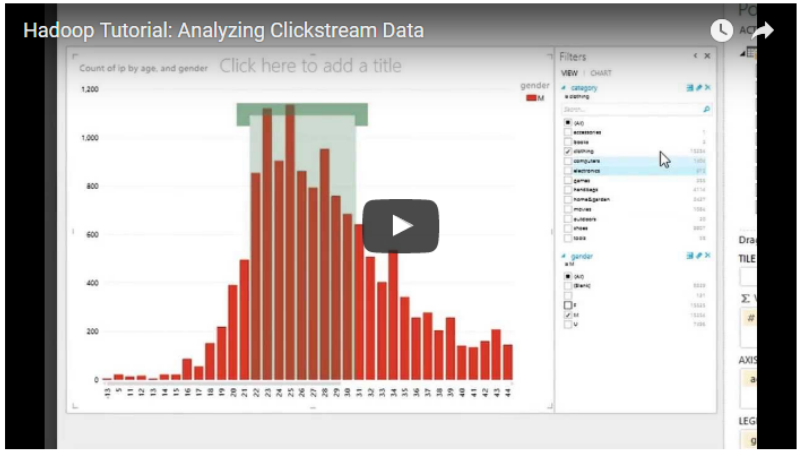
1. Indexing and Searching text within images with Apache Solr
2. Incremental Backup of Data from HDP to Azure using Falcon for Disaster Recovery and Burst capacity
3. Realtime Event Processing in Hadoop with NiFi, Kafka and Storm
4. **Visualize Website Clickstream Data**
   - Introduction
   - Lab 1: Perform Web Log Analysis with Hive
   - Lab 2: Visualize Clickstream Logs with Excel
5. Refine and Visualize Server Log Data
6. Analyzing Social Media and Customer Sentiment With Apache NiFi and HDP Search
7. Analyze HVAC Machine and Sensor Data
8. Natural Language Processing and Sentiment Analysis for Retailers using HDP and ITC Infotech Radar

VISUALIZE WEBSITE CLICKSTREAM DATA

INTRODUCTION

Your home page looks great. But how do you move customers on to bigger things—like submitting a form or completing a purchase? Get more granular with customer segmentation. Hadoop makes it easier to analyze, visualize and ultimately change how visitors behave on your website.

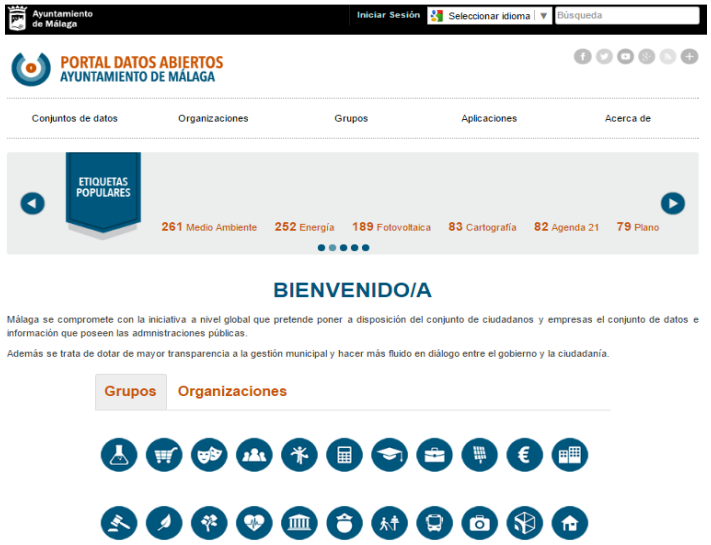
In this demo, we demonstrate how an online retailer can optimize buying paths to reduce bounce rate and improve conversion.



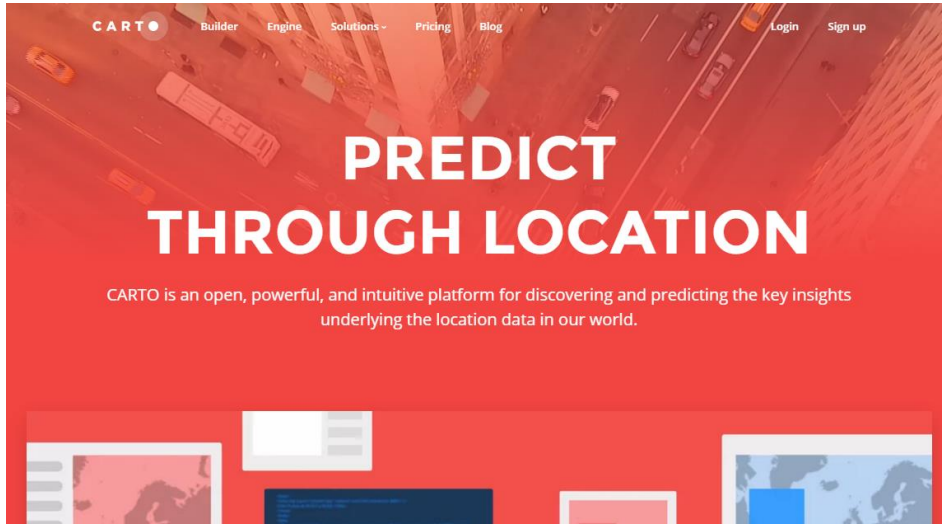
https://github.com/hortonworks/hadoop-tutorials/blob/master/Community/T01_RHadoop_visitors_prediction.md

