

Bases de datos Unidad 1

**Universidad de Los Andes
Escuela de Ingeniería de Sistemas
Departamento de Computación**

Tema 2. Los sistemas de gestión de bases de datos

Tema 2. Los sistemas de gestión de bases de datos

▶ **Contenidos:**

- ▶ Reseña histórica
- ▶ Objetivos
- ▶ Conceptos generales
- ▶ Arquitectura de referencia
- ▶ Ejemplos de las estructuras de varios Sistemas de Gestión de Bases de Datos (SGBD) comerciales

▶ **Objetivo:**

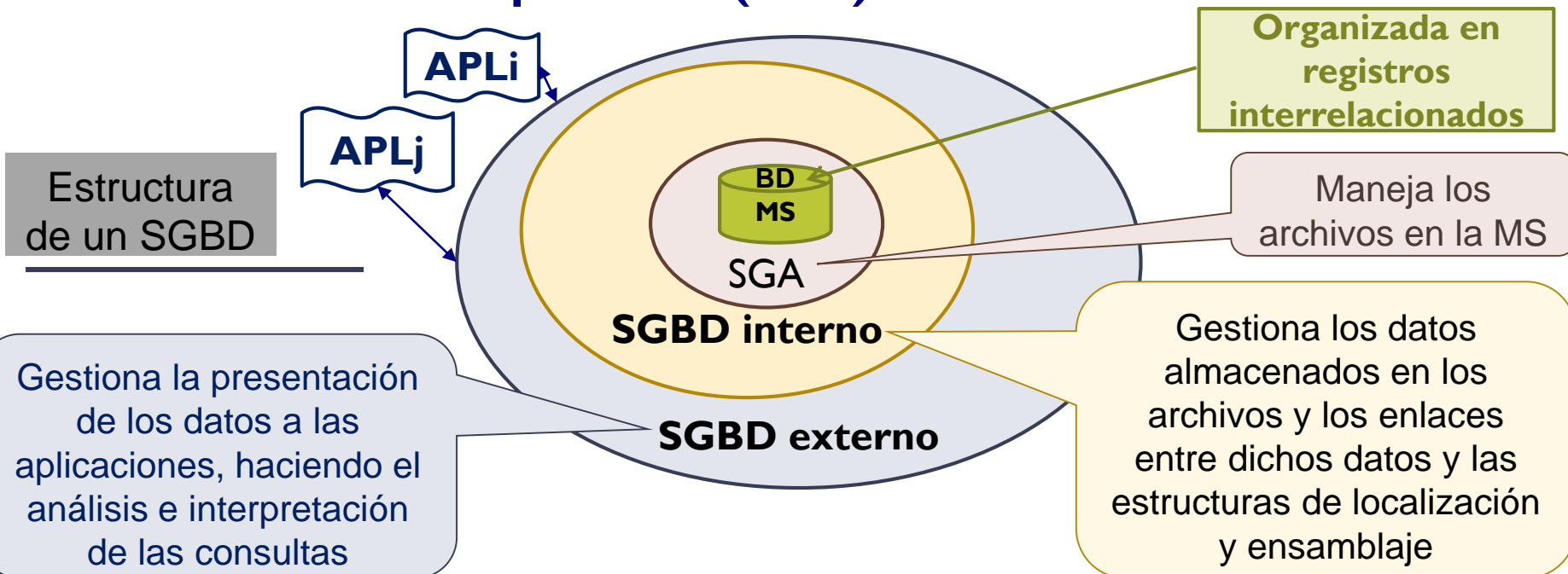
- ▶ Lograr una visión general de la estructura de los SGBD y de su evolución

▶ **Actividades:**

- ▶ Leer: Elmasri y Navathe, cap. 1 y 2
- ▶ Realizar el ejercicio 1

Bases de datos

- ▶ **SGBD: Software que permite insertar, modificar, eliminar y recuperar eficazmente los datos específicos dentro de una gran masa de información compartida por muchas aplicaciones o sistemas**
- ▶ **Base de datos (BD): colección de datos gestionados por un SGBD, que capturan la representación abstracta del dominio de una aplicación (APL)**



▶ **¿Por qué estudiar BD?**

1. Su uso expandido desde los celulares inteligentes hasta Wikipedia
2. Impacto en el mundo: mercado de software (aprox. igual al de los SO) sitios web, *eBusiness*, proyectos científicos, minería de datos, inteligencia de negocio, etc.

▶ **Principal objetivo: Simplificar el almacenamiento y acceso a los datos**

▶ **Ejemplo:**

Realización de las inscripciones y control de los estudios realizados por los estudiantes de la ULA

- ▶ No automatizado vs automatizado

Desarrollo de los SGBD

| Inicio de los años 60 | Finales de los 60 (SGA) | Años 70 y 80. 2da. 3ra. y 4ta. generación de SGBD | Años 90. 5ta. generación de SGBD | Años 00. SGBD híbridos y no relacionales |
|-----------------------------|---|---|---|---|
| Sólo archivos secuenciales | Aparecen los métodos de acceso | Se mantienen los métodos de acceso | Se mantienen los métodos de acceso | Se mantienen los métodos de acceso |
| Procesamiento en lotes | Procesamiento en lotes, en línea o en tiempo real | Procesamiento distribuido y/o paralelo | Se mantienen todos los tipos de procesamiento | Se mantienen todos los tipos de procesamiento |
| Archivo lógico = físico | Diferencia entre archivo lógico y físico | Tipos básicos de datos: entero, real, carácter, fecha, etc. | Otros tipos de datos: imágenes, audio, voz, animación, etc. | Se mantienen los tipos de datos con énfasis en el BLOB |
| Dependencia lógica y física | Independencia física | Hay independencia lógica, física y de localización | Manejo de objetos compuestos y complejos | Manejo de documentos, registros extensibles y por clave |

Desarrollo de los SGBD

| Inicio de los años 60 | Finales de los 60 (SGA) | Años 70 y 80. 2da. 3ra. y 4ta. generación de SGBD | Años 90. 5ta. generación de SGBD | Años 00. SGBD híbridos y no relacionales |
|---|---|---|--|---|
| No hay compartición de datos | No hay compartición de datos | Hay compartición de datos | Manejo de cualquier tipo de datos | Manejo de cualquier tipo de datos |
| Alto nivel de redundancia de datos, inconsistencia de datos | Se mantiene la redundancia y la inconsistencia de datos | Redundancia y consistencia de datos controlada | No redundancia. Soporte de evolución del esquema. Manejo de versiones y reglas | No redundancia. Soporte de varios tipos no relacionales |
| No hay recursos de seguridad | Se usan algunos recursos de seguridad | Manejo de la integridad y seguridad de datos. Transacciones ACID* | Transacciones ACID de larga duración y cooperativas | Transacciones ACID o BASE** |

