THE BRIDGE

Data Science

Part-time Bootcamp

Inventariamos stock

Hacemos balance de caja

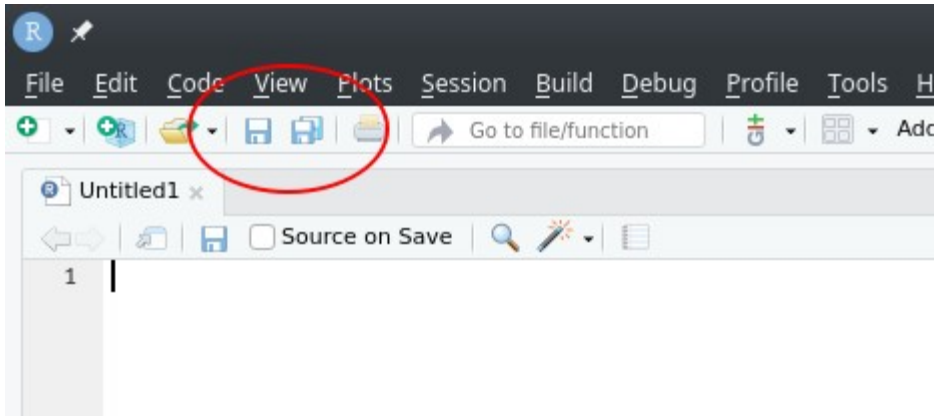
Ajustamos los precios

Gestionamos pedidos

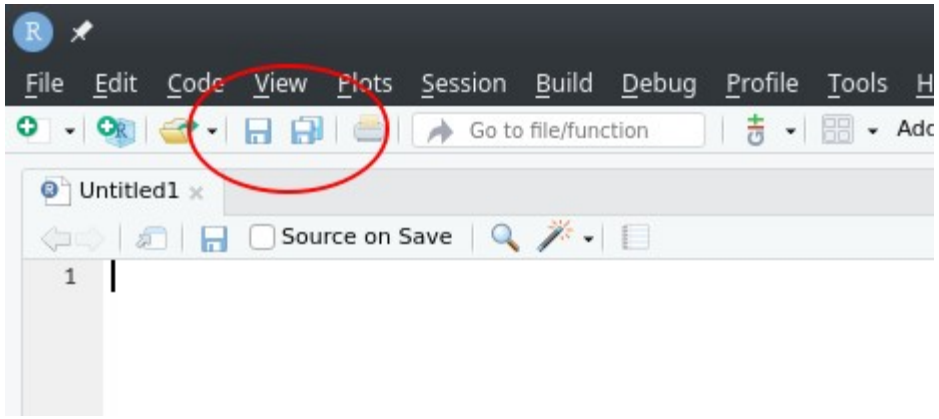
Pagamos nóminas

Organizamos turnos

Las empresas en su día a día manejan multitud de información relativa a su actividad.

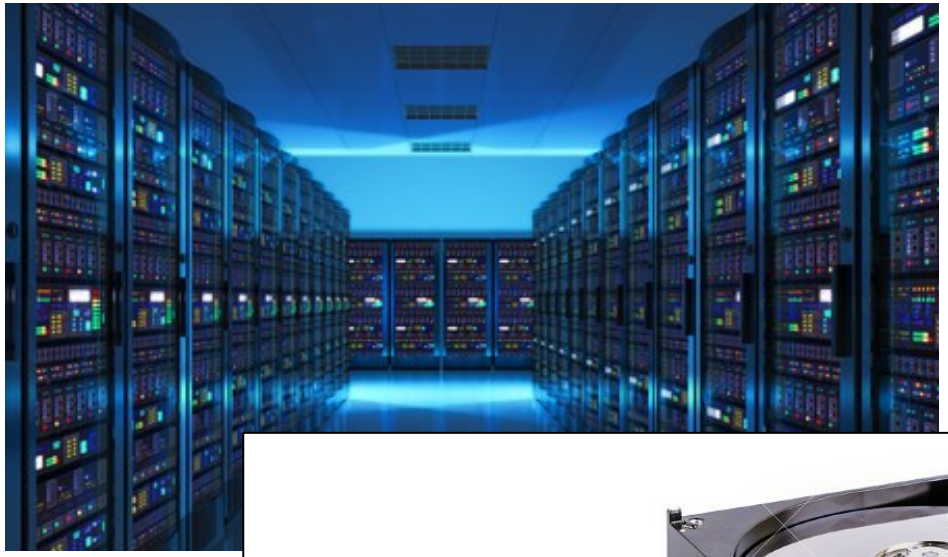


Nuestros sistemas actuales son herencia de la evolución de estos desde sus inicios.

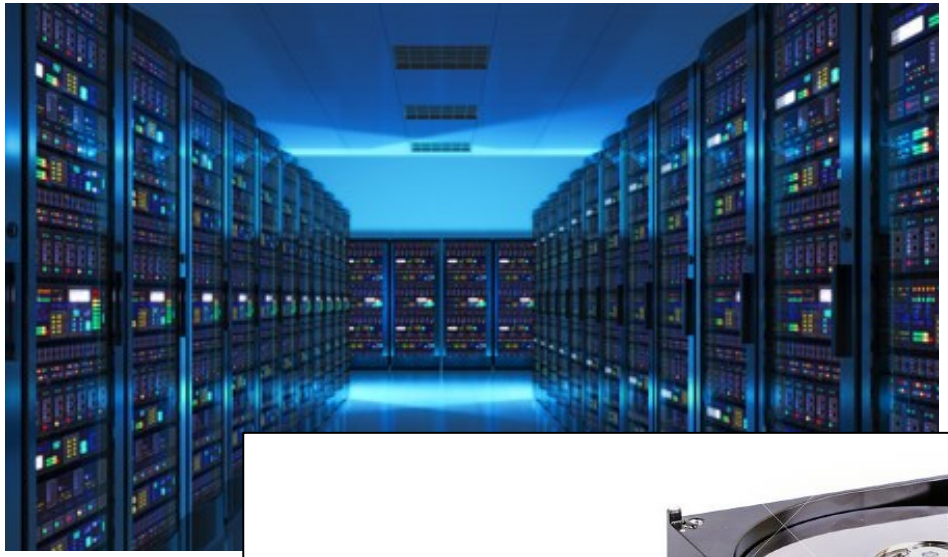


Nuestros sistemas actuales son herencia de la evolución de estos desde sus inicios.

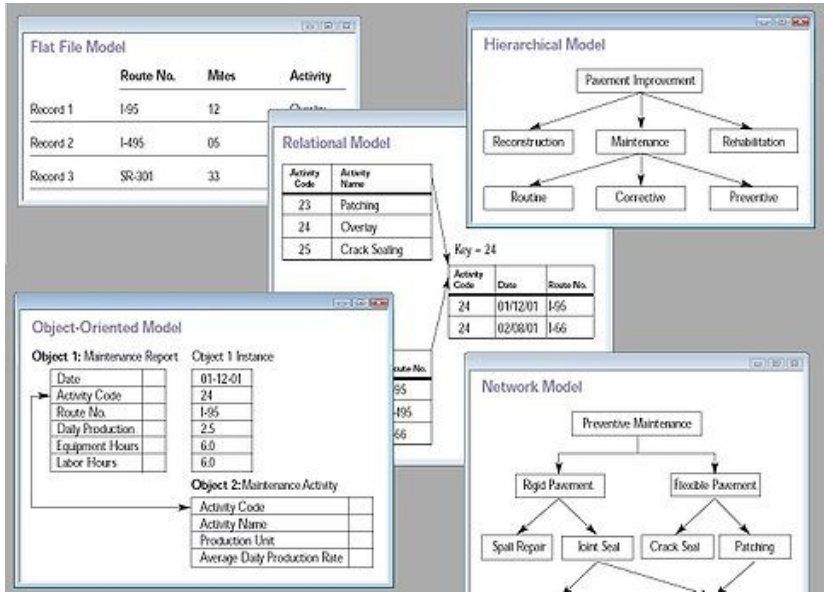
En sus inicios los sistemas eran muy básicos, de bajo nivel, operando sobre los recursos más crudos en las máquinas.

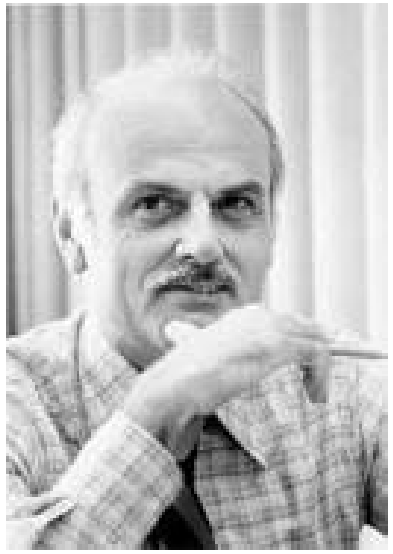


En sus inicios los sistemas eran muy básicos, de bajo nivel, operando sobre los recursos más crudos en las máquinas.



Se buscaban abstracciones que permitieran un acceso y operación más sencilla a la información existente.





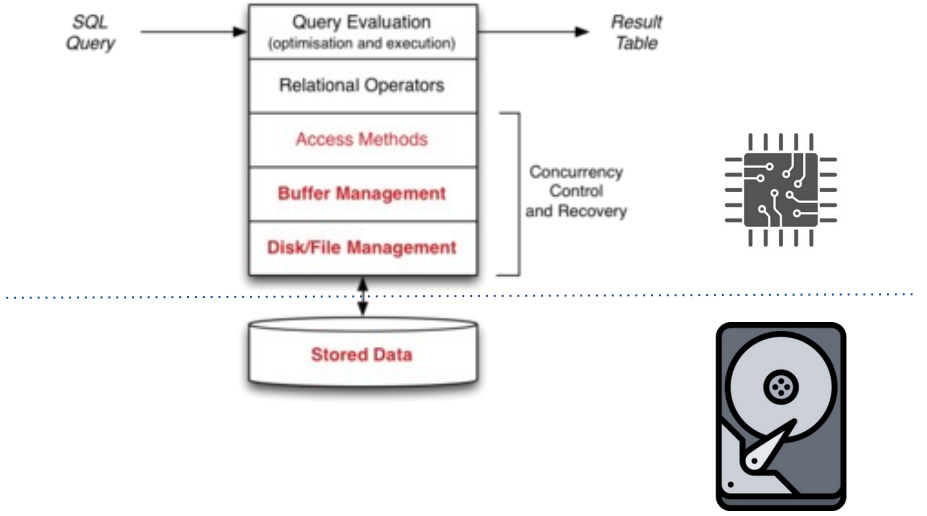
Edgar F. Codd (1923 - 2003)

[...] Publicó *Un modelo relacional de datos para grandes bancos de datos compartidos* (título original: *A Relational Model of Data for Large Shared Data Banks*) en 1970.

[...] Larry Ellison diseñó la base de datos Oracle basándose en las ideas de Codd.

Base de datos: Conjunto de datos estructurados pertenecientes a un contexto.

Sistema gestor de base de datos: Software que gestiona y opera una base de datos.

