



Nociones de Arquitectura de la Información (AI)

Modelos para representar la estructura que soporta el comportamiento



¿Cómo construir
cada vez
estructuras más
complejas
cumpliendo
ciertos criterios?

La AI brinda herramientas para estructurar los datos con el fin de facilitar su uso y localización (findability).

- Es una disciplina para el estudio, análisis, organización, disposición y estructuración de la información, y de la selección y presentación de los datos en los sistemas de información interactivos y no interactivos.



¿El proceso de desarrollo de software es una ingeniería, ciencia, arte o manualidad?

Existe un consenso general...



<https://www.codeproject.com/Articles/1130886/The-Software-Development-Process-Science-Engineeri>

El proceso de desarrollo de software es imperfecto:

- Razones humanas (administración, habilidades, incompetencias, comunicación, claridad, etc.)
- Razones tecnológicas (herramientas, soporte, documentación, confiabilidad, etc.).

La literatura sobre desarrollo de software está llena de frases que hablan sobre metodologías

Las Falsas ilusiones y expectativas

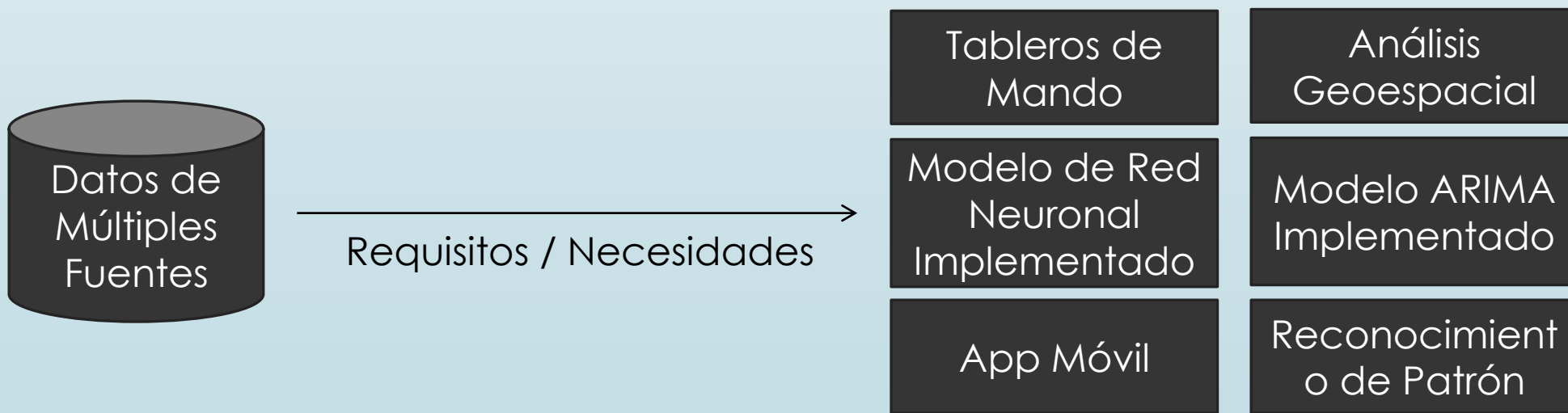
- La literatura da falsas ilusiones a los entusiastas, a las masas, programadores novatos, gerentes e incluso desarrolladores senior la cálida y difusa ilusión de que hay un **proceso repetible** para el desarrollo de software que **garantiza palabras como “ciencia” e “ingeniero.”**

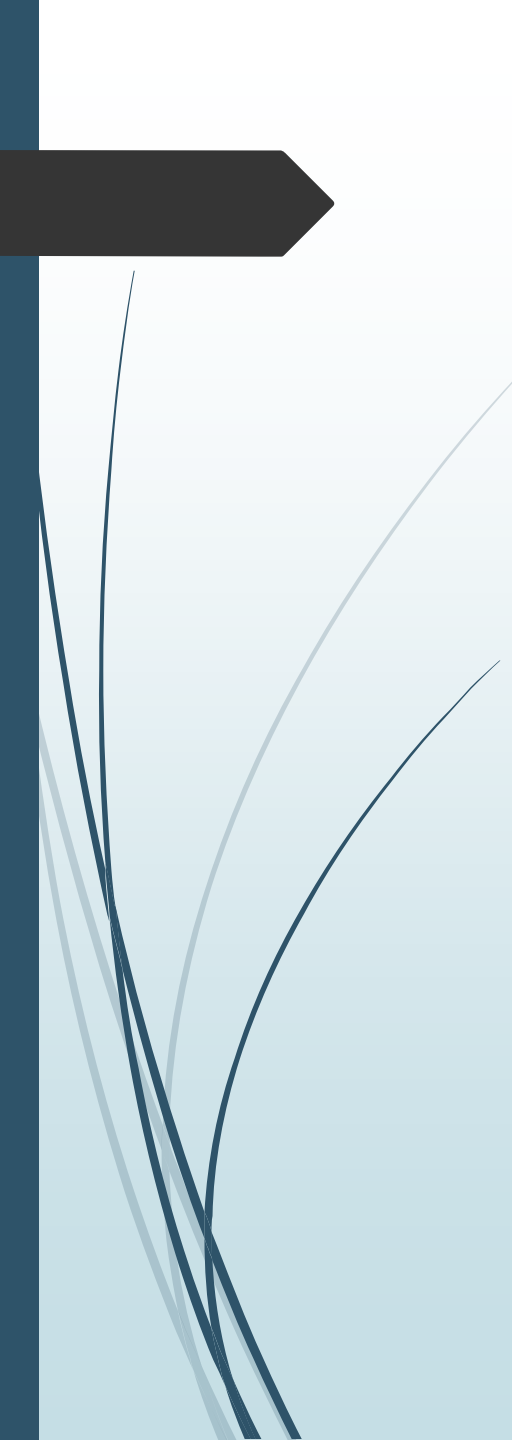
Los experimentados no se dejan engañar

- Aquellos que son consientes o han estado en un ambiente de desarrollo productivo “real” y reconocen la **gran holgura o comodidad de las metodologías** se sienten más cómodos describiendo el proceso de desarrollo de software como “arte” o “manualidad” y con un toque de “brujería.”

¿Qué es un producto de datos? (aplicaciones basadas en datos)

- Un producto, en este contexto, es una pieza de software, es decir, la implementación de un algoritmo en un dispositivo (PC, Móvil, PC On Board, Smart TV, Manilla Inteligente...); también los conocemos como aplicaciones.
- Entonces, un producto de datos, es por ende una pieza de software para resolver un problema basado en datos... se fundamentan en datos históricos y en tiempo real que se capturan, procesan, almacenan, analizan y se modelan...





Los productos de datos deben ser replicables y para un mercado o contexto específico.

En general, el desarrollo de un producto requiere de 3 fases:

1. Definición. Entender el mercado, contexto o dominio, elicitar los requisitos (problemas o necesidades) y definir los que el producto va a resolver
2. Diseño. Conceptualizar la solución y elaborar la especificación funcional y la experiencia de usuario
3. Implementación. Elaborar la especificación técnica, construir y probar la solución

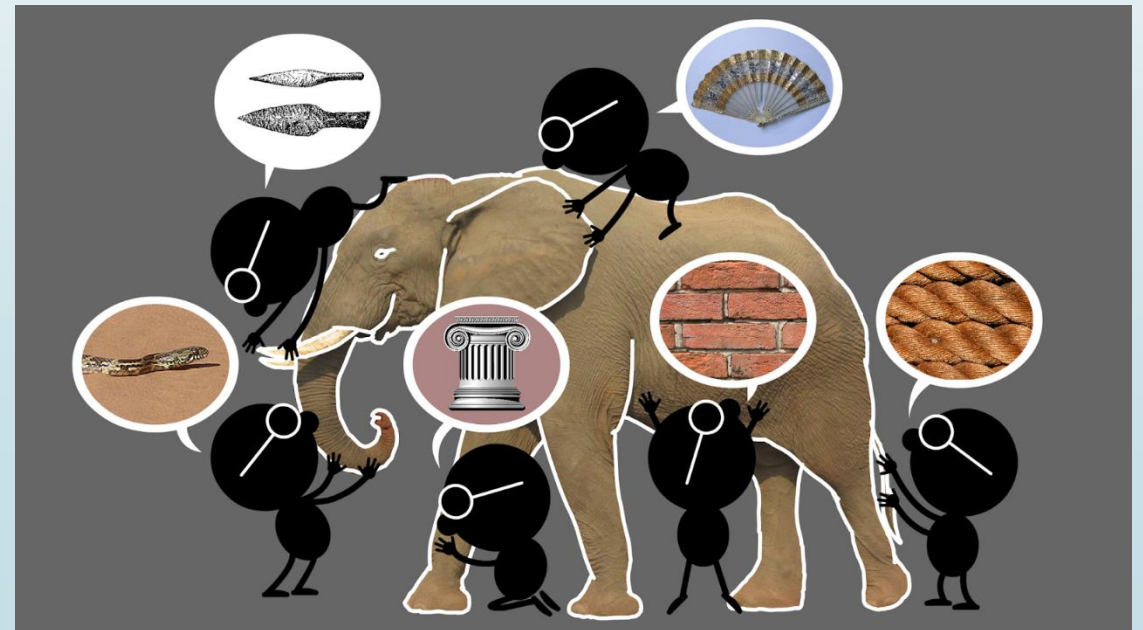
El desarrollo de productos basados en datos o desarrollo de software basado en datos

- ¿Cómo crear, desarrollar, implementar algo que es tan complejo?.

Evidentemente Desarrollar software NO es fácil.



Los Ciegos
y el Elefante





Como el cliente lo
explicó



Como el lider del
proyecto lo entendió



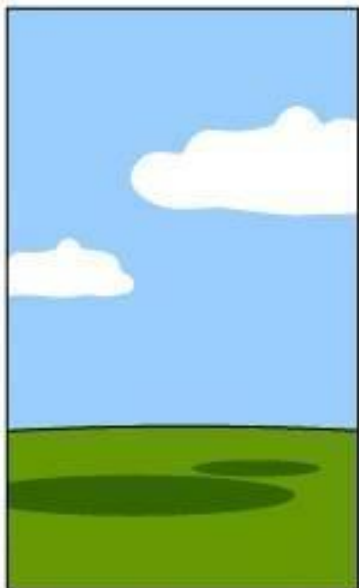
Como el analista lo
diseñó



Como el programador
lo escribió



Como el vendedor lo
describió



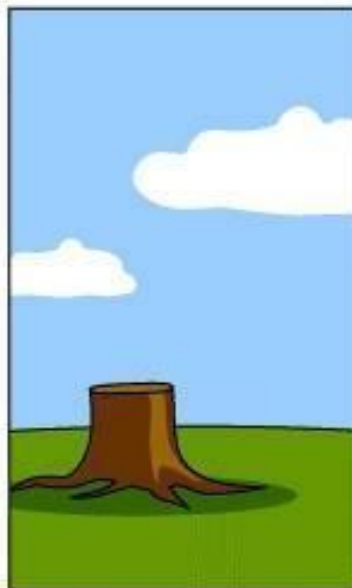
Como fue documentado
el proyecto



Que aplicaciones
se instalaron



Como le fue facturado
al cliente



Como se le dio soporte



Lo que el cliente
realmente necesitaba

El Clásico
dilema de
cualquier
desarrollo de
productos...

LA VIDA DE UN INGENIERO
DE SOFTWARE.

COMENZAMOS DE CERO.
UNOS BUENOS CIMIENTOS. ESTA VEZ
CONSTRUIRÉ LAS COSAS
CORRECTAMENTE



MUCHO MÁS TARDE...

AY, MADRE. YA LO
HE VUELTO A
HACER, ¿NO?



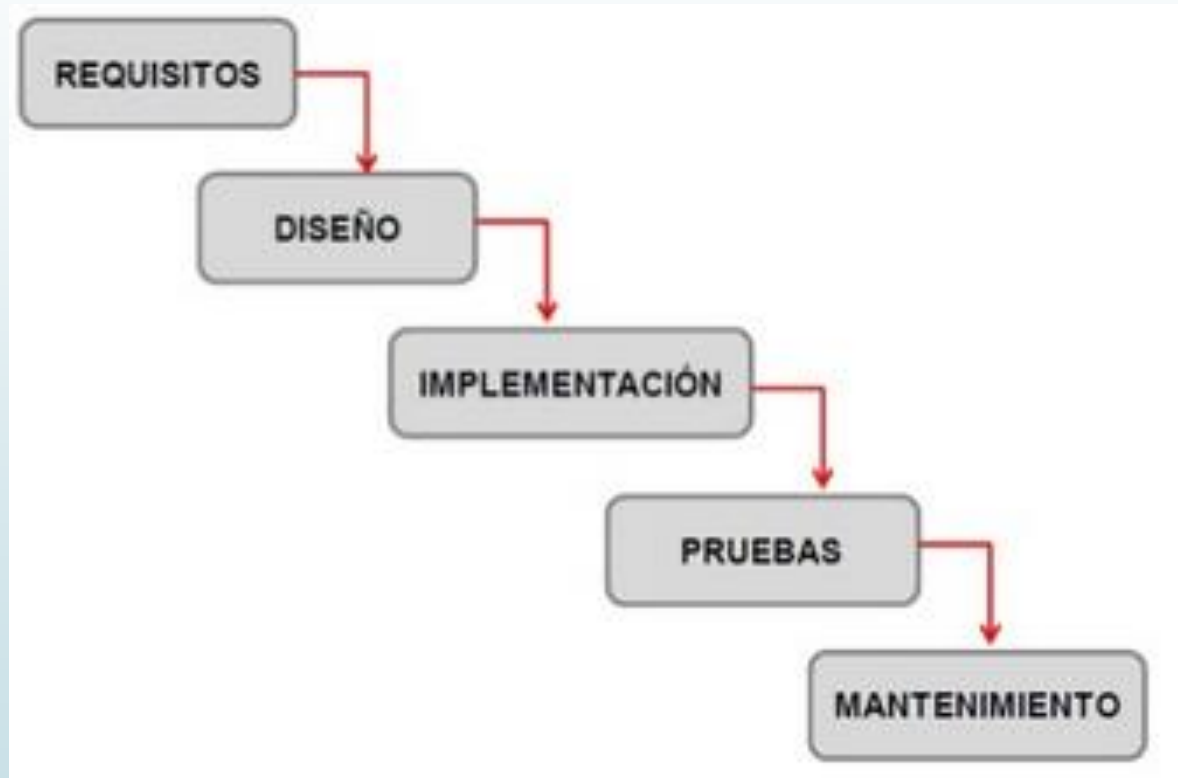
A la
mayoría
nos ha
pasado

Sobre el Ciclo de Vida (Nacimiento hasta la defunción)

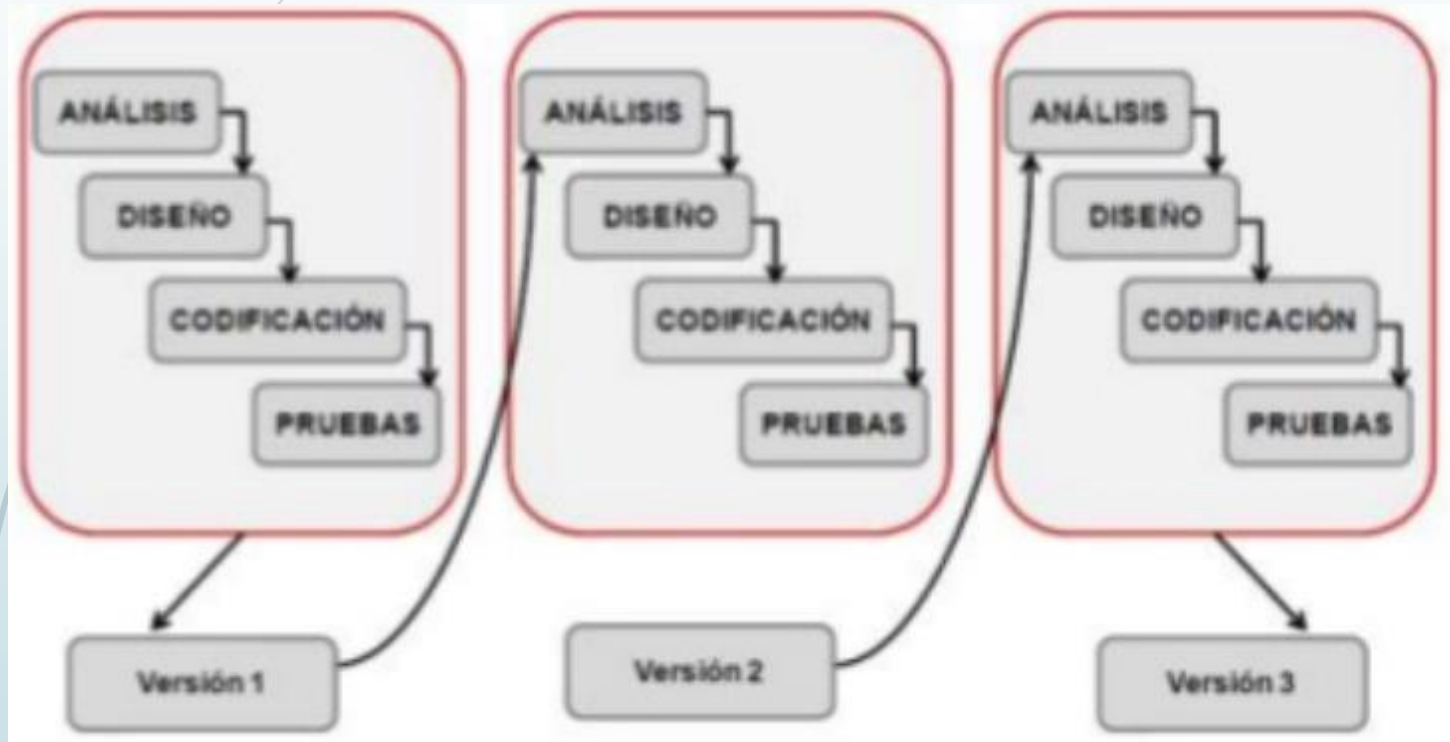
- El ciclo de vida es el conjunto de fases por las que pasa el objeto que se desea desarrollar, desde que nace la idea hasta que el objeto es retirado o remplazado (muere).