* Atomicidad, Consistencia, aislamiento y Durabilidad

** Básicamente disponibles (Available), consistencia-Suave y Eventual

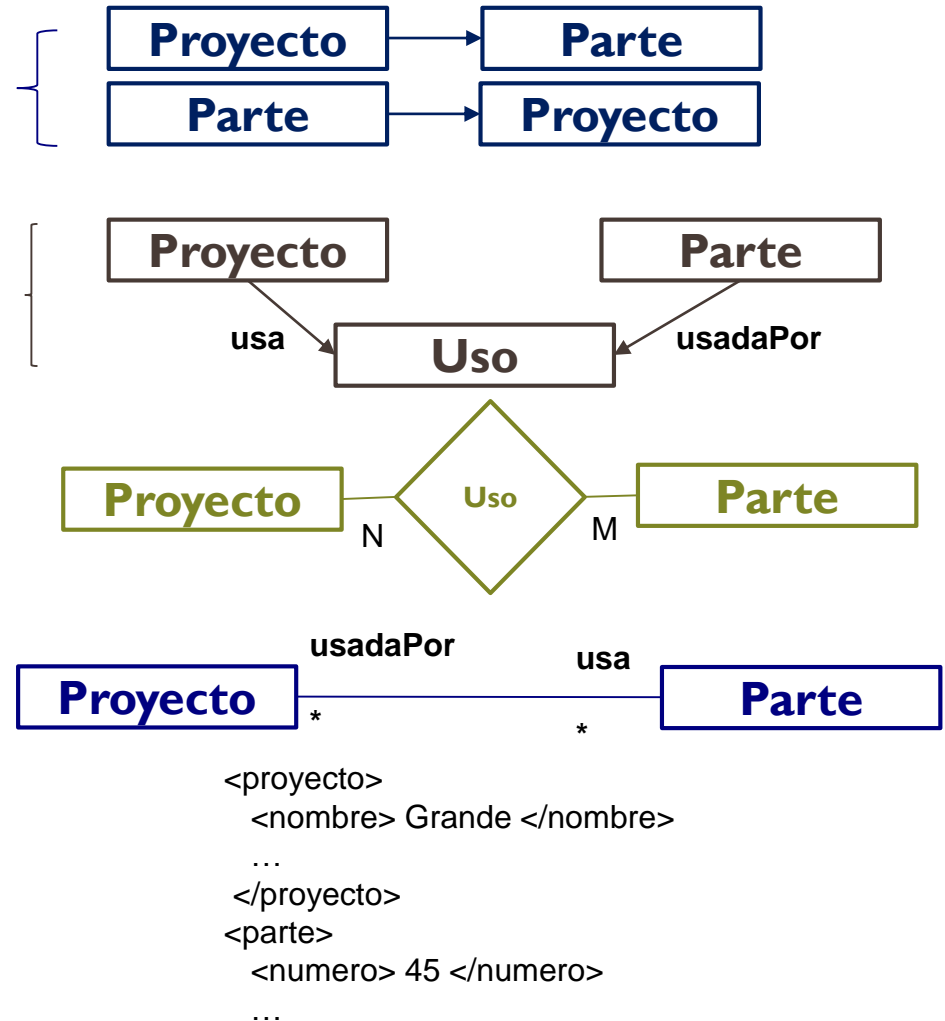
Desarrollo de los SGBD

| Inicio de los años 60 | Finales de los 60 (SGA) | Años 70 y 80. 2da. 3ra. y 4ta. generación de SGBD | Años 90. 5ta. generación de SGBD | Años 00. SGBD híbridos y no relacionales |
|--|--|---|--|--|
| Datos direccionables a nivel de archivos | Datos direccionables a nivel de registros | Datos direccionables a nivel de campo y grupo | Encapsulación de datos y de programas | Se mantiene la encapsulación de datos y programas |
| Lenguajes de programación con las primitivas de manejo de archivos | El SGA tiene sus primitivas de manejo de archivos utilizadas desde los lenguajes | Aparecen los lenguajes declarativos de descripción y de manipulación de datos | Uso de los lenguajes orientados por objetos para descripción y manipulación de objetos | Se mantiene el uso de los lenguajes de programación, orientados por objetos y declarativos |
| No hay recuperación por varias claves | No hay recuperación por varias claves | Recuperación por varias claves concatenadas o no | Adicionalmente hay recuperación por el identificador del objeto | Se mantienen los tipos de recuperación |

Reseña histórica

► **Eras:** [Stonebraker, 2014]

1. **Jerárquico (IMS) 1968-1970**
2. **Redes (CODASYL, Committee on Data Systems Languages) 1969-1973**
3. **Relacional (ER 1976) 1970-1983 (R, INGRES, DB/2)**
4. **Objeto (Semántica, 1977) 1983-1990 (GEM, etc.)**
5. **Objeto-Relacional 1990 (PostgreSQL, MySQL, etc.)**
6. **Semi-estructurado (XML) 1990-2014**



No estructuradas (NOSQL)

- ▶ **Almacenamiento por columna (entidad/atributo/valor)**
 - ▶ BigTable, Google, 2006
 - ▶ HBase, Facebook, Yahoo, 2007
 - ▶ Hypertable, Baidu, Rediff, 2007
- ▶ **Documental**
 - ▶ CouchDB, Apple, BBC, Cern, 2008
 - ▶ MongoDB, FourSquare, 2010
- ▶ **Basadas en Grafos**
 - ▶ Neo4j, Box.net, 2003
 - ▶ FlockDB, Twitter, 2010
- ▶ **Almacenamiento por clave/valor (tablas hash)**
 - ▶ Dynamo, Amazon, 2007
 - ▶ Membase, Zynga, 2009
 - ▶ Kyoto Cabinet
 - ▶ Redis, 2009
 - ▶ Cassandra, Facebook, Twitter, 2008
 - ▶ Voldemort, LinkedIn, 2008
 - ▶ Riak, Comcast, 2008
 - ▶ Hadoop, Lucene, 2011