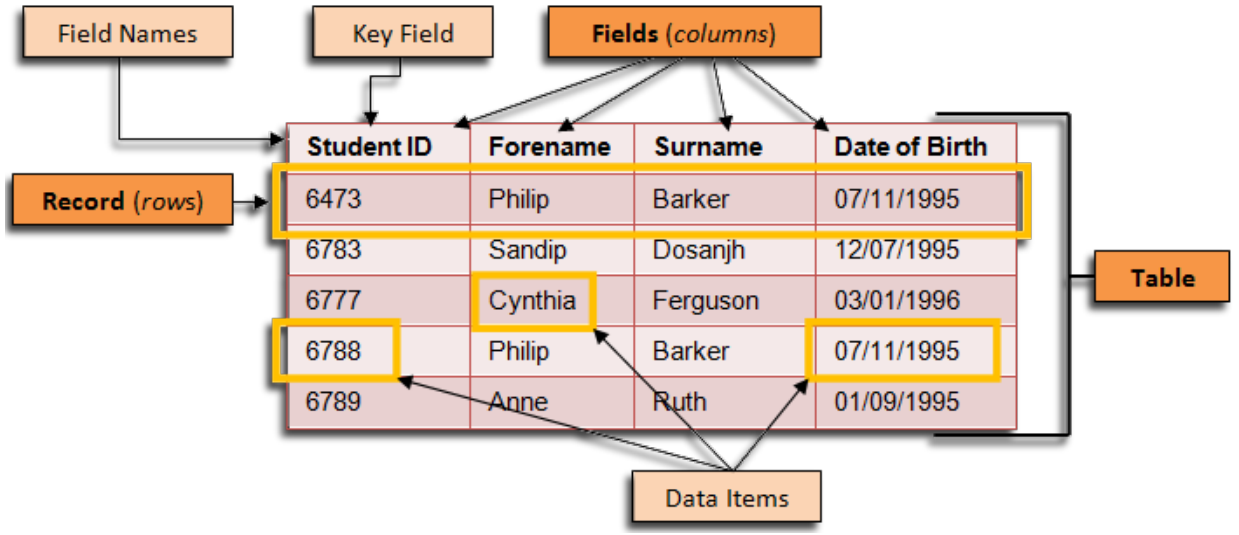


Sistema gestor de base de datos relacional (RDBMS)

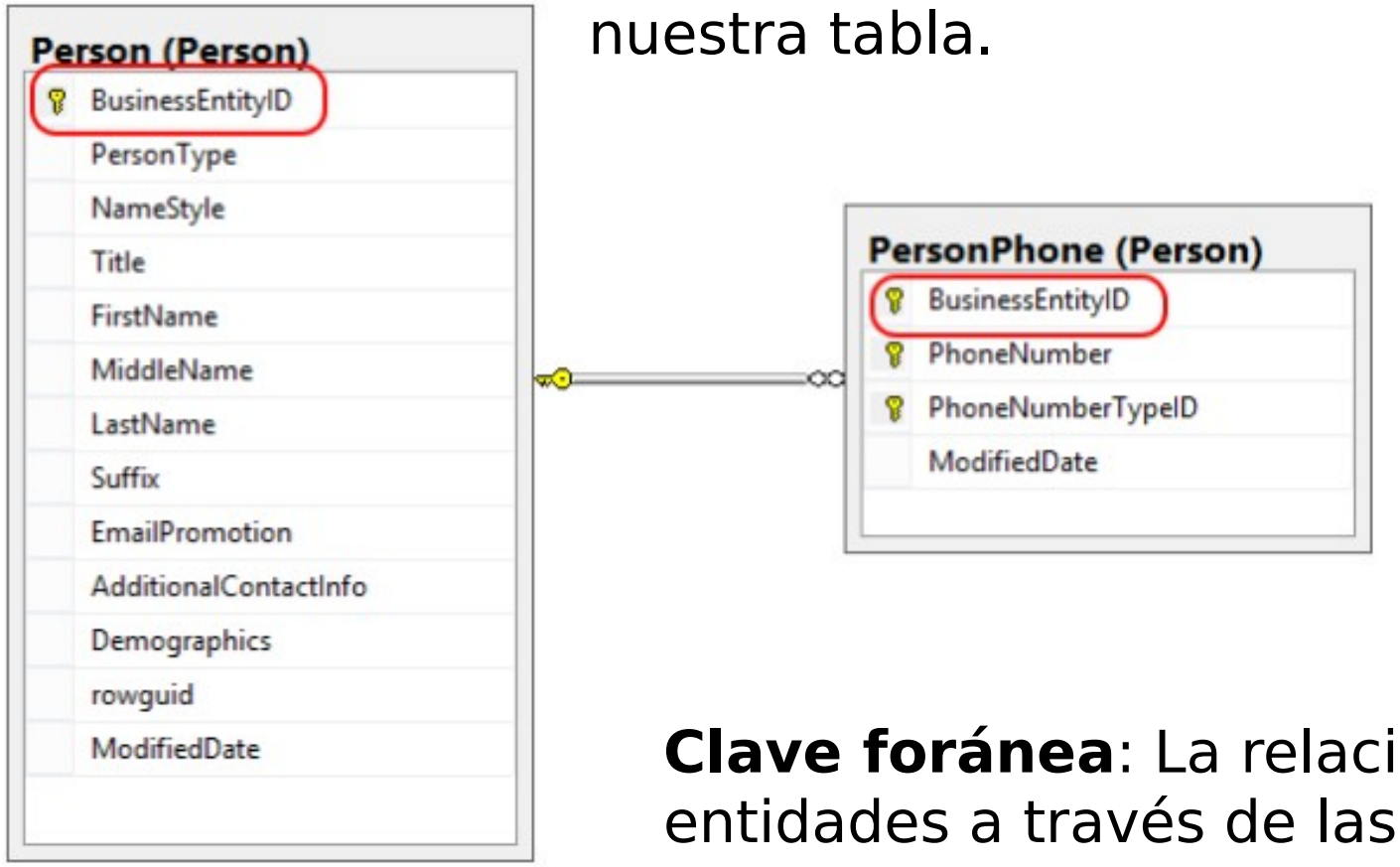
La unidad base es la **tabla** (relation)

Se dividen en **columnas**, llamadas **atributos o campos**.

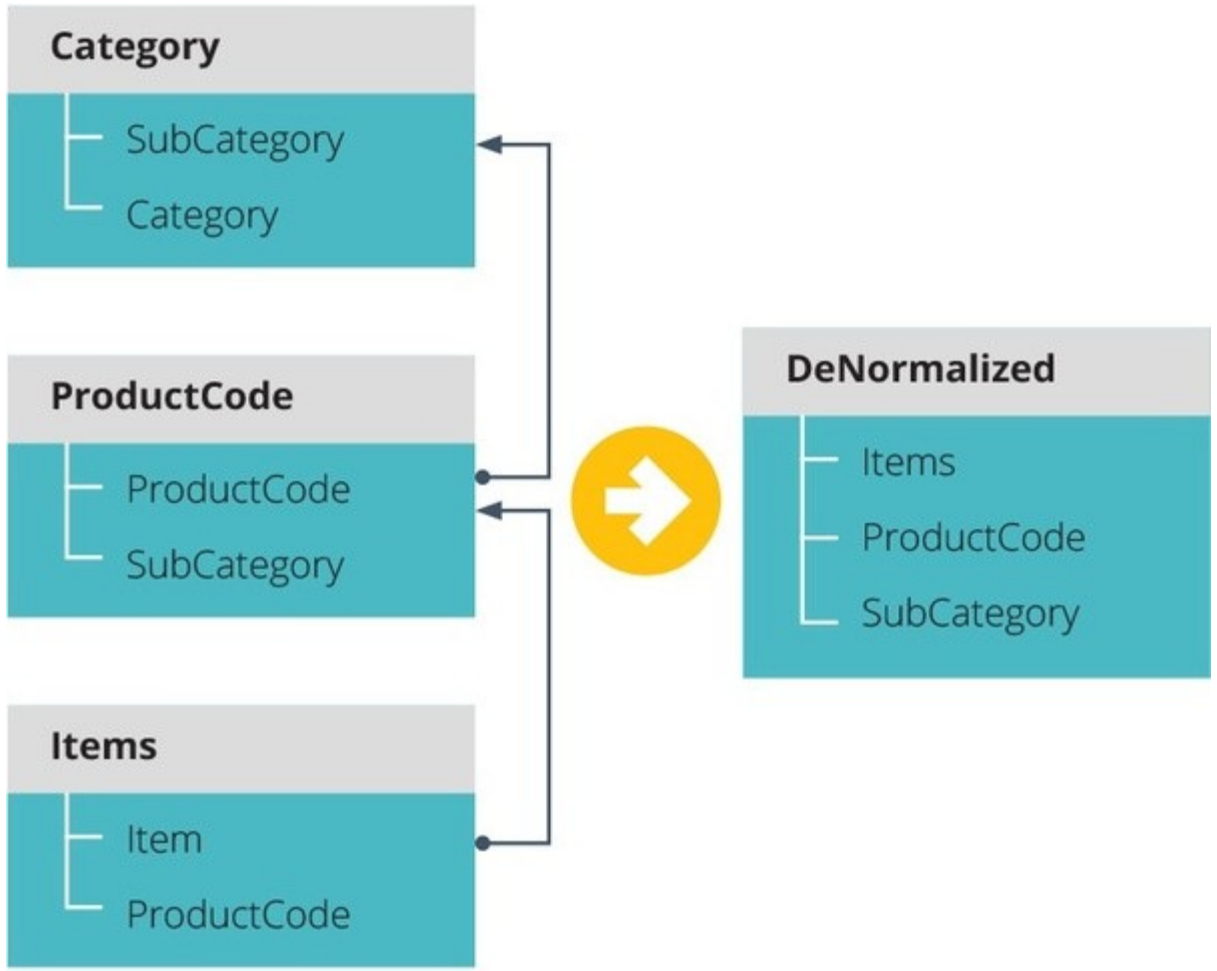
Y en filas de datos acorde a esas columnas, también llamadas **tuplas**.



Clave primaria: Identifica unívocamente cada fila de nuestra tabla.



Clave foránea: La relación existente entre distintas entidades a través de las claves primarias de estas.



Además del formato lógico de los sistemas, se definieron ciertas prácticas de cómo la información debería estar almacenada para evitar inconsistencias y duplicidades de la información.

Esto dio como origen a las formas normales en las que nos encontramos la información y los sistemas tradicionales.

Primera forma normal: No pueden existir campos compuestos o multivalor

Segunda forma normal: No pueden existir dependencias parciales

Tercera forma normal: No pueden existir dependencias transitivas (A implica B y B implica C)

Esto permite resolver las distintas formas en las que la información se referencia, manteniendo la consistencia de los datos y su rendimiento en general.

1NF

Students		
FName	**SName**	**Class**
Timothy	Smith	Computer Science
Jessica	Green	Computer Science
Jessica	Green	Maths
Mark	Lynch	Maths

2NF

Students			
ID	**FName**	**SName**	**Class**
1	Timothy	Smith	1
2	Jessica	Green	1
3	Jessica	Green	2
4	Mark	Lynch	2

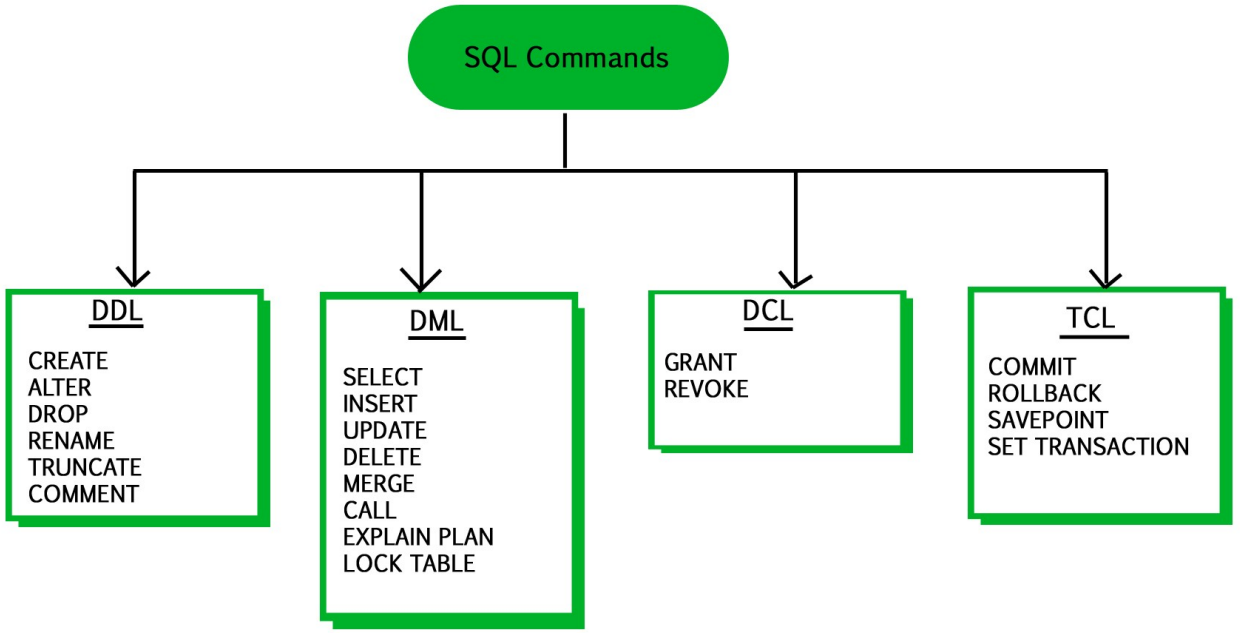
Class	
ID	**Class**
1	Computer Science
2	Maths