Modelo en Cascada, o “ciclo de vida en cascada”



Modelo iterativo



- Se iteran sobre varios ciclos de vida en cascada.
- Al finalizar cada iteración se entrega una versión mejorada. (No es necesariamente incremental)

Modelo en espiral



- Las actividades se planean tal que se pueda avanzar según la espiral.
- De adentro hacia afuera.
- Cada ciclo representa un conjunto de actividades.

Modelo de prototipos



- También diseño rápido.
- Se requiere desarrollar un prototipo que debe ser evaluado para refinar los requisitos.

¿En un sentido simple qué es arquitectura?

Definición Base

- La arquitectura consiste en la gestión, configuración e introducción de la estructura, directrices, principios y liderazgo de los aspectos técnicos de un proyecto de desarrollo.
- Se requiere de alguien que pueda asumir la propiedad, responsabilidad y liderazgo del proceso de definición de la arquitectura.

Que implica una arquitectura

- Tener la visualización holística del sistema.
- Crear los planos del sistema.
- Definir los componentes del sistema.
- Definir el modo como interactúan todos los componentes del sistema.
- Integrar los requisitos no-funcionales en el sistema.
- Producir los artefactos necesarios para el sistema: documentos de descripción de la arquitectura.

A nivel de sistemas la arquitectura tiene muchos apellidos...

Componentes de la Arquitectura Empresarial

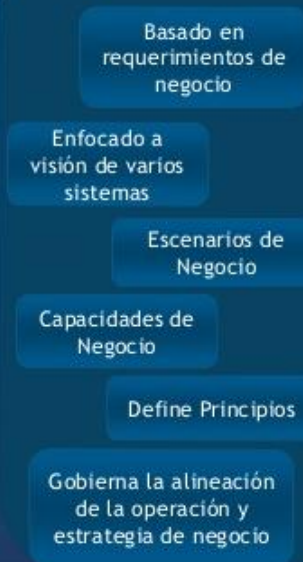


Fuente: Adaptación de Colombia Digital del gráfico desarrollado por Amazing Consultores

SAI

Coincidencias y diferencias entre Arquitectura Empresarial y de Software

Arquitectura Empresarial



Arquitectura de Software

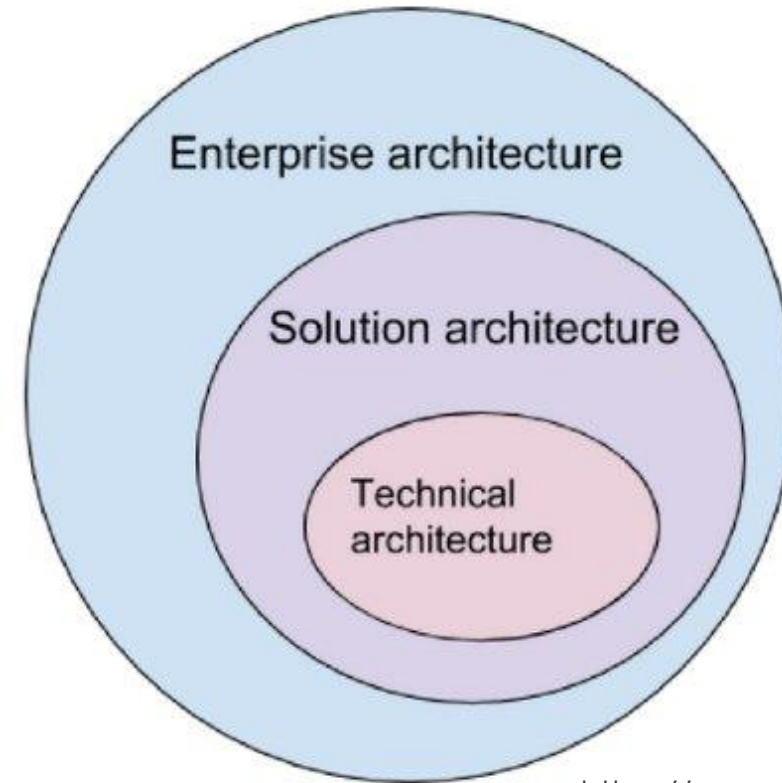


El apellido “de soluciones” ... “*multi solution architect*”



<https://www.pedromonjo.com/2016/08/multi-solution-architect-role/>

18



» <https://www.altexsoft.com/blog/engineering/solution-architect-role/>

System and Solution Architect / Engineering

- <https://www.scaledagileframework.com/system-and-solution-architect-engineering/>

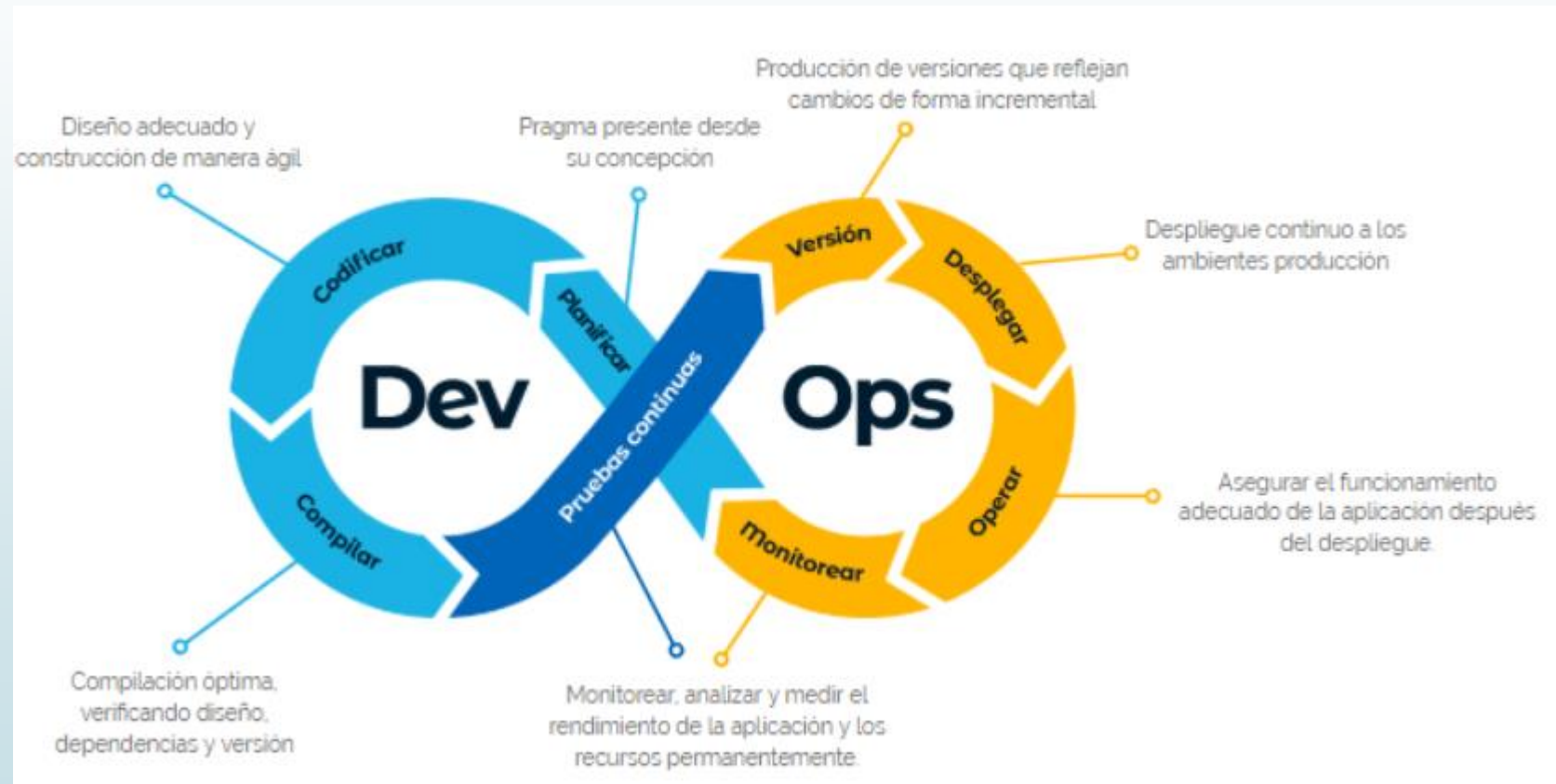
Detalladamente se define el rol del arquitecto de soluciones

SOLUTION ARCHITECT ROLE

Responsibilities	<ul style="list-style-type: none">✓ Analyzing technology environment✓ Analyzing enterprise specifics✓ Analyzing requirements✓ Setting collaboration framework✓ Creating a solution prototype✓ Participating in technology selection✓ Solution development control✓ Project management support
Experience	<ul style="list-style-type: none">✓ At least eight years of working experience in one or multiple IT areas✓ IT infrastructure and cloud development✓ Engineering and software architecture design✓ Business analysis✓ DevOps✓ Project and product management
Personal Attributes	<ul style="list-style-type: none">✓ Excellent communication skills✓ Deep analytical skills✓ Project and resource management skills✓ Work collaboratively✓ Influence and negotiate

Es difícil encontrar Conceptualmente en el mercado, la industria y la academia, un Consenso entre arquitecto de software y de soluciones.

En vez de hablar de Arquitecto de Software se habla de Soluciones para hacer énfasis que tiene competencias, entre otras, en



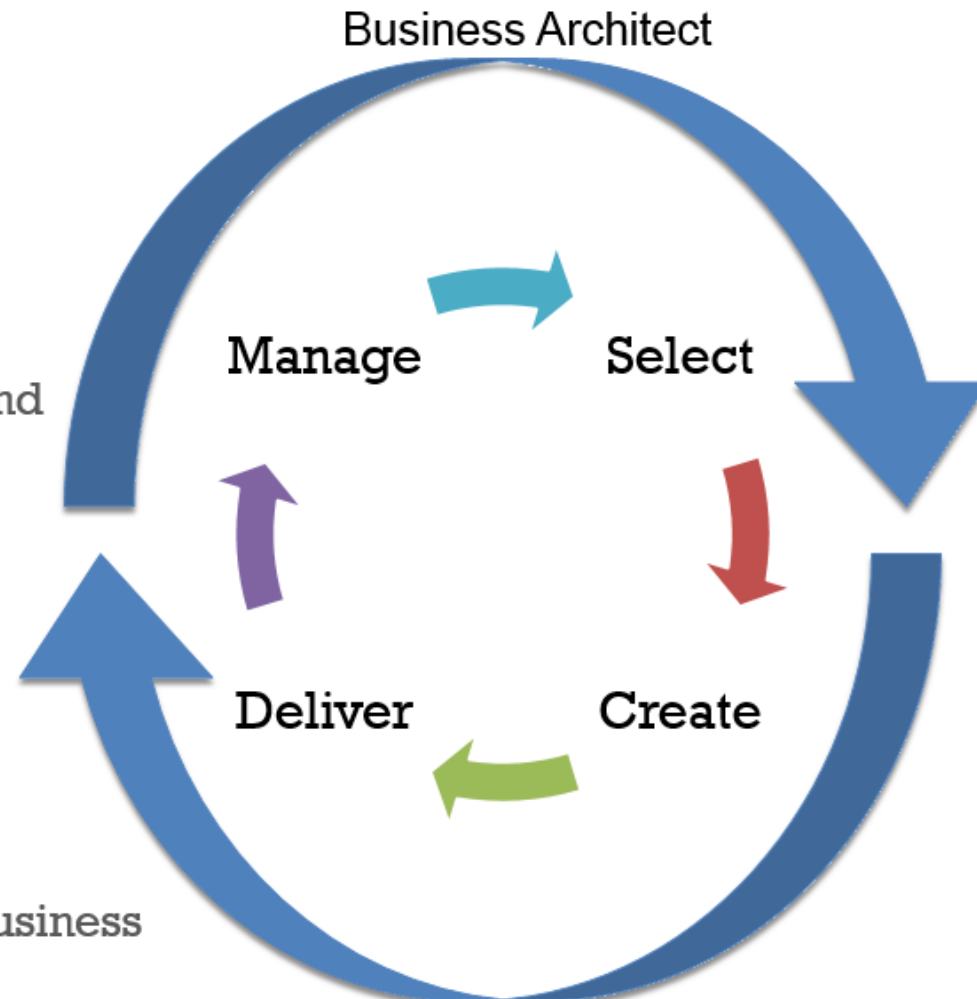
desarrollo de software (Dev) y la operación del software (Ops)

<https://www.pragma.com.co/academia/conceptos/devops>

Arquitecto de Negocios vs Arquitecto de Soluciones

Architecture Transitions

- Business Architect
 - Business Strategy
 - Capability Map
 - Business Case
 - Valuation Guidelines
 - Target State (business and technical)
- Solution Architect
 - Solution Architecture
 - Context/Logical
 - Value Decisions
 - Reasoning Views and Perspectives
 - Updated Target State (business and technical)



Un arquitecto de negocios solo se preocupa por el retorno de la inversión y la administración de la operación empresarial...

Un arquitecto debe "escuchar las sugerencias del desarrollador para mejorar la arquitectura".

- Frederick P. Brooks, Jr

El Arquitecto de Soluciones debe ser...

Una persona con

- Amplios conocimientos técnicos,
- Gran experiencia en programación,
- Liderazgo
- Proactividad
- Alto Nivel de Concreción
- Asertivo
- Diplomático
- Honesto
- Ético

Funciones

- Gestión de los requisitos no funcionales y definición de la *Arquitectura de Software*
- Selección de la Tecnología
- Mejora continua de la Arquitectura
- Facilitador
- Líder y Formador
- Aseguramiento de la Calidad
- Crítico de los requisitos no funcionales, de rendimiento, la escalabilidad, la disponibilidad, auditoría
- Validar la arquitectura y sus artefactos.

El Arquitecto de Soluciones debe asumir la dirección técnica

El deber ser del Arquitecto

- Un arquitecto “sugiere” (“no dicta”) la implementación porque el programador / codificador / constructor tiene la “responsabilidad creativa y creativa”. Frederick P. Brooks, Jr
- Un arquitecto debe estar “listo para renunciar al crédito por las mejoras sugeridas”. Frederick P. Brooks, Jr

El **Arquitecto de Software** debe proporcionar orientación técnica y dar apoyo al equipo de desarrollo; debe estar preparado para entrenar o facilitar al equipo las tecnologías seleccionadas (Formador / Facilitador) y también debe estar abierto a sugerencias,

Garantizar la calidad es también responsabilidad de un *Arquitecto de Software*,

- Un arquitecto debe esforzarse por que el trabajo sea “sobrio y limpio”, evitando la “ornamentación funcional” - Frederick P. Brooks, Jr
- Códigos y Modelos -> Simples y Concretos
- El cual debe apoyarse en procesos de integración continua que utilicen herramientas automatizadas de análisis de código fuente, pruebas unitarias y cobertura de código, para asegurar el cumplimiento de las normas, políticas y mejores prácticas establecidas.

En términos generales, la arquitectura se guía por los “drivers arquitectónicos” los cuales están dados por los requisitos...

Comportamiento / Funcionalidades

Encender

Despegar

Volar

Aterrizar



Desempeño

Aerodinámica

Autonomía

Poder de los motores

Estructura / Componentes

Motores

Alas

Fuselaje

Tren de aterrizaje

Aspectos Específicos de Análisis

Modelo de
Costos

Modelo de
Seguridad

Modelo de Producción

Mercado

Los esquemas Preconceptuales permiten representar tanto la estructura como el comportamiento.