Propiedades de un SGBDR

- ▶ **Un SMBD relacional debe cumplir con al menos estas tres propiedades:**
 1. Debe almacenar los datos como relaciones donde cada columna debe ser almacenada independientemente e identificada con su nombre y el orden de sus filas es irrelevante
 2. Las operaciones disponibles deben ser verdaderamente relacionales, esto es que ellas deben generar nuevas relaciones a partir de las relaciones operandos
 3. Debe soportar al menos una variante de la operación reunión (*join*)

Objetivos de un SGBDR

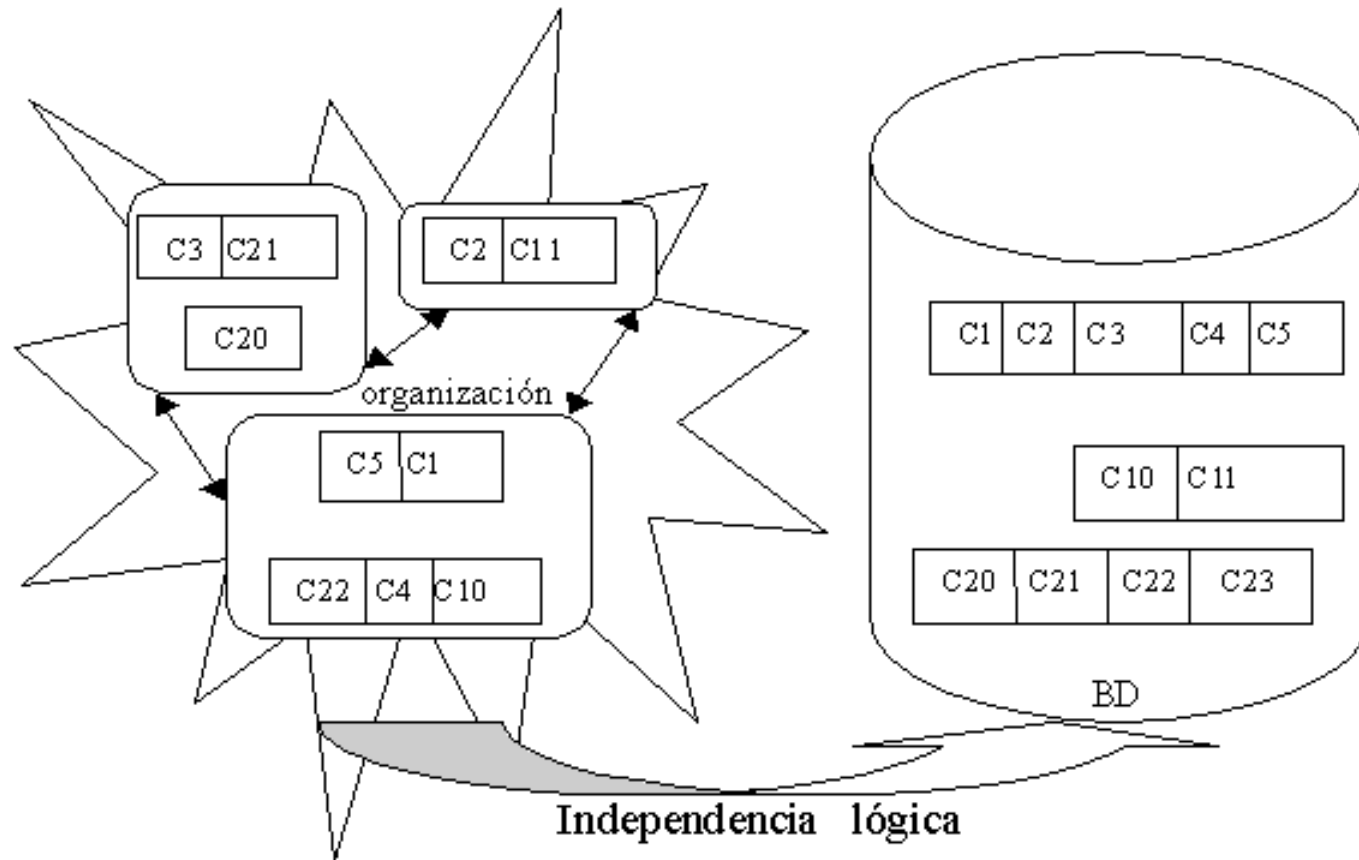
1. **Independencia física: debe permitir la realización de estructuras de almacenamiento de datos en forma independiente de su estructura lógica en la realidad**

Ventajas: los cambios en la estructura lógica no implican cambios en la de almacenamiento, las consideraciones sobre el mejor manejo de los datos almacenados quedan a cargo del SGBD y los cambio en la estructura de almacenamiento no implican cambios en las APLs

2. **Independencia lógica: debe permitir una cierta independencia entre los datos vistos por las aplicaciones y la estructura lógica de ellos en la realidad**

Ventajas: soporte de la evolución de los datos y que cada grupo de trabajo vea esos datos como cada grupo lo desea

Independencia lógica



Objetivos de un SGBDR

3. Manipulación de los datos por personas no especializadas en computación:

se logra permitiendo que cualquier persona consulte y actualice (si está autorizada) los datos en la base de datos, por medio de lenguajes de programación no procedimentales, que permiten expresar lo que se desea obtener, sin describir la forma de hacerlo

4. Eficacia en el acceso a los datos:

se permite el acceso eficaz a los datos sin tener en cuenta que el que hace el acceso sea especialista o no en el área y que conozca o no la estructura interna de los datos. El tiempo de respuesta está íntimamente ligado al número de accesos a disco, lo que será soportado por el SGBD

Objetivos de un SGBDR

5. **Administración centralizada de los datos:**

debe existir un grupo de trabajo dentro de la organización que se dedique a la definición de las estructuras de almacenamiento y de los datos de la base siguiendo su evolución a través del tiempo. Dicho grupo debe formarse con personas especializadas tanto en el área como en la misma organización

6. **Redundancia de datos controlada:**

la administración centralizada debe velar por la no duplicación física de los datos que serán compartidos por todos los usuarios, y en caso que sea absolutamente necesaria un cierto nivel de redundancia, ésta debe estar estrictamente controlada por el SGBD

Objetivos de un SGBDR

7. **Coherencia de los datos:**

el SGBD debe tener ciertas facilidades o útiles para hacer que las APL respeten las reglas de integridad de la BD, en particular aquellas correspondientes a la modificación de los datos

8. **Posibilidad de compartir los datos:**

una APL puede usar los datos almacenados en la BD como lo desee y como si fuera el único en usarlo, sin saber que otro puede estar modificándolo concurrentemente