Use Case 2: Open Data Visualization with Carto

Open Data from
http://datosabiertos.malaga.eu/



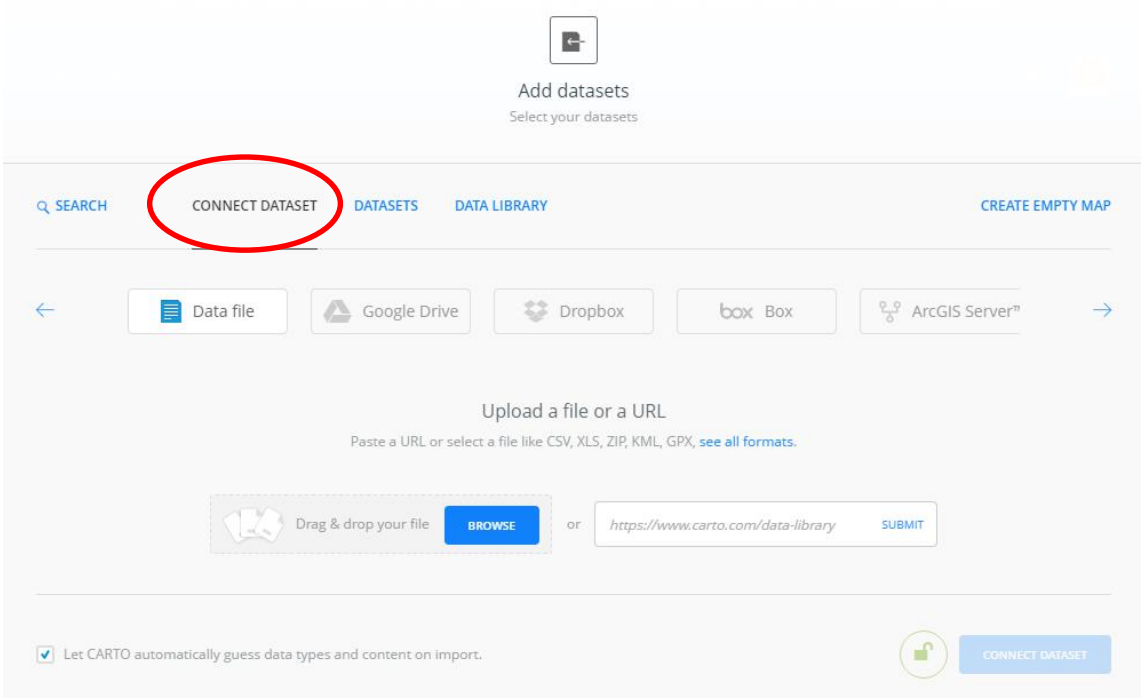
Registration on Carto https://carto.com/