La simbología básica del Esquema Preconceptual se muestra en la Figura 2.2.

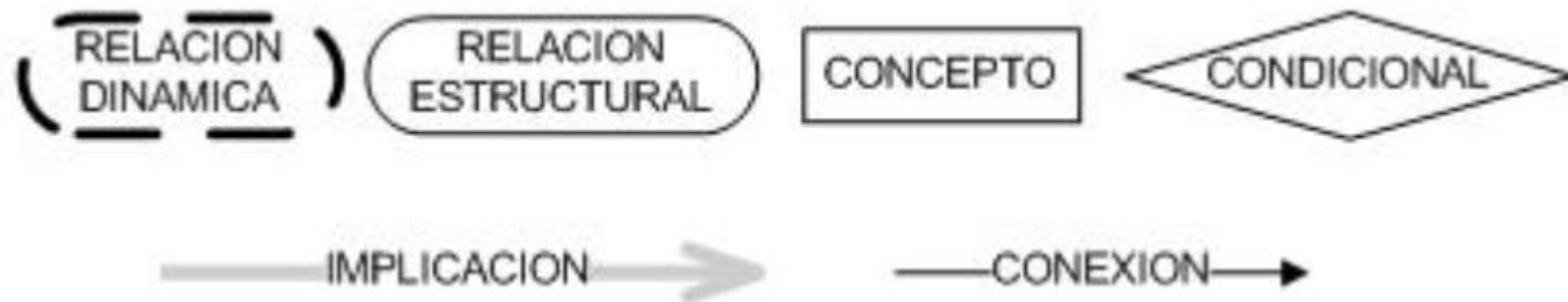


Figura 2.2. Simbología básica del Esquema Preconceptual.

- **Ejemplo compra de Tiquetes (entrada) para Cine**

El modelo del dominio sólo permite representar la estructura

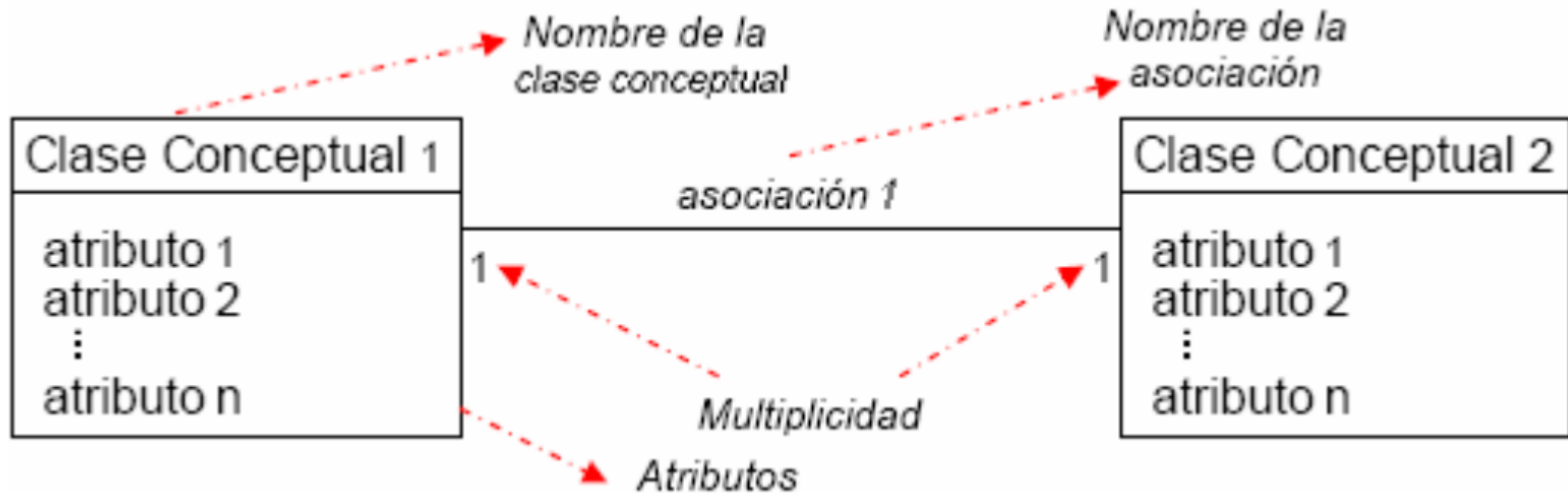
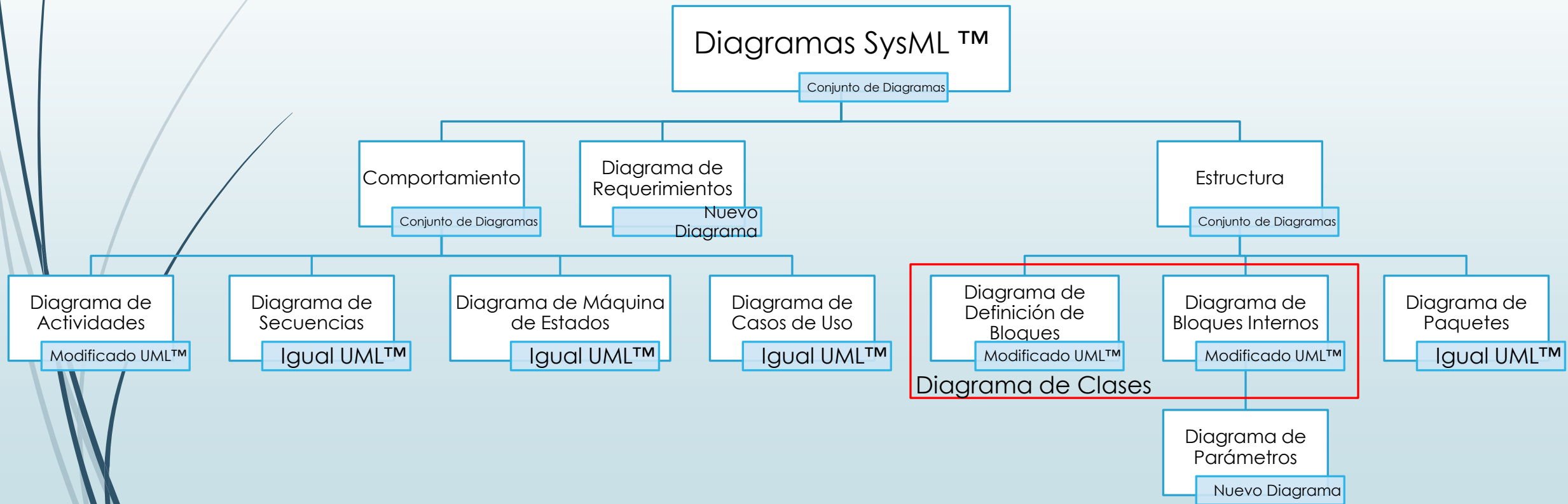
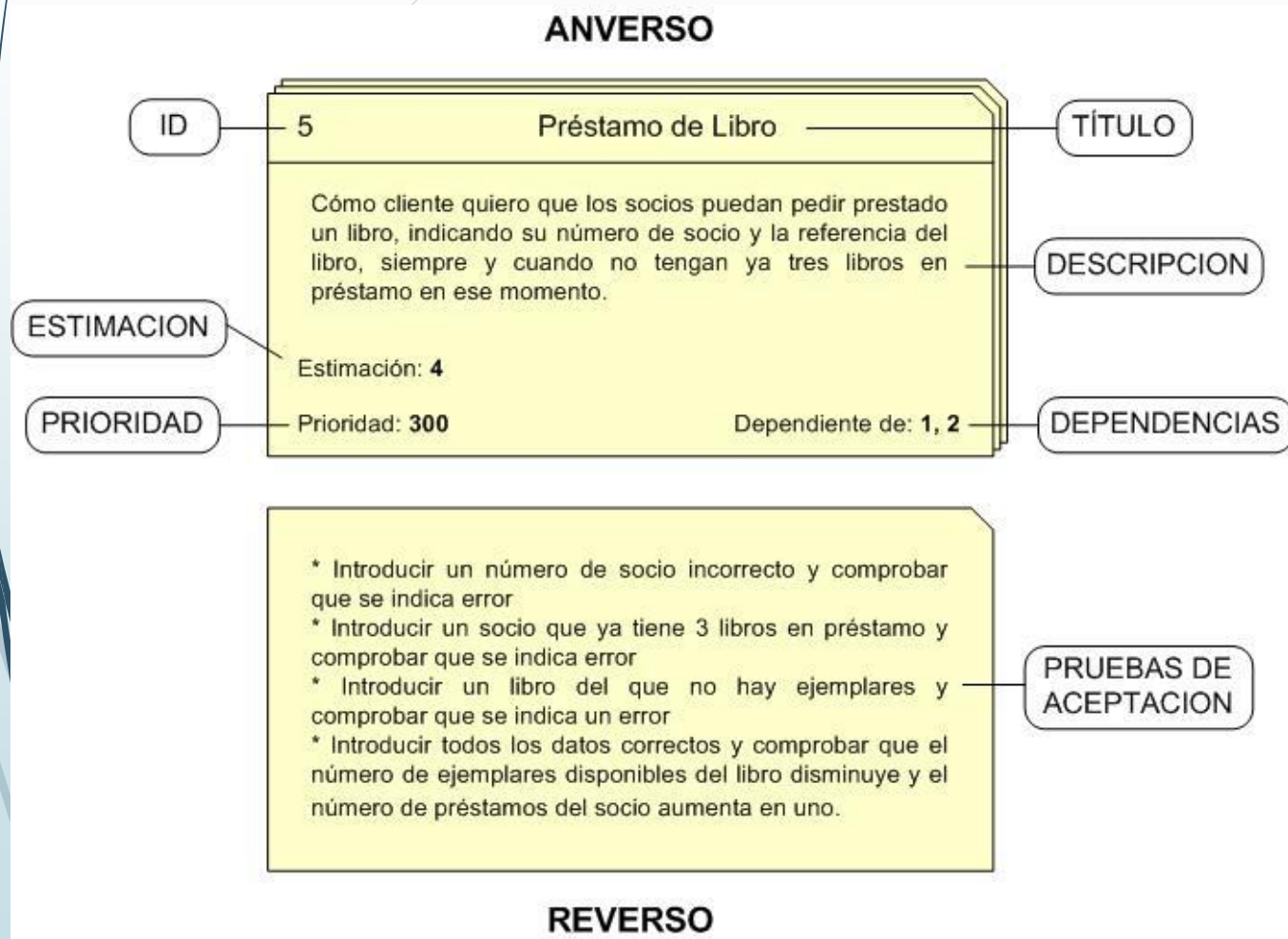


Figura 2.4. Esquema de un modelo del dominio

Según el Lenguaje de Modelado Unificado (UML) y Lenguaje de Modelado para Sistemas (SysML) se tiene



Esquema general de una Historia de Usuario...



*Como
[rol del usuario]
quiero
[objetivo]
para poder
[beneficio]*

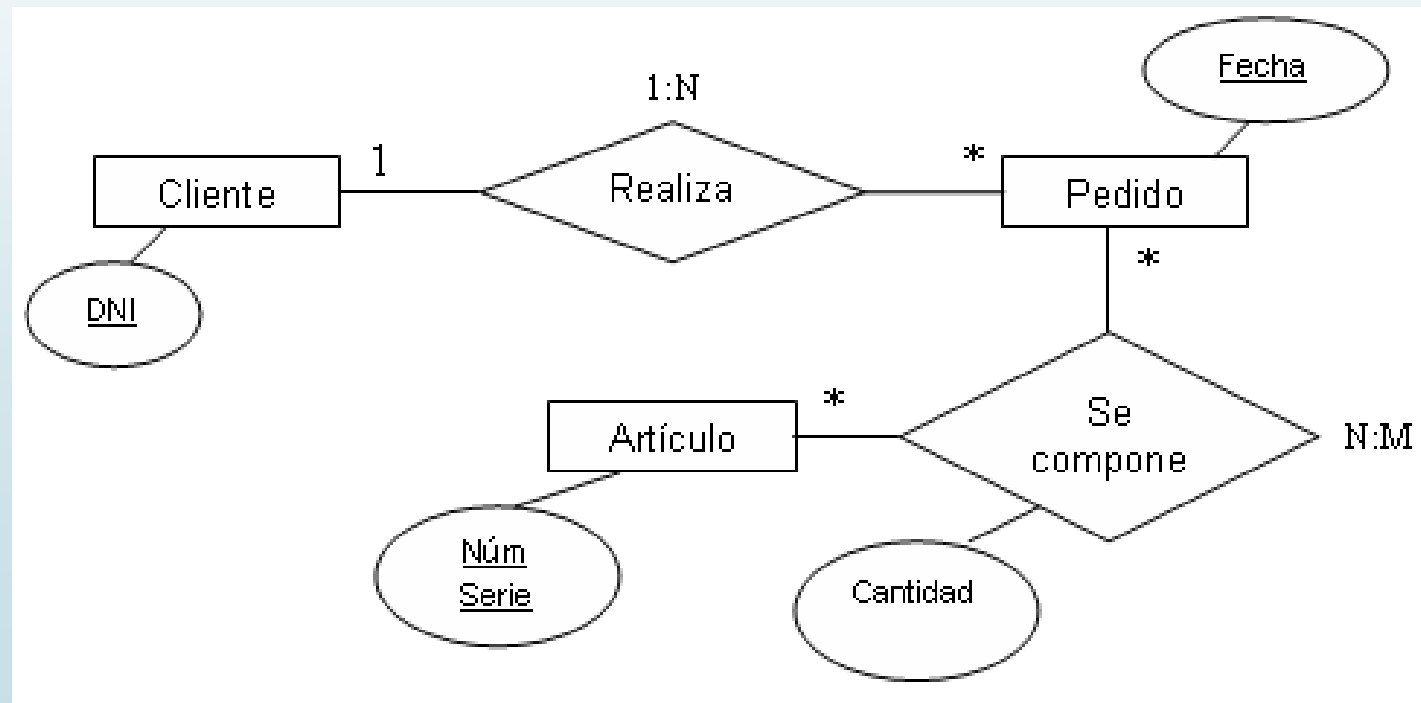


En general, un modelo de base de datos relacional estar compuesto por:

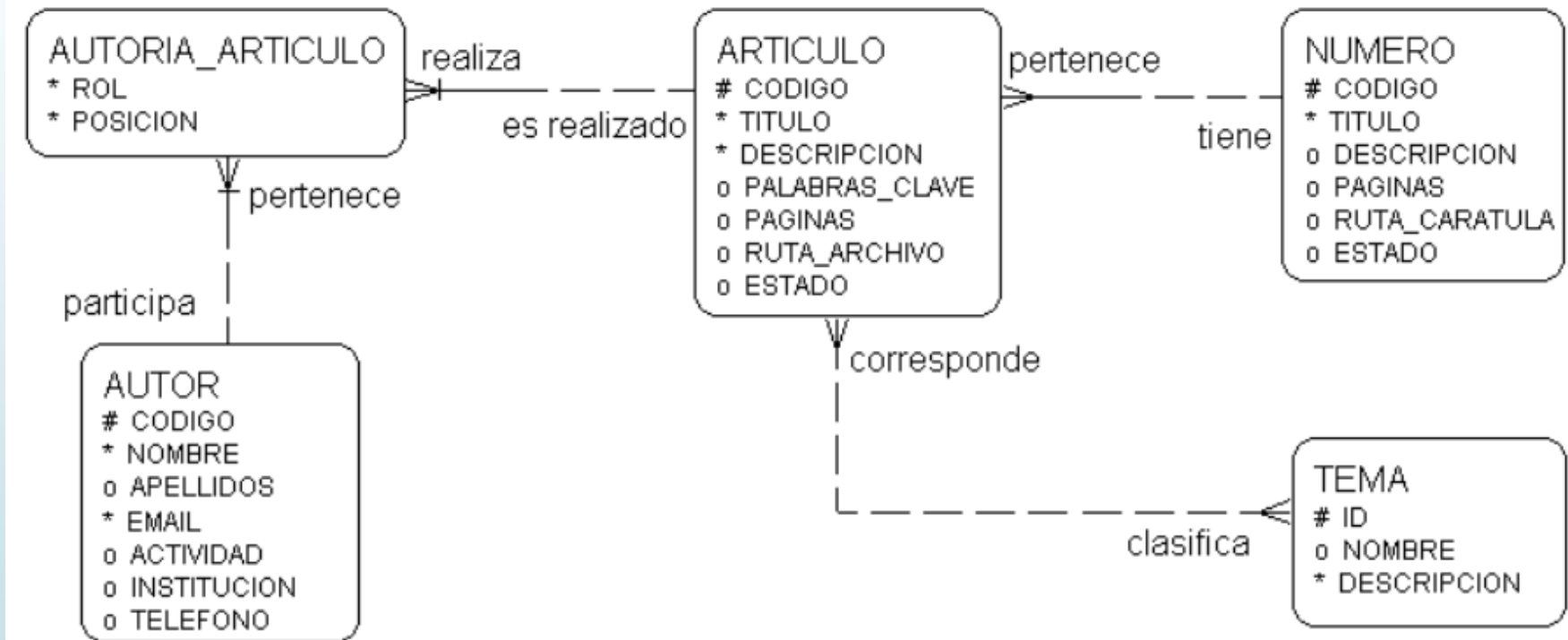
- **Entidades:** todo lo que existe y es capaz de ser descrito (sustantivo).
- **Atributos:** es una característica (adjetivo o sustantivo) de una entidad; estos sirven para identificarla, relacionarla y describirla.
- **Relaciones:** la conexión que existe entre 2 entidades (verbo).
- **Cardinalidad:** número de ocurrencias que pueden existir entre un par de entidades.
- **Clave:** uno o más atributos que "juntos" identifican de manera única a una entidad. *"Atributos que Identifican"*.
- **Clave Candidata:** son las posibles claves que pueden identificar de manera única una entidad.
- **Clave Primaria:** es la clave candidata con menos cantidad de atributos.
- **Clave foránea:** es un atributo que relaciona una entidad con otra. *"Atributos que relacionan"*

Para construir modelos de datos existen diversas notaciones:

- **Notación Chen:** Fue planteada por Peter Chen en los setenta.