9. **Seguridad de los datos:**

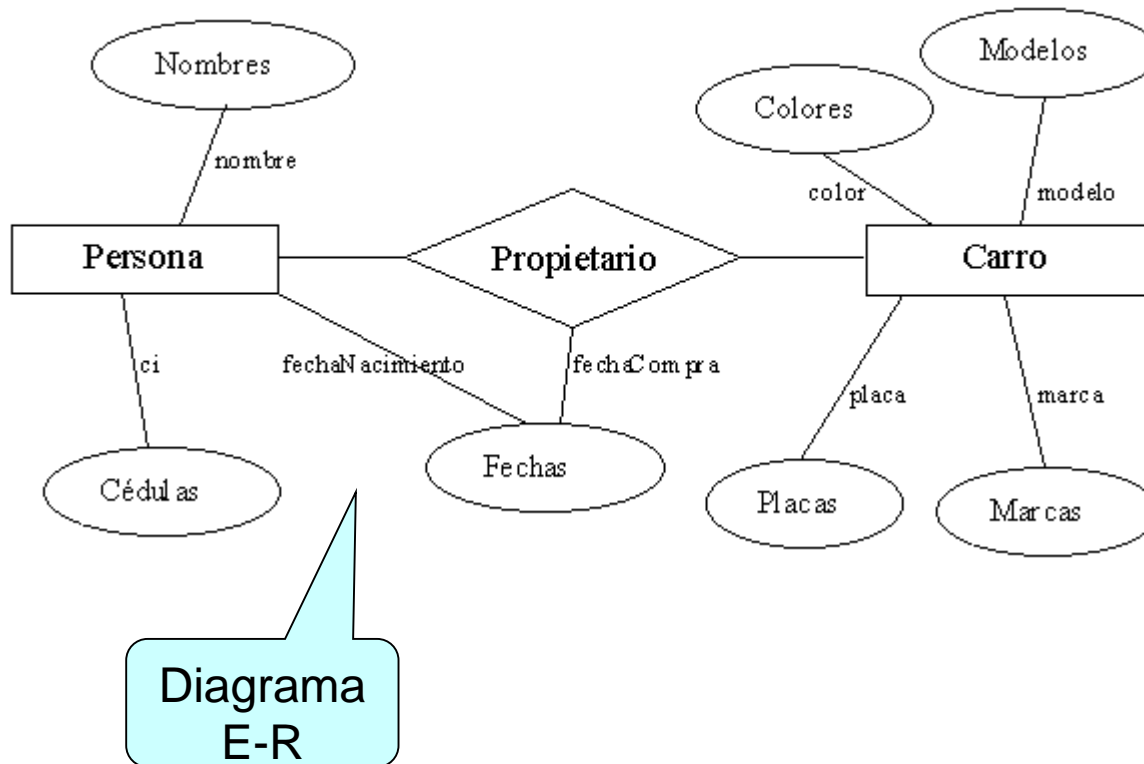
se debe permitir la protección contra accesos no autorizados o malintencionados

Objetivos de un SGBDOR

10. **Habilidad para representar y manipular objetos complejos y compuestos**
11. **Posibilidad de almacenar y recuperar datos arbitrariamente grandes**
12. **Habilidad para definir y manipular cualquier tipo de dato**
13. **Posibilidad de representar y manejar los cambio en la base de datos a través del tiempo**
14. **Representación y manipulación de varios conceptos de modelado semántico, útiles en la aplicaciones**
15. **Posibilidad de especificar reglas y restricciones de integridad para soportar inferencia en las aplicaciones basadas en conocimiento**
16. **Habilidad para manejar transacciones cooperativas de larga duración**

Modelo de datos

- **Conjunto de conceptos y reglas de composición de los mismos que permiten describir los datos**



Conceptos básicos

- ▶ **Lenguaje de descripción de datos (LDD):** permite describir los datos de un modelo de una BD entendible por una máquina
- ▶ **Esquema:** descripción de un conjunto de datos particulares hecha por medio del LDD

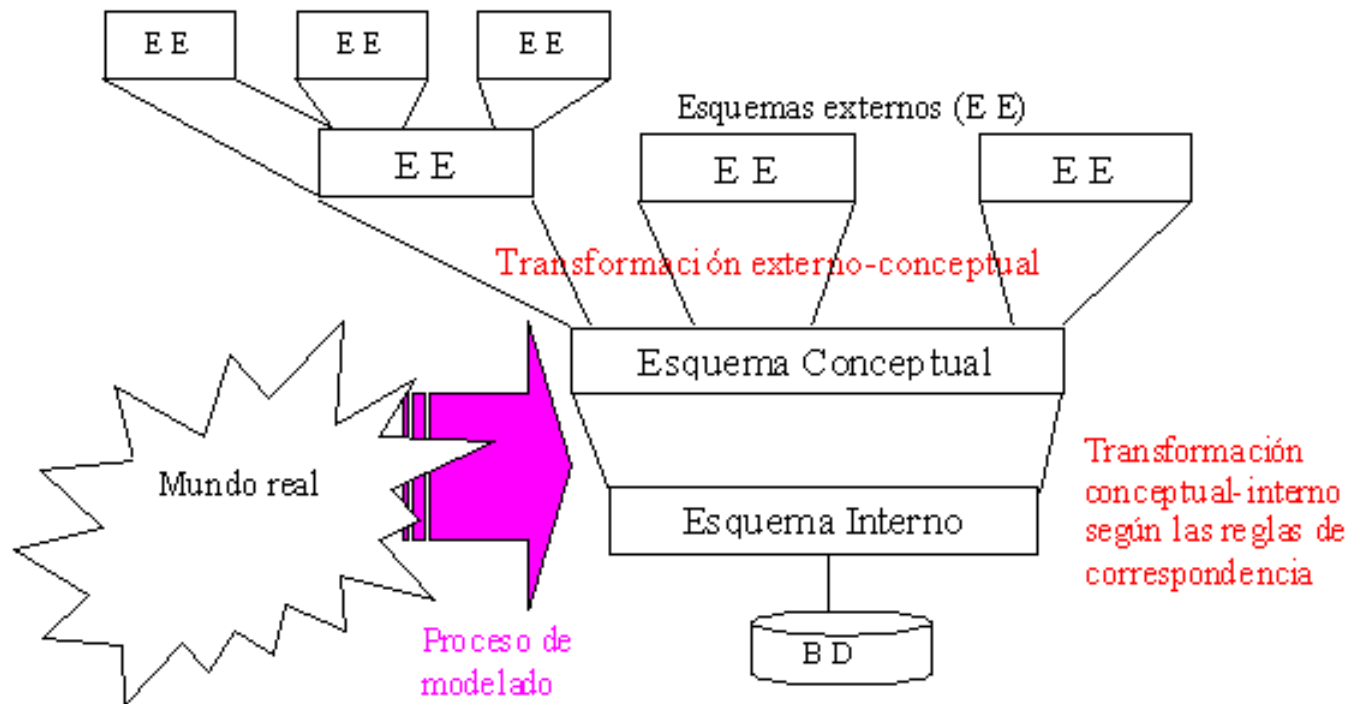
```
CREATE TABLE Persona
```

```
(cédula: String 10,  
 nombre: String 50,  
 fechaNacimiento: Date, ..... )
```