David Bueno (CEMI) : https://www.youtube.com/watch?v=DQjNZ4baSsw

Use Case 2: Open Data Visualization with Carto

- Connecting Dataset

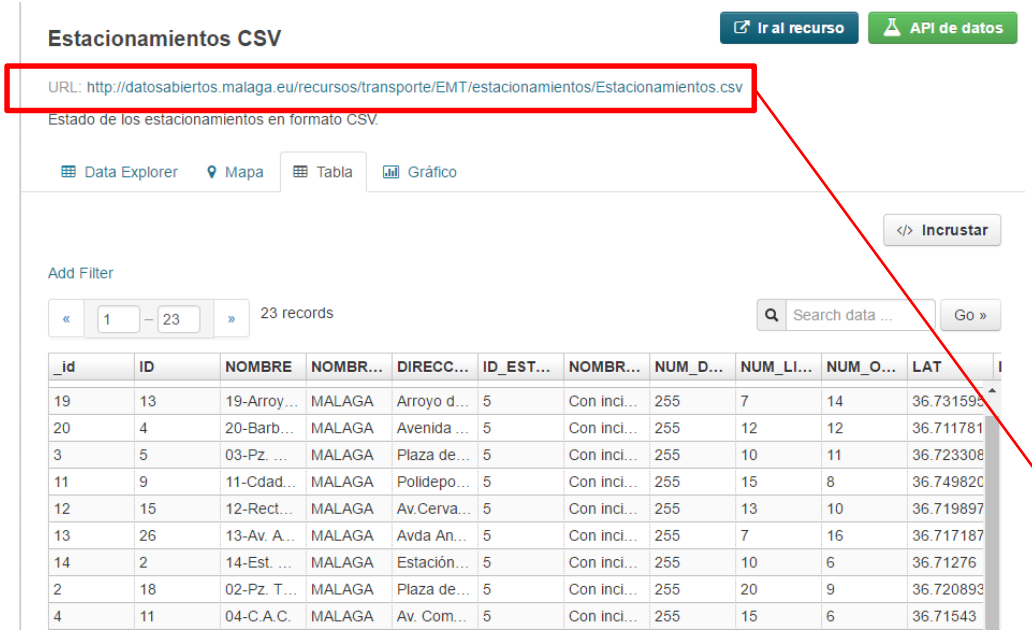


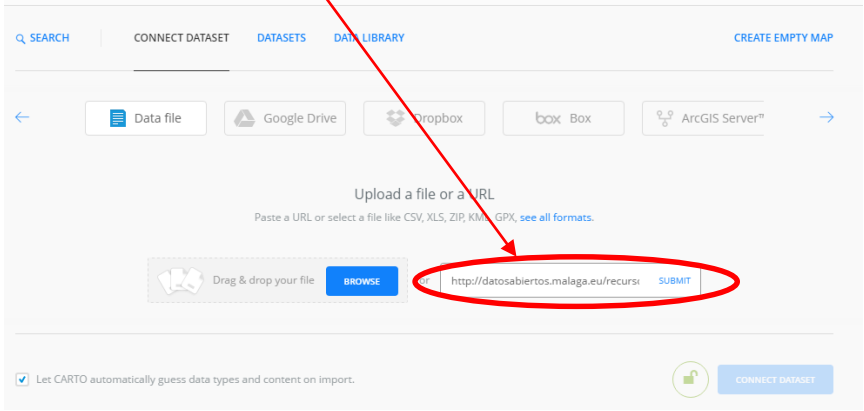
Use Case 2: Open Data Visualization with Carto