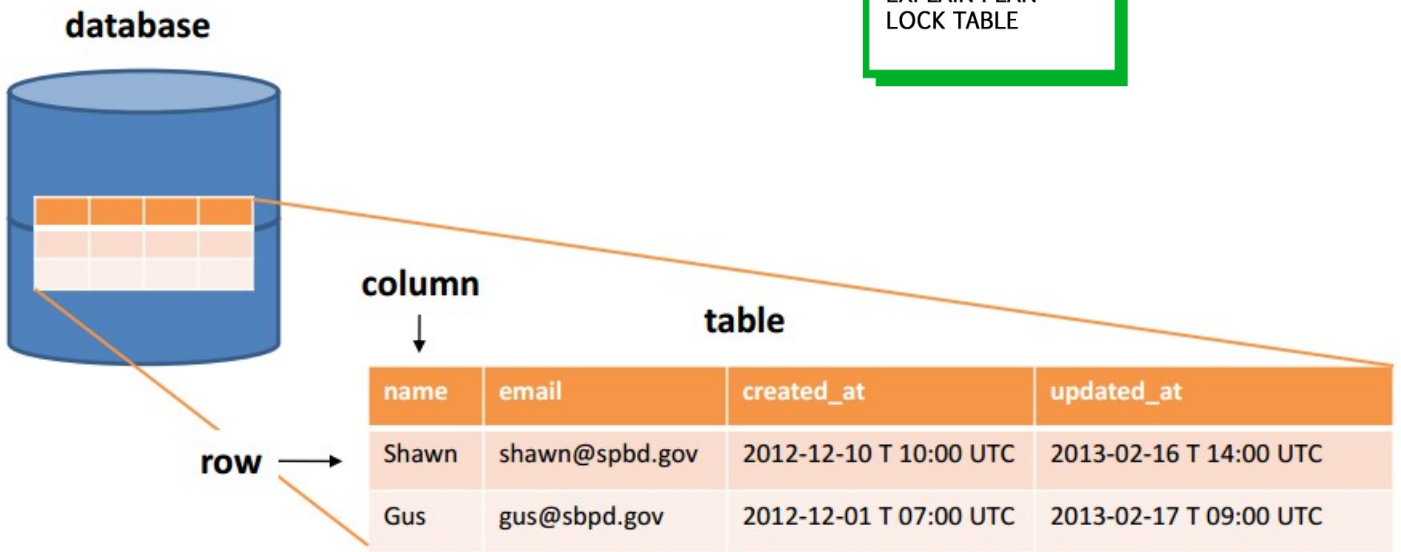
3NF

Students		
ID	**FName**	**SName**
1	Timothy	Smith
2	Jessica	Green
3	Mark	Lynch

Subject	
ID	**Class**
1	Computer Science
2	Maths

Class		
ID	**Student**	**Class**
1	1	1
2	2	1
3	2	2
4	3	2


SQL Commands
DDL
CREATE
ALTER
DROP
RENAME
TRUNCATE
COMMENT
DML
SELECT
INSERT
UPDATE
DELETE
MERGE
CALL
EXPLAIN PLAN
LOCK TABLE
DCL
GRANT
REVOKE
TCL
COMMIT
ROLLBACK
SAVEPOINT
SET TRANSACTION


database
column
table
row
name
email
created_at
updated_at
Shawn
shawn@spbd.gov
2012-12-10 T 10:00 UTC
2013-02-16 T 14:00 UTC
Gus
gus@sbpd.gov
2012-12-01 T 07:00 UTC
2013-02-17 T 09:00 UTC


Lower Costs
Reduced Risk
Improve Customer Satisfaction
Improve Quality
Enhance Process Consistency
Maintain Security
Simplify Operations
Business Process Management

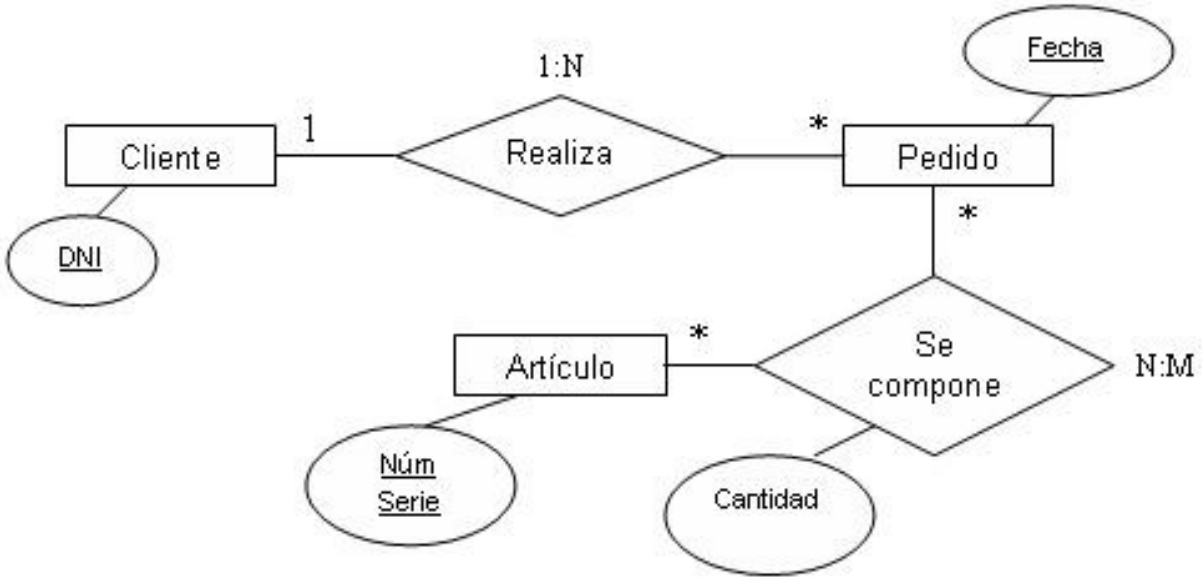
Nos solemos referir a los comandos del *Structured Query Language* (SQL) como queries. Las opciones más comunes son la definición (*Data Definition Language*):

- **CREATE**: Creación de un elemento con un nombre dado.
- **ALTER**: Alterar su estructura
- **TRUNCATE**: Vaciar su contenido
- **DROP**: Eliminar por completo

Y la manipulación (*Data Manipulation Language*):

- **SELECT**: Obtener un subconjunto de los datos
- **INSERT**: Insertar nuevos datos
- **UPDATE**: Actualizar campos concretos
- **DELETE**: Eliminar tuplas

Iniciaremos el diseño de nuestra base de datos con un modelo **entidad/relación**.



Este primer trabajo permite identificar la **cardinalidad** y cómo deberá traducirse el **esquema de datos** para no generar problemas de rendimiento en el normalizado de los datos.

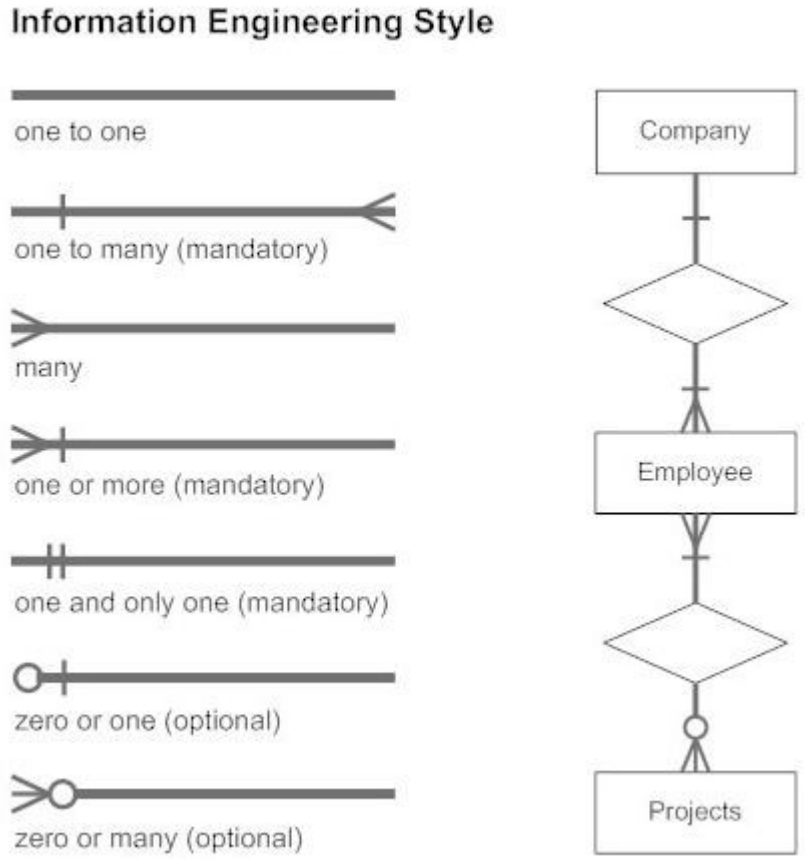
JOIN!

Se emplea una sintaxis estándar para indicar esta relacionalidad.

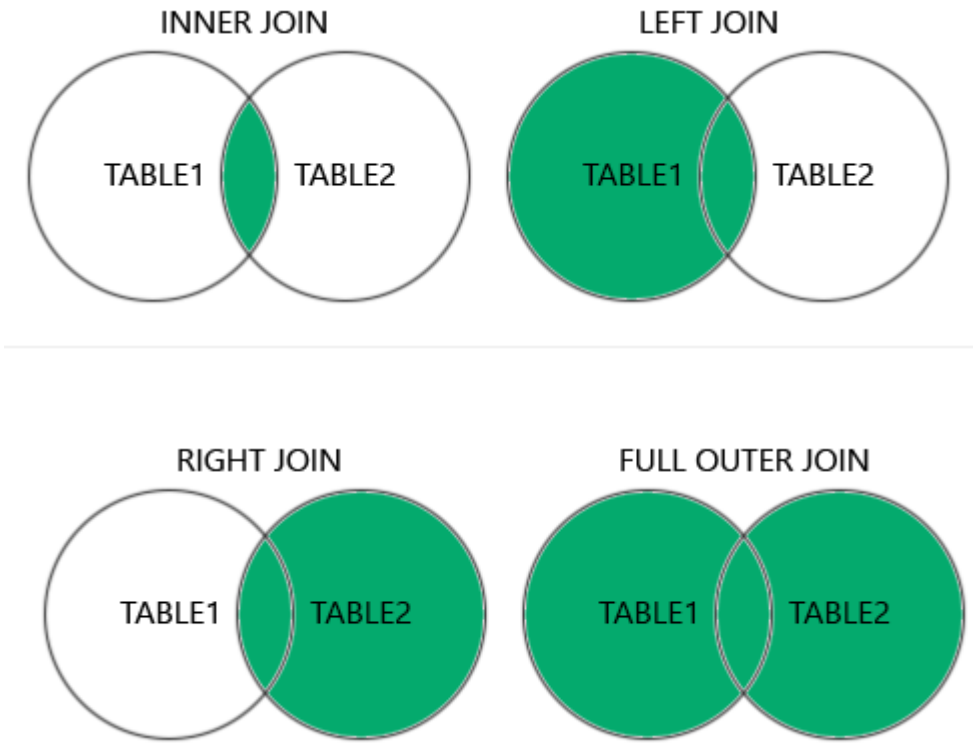
Product Name	Supplier ID
Planet Oat Oatmilk	1
Honey Nut Frosted Flakes	2
Magnum Double Tub	5
Sour Patch Marshmallows	3
Ferrero Eggs	4

Supplier ID	Supplier Name
1	John
2	Anne
3	Robert
4	Jerry
5	Tim

Product Name	Supplier Name
Planet Oat Oatmilk	John
Honey Nut Frosted Flakes	Anne
Sour Patch Marshmallows	Robert
Ferrero Eggs	Jerry
Magnum Double Tub	Tim