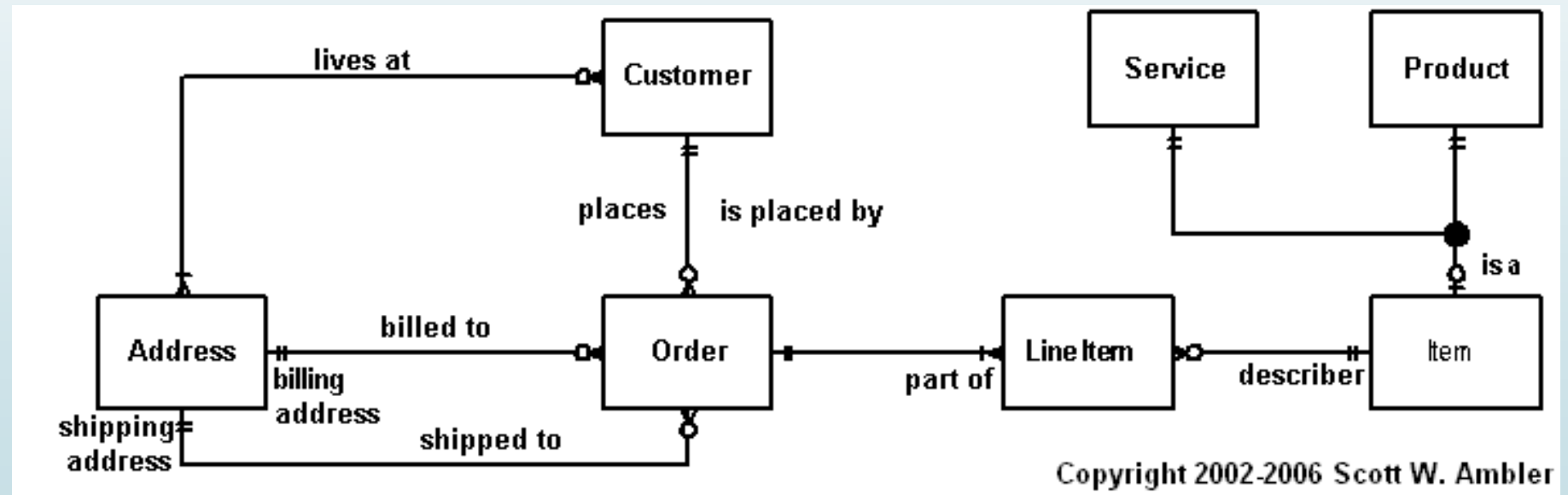


Notación Barker: fue adoptada por Oracle Corporation en sus productos de modelado de datos.

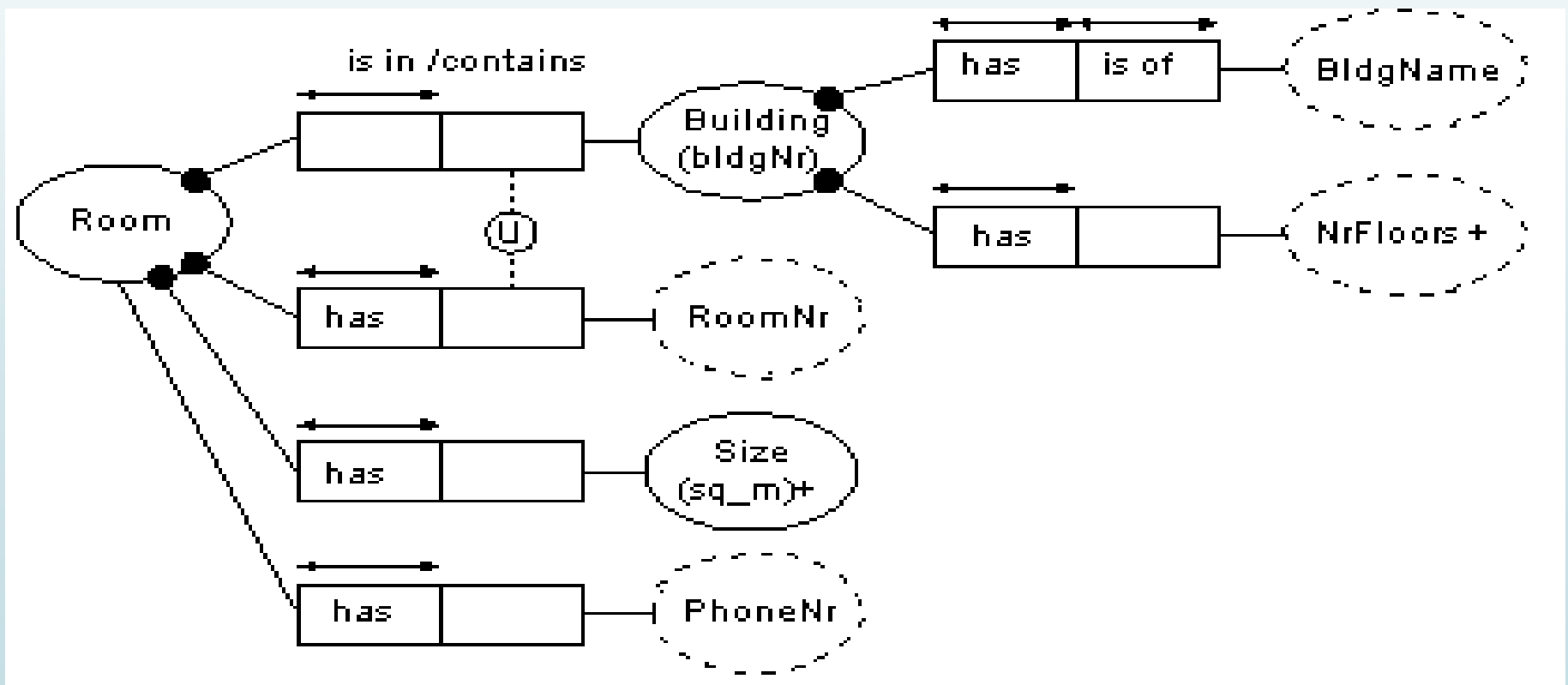


IE (Information Engineering): desarrollada inicialmente por Clive Finkelstein quien luego la refinó con el apoyo de James Martin;

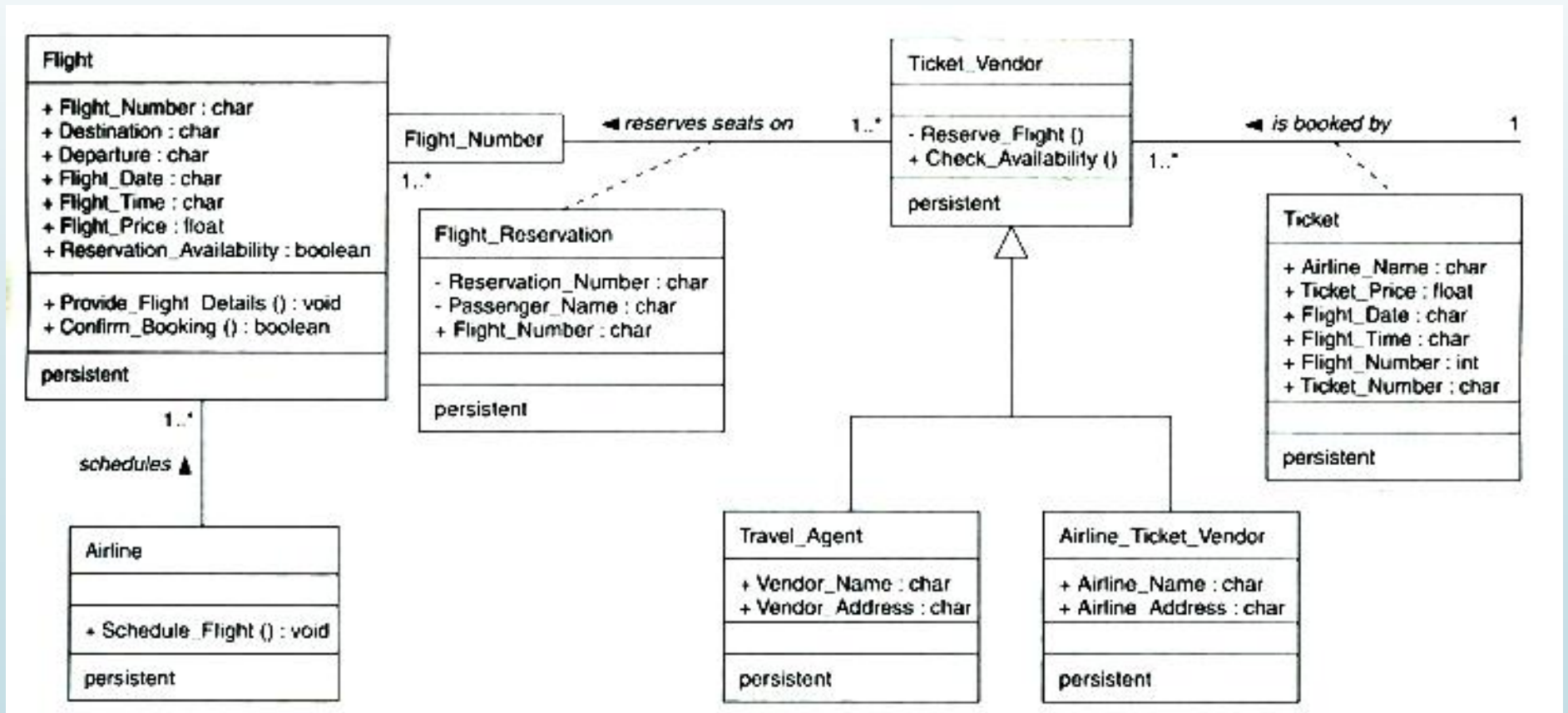
- Aunque es clara e intuitiva, sólo sirve para modelos de alto nivel de abstracción (modelos lógicos), pues no permite modelar los atributos de las entidades .



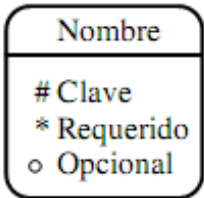
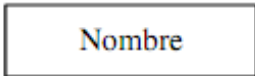
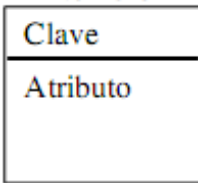
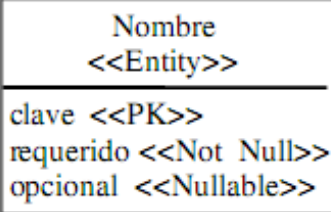
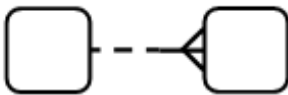
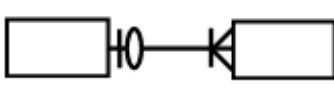
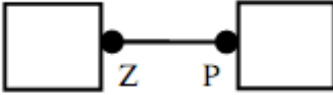
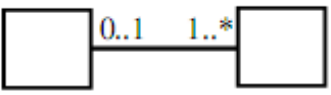
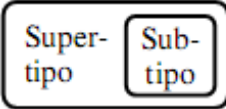
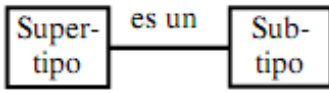
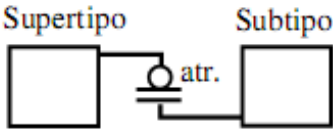
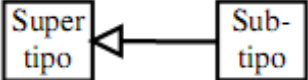
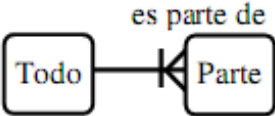
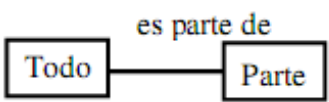
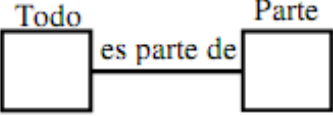

IDEF1X (Data Modeling): desarrollada por el Departamento de Defensa de los Estados Unidos dentro de una familia de estándares llamada IDEF(Integrated DEFinition Method).



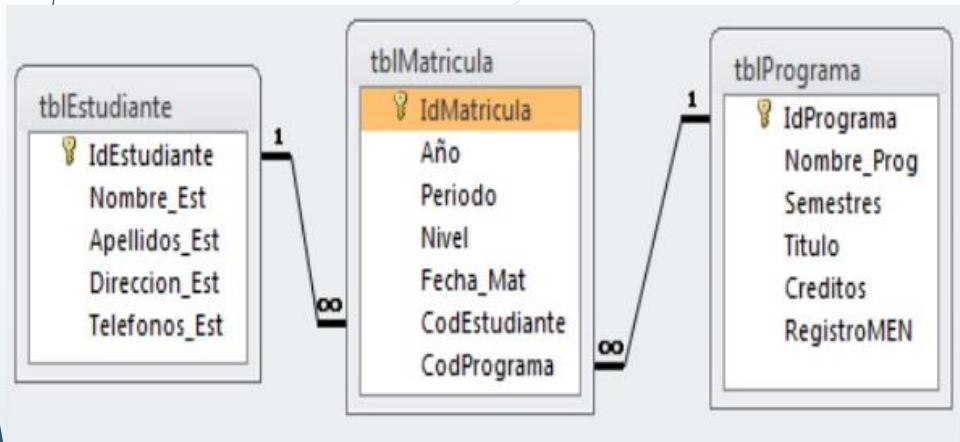
UML (Unified Modeling Language): si bien es un lenguaje de modelado objetual, se puede extender a través de perfiles para soportar otro tipo de modelos.



Estas notaciones y otras, son relevantes considerando que las bases de datos NO son Sólo Relacionales, Ni Sólo SQL.