- ▶ **Administrador de datos:** persona encargada de la definición de los esquemas y de sus reglas de correspondencia e integridad

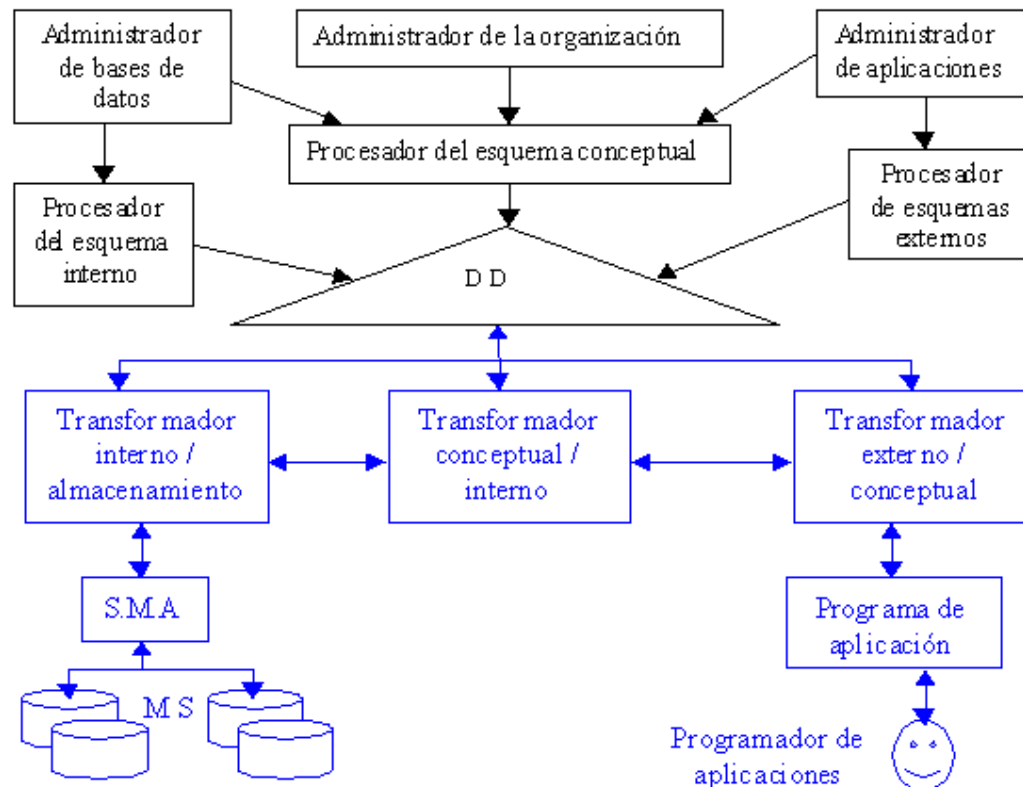
Conceptos básicos

- **Niveles de descripción de un SBD: las diferentes etapas que se deben efectuar para describir una BD**