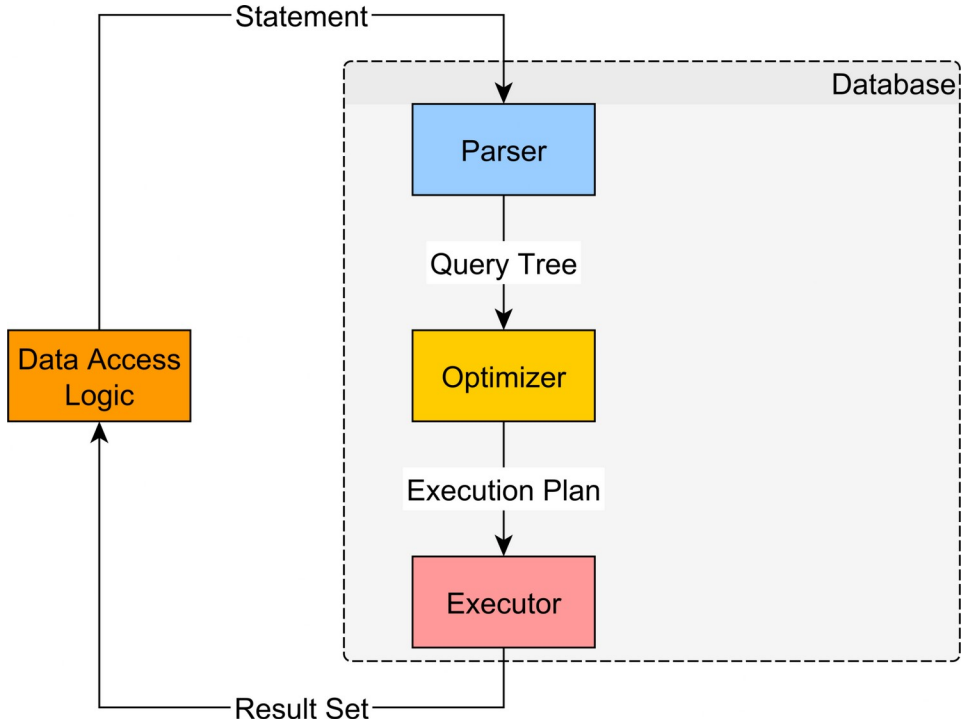


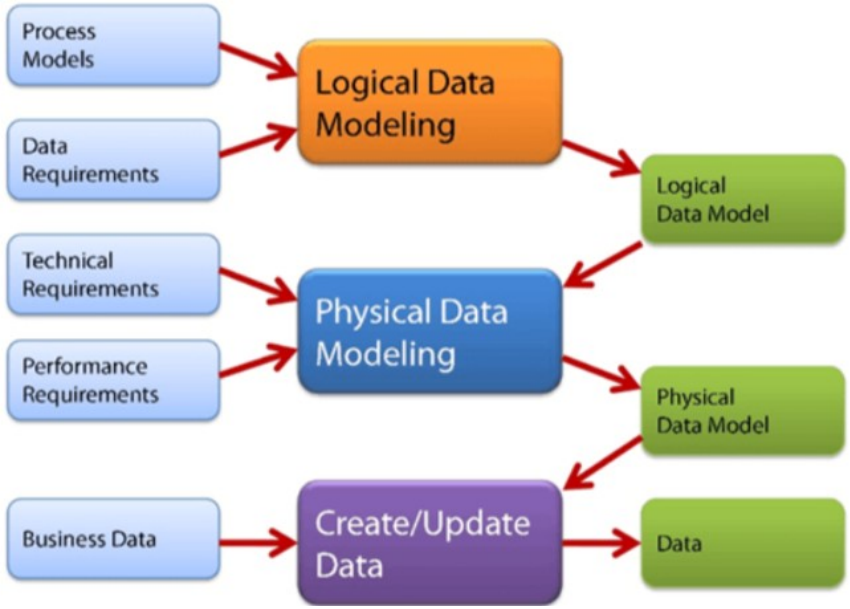
JOIN!



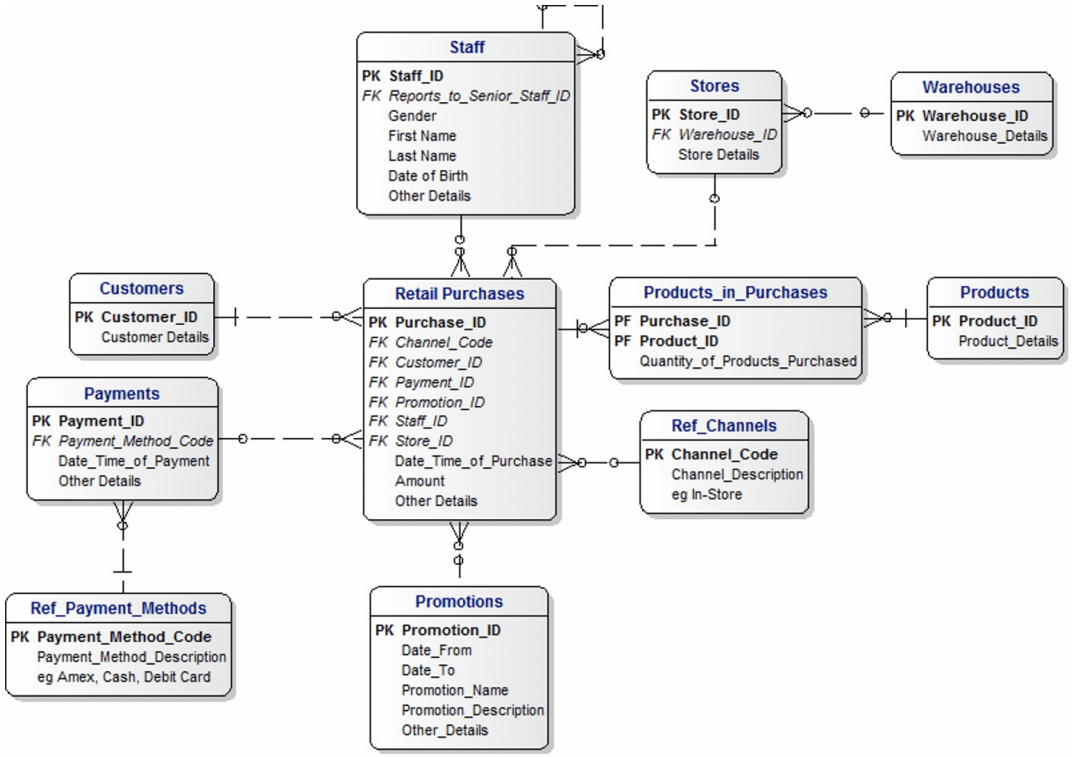
https://www.w3schools.com/sql/sql_join.asp

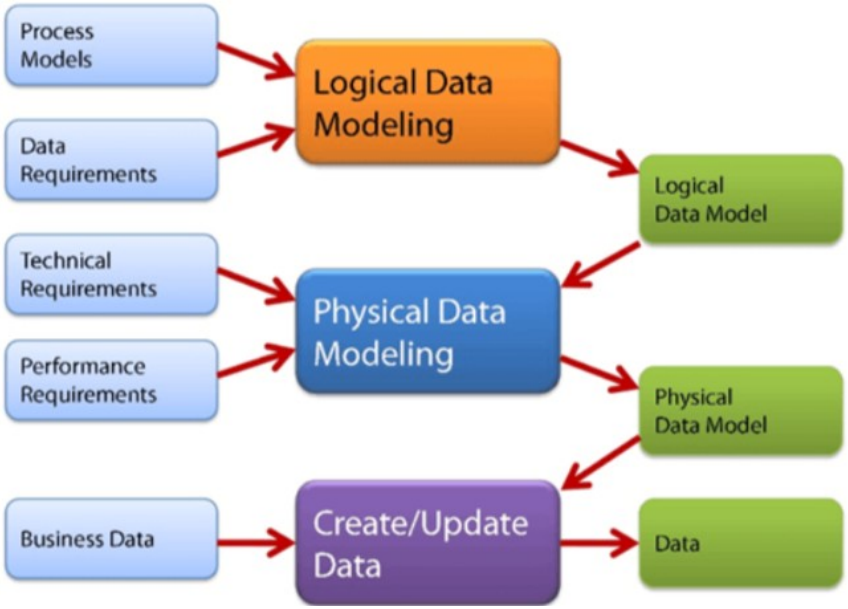
```
Select student.name, COUNT(class.class)
from student
        join class on ...
        join subject on ...
where subject.name = "Math"
group by student.name
```



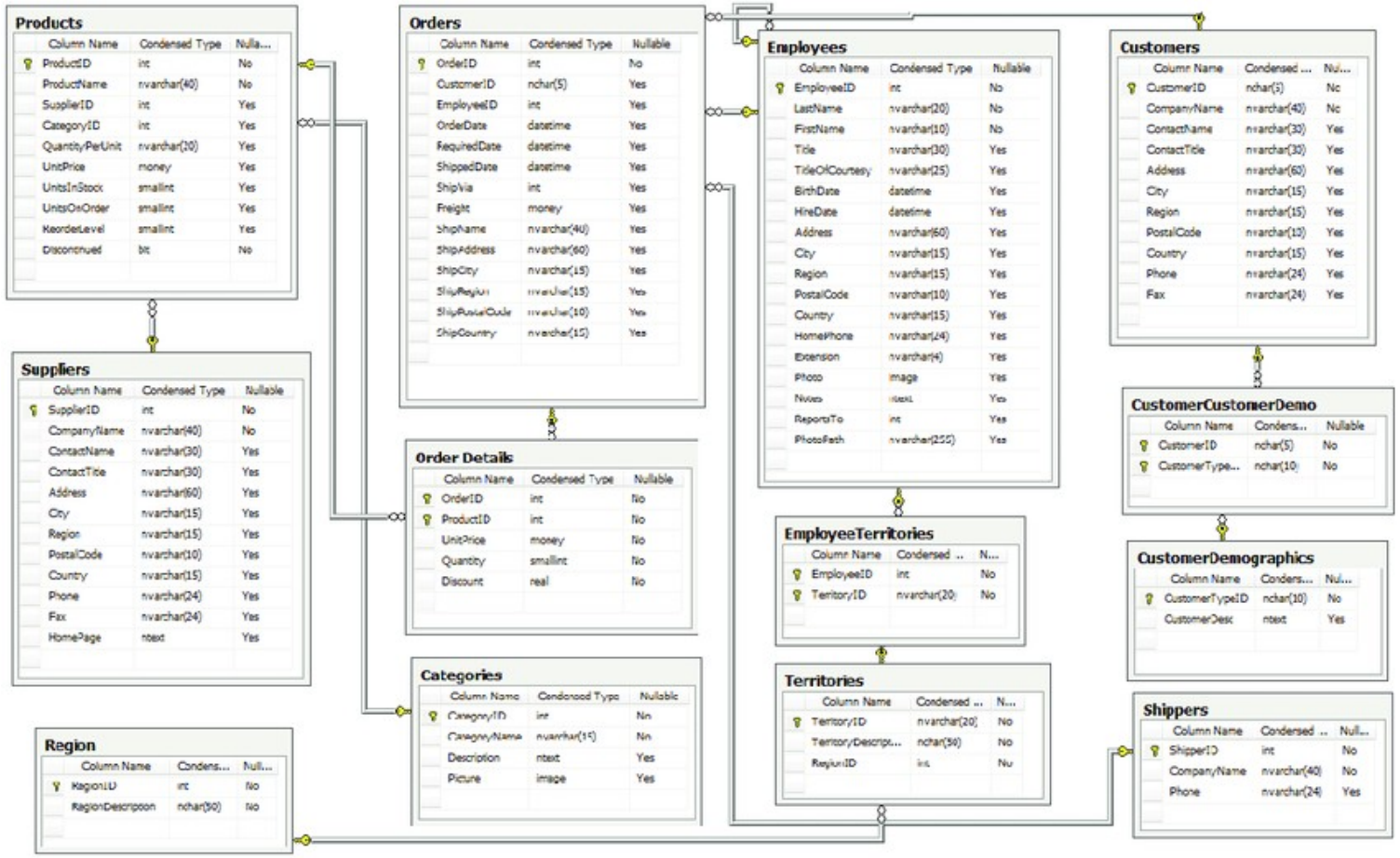


El **modelo lógico**, identifica una estructura base de cómo se deberán implementar las tablas en base a la relacionalidad anteriormente vista.





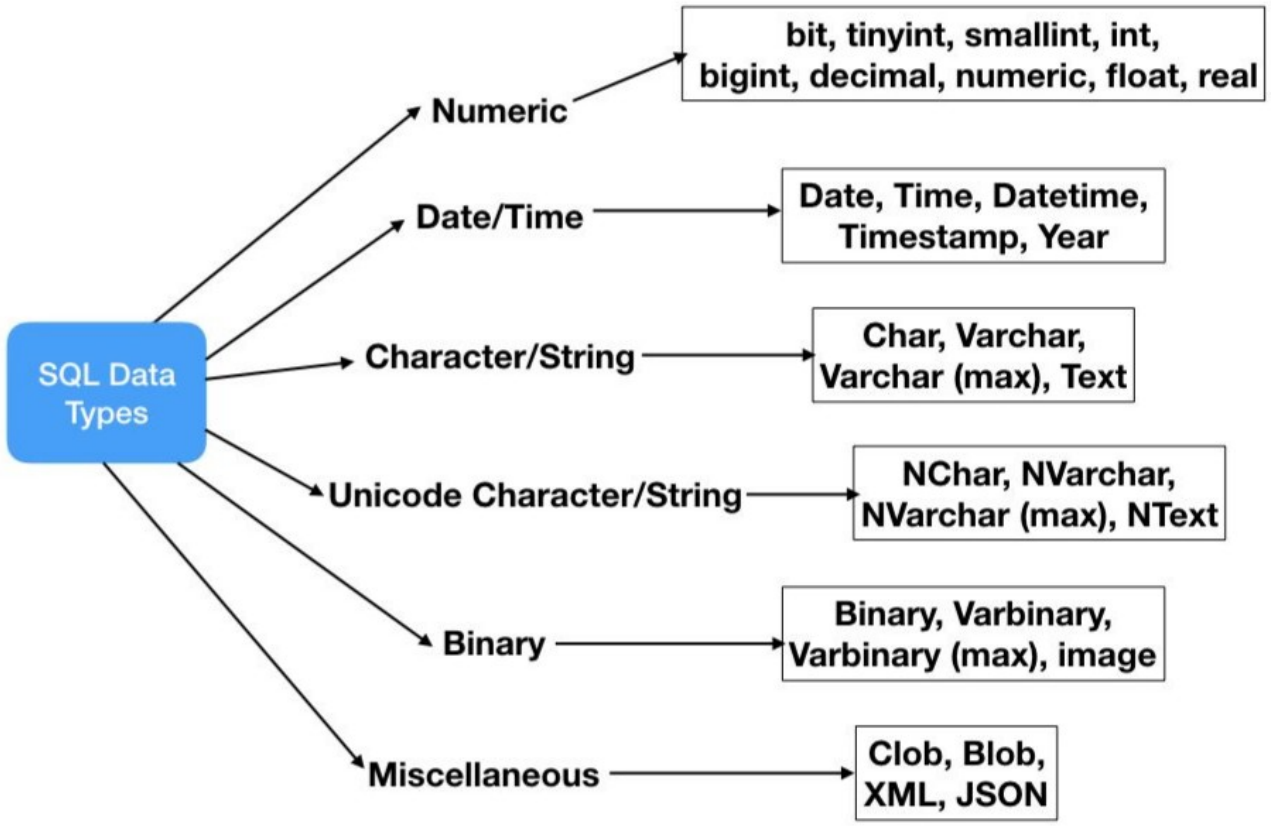
El **modelo físico** implementa los tipos de datos necesario y expresa el código que construirá el modelo objetivo.