- Connecting Dataset: Málaga Bici

Use Case 2: Open Data Visualization with Carto

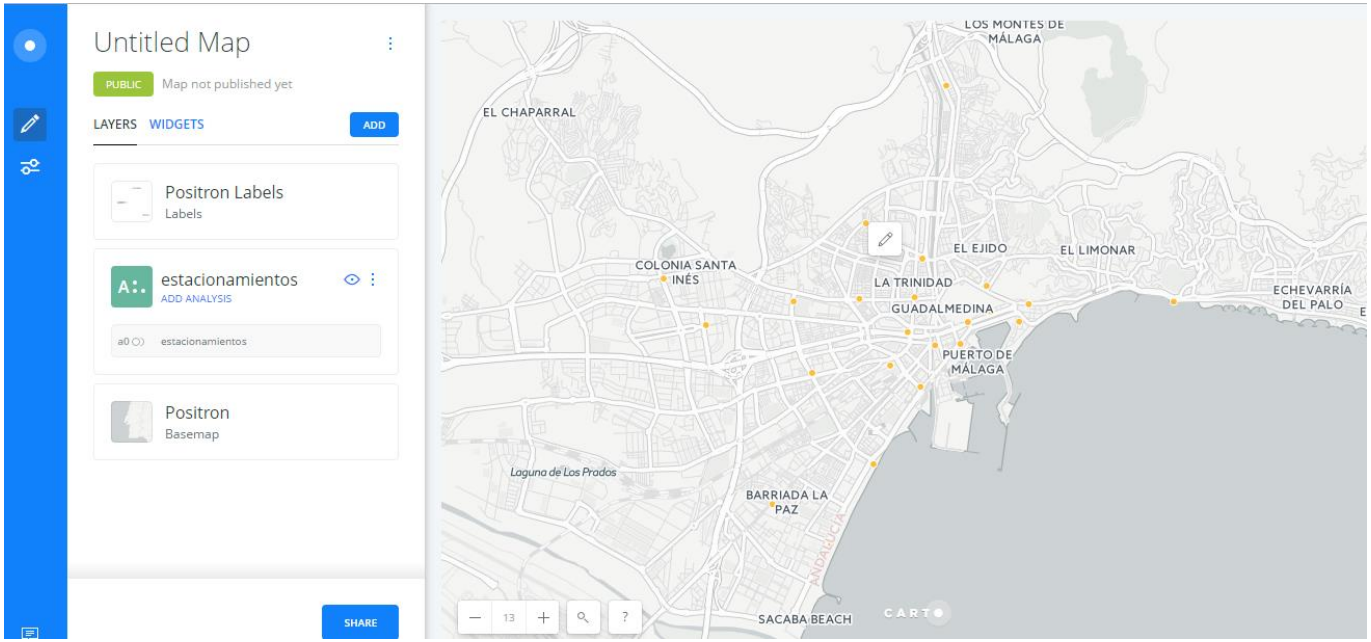
- Connecting Dataset: Málaga Bici





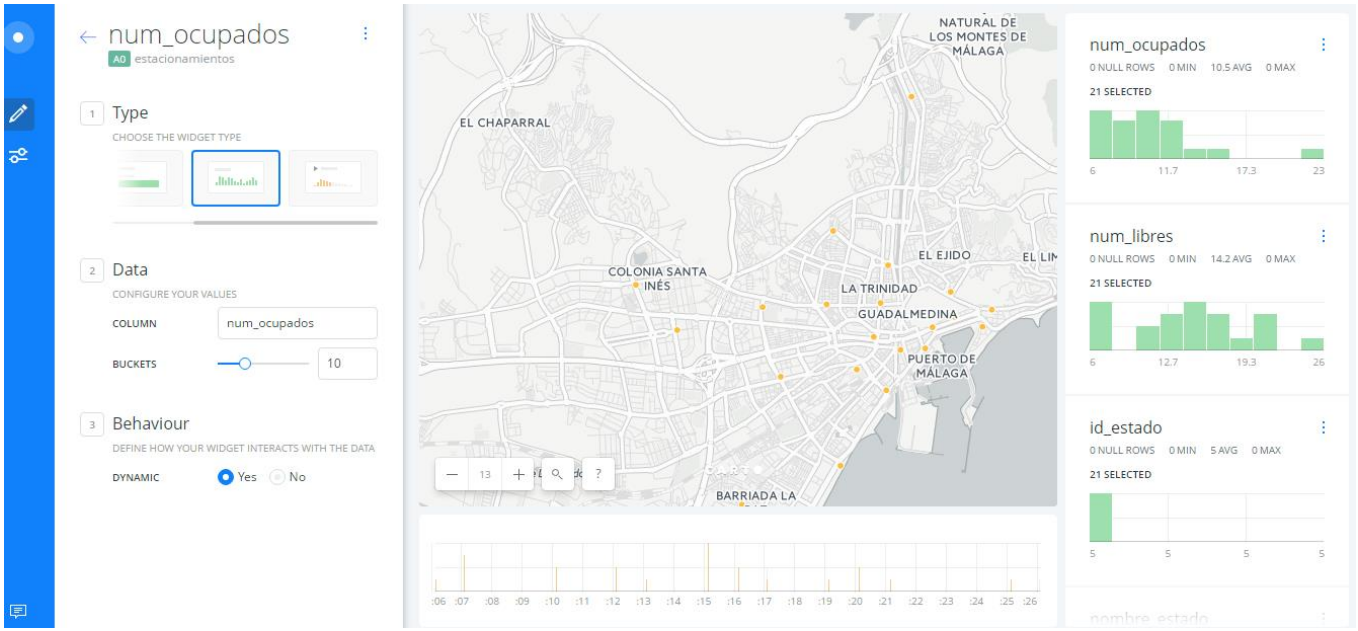
Use Case 2: Open Data Visualization with Carto