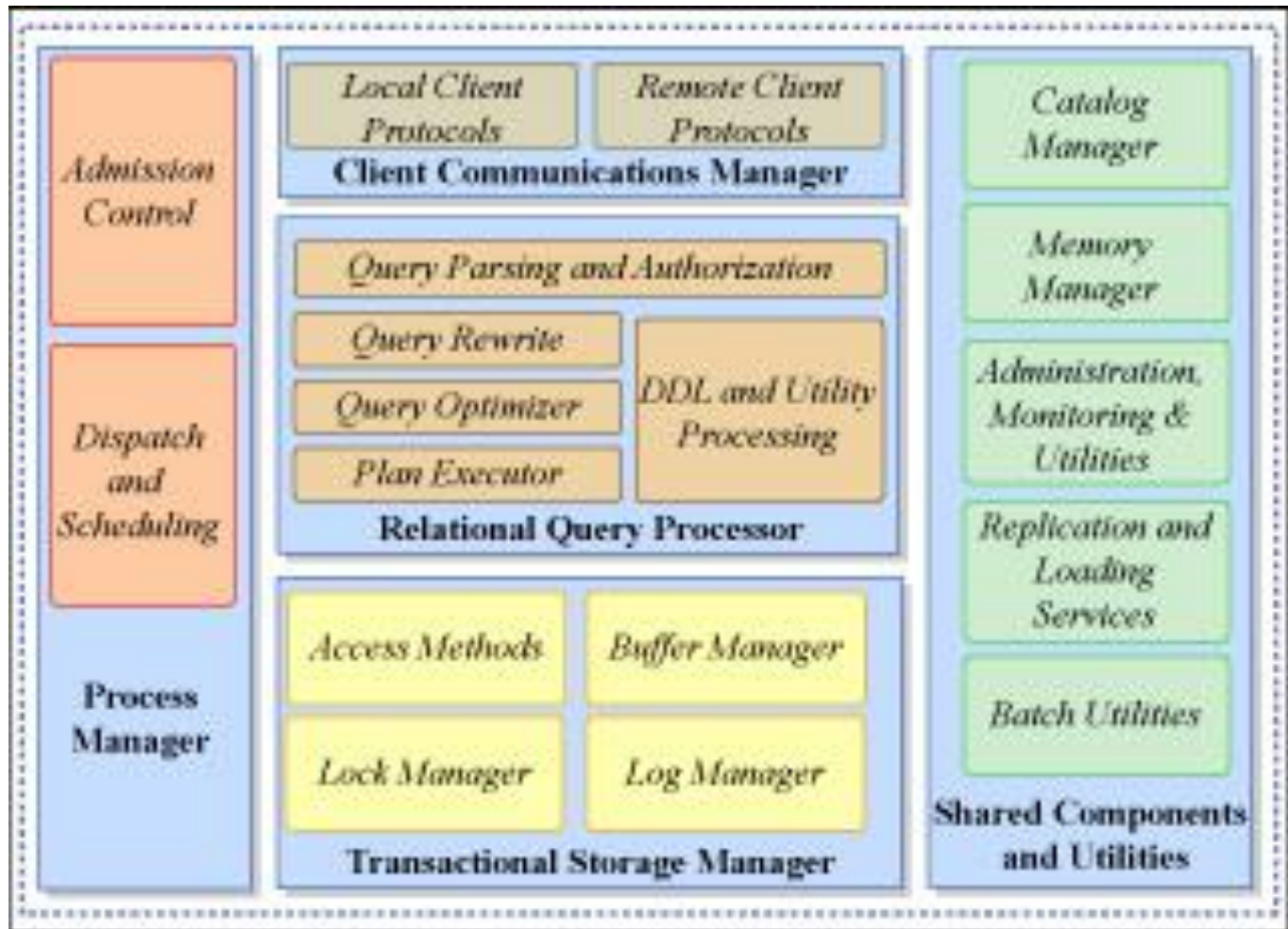


Arquitectura de referencia

- ▶ **Catálogo o Diccionario de Datos (DD):** conjunto de esquemas y reglas de correspondencia entre dichos esquemas, combinado con una descripción del significado de los datos en la BD
- ▶ Según el grupo de estandarización **ANSI/X3/SPARC**

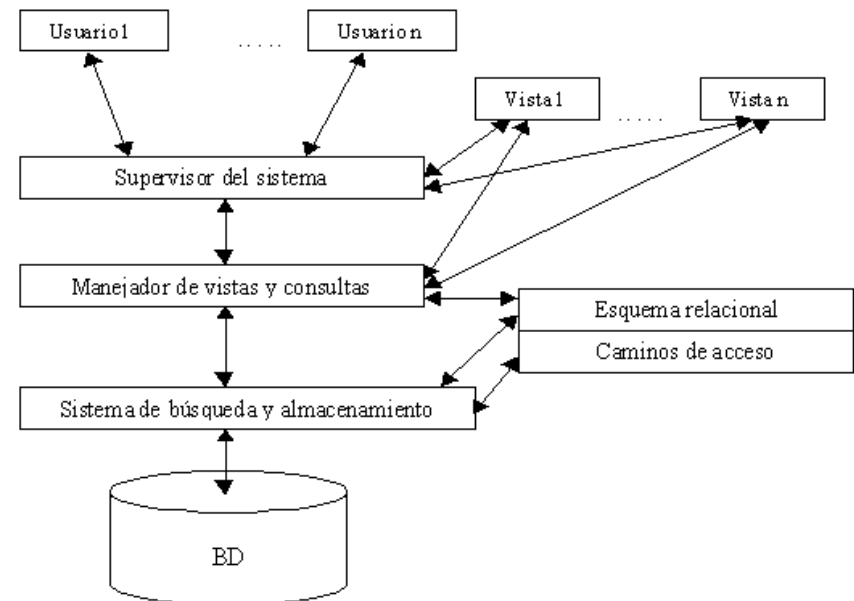


Subsistemas de un SGBDOR



SGBDR comerciales. Sistema R

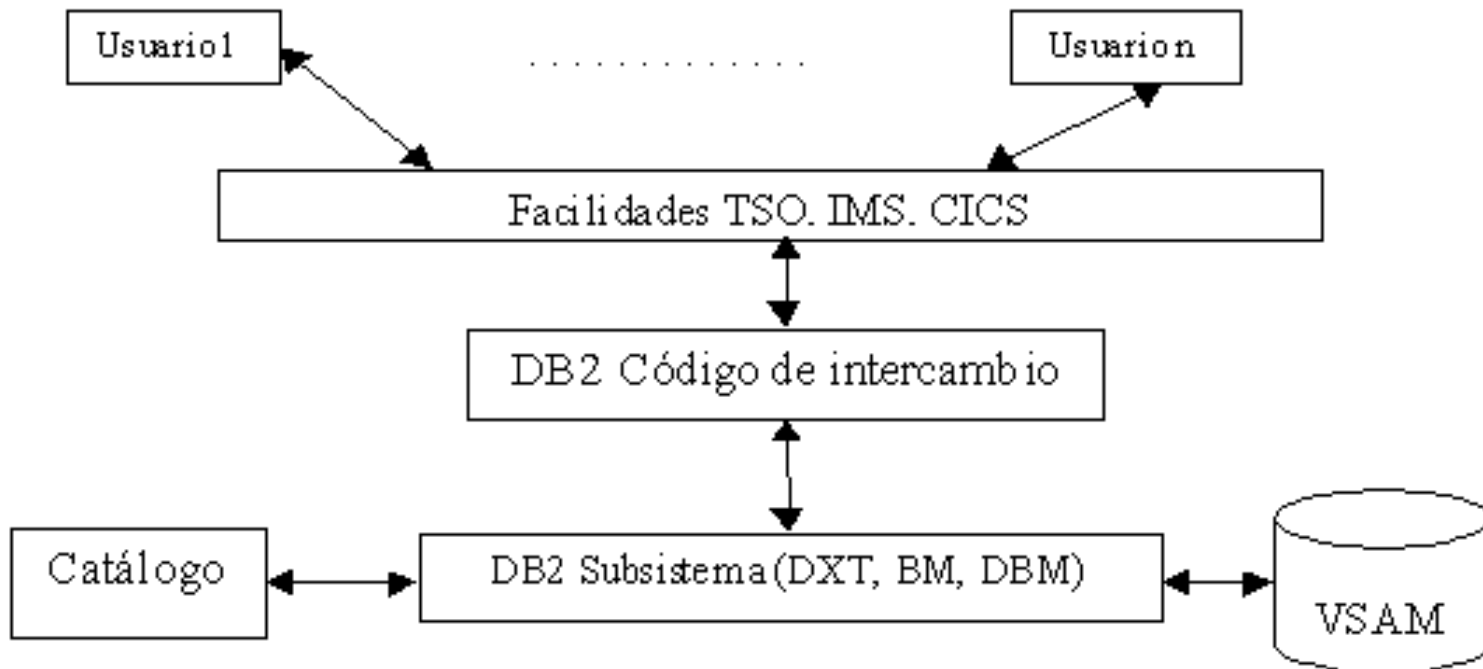
- ▶ Diseñado y desarrollado entre 1974 y 1979 en uno de los laboratorios de investigación de la compañía IBM [Ast-76]
- ▶ Es uno de los primeros SGBD relacionales
- ▶ Lenguajes de definición y manipulación de datos:
- ▶ SQL (*Structured Query Language*), basado en el lenguaje SEQUEL y en el álgebra relacional
- ▶ QBE (*Query By Example*), representante del cálculo relacional de dominios