Solo ahora podemos empezar a usar nuestro sistema y generar datos en él.

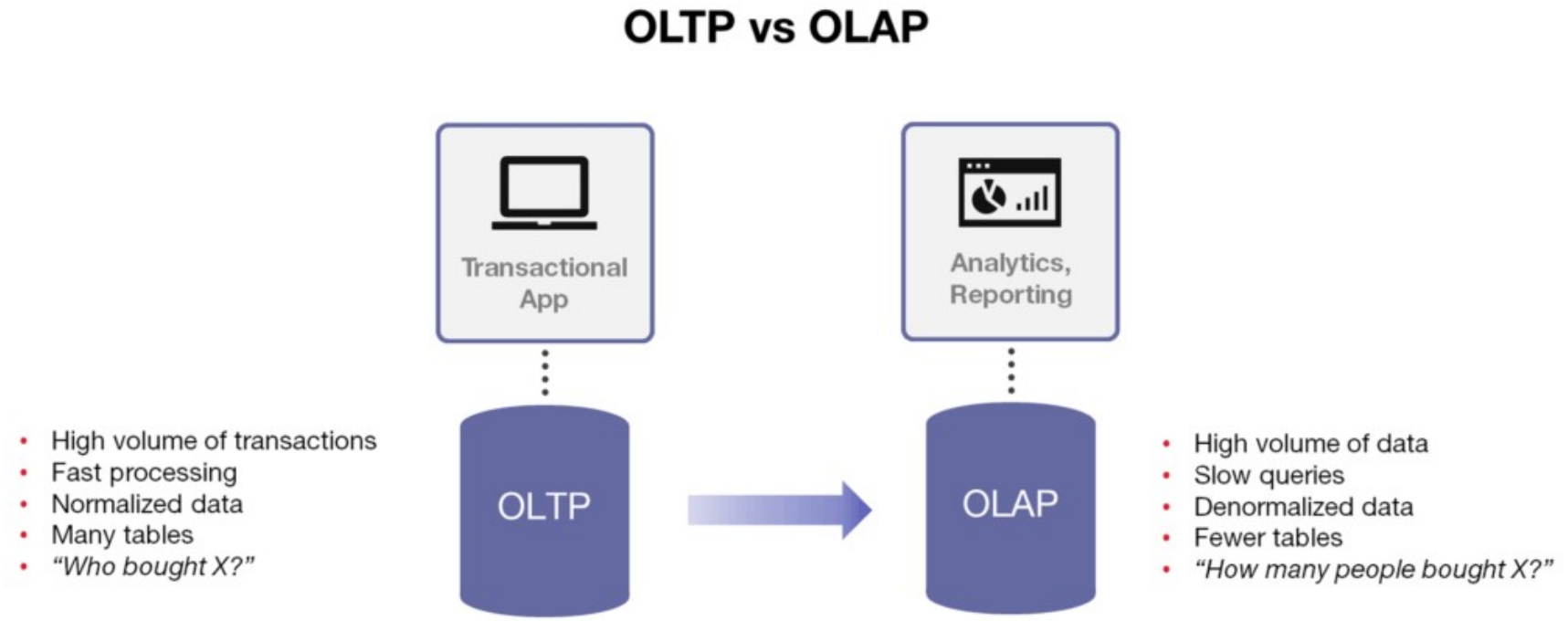
Deberemos identificar el tipo de dato en base al sistema gestor de base de datos que empleemos.

- Oracle
- PostgreSQL
- MySQL

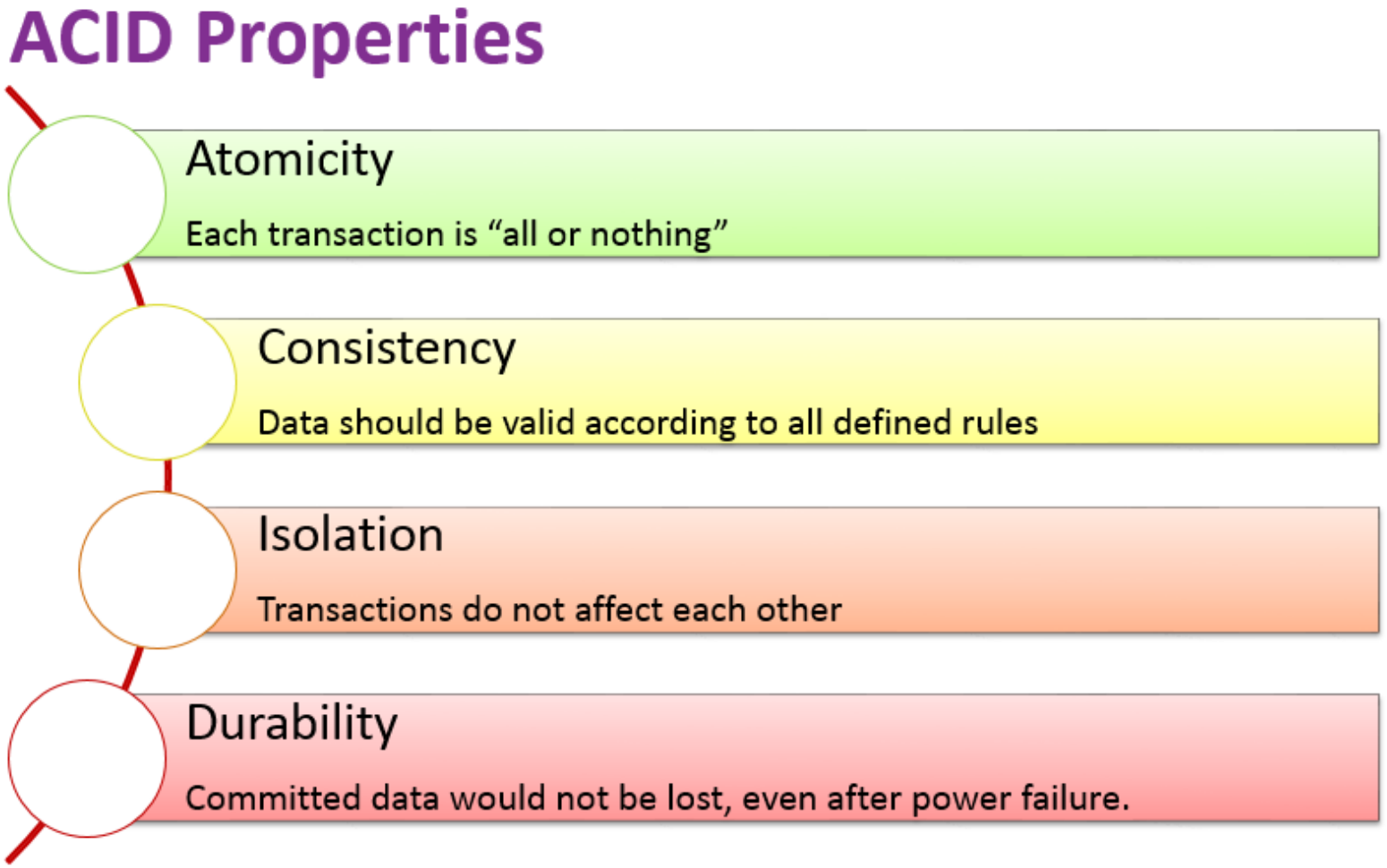


Dividiremos las bases de datos relacionales (RDBMS) en dos tipologías:

- **Operacionales** (transaccionales): Destinadas a actividades clave de nuestro negocio
- **Informacionales** (analíticas): Destinadas a realizar nuestras tareas de análisis

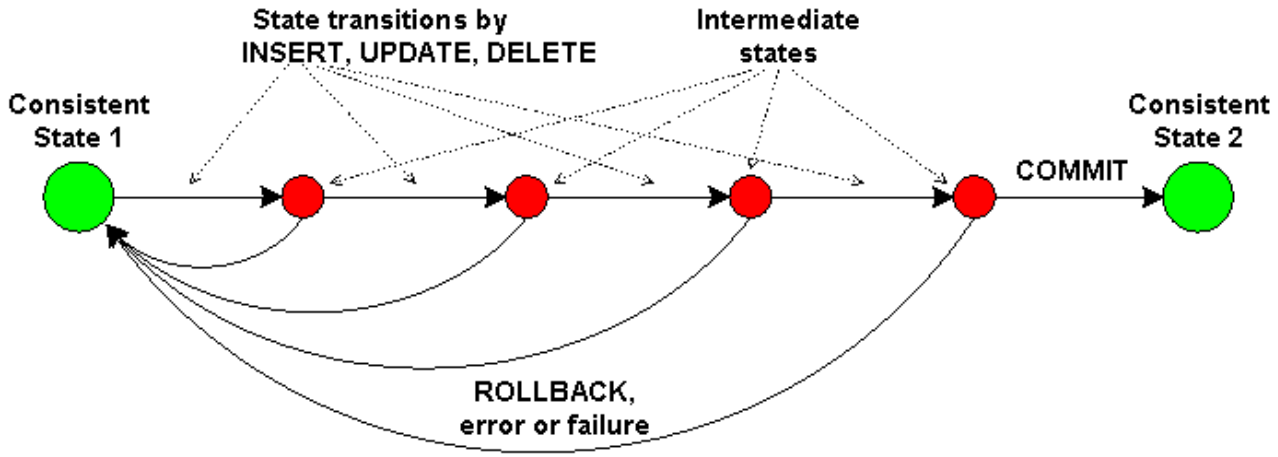


Transacciones (operacionales)



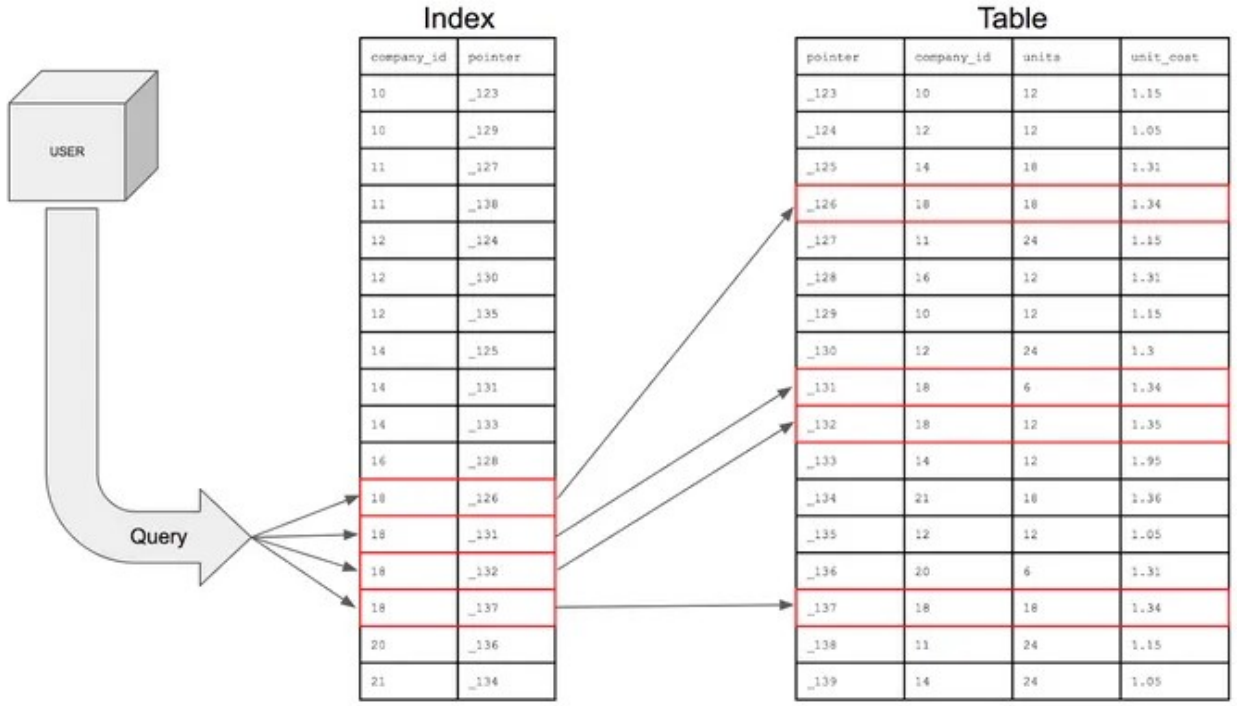
Transacciones (operacionales)

Representan la lógica de operación de la entidad y tienen que ofrecer garantías para asegurar que nuestros negocios funcionen de forma correcta.