- Connecting Dataset: Málaga Bici



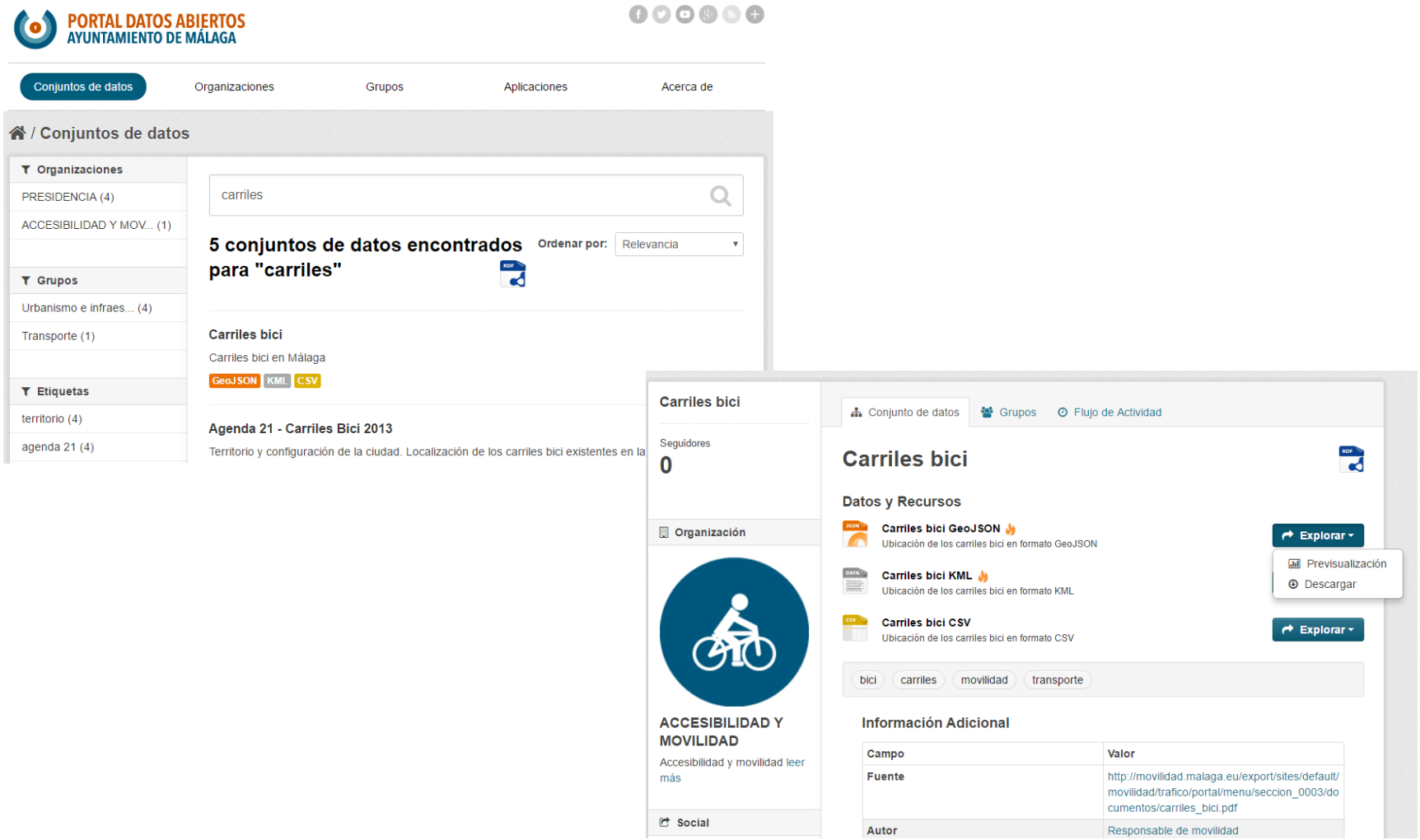
Use Case 2: Open Data Visualization with Carto