Elemento de Modelado	Notación			
	Barker	IE	IDEFIX	UML
Entidad y características de los atributos				
Relación con multiplicidad de 0..1 a 1..n				
Herencia				
Agregación/ Composición				

Ejercicio en Clase, dado el modelo convertirlo en IE y UML




Elemento de Modelado	Notación			
	Barker	IE	IDEFIX	UML
Entidad y características de los atributos	<div>Nombre</div> <div># Clave</div> <div>* Requerido</div> <div>o Opcional</div>	<div>Nombre</div>	<div>Nombre</div> <div>Clave</div> <div>Atributo</div>	<div>Nombre</div> <div><<Entity>></div> <div>clave <<PK>></div> <div>requerido <<Not Null>></div> <div>opcional <<Nullable>></div>
Relación con multiplicidad de 0..1 a 1..n	<div>□</div> - <div>◻</div>	<div>□</div> - <div>◻</div>	<div>□</div> - <div>◻</div> <div>Z P</div>	<div>□</div> - <div>□</div> <div>0..1 1..*</div>
Herencia	<div>Super-tipo</div> <div>Sub-tipo</div>	<div>Super-tipo</div> es un <div>Sub-tipo</div>	<div>Supertipo</div> <div>Subtipo</div> <div>atr.</div>	<div>Super tipo</div> <div>Sub-tipo</div>
Agregación/ Composición	<div>Todo</div> es parte de <div>Parte</div>	<div>Todo</div> es parte de <div>Parte</div>	<div>Todo</div> es parte de <div>Parte</div>	<div>Todo</div> <div>Parte</div>



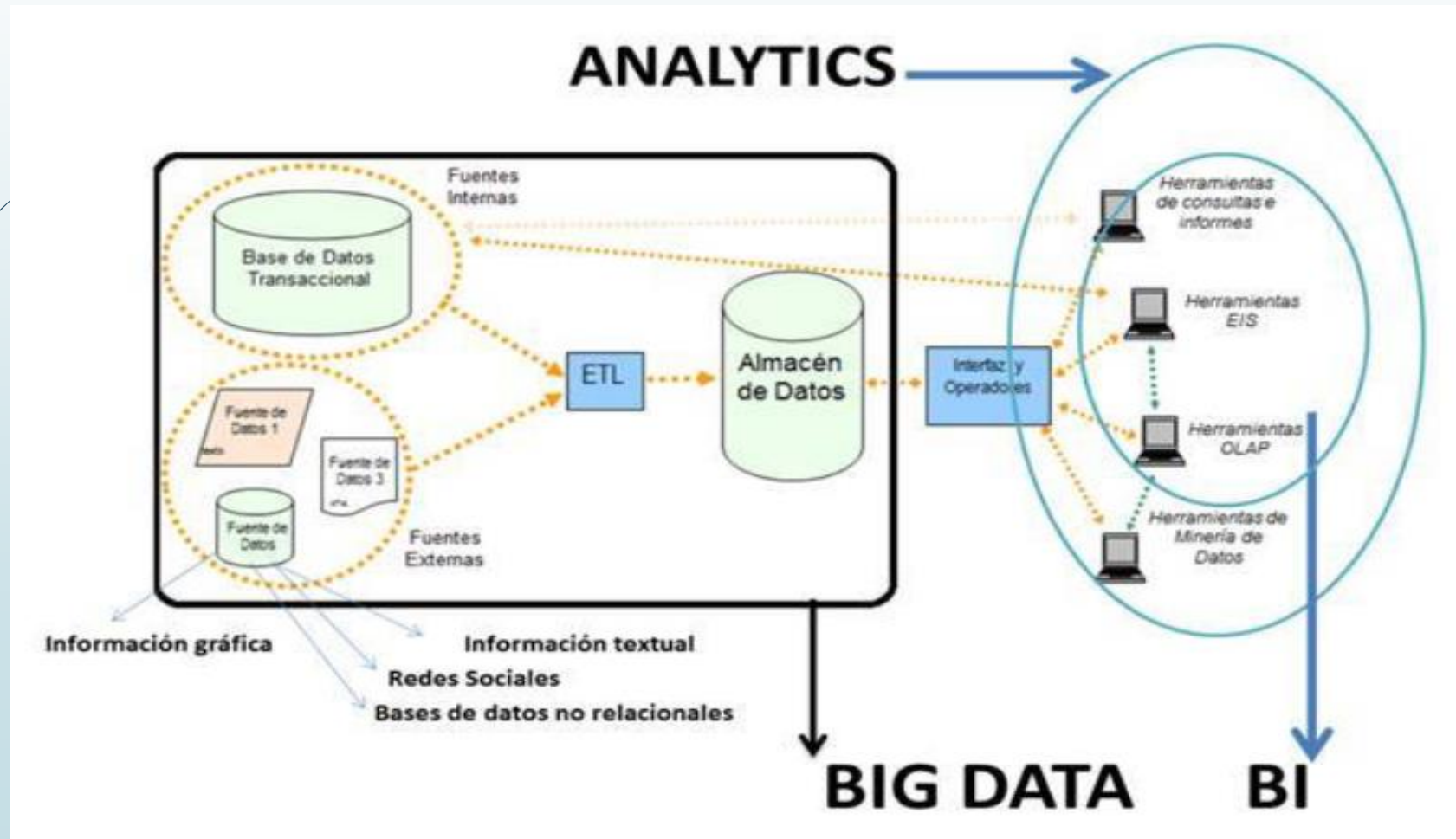
Lecturas Recomendadas

- Data Modeling
 - <http://www.agiledata.org/essays/dataModeling101.html>
- Directrices para la construcción de Artefactos de persistencia en el Proceso de desarrollo de software
 - <http://revista.eia.edu.co/articulos9/articulo%206.pdf>



Entonces, Cómo se relaciona la arquitectura de la información con Analítica, BigData, etc.

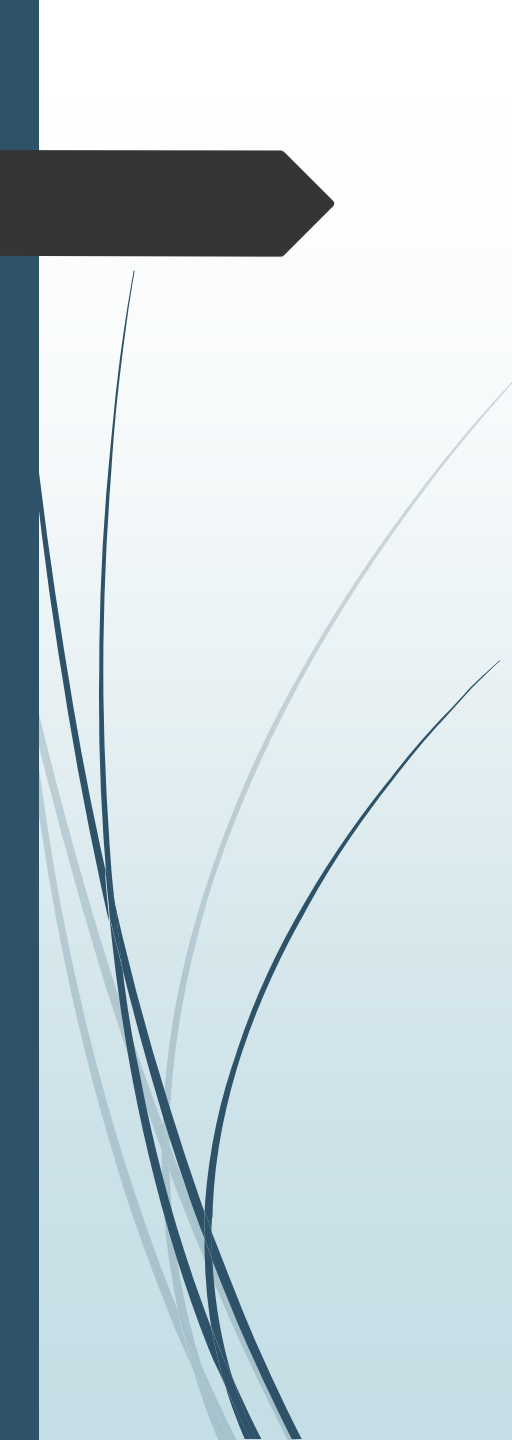
La arquitectura de la información girará en torno cómo crear y mantener el “Almacén de Datos”





Entonces, ¿por qué hasta ahora los conocemos?

- Masificación de Internet
- La potencia de los computadores
- Cantidad de Datos Disponibles
- La solución a la mayoría de los problemas es multidisciplinar.
- Grandes Empresas que se enfocaron en proponer herramientas para usuario final. Ej: Enterprise Miner (SAS), Modeler (SPSS), Power BI (Microsoft).
- El desarrollo de metodologías propias para resolver este tipo de problemas (SEMMA, CRISP-DM).

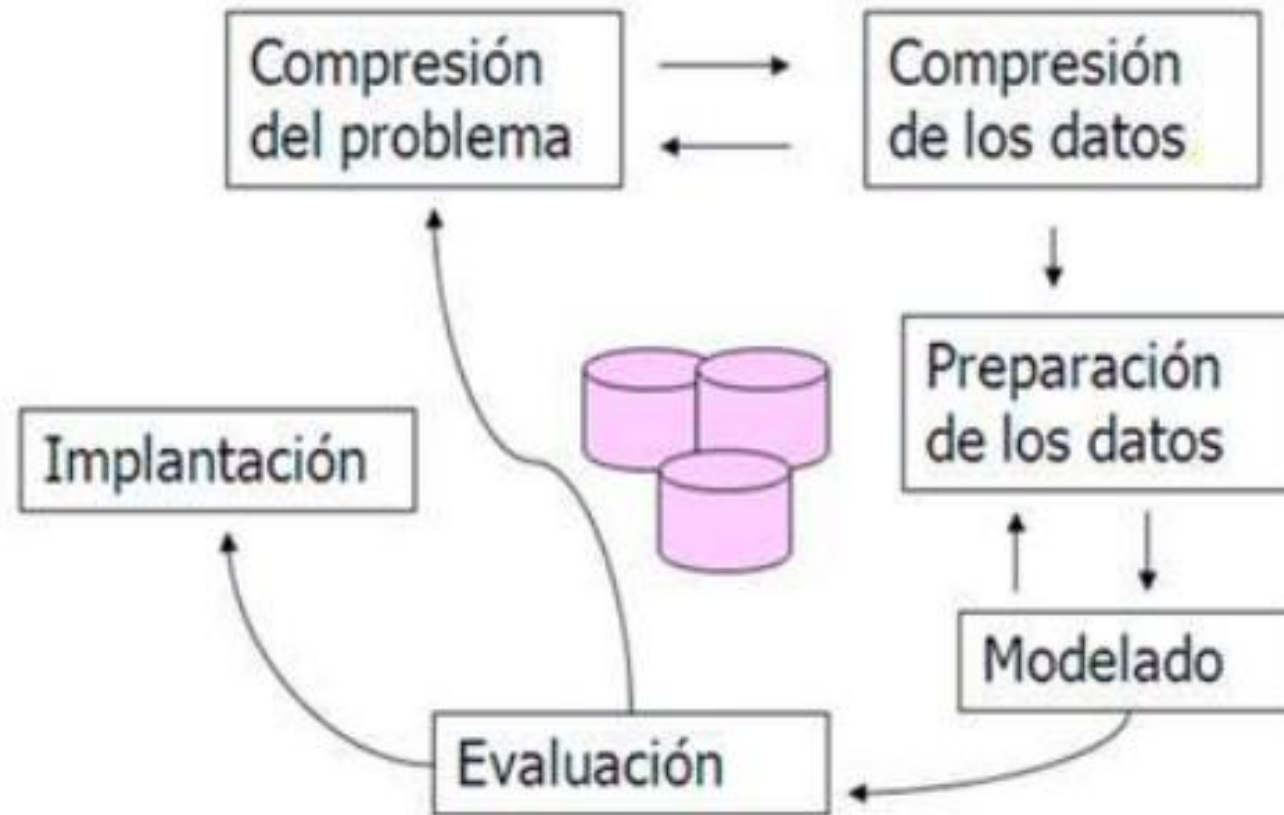


Para aplicar analítica, evidentemente debe estar soportada en una excelente gestión de la información (dato), requiere:

- Un acercamiento **holístico** para el manejo y gestión de una organización.
- Se debería adoptar una **vista integral** del negocio que cubra:
 - Los procesos de negocio.
 - Los sistemas de información.
 - Los datos e Información.
 - La Infraestructura tecnológica.
 - ... ¿Otros más?

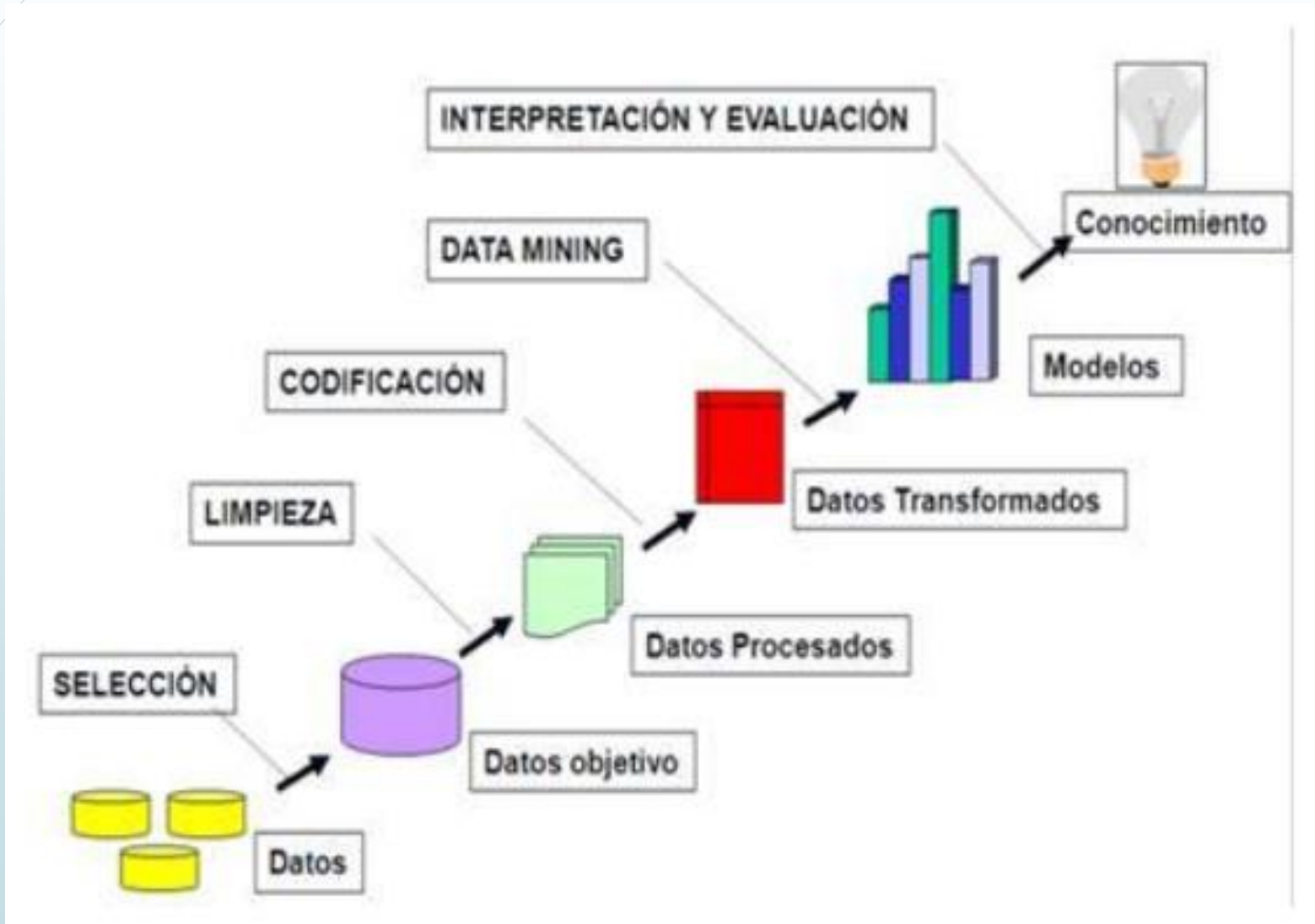
Toda Base de Datos debería soportar

Ciclo del Análisis de Datos



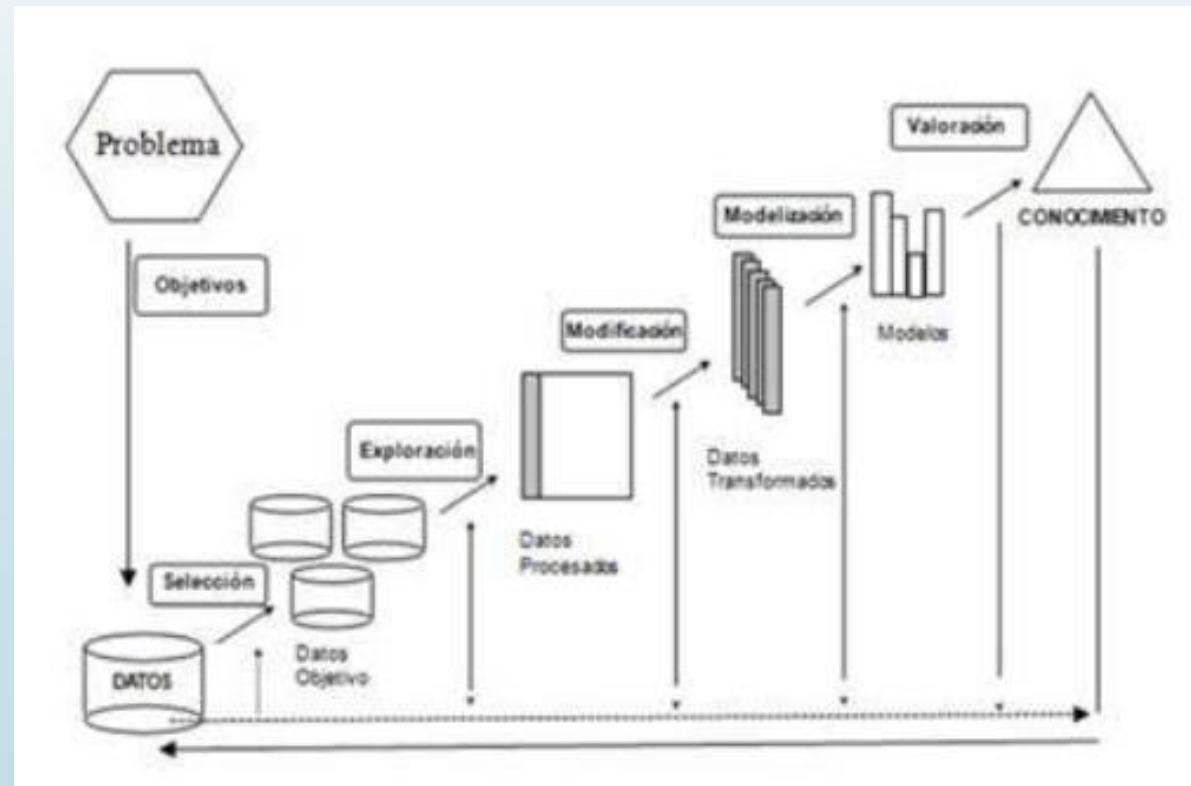
Caso Calidad del Aire.