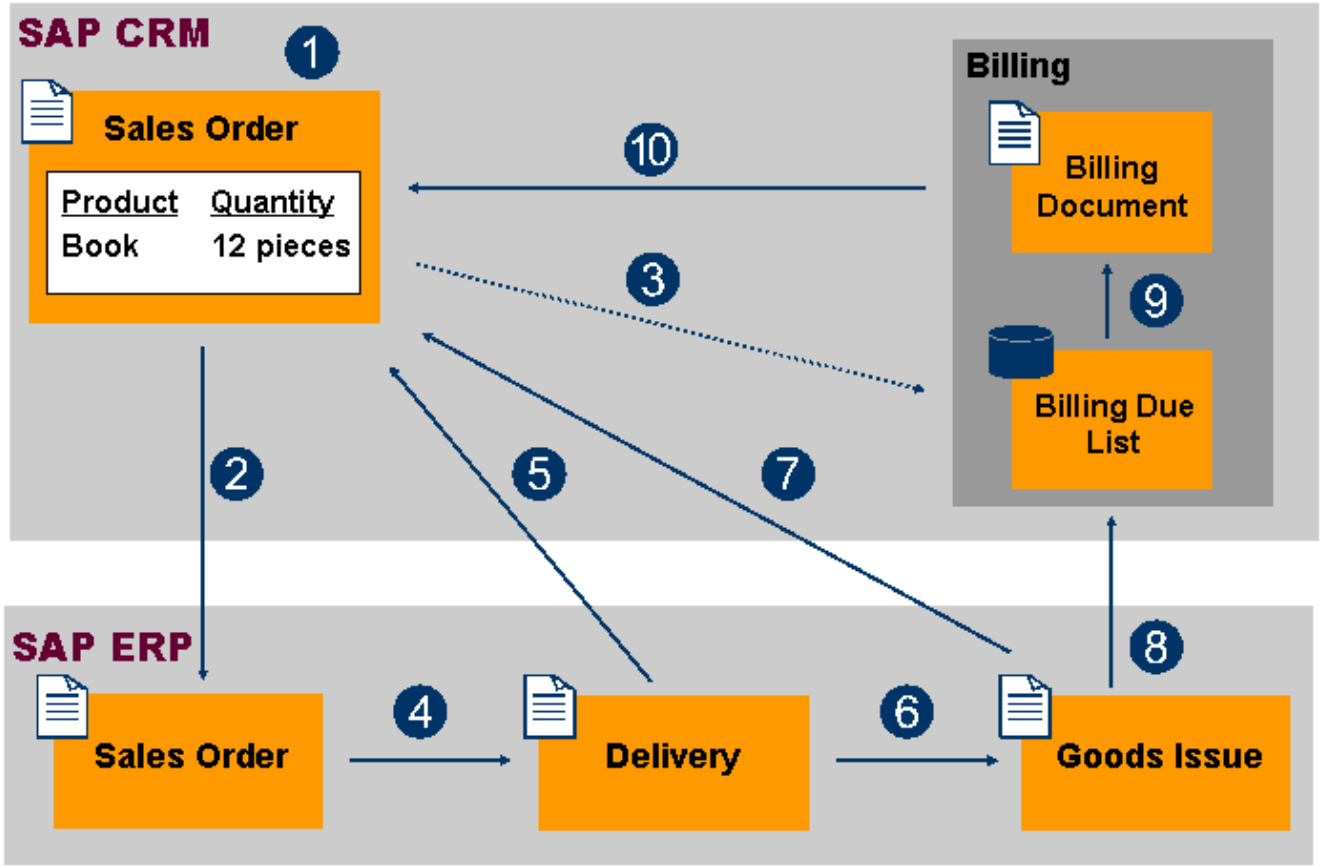


Índices

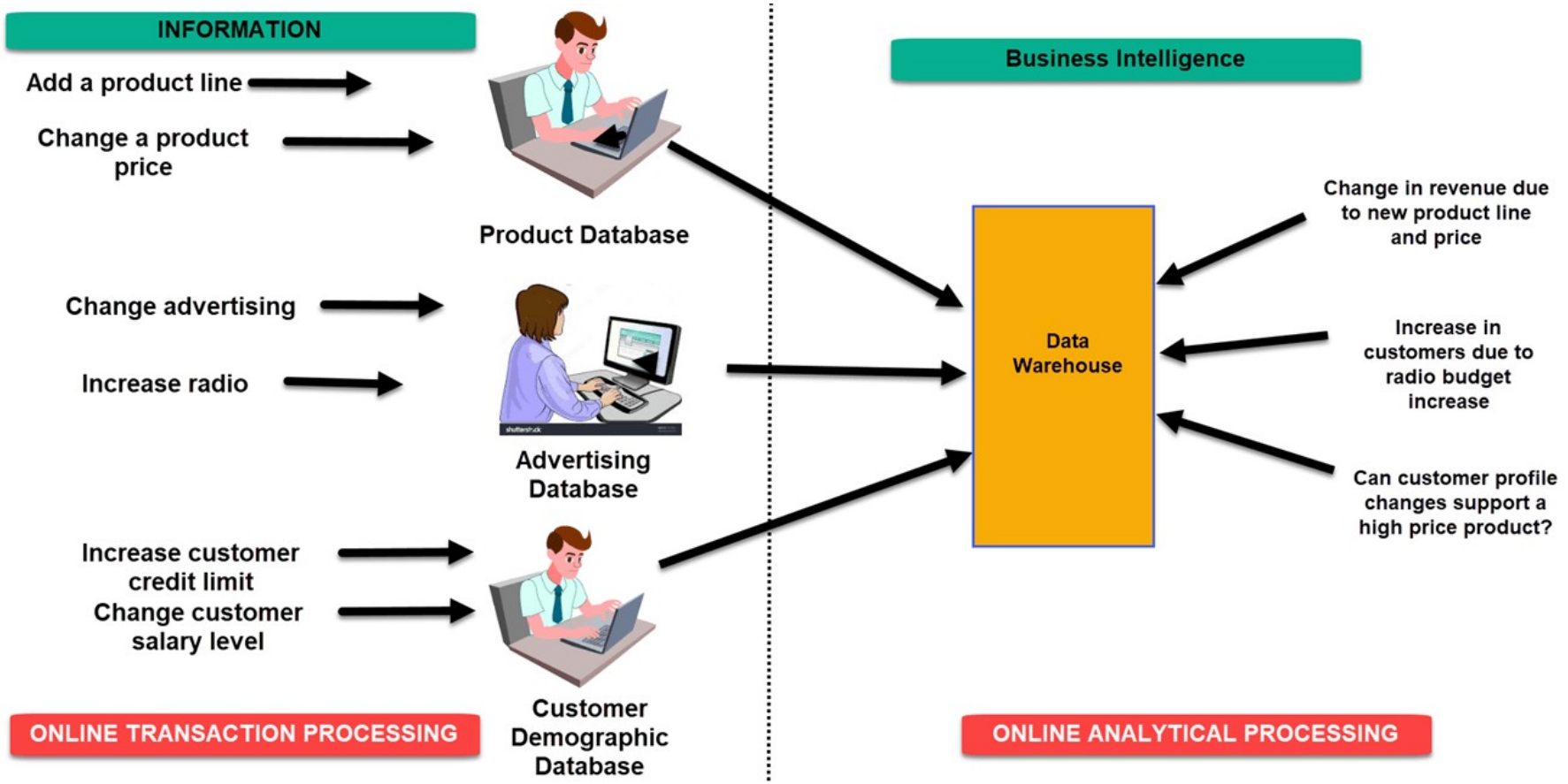
Buscar en tablas puede ser costoso. Los índices nos ayudan a prepararnos algunas de las consultas más habituales.



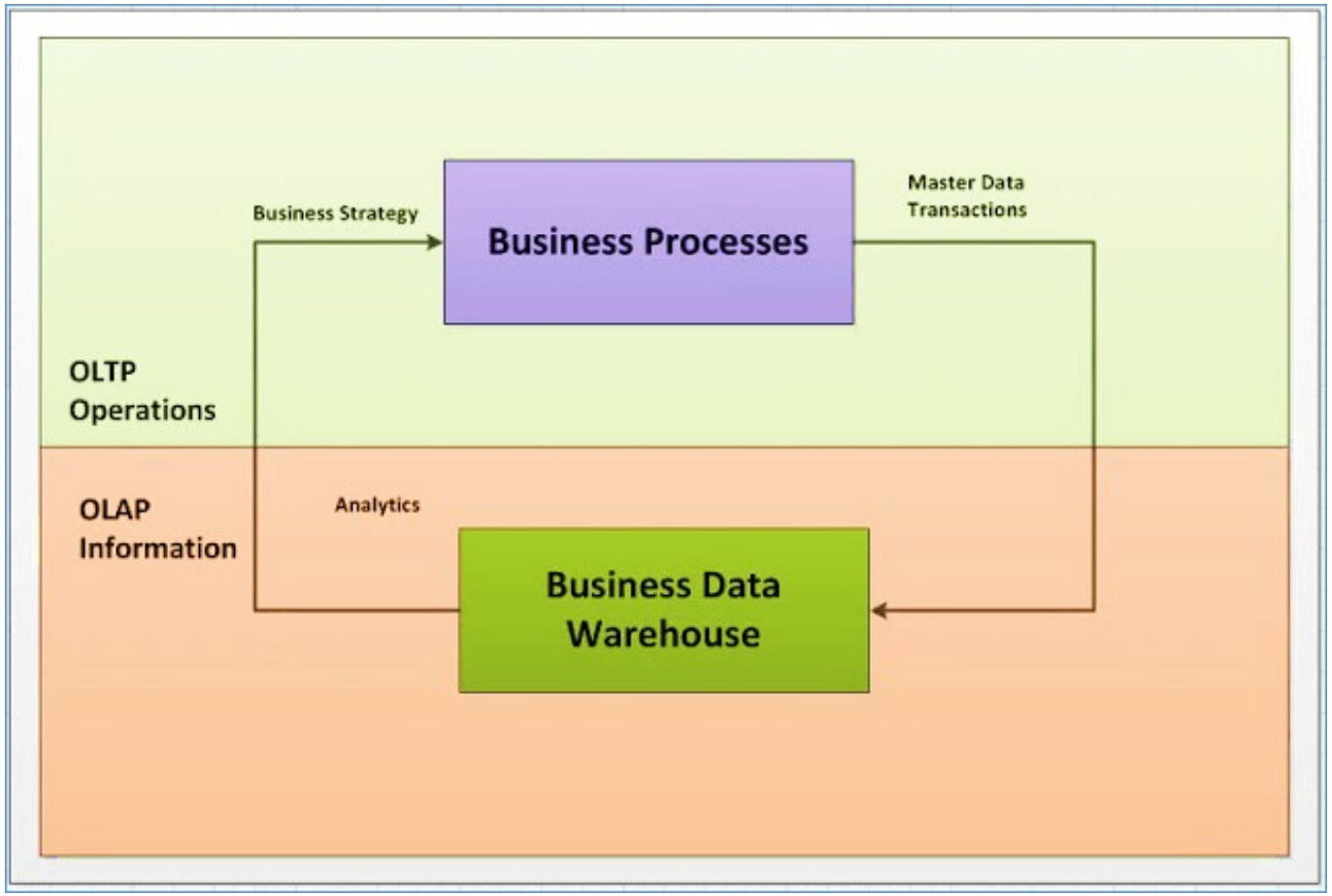
Para no tener que realizarlo desde cero, existen soluciones de mercado ya preparadas.



Realizar operaciones pesadas sobre nuestros sistemas críticos puede poner en riesgo el buen funcionamiento de nuestra empresa.



Realizar operaciones pesadas sobre nuestros sistemas críticos puede poner en riesgo el buen funcionamiento de nuestra empresa.