- Adding Widgets: Málaga Bici



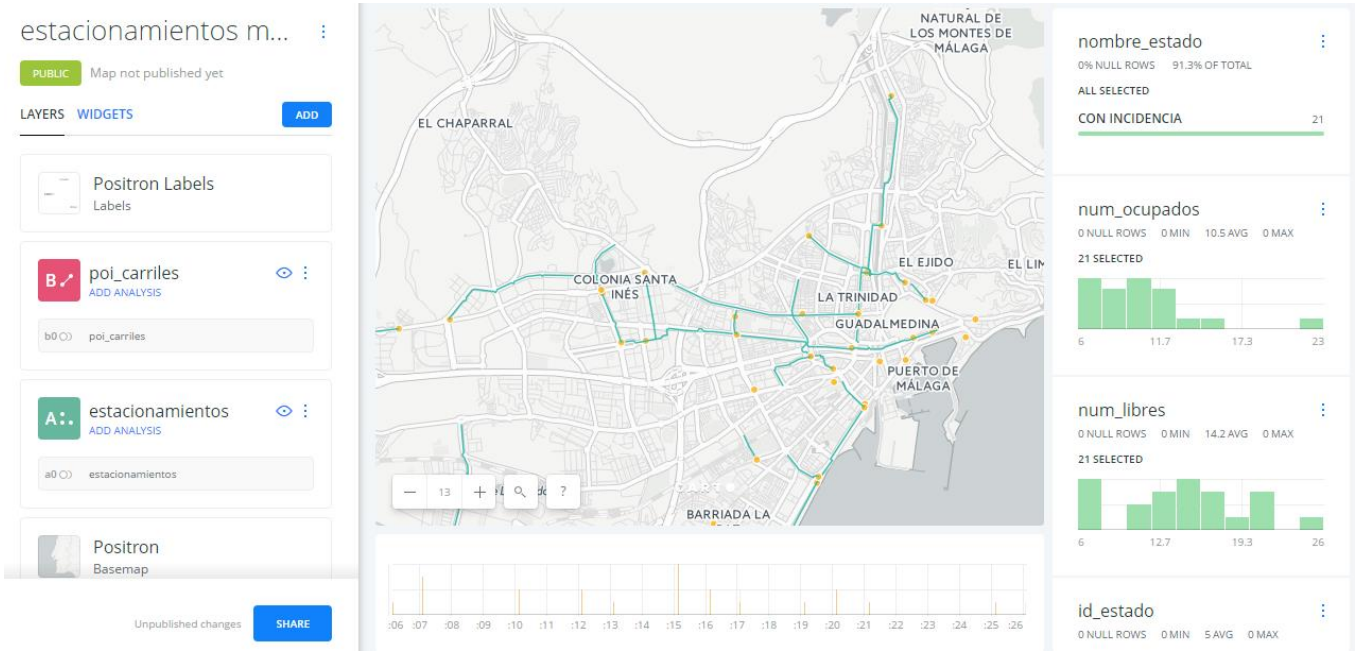
Use Case 2: Open Data Visualization with Carto

- Adding new layer: Málaga Carriles Bici



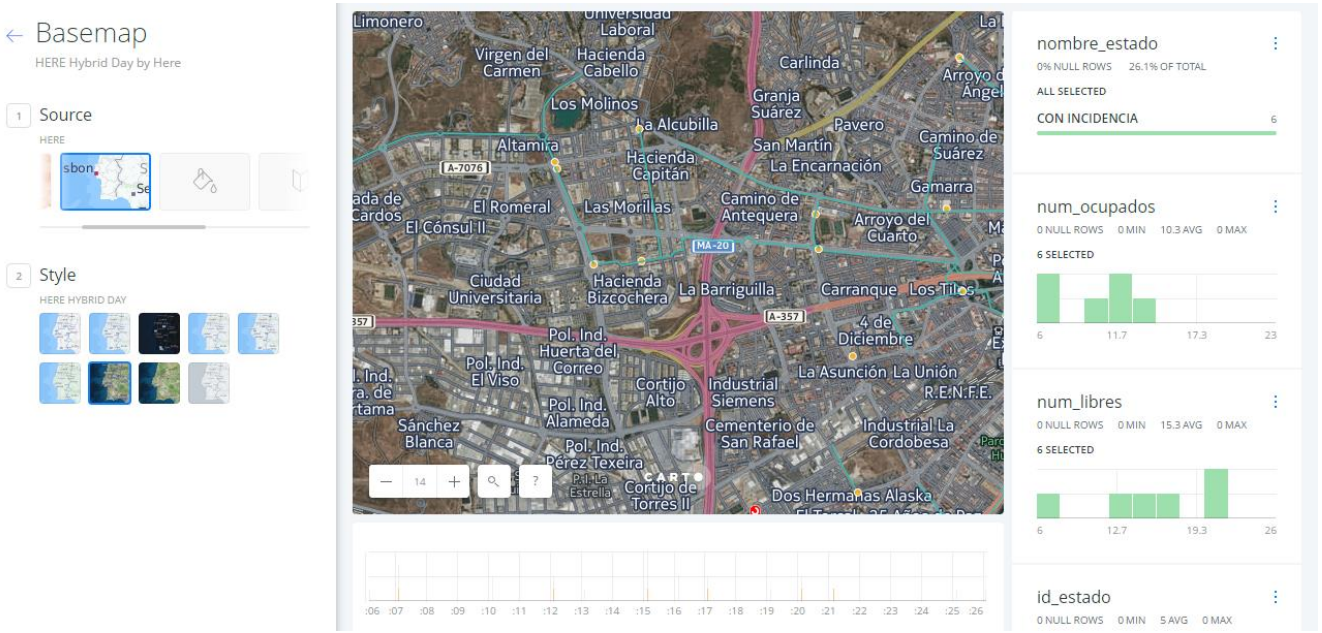
Use Case 2: Open Data Visualization with Carto