Y se debe considerar la escala

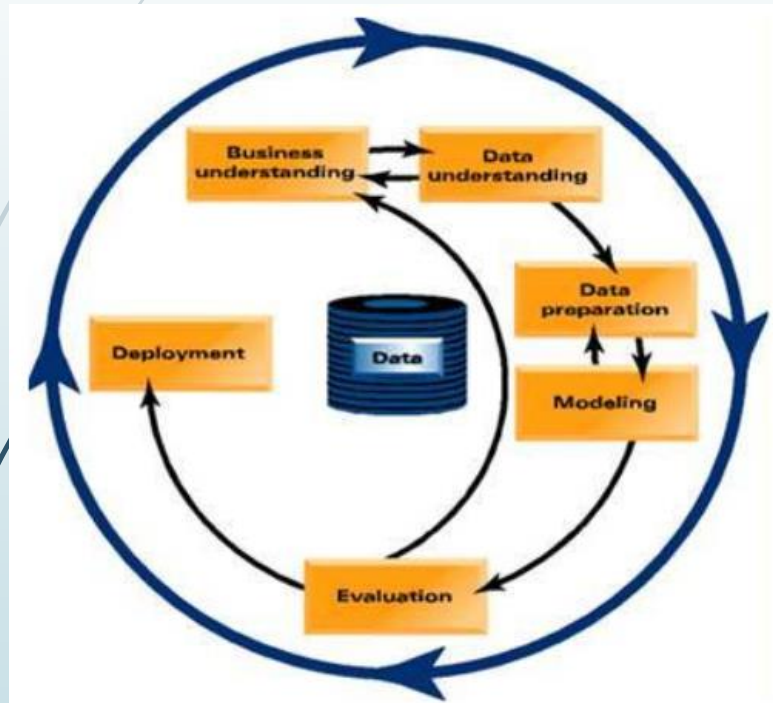


Existen metodologías que guían el proceso de construcción y análisis de la base de datos...

- SEMMA de SAS Institute. Esta divide el proceso en 5 fases: SAMPLE, EXPLORE, MODIFY, MODEL y ASSES. (Muestreo o Selección, exploración, modificación, modelización y evaluación o valoración).



Ahora CRIPS-DM de IBM, Cross Industry Standard Process for Data Mining



CRISP-DM: Fases

- **Comprensión del negocio**
Entendiendo los objetivos y requerimientos del proyecto, Data mining problem definition
- **Comprensión de los datos**
Recogida inicial de datos. Identificación de los problemas de calidad
- **Preparación de los datos**
Tablas, registros y selección de atributos, Transformación y limpieza de los datos
- **Modelado**
Selección y aplicación de técnicas de modelización, calibración de parámetros
- **Evaluación**
Objetivos del negocio y técnicas de evaluación
- **Desarrollo**
Desarrollo de los resultados del modelo, Implementación del proceso de minería de datos



En términos prácticos la Base de Datos debe estar en la capacidad de soportar la respuesta a siguientes preguntas:

- **Observar** ¿qué está ocurriendo?
- **Comprender** ¿por qué ocurre?
- **Predecir** ¿qué ocurriría?
- **Colaborar** ¿qué debería hacer el equipo?
- **Decidir** ¿qué camino se debe seguir?



El papel de la BD para el análisis de volúmenes de datos



El análisis de los datos día requiere de:

- El uso de técnicas estadísticas para aprender de los datos.
- La búsqueda de patrones y anomalías
- Predicciones con alto margen de fiabilidad
- Alta capacidad de procesamiento
- Extracción del conocimiento embebido en los datos.
- Experiencia en el Negocio.
- Profesionales con competencias para responder el Cómo y Cuándo se hace (Pensar antes de Hacer, Hacer y Hacer Pensando).



¿Cuál es el tipo de profesionales que se requiere?

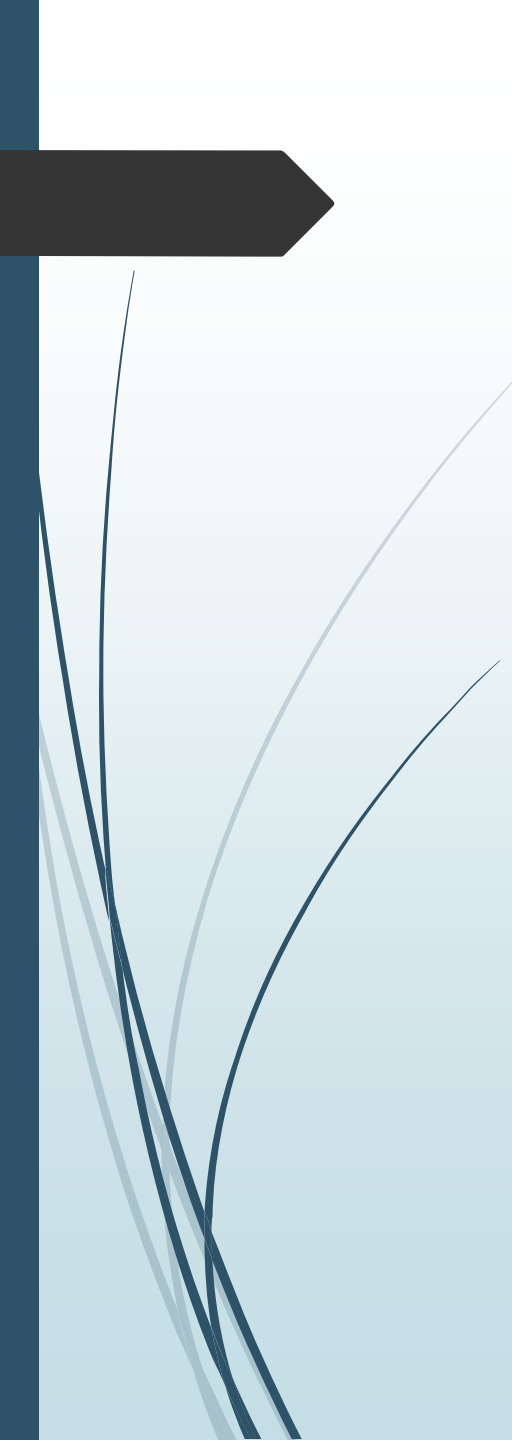
Profesionales, con habilidades en matemáticas, estadística e ingeniería informática, que sean capaces de extraer el máximo valor de los datos de la organización.



¿Dónde se requiere la gestión y análisis de volúmenes de datos?

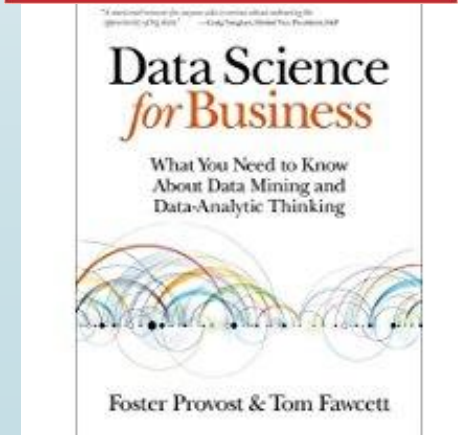
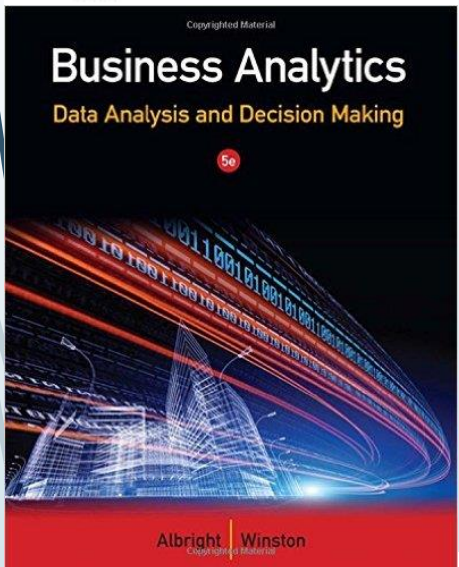
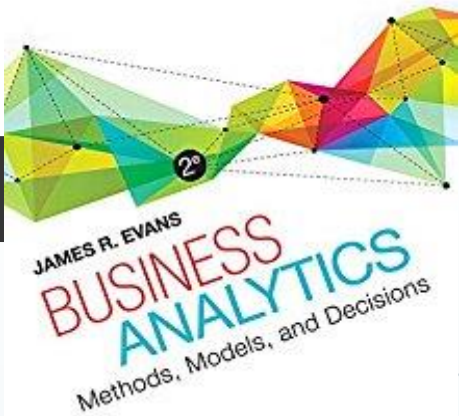
- Investigación científica
- Sanidad
- Seguridad
- Medios de Comunicación
- Finanzas
- Seguridad
- Política

*En general donde se requieran herramientas para **ganar competitividad** y/o **mejorar la calidad de vida** de las personas*



El sistema de gestión de base de datos a parte de almacenarlos debería soportar según el dominio o problema:

- Modelar datos.
- Segmentación o perfilado a través de patrones.
- Descubrir relaciones.
- Identificar regularidades. (Espacio)
- Identificar tendencias. (Tiempo)
- Reglas de Asociación.
- Pronósticos.
- Simulaciones.



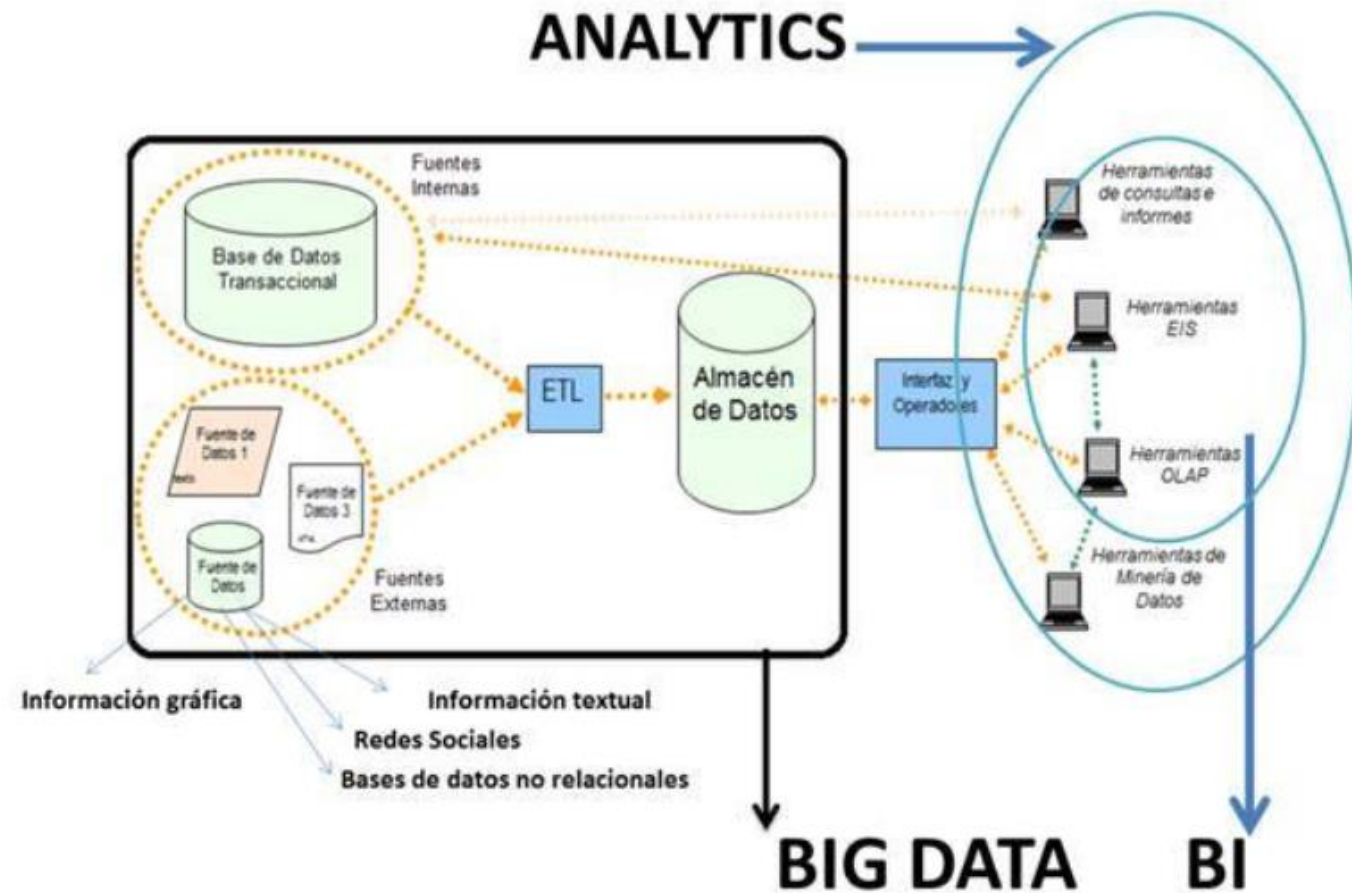
Además, explicar el Pasado y Predecir el Futuro

- Tradicionalmente, un sistema tiene la capacidad de decir que las ventas de abrigo aumentaron en Barranquilla, pero no explicará por qué.
- El concepto de debe evolucionar... y el sistema deberá estar en la capacidad de explicar que en las madrugadas en Barranquilla se presentan oleadas de frío intenso, las cuales posiblemente durarán por un período de 3 meses... y se presentarán cada dos años... lo cual es un reflejo del cambio climático extremo atribuido a la contaminación ambiental...

La BD debe ser transversal a Data Mining, Analytics y BigData.

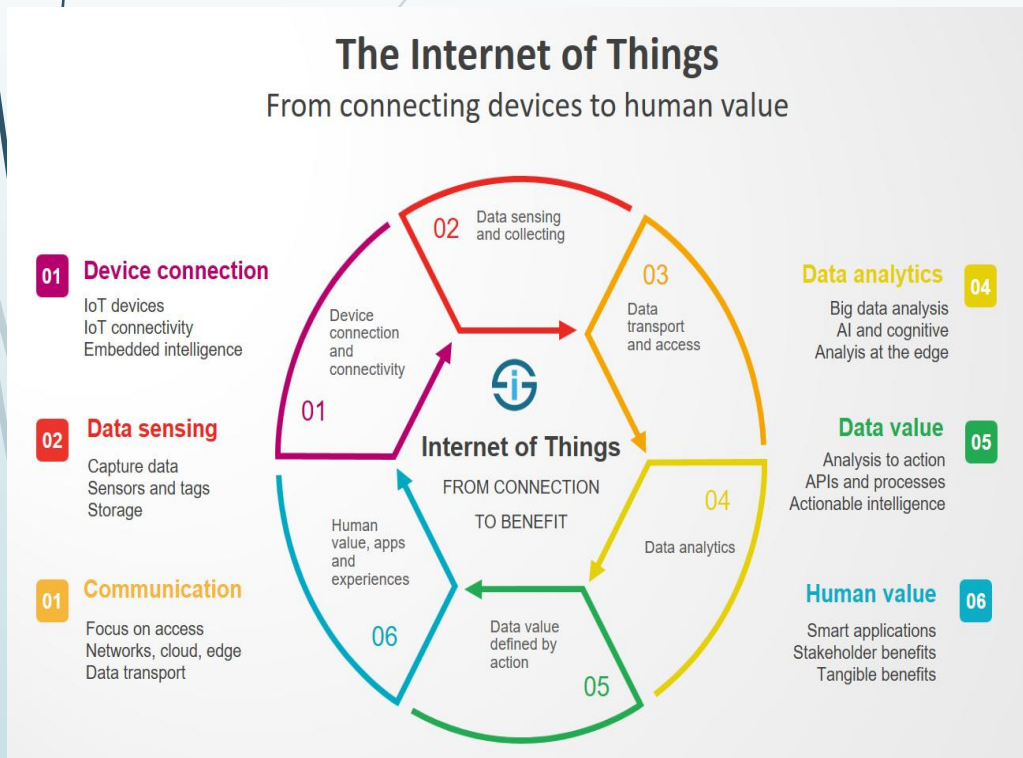
No Torture los Datos, para extraer algo que no tienen

Nuevamente, la base de datos debe soportar la integración...