- ▶ **Producto IBM (*San Jose Research Center, ahora Almadem*) desarrollado en 1983 siguiendo el modelo relacional y presentando mejoras con respecto a su antecesor, el sistema R**
- ▶ **Lenguajes: Cobol, PL/I, Fortran, C, Prolog y ensamblador IBM**
- ▶ **Tablas:**
 - ▶ básicas existen físicamente y consisten de uno o varios archivos VSAM y
 - ▶ vistas son tablas virtuales, no existen físicamente
- ▶ **Última versión: DB2 UNIVERSAL Database**
- ▶ **Soporta el enfoque objeto-relacional (*Object-Relational*)**
- ▶ **Permite realizar BD distribuidas**
- ▶ **Es uno de los mejores en el soporte de paralelismo en BD**

SGBDOR-DB2

- ▶ El almacenamiento de los datos se realiza en un espacio de tablas (*tablespace*) y un espacio de índices (*indexspace*)
- ▶ **Página:** unidad de transferencia entre la memoria principal y la secundaria, tienen una longitud fija entre 4 KB y 32 KB

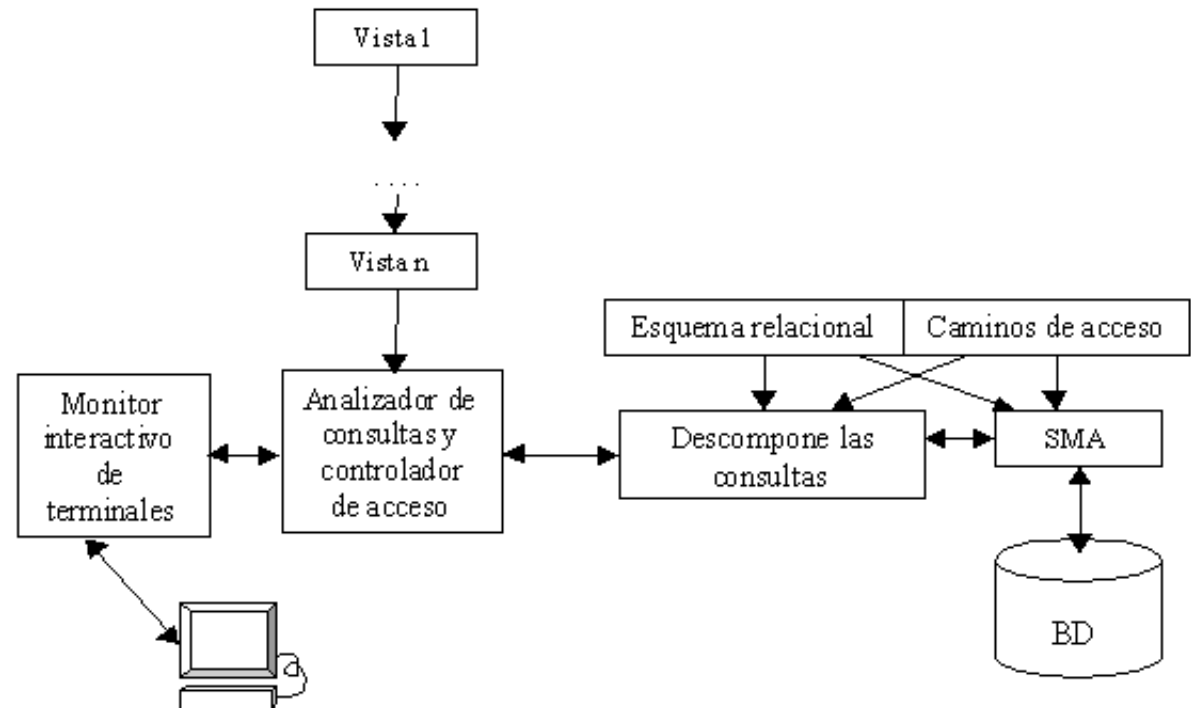


- ▶ **Espacio: colección dinámica y extensible de páginas, cuya longitud va desde 4 KB hasta 64 GB con bloqueos mínimos de 1/4 de página**
- ▶ **Cada espacio pertenece a un grupo de espacios que es una colección de áreas de acceso directo en el mismo dispositivo**
- ▶ **Los grupos de almacenamiento se manejan como VSAM**
- ▶ **Varias tablas por *tablespace***
- ▶ **Cada tabla tiene un índice árbol_B+ agrupado por la clave primaria cuyo nodo es una página y sus hojas están enlazadas**
- ▶ **Almacena todos los tipos de datos como cadenas de bytes**
- ▶ **Los registros dentro de una página se pueden modificar sin cambiar su identificador (RID)**

- ▶ **Desarrollado en la Universidad de Berkeley-California, proyecto de 1970-1985, Actian Corp. Versión 2010**
- ▶ **Estructura similar al Sistema R**

▶ **Diferencias:**

- ▶ Procesamiento y optimización de las consultas o interrogaciones a la BD
- ▶ Lenguaje QUEL (*Query Language*) basado en el cálculo relacional de tuplas