- Adding new layer: Málaga Carriles Bici



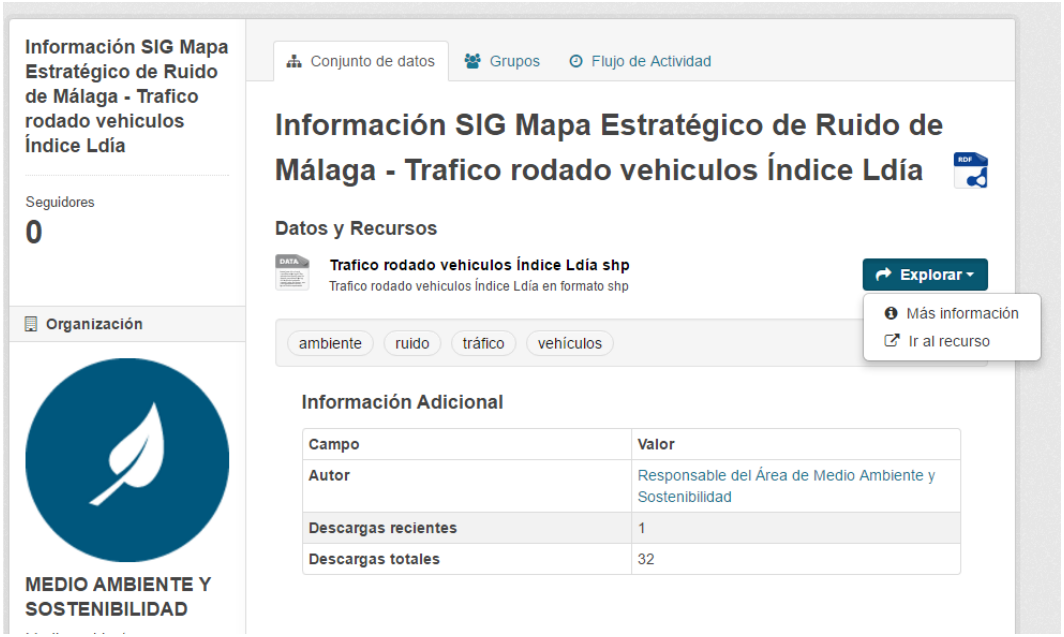
Use Case 2: Open Data Visualization with Carto