- **Consultas e informes simples** (*Querys y reports*).
- **Cubos OLAP** (*On-Line Analytic Processing*). Exploración, tablas dinámicas, etc.
- **EIS (Executive Information Systems)**: Soluciones que permiten visualizar, de una forma rápida y fácil, el estado de una determinada situación empresarial, presente o pasada, y que permite detectar anomalías u oportunidades.
- **DSS (Decision Support Systems)**: Aplicación informática que basándose en modelos matemáticos y mediante análisis de sensibilidad permite ayudar a la toma de decisiones.
- **Data Mining o minería de datos**: Pueden considerarse sistemas expertos que nos permiten “interrogar a los datos” para obtener información
- **ETL**: Extract, Transform and Load («extraer, transformar y cargar», procesor que permite mover datos desde múltiples fuentes, reformatearlos y limpiarlos, y cargarlos en otra base de datos,

Los datos y la información ha existido desde siempre, y su registro desde el milenio IV A.C.



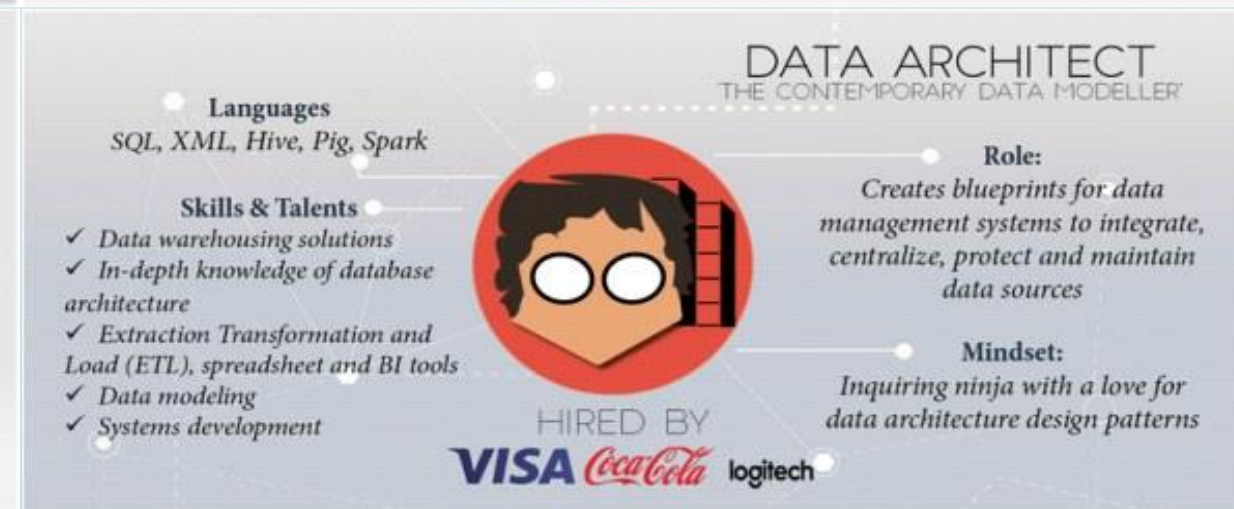
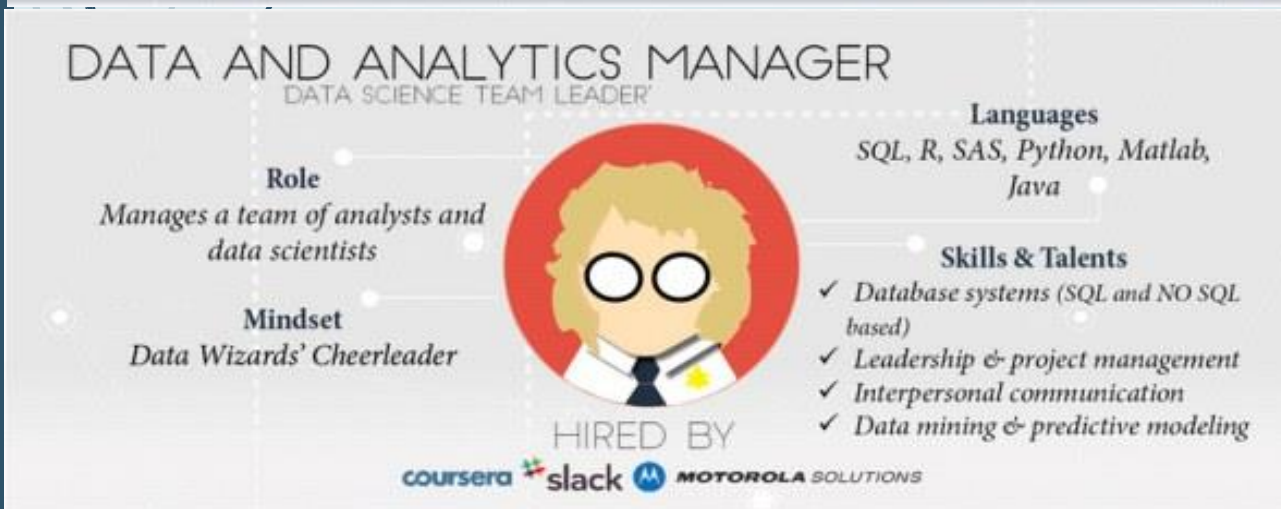
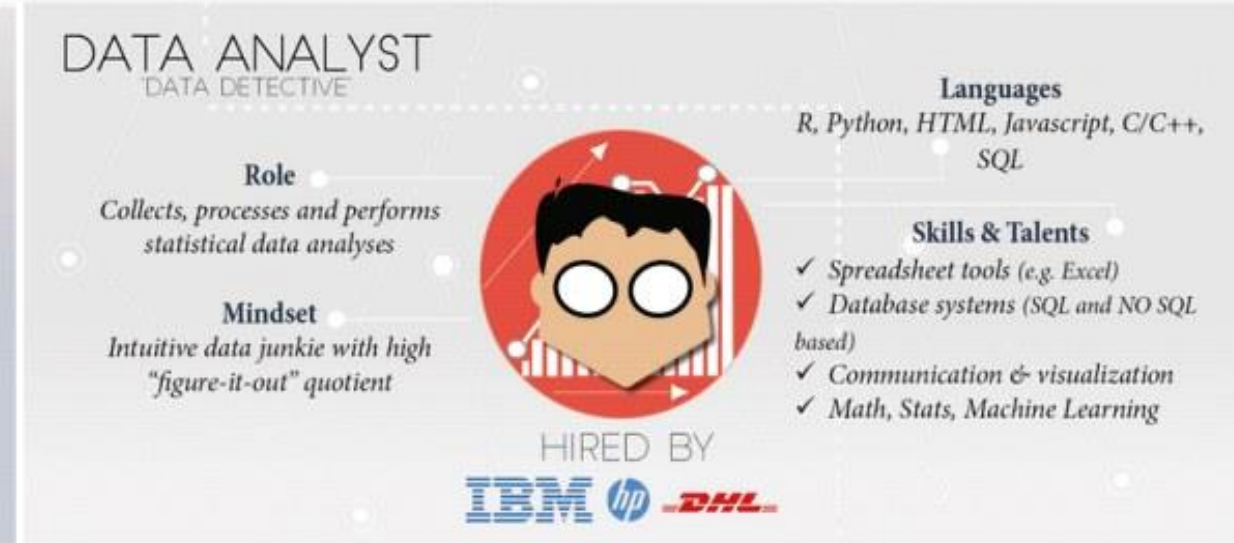
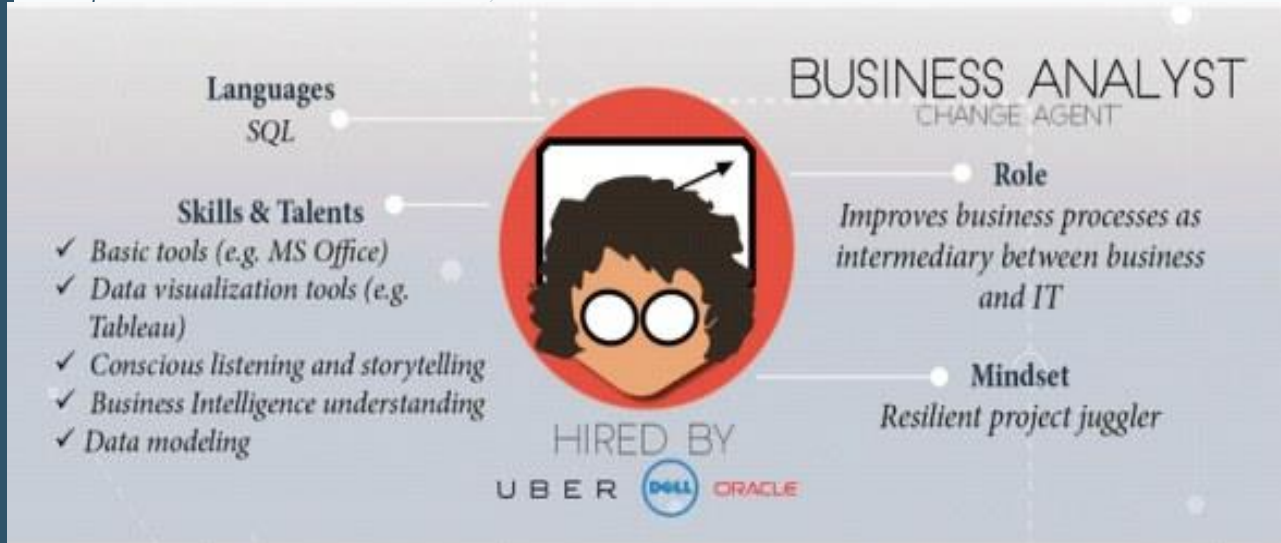
Adaptado de: "Defining the Big Data Architecture Framework", Demchenko, University of Amsterdam, 2013



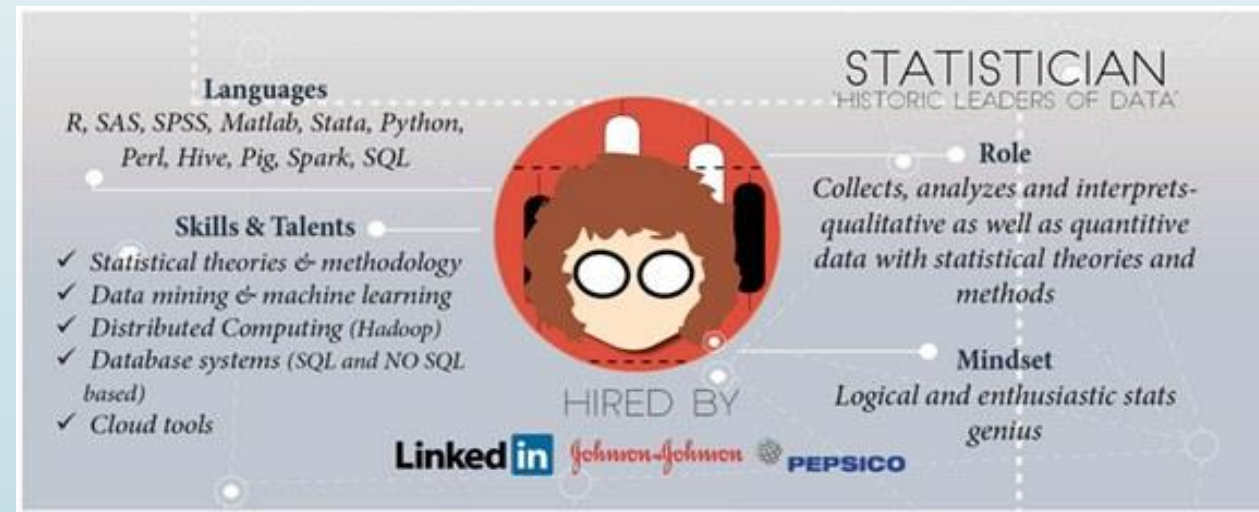
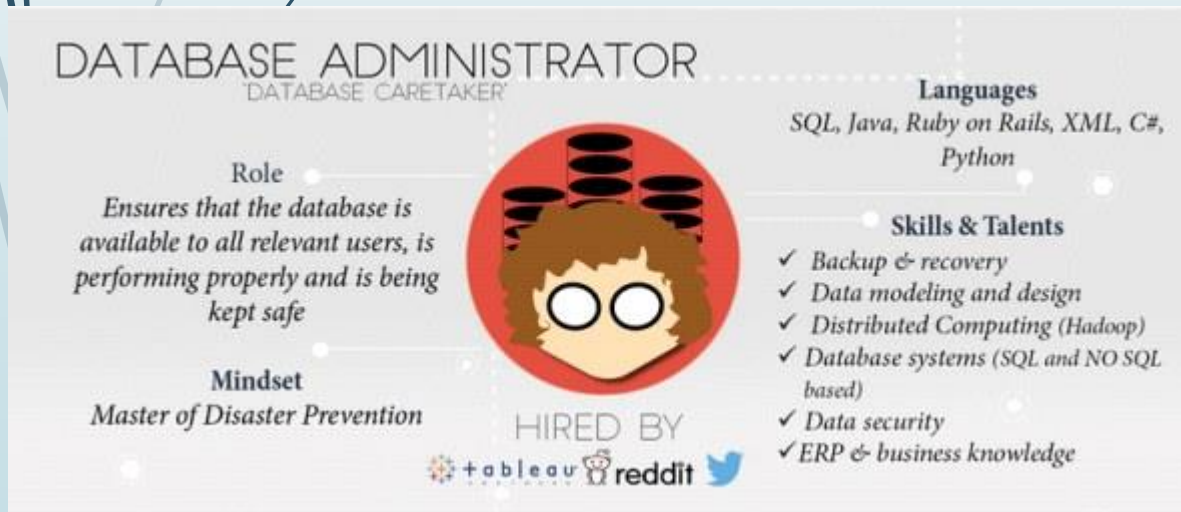
Sobre el Big Data

- Las organizaciones requieren adquirir ventajas competitivas, aprovechando el alto flujo de información de hoy en día (georeferenciamiento, redes sociales, clics en una página, fotos, videos, música, sensores, ...).
- Para BI y BD se requieren las siguientes 5 propiedades:
 - **Volumen:** un motor de un avión puede generar 10TB de datos en 30 minutos de vuelo.
 - **Velocidad (alta frecuencia):** 140 caracteres por tweet, generan alrededor de 333 GB por hora, aproximadamente 8TB por día
 - **Variedad:** fotos, música, páginas web, ...
 - **Valor:** identificar la información valiosa, transformarla y extraer más información para analizarla, transformarla y generar conocimiento.
 - **Veracidad:** Se deben llegar a conclusiones objetivas. Los datos usados no fueron manipulados.

Desde este enfoque emergen los siguientes perfiles profesionales, posiblemente tú lo seas y no lo sabías:



Adicionalmente, se





Prescriptive

Descriptive

Experimental

Theoretical

Data Analysis/Analytics

Machine Learning

Big Data

Methods and Algorithms

Big Data Analytics

Data Mining

Software Tools

2015
Dahl Winters

- Si generas información con una técnica de Machine Learning, sus resultados deben retroalimentar la base de datos

Un ejemplo de tal integración es el esfuerzo de las grandes empresas por ofrecer herramientas de BI que incorporan Minería de Datos.