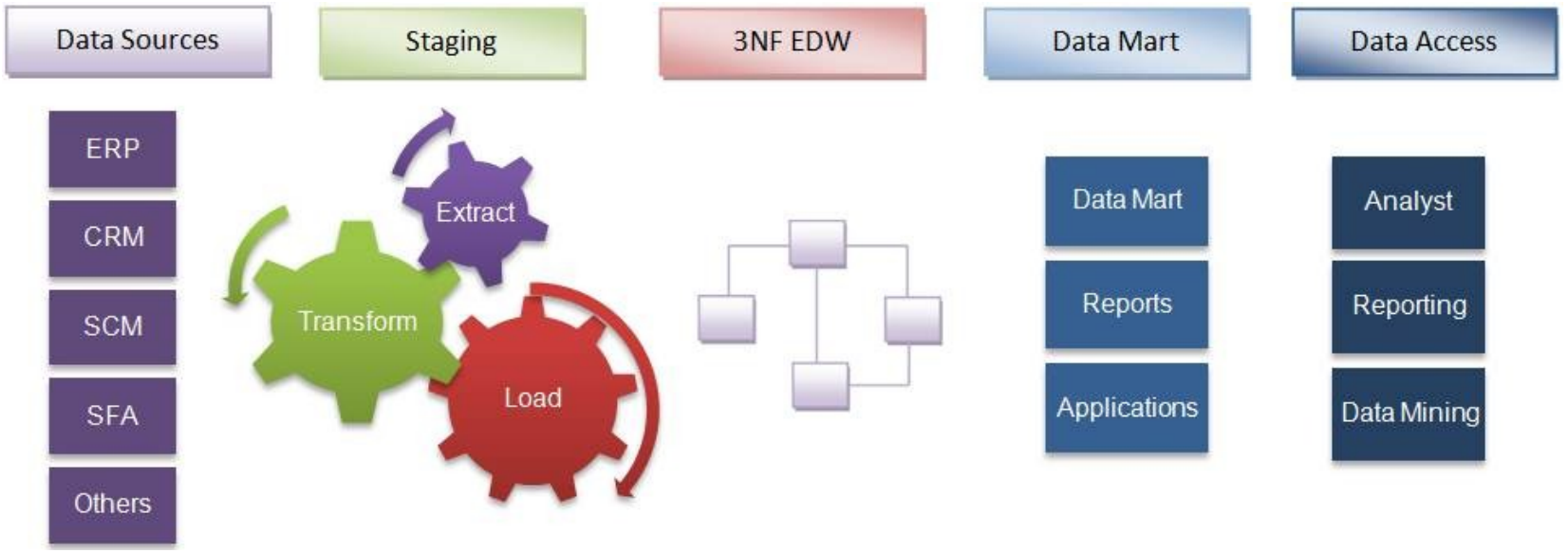


Bill Inmon (1945 -)

[...] el padre del concepto del data warehousing.

[...] creó la definición más aceptada de del data warehouse:

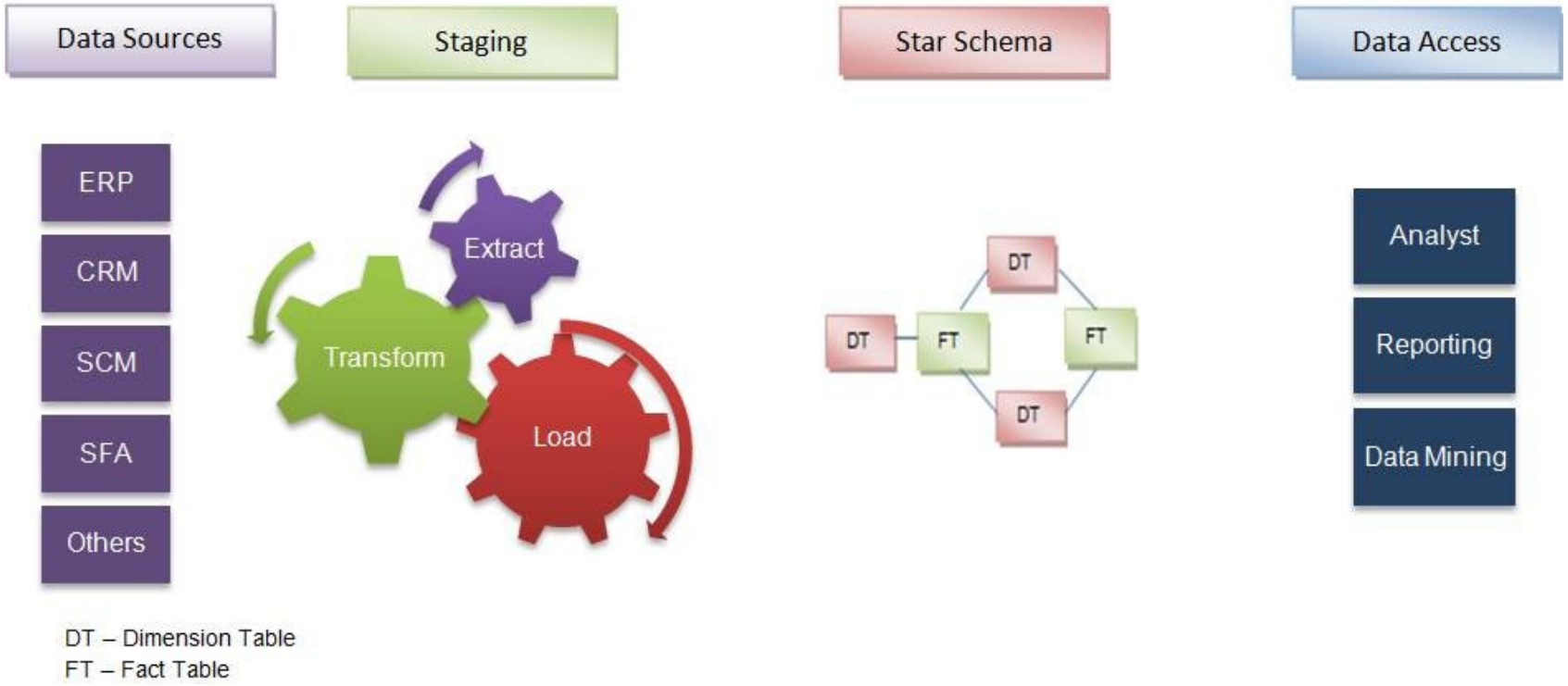
un sistema de soporte a las decisiones unificado, no volátil, cambiante en el tiempo y orientado a aspectos concretos de negocio.



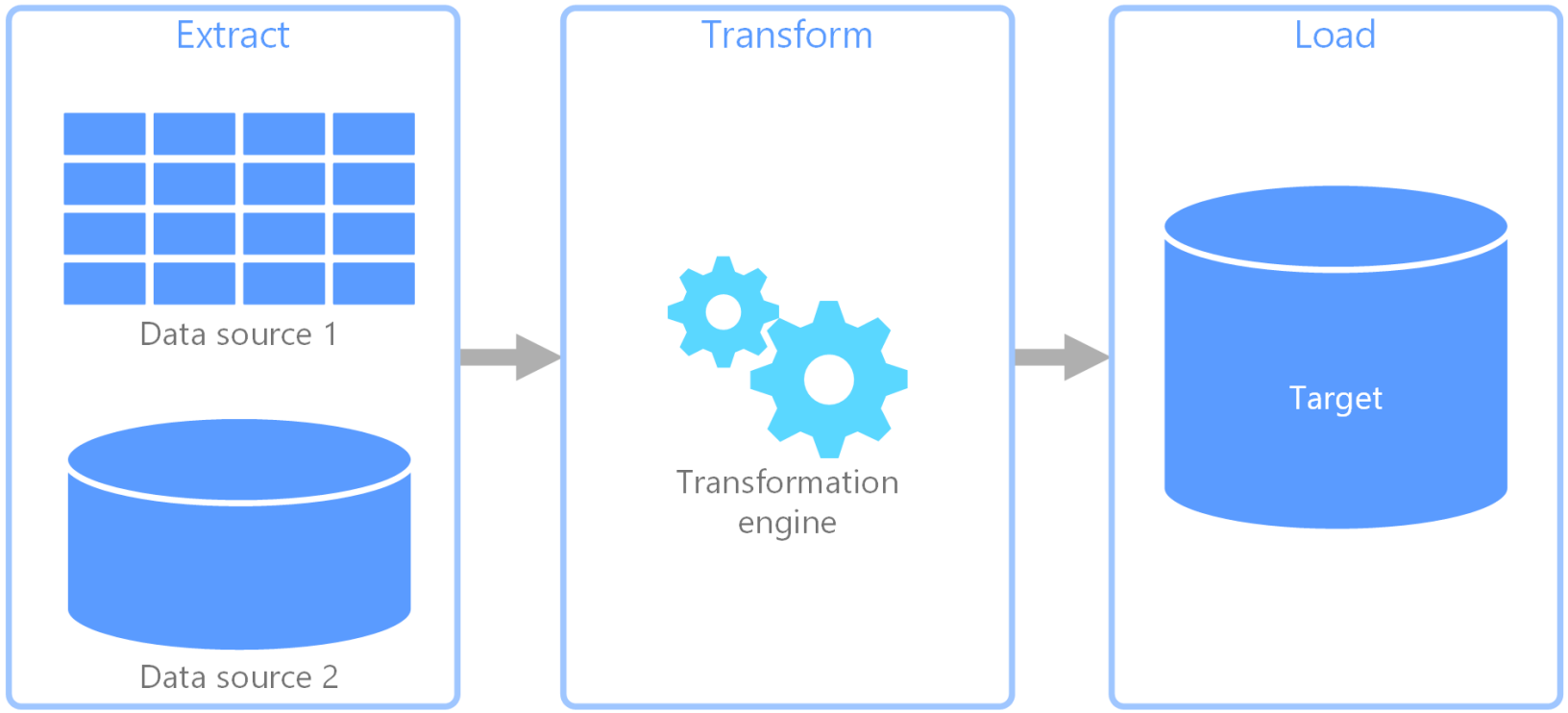
Ralph Kimball

[...] promotor también de conceptos como el data warehousing o el business intelligence.

[...] su metodología, también conocida como modelado dimensional o la metodología Kimball, se ha convertido en el estándar de los sistemas de soporte a las decisiones.