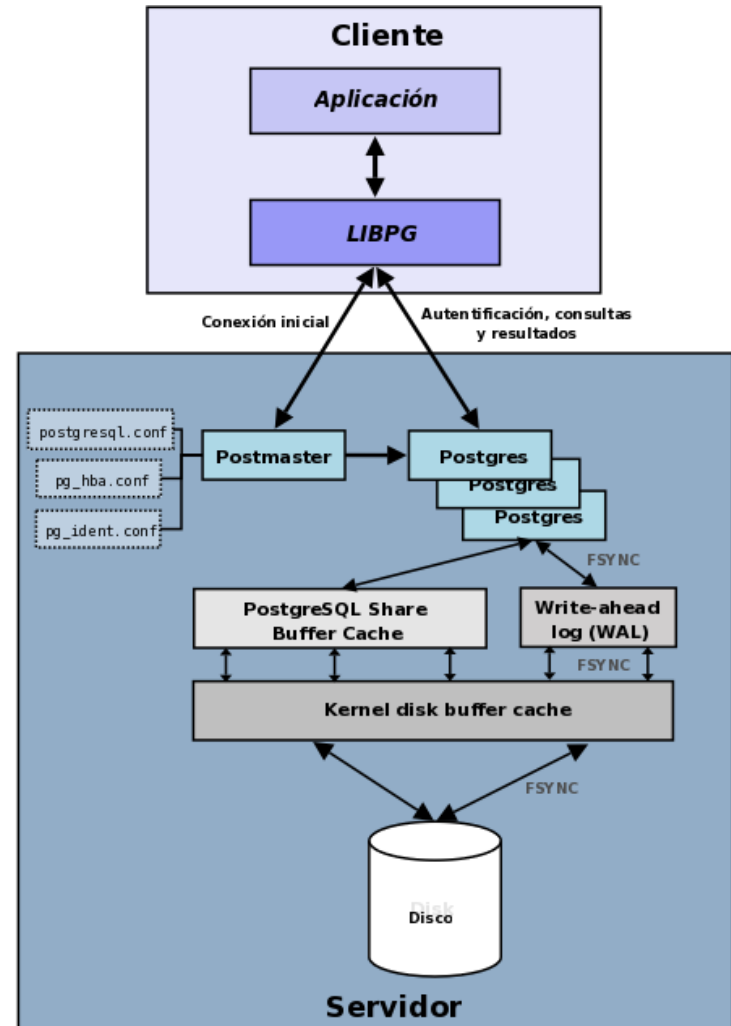


SGBDOR-PostgreSQL

- ▶ **Post Ingres, 1995. Estructura basada en el modelo cliente-servidor con un proceso por usuario**
- ▶ **Una sesión de usuario contiene procesos cooperativos de Unix:**
 - ▶ Un proceso supervisor (demonio) denominado *postmaster*
 - ▶ La aplicación *frontend* del usuario (e.g., el programa *psql*)
 - ▶ Uno o más servidores de bases de datos (*backend*), el proceso *postgres* en si mismo
 - ▶ Un proceso supervisor simple maneja una colección de bases de datos en un *host*. Esta colección se denomina un *cluster* (de bases de datos)
 - ▶ Supervisor y servidores están en el servidor de BD con el *user-id* *postgres* diferente del *super-usuario* de *unix* o *linux*
 - ▶ Aplicaciones normalmente están en los clientes

SGBDOR-PostgreSQL

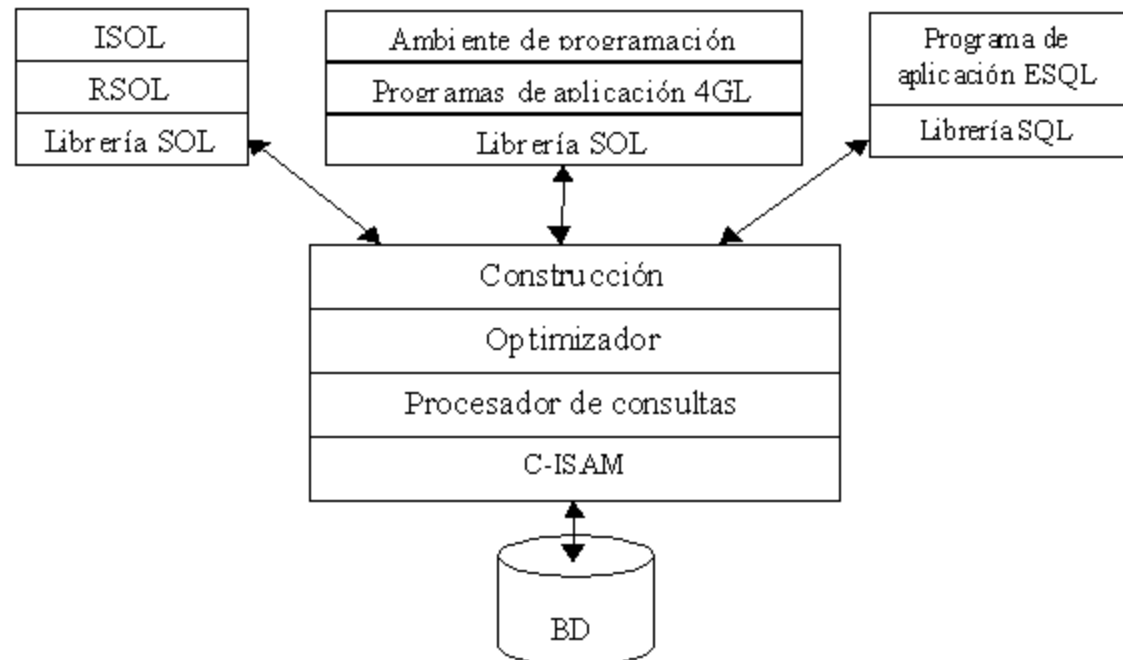
- ▶ El proceso supervisor está siempre corriendo, esperando por nuevos requerimientos, mientras que las aplicaciones y los servidores vienen y se van
- ▶ libpq permite hacer múltiples conexiones desde una aplicación hacia varios servidores
- ▶ Cada aplicación es un proceso de un hilo
- ▶ Multiprocesos en vez de multihilos, si un proceso falla no afecta al resto de los procesos



SGBDR-INFORMIX

- ▶ **Desarrollado por Informix Software, Inc. (1980-2005)**
- ▶ **Lenguaje de manipulación de datos: basado en SQL**
- ▶ **Lenguaje de cuarta generación: 4GL, permite el desarrollo de los PA sin el uso de lenguajes anfitriones**

- ▶ Nueva versión: **INFORMIX UNIVERSAL SERVER**, soporta las mismas características de su homólogo DB2
- ▶ Versión orientada por objetos: **INFORMIX-ILLUSTRRA**, fusión de Informix Software, Inc y la compañía Illustra
- ▶ IBM-Informix, 2001



Autoevaluación

1. **¿Qué es una BD?**
2. **¿Qué es un SGBD y cuál es su estructura?**
3. **¿Cómo ha sido el desarrollo de los SGBD?**
4. **¿Cuáles son los objetivos de un SGBD?**
5. **Cuál es la diferencia entre independencia física y lógica?**
6. **¿Qué es un modelo de datos?**
7. **¿Cuáles son los niveles de descripción de un SBD?**
8. **¿Cuáles son las propiedades de un SGBDR?**
9. **¿Cuál es la arquitectura de referencia de un SGBD?**
10. **¿Cuáles son las arquitecturas del sistema R, DB2, INGRES, PostgreSQL e INFORMIX?**