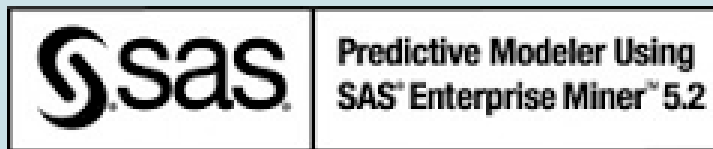
- SQL Analysis Services



- Oracle Data Mining



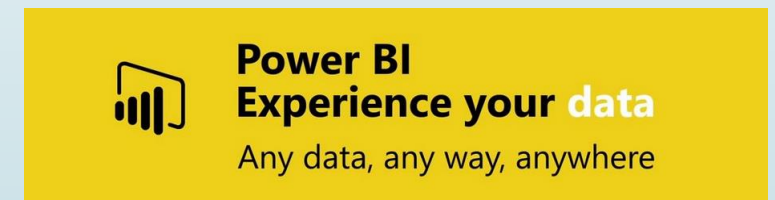
- SAS Enterprise Miner



- IBM SPSS Modeler



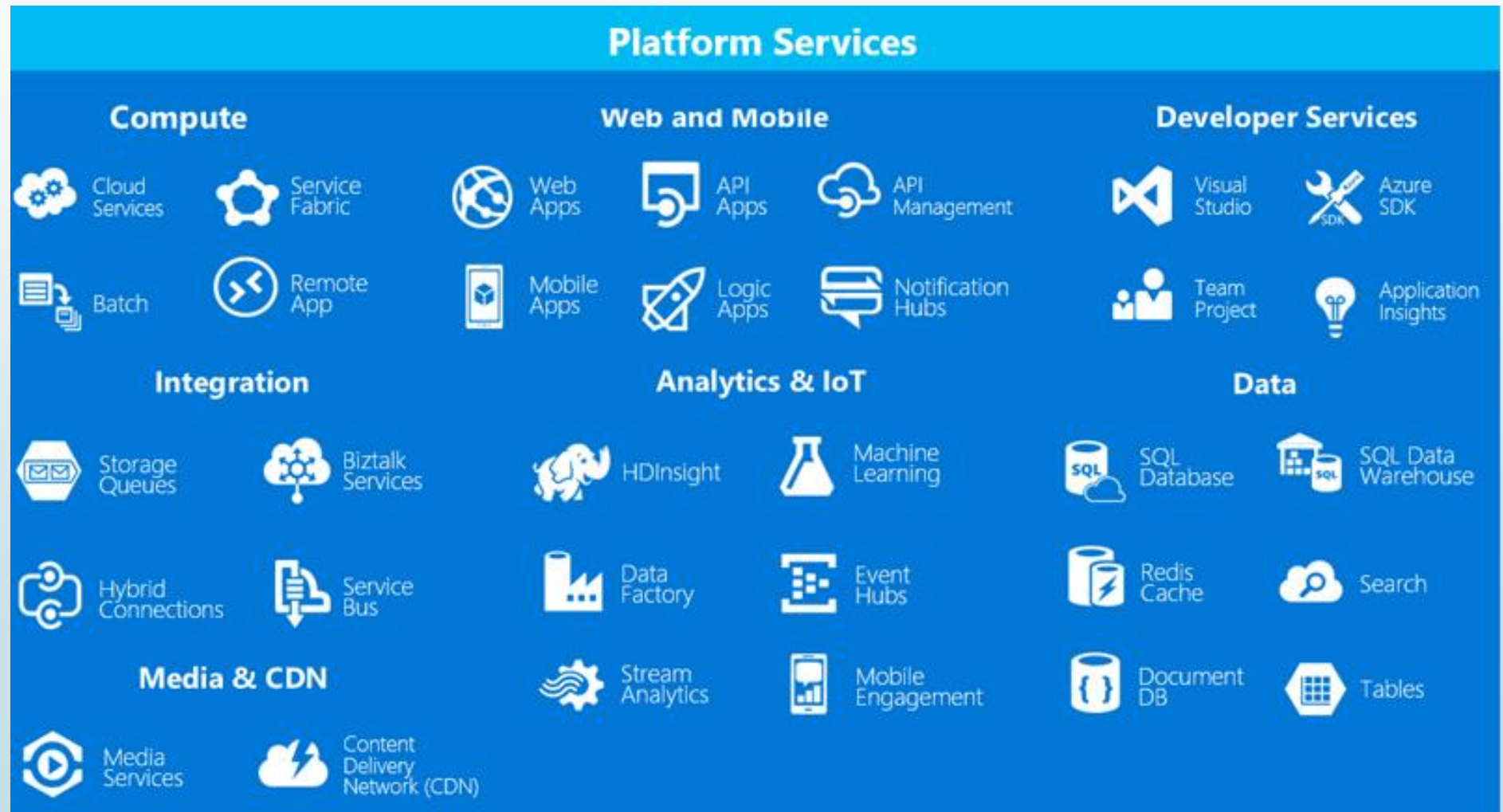
- Microsoft Power BI (visualización)



- Entornos de Desarrollo Integrado IDE's

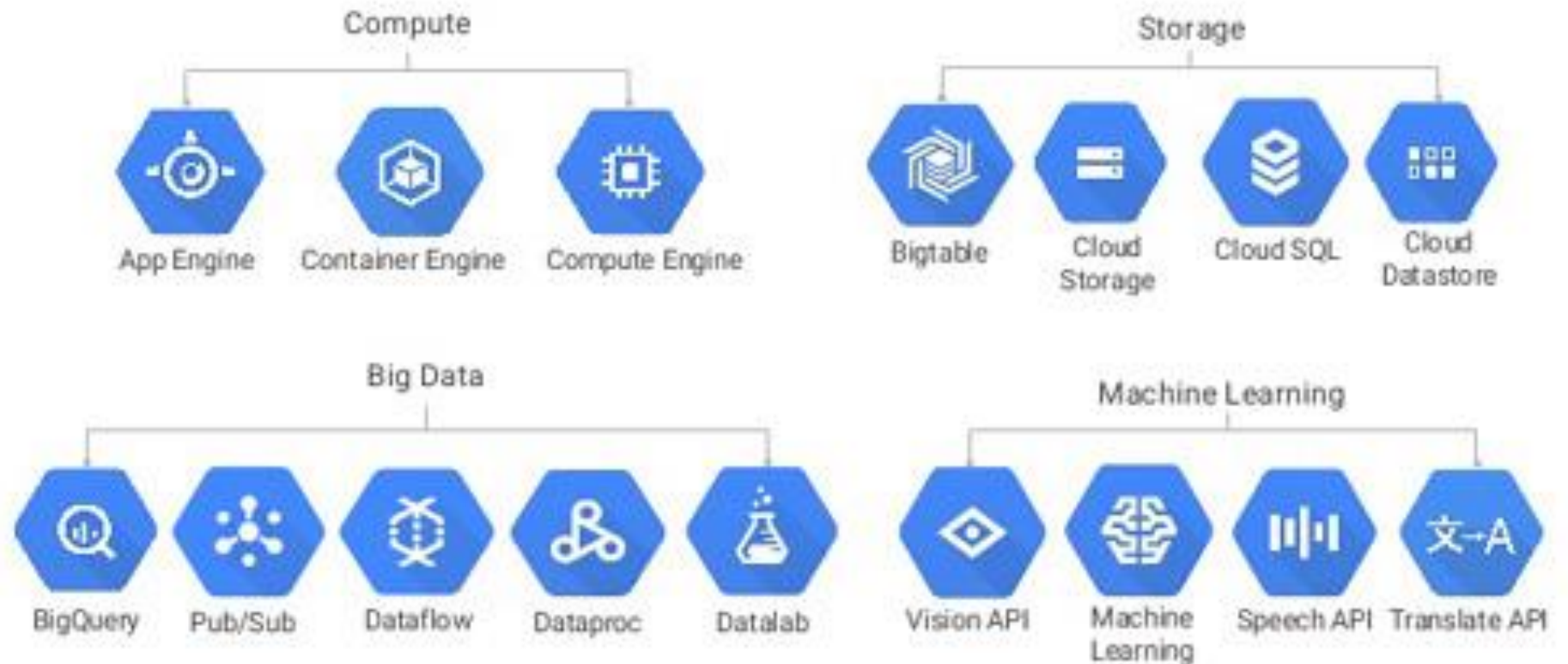


Servicios en la Nube como Microsoft Azure:



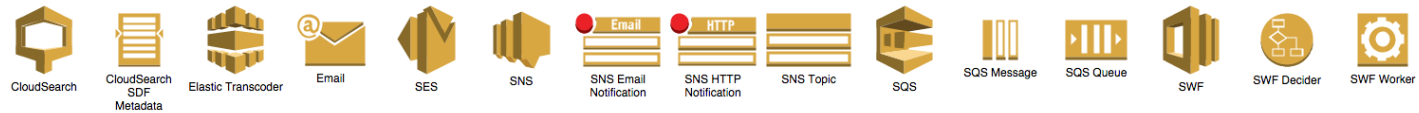


Google Cloud Platform

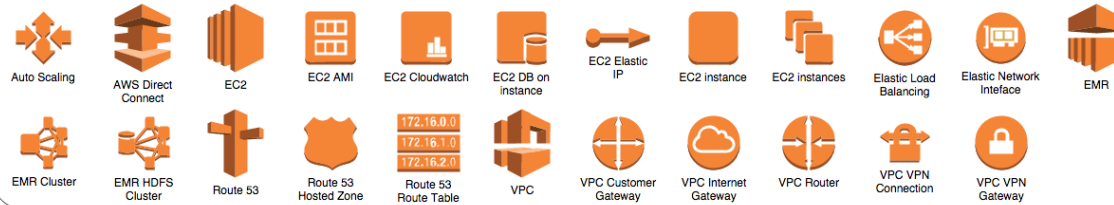




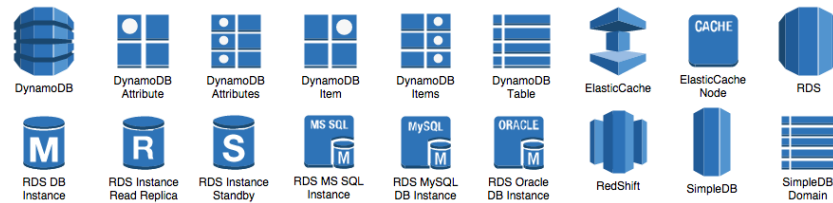
Application Services



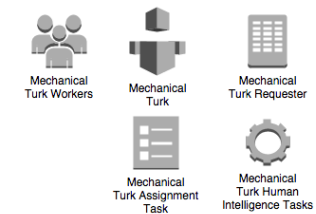
Compute and Networking



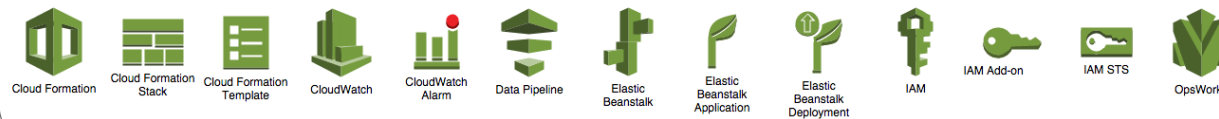
Database



On-Demand Workforce



Deployment and Management





No todo es SQL ni Relacional



No todo se puede representar mediante el esquema relacional, forzamos la realidad a adaptarse al modelo...

- Las aplicaciones web que utilizan algún tipo de bases de datos para funcionar.
- Tradicionalmente se usan sistemas de gestión de bases de datos SQL como MySQL, Oracle o MS SQL.
- Emergen NoSQL (Not only SQL – No solo SQL). Con la llegada de aplicaciones como Facebook, Twitter o Youtube, cualquier usuario podía subir contenido, provocando así un crecimiento exponencial de los datos.
- El mundo es más social que transaccional. Se generan mas twits que transferencias de dinero.

THE WORLD OF DATA

NUMBER
OF EMAILS
SENT
EVERY SECOND

2.9
MILLION

DATA
CONSUMED BY
HOUSEHOLDS
EACH DAY

375
MEGABYTES

VIDEO
UPLOADED TO
YOUTUBE EVERY
MINUTE

20
HOURS

DATA PER
DAY
PROCESSED
BY GOOGLE

24
PETABYTES

TWEETS
PER
DAY

50
MILLION

TOTAL MINUTES
SPENT ON
FACEBOOK
EACH MONTH

700
BILLION

DATA SENT
AND RECEIVED
BY MOBILE
INTERNET USERS

1.3
EXABYTES

PRODUCTS
ORDERED ON
AMAZON PER
SECOND

72.9
ITEMS



SOURCES: Cisco, comScore, MapRadium, Radicati Group, Twitter, YouTube

IN THE 21ST CENTURY, we live a large part of our lives online. Almost everything we do is reduced to bits and sent through cables around the world at light speed. But just how much data are we generating? This is a look at just some of the massive amounts of information that human beings create every single day.

All in the NoSQL Family

NoSQL databases are geared toward managing large sets of varied and frequently updated data, often in distributed systems or the cloud. They avoid the rigid schemas associated with relational databases. But the architectures themselves vary and are separated into four primary classifications, although types are blending over time.



Document databases

Store data elements in document-like structures that encode information in formats such as JSON.



Common uses include content management and monitoring Web and mobile applications.



EXAMPLES:

Couchbase Server, CouchDB, MarkLogic, MongoDB



Graph databases

Emphasize connections between data elements, storing related "nodes" in graphs to accelerate querying.



Common uses include recommendation engines and geospatial applications.



EXAMPLES:

Allegrograph, IBM Graph, Neo4j



Key-value databases

Use a simple data model that pairs a unique key and its associated value in storing data elements.



Common uses include storing clickstream data and application logs.



EXAMPLES:

Aerospike, DynamoDB, Redis, Riak



Wide column stores

Also called table-style databases—store data across tables that can have very large numbers of columns.



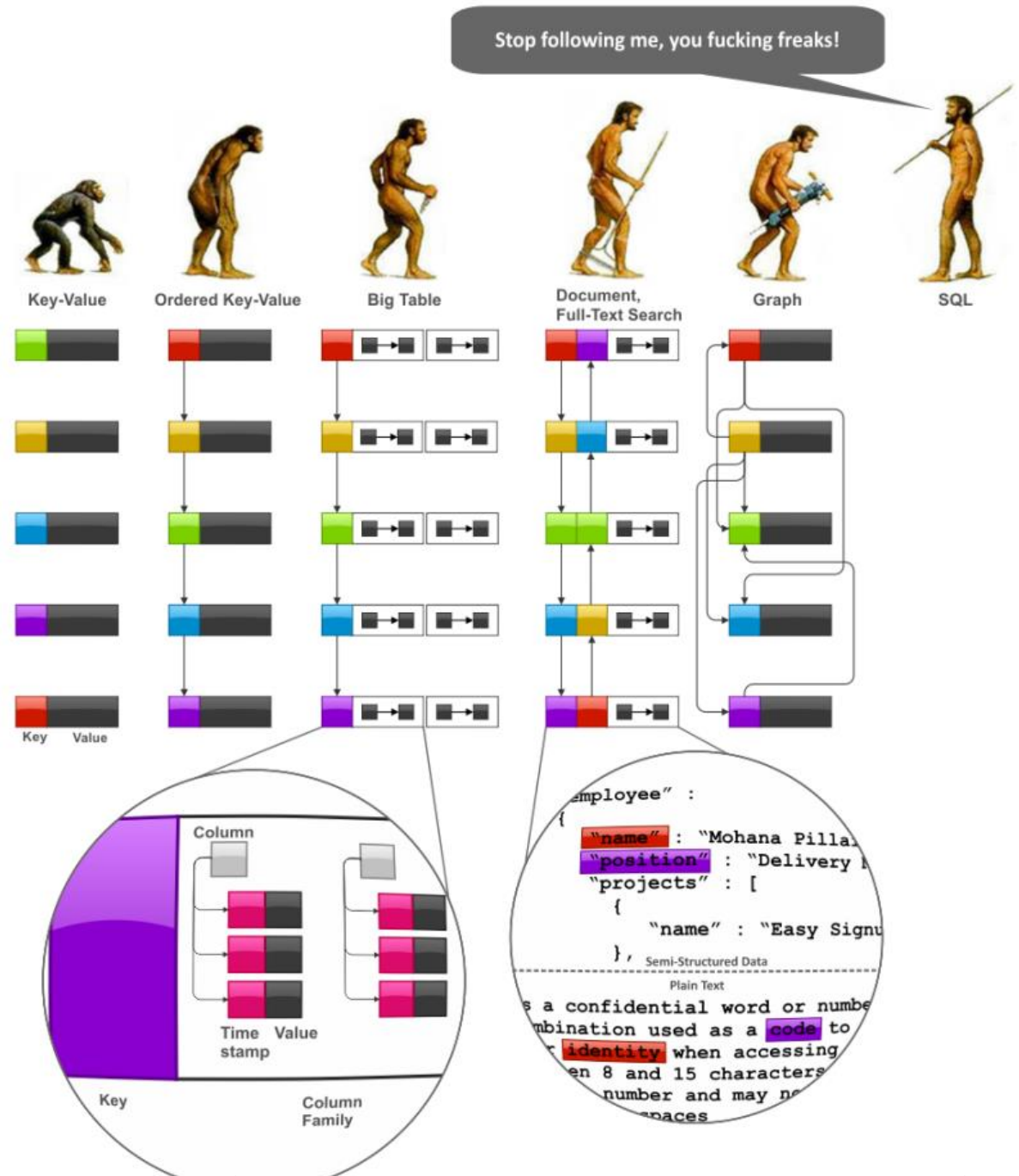
Common uses include Internet search and other large-scale Web applications.



EXAMPLES:

Accumulo, Cassandra, HBase, Hypertable, SimpleDB

La Clave
Valor es
la base
para
estos
SGBD





Gracias