- Changing Basemap: Málaga Carriles Bici



Use Case 2: Open Data Visualization with Carto

- Adding new layer: Málaga Traffic Noise

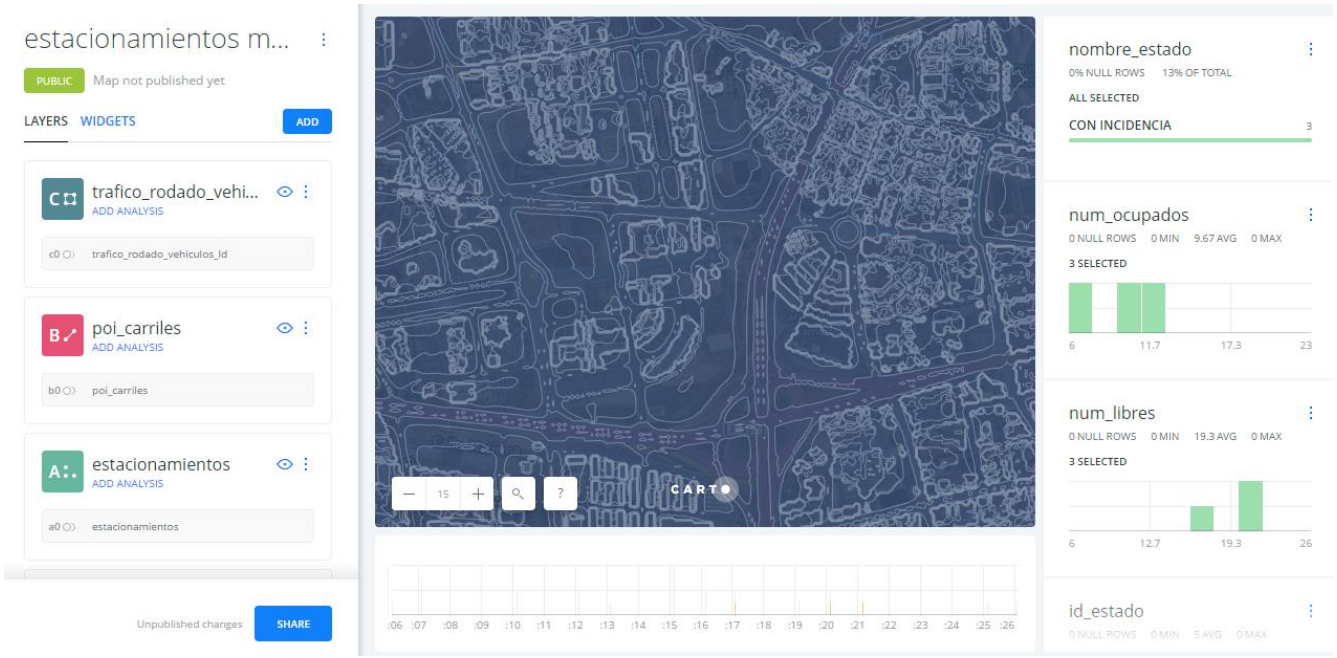


Use Case 2: Open Data Visualization with Carto

- Adding new layer: Málaga Traffic Noise

Use Case 2: Open Data Visualization with Carto

- Adding new layer: Málaga Traffic Noise



Use Case 2: Open Data Visualization with Carto

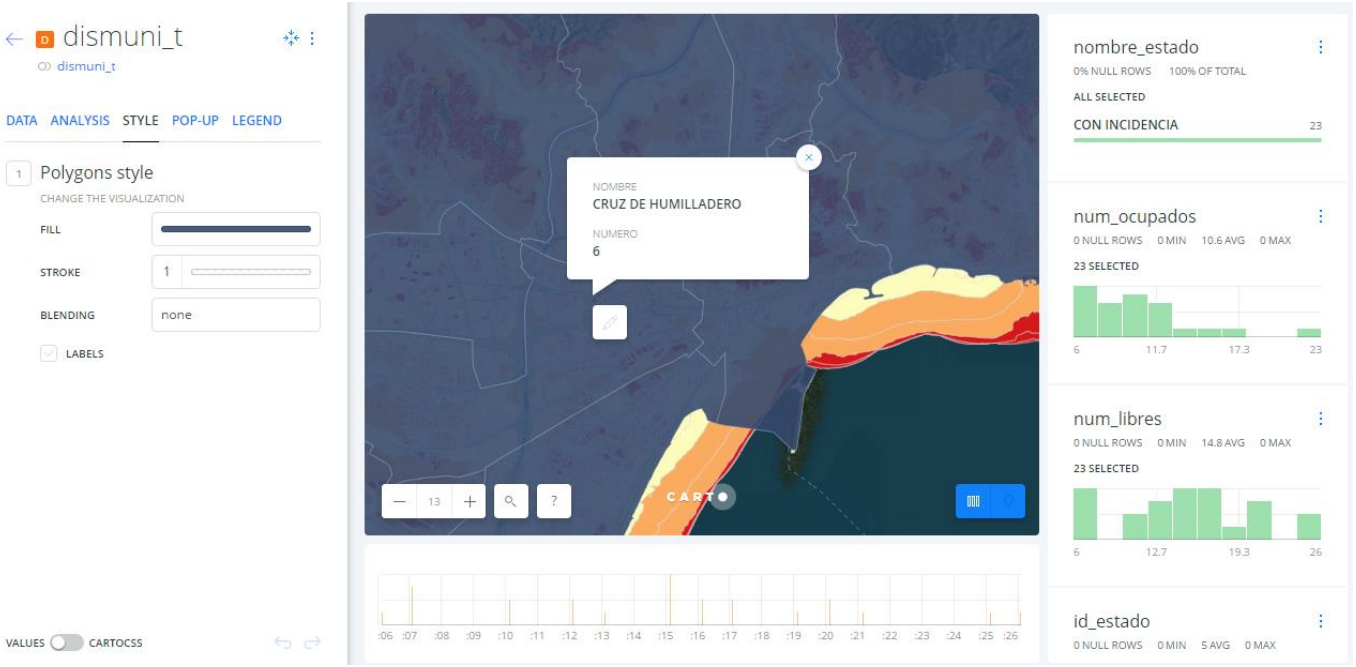
- Adding new layer: Málaga Traffic Noise

Use Case 2: Open Data Visualization with Carto

- Adding new layer: Malaga Council Districts

Use Case 2: Open Data Visualization with Carto

- Adding new layer: Malaga Council Districts



Use Case 2: Open Data Visualization with Carto

- Adding new layer: Malaga Council Wifi Points

Use Case 2: Open Data Visualization with Carto

- Adding new layer: Malaga Council Wifi Points