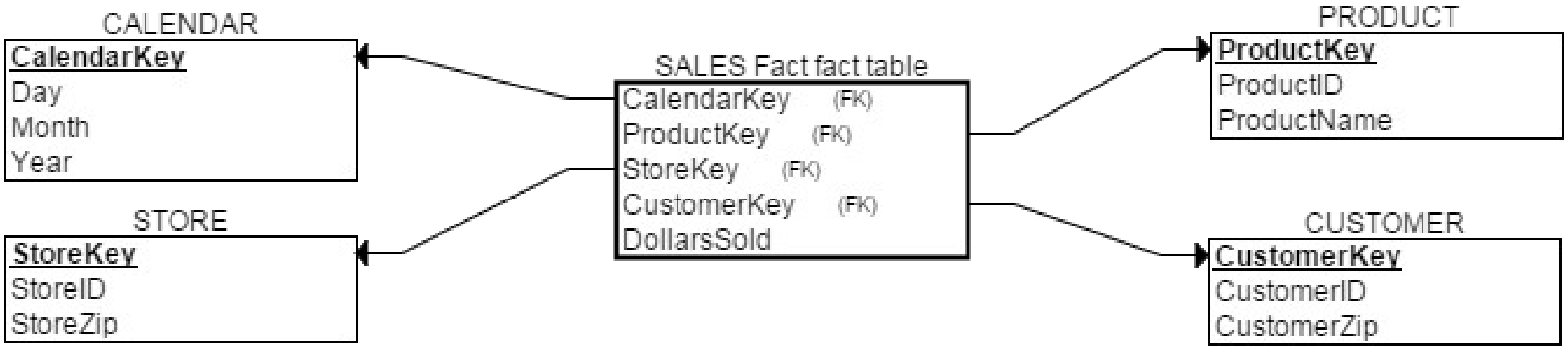


ETL


Extract
Data source 1
Data source 2
Transform
Transformation engine
Load
Target

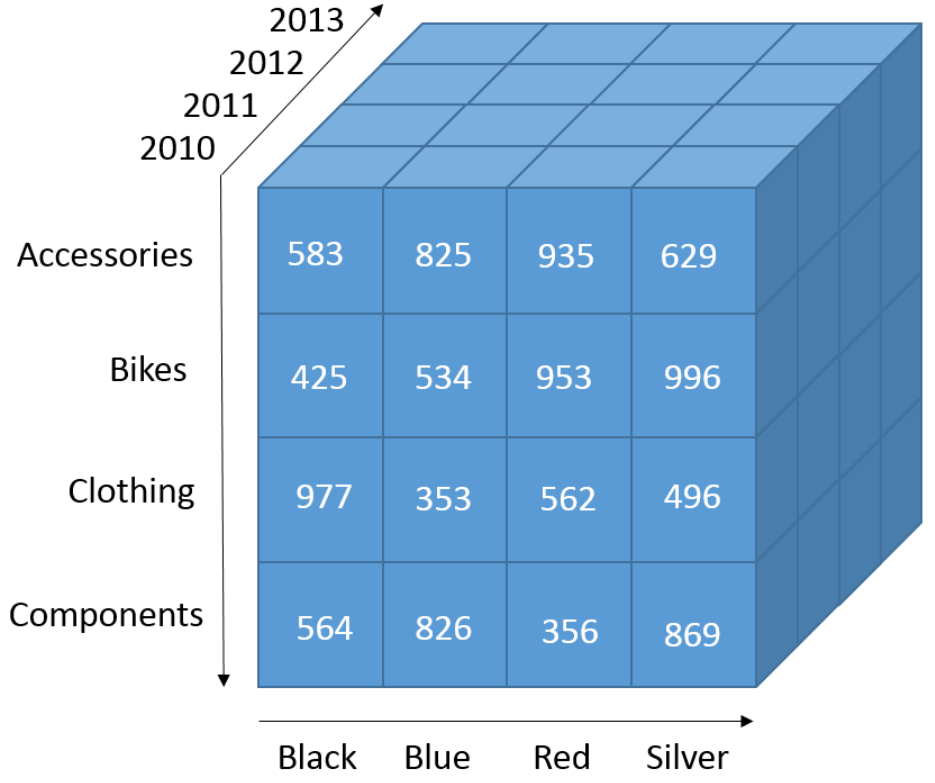
Modelo dimensional o *star schema*

*[...] Un **cliente** **compró** un **producto** de una **tienda** ayer*



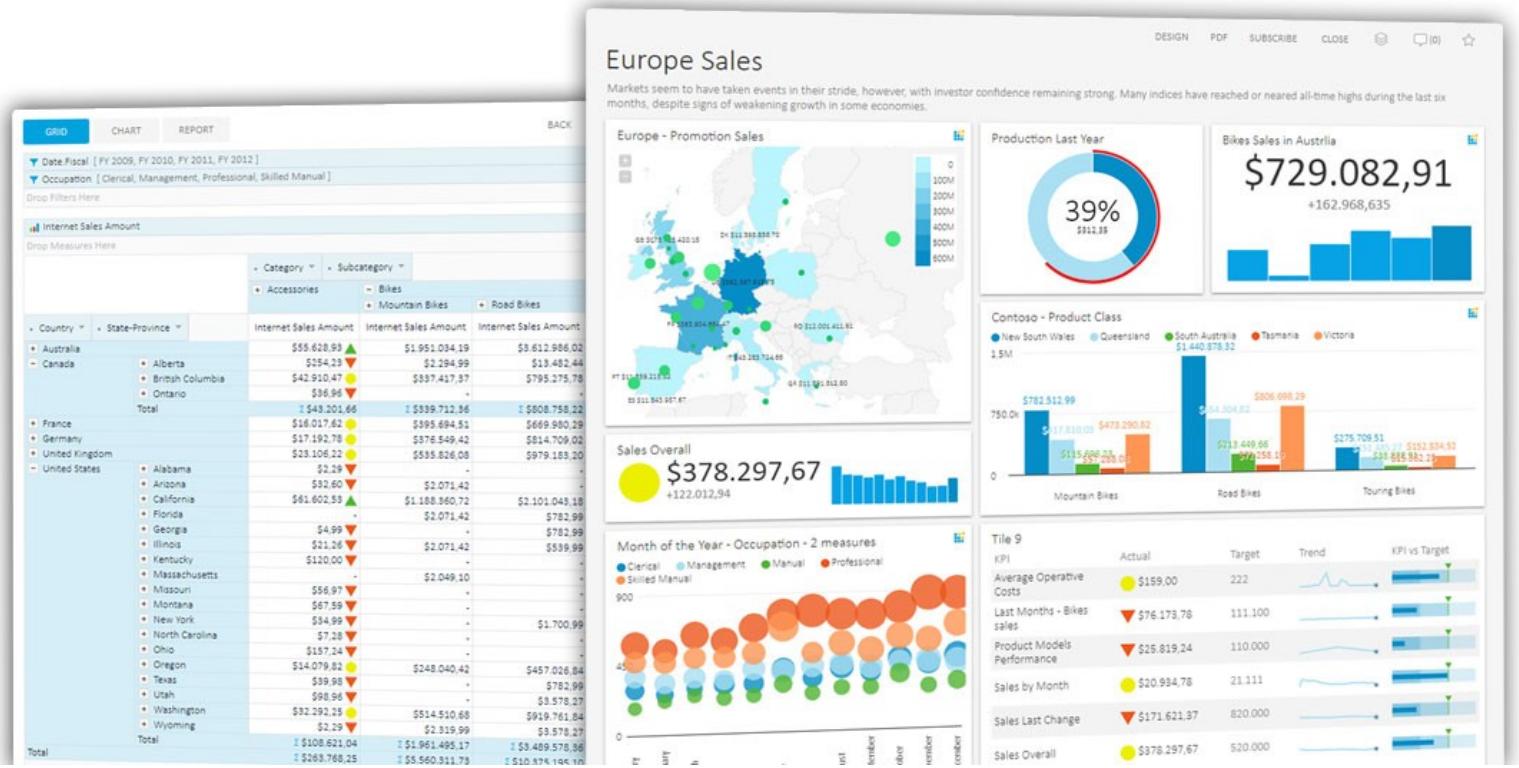
Cubos OLAP

Se conoce como cubos a estructuras ya calculadas para un uso ágil por parte de los sistemas de cuadro de mando.



Business Intelligence

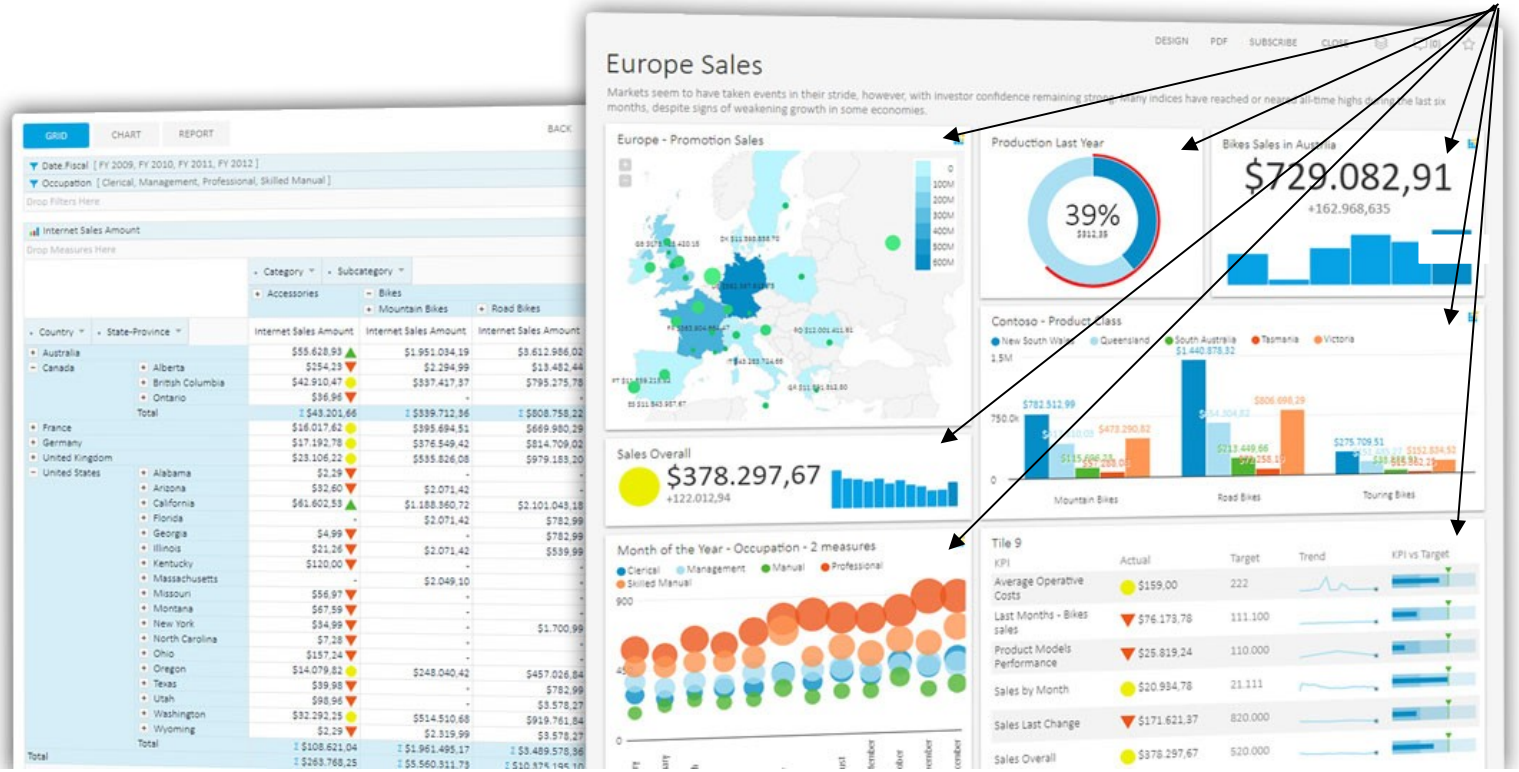
Existen soluciones que nos permiten crear cuadros de mando con esta información.

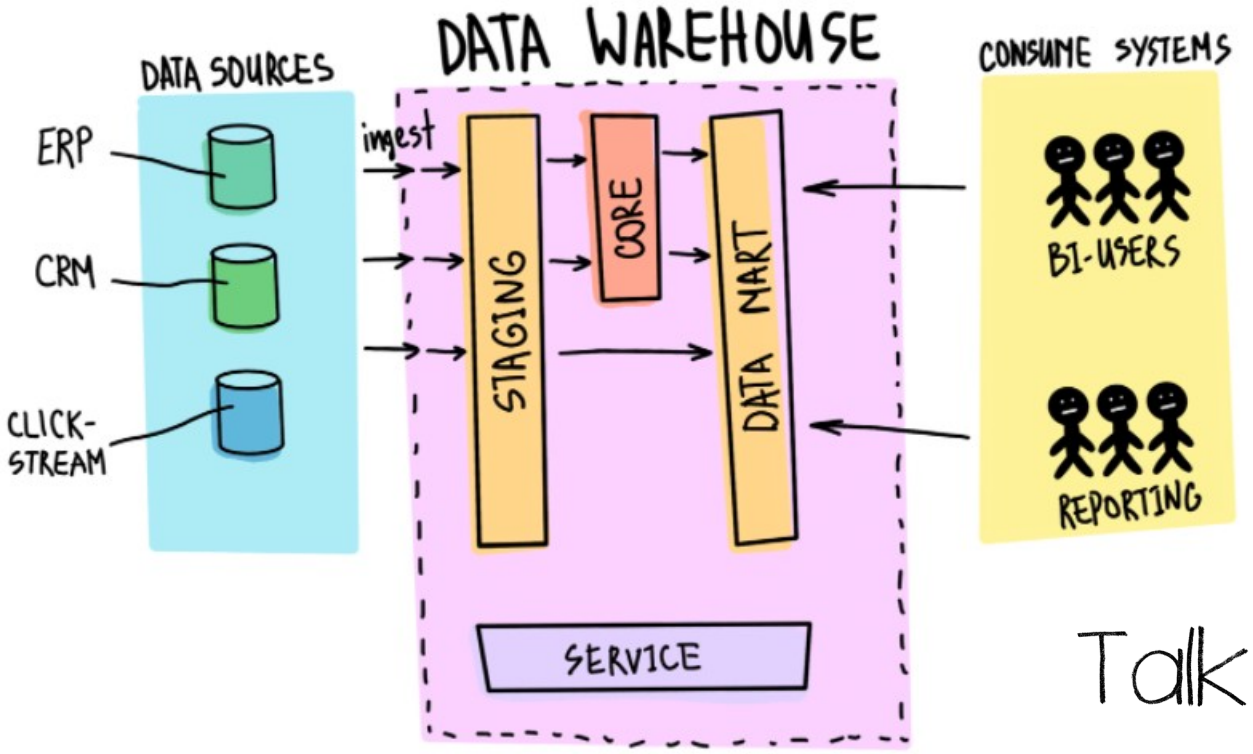


Business Intelligence

Existen soluciones que nos permiten crear cuadros de mando con esta información.

SQL





Talk is cheap

Show me the

CODE


THE
BRIDGE