

**INSTITUTO POLITÉCNICO NACIONAL**

**UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA  
Y TECNOLOGÍAS AVANZADAS**

**BASES DE DATOS DISTRIBUIDAS 3TM3**

**Reporte de lectura del tema arquitecturas de BDD**

Integrantes:

- Lisardo René Morgado Resendiz

## Conceptos relevantes de la lectura

### - *Independencia de datos:*

Es un concepto de bases de datos que nos habla de una forma de gestión en la cuál nuestros datos están separados de otros programas que podrían estarlos usando. Esto a forma de protección de datos.

### - *Arquitectura ANSI/SPARC:*

Creada por un grupo de estudio del ANSI (American National Standards Institute) propone que las interfaces sean estandarizadas, y define una arquitectura con 43 interfaces, de las cuales 14 trabajarán directamente con el subsistema físico de almacenamiento de la computadora y por lo cuál, no serán parte esencial de la arquitectura.

Una versión simplificada de esta arquitectura nos dice que debe haber 3 vistas simplificadas de los datos, una vista “externa” para los programadores o usuarios finales, una vista “interna” para la máquina o el sistema y una vista “conceptual” para la empresa, por lo cuál se requiere un esquema definido para cada vista.

En el nivel más bajo de la arquitectura tenemos la vista interna, que trabaja con la definición física y la organización de los datos. Los problemas a resolver en este nivel de la arquitectura son los mecanismos de acceso a los diferentes dispositivos de almacenamiento y la ubicación de los datos en estos.

En la vista externa se gestiona la vista que tendrá un usuario y las relaciones de datos a las que podrá acceder. Una vista puede ser compartida por diferentes usuarios creando así con la colección de estos la vista externa general.

En medio de estas dos definiciones tenemos el esquema conceptual, que es una definición abstracta de la base de datos, la cuál deberá representar los datos y las relaciones entre estos considerando las restricciones físicas de los dispositivos de almacenamiento y las aplicaciones individuales.

Estas tres definiciones se complementan por medio de mapeos en los cuáles se especifica como una definición de un nivel puede ser obtenida mediante una de otro. Esta perspectiva es importante ya que provee las bases para la **independencia de datos**.

### - DBMS

Un DBMS (Data Base Manager System) es un programa compartido con múltiples procesos llamados transacciones el cuál tiene dos dependencias, el **subsistema de comunicación** y el **sistema operativo**.

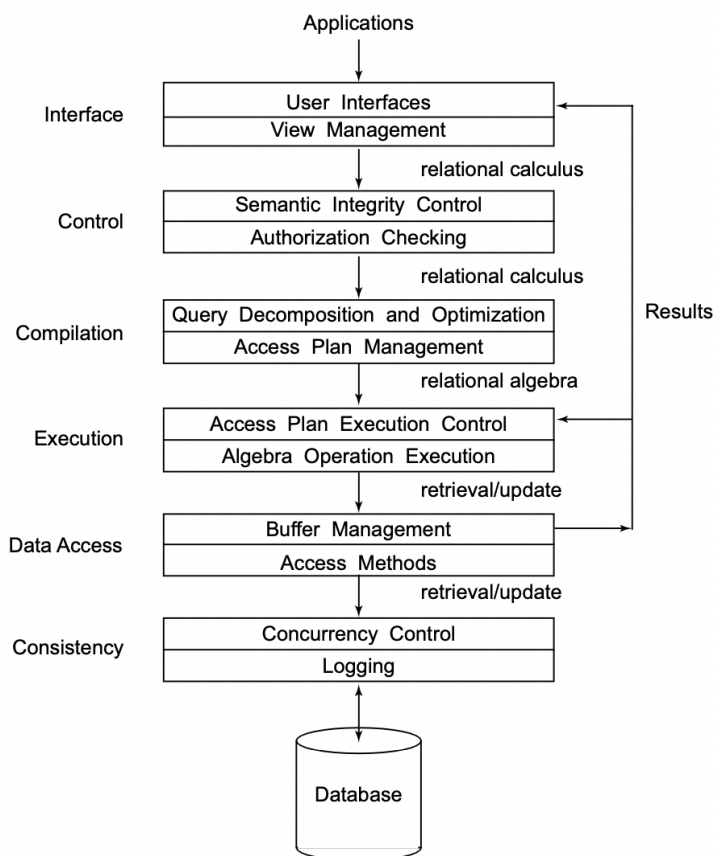
- Función del subsistema de comunicación:

Este permite la comunicación mediante interfaces del DBMS con otros subsistemas para comunicarse con otras aplicaciones.

- Función del sistema operativo:

El sistema operativo nos proporciona una manera de comunicar nuestro DBMS con los recursos del ordenador.

- Capas funcionales de un DBMS Centralizado:



La **capa de la interfaz** maneja lo que es transmitido a las aplicaciones, ya sea mediante lenguajes de programación, consultas realizadas aplicando operaciones de álgebra relacional, etc. Prácticamente consiste en traducir la búsqueda del usuario de datos externos a datos conceptuales.

La **capa de control** controla las búsquedas añadiendo predicados semánticos y de autorización en un lenguaje declarativo. Enriquece las búsquedas en lenguaje de alto nivel hechas **por la capa de interfaz**.

La **capa de compilación o procesamiento de consultas** convierte nuestro código en operaciones de bajo nivel mediante la descomposición de la búsqueda en un árbol de operaciones algebraicas y la optimización de dichas búsquedas. El resultado de esta capa se guarda en un plan de acceso con instrucciones de bajo nivel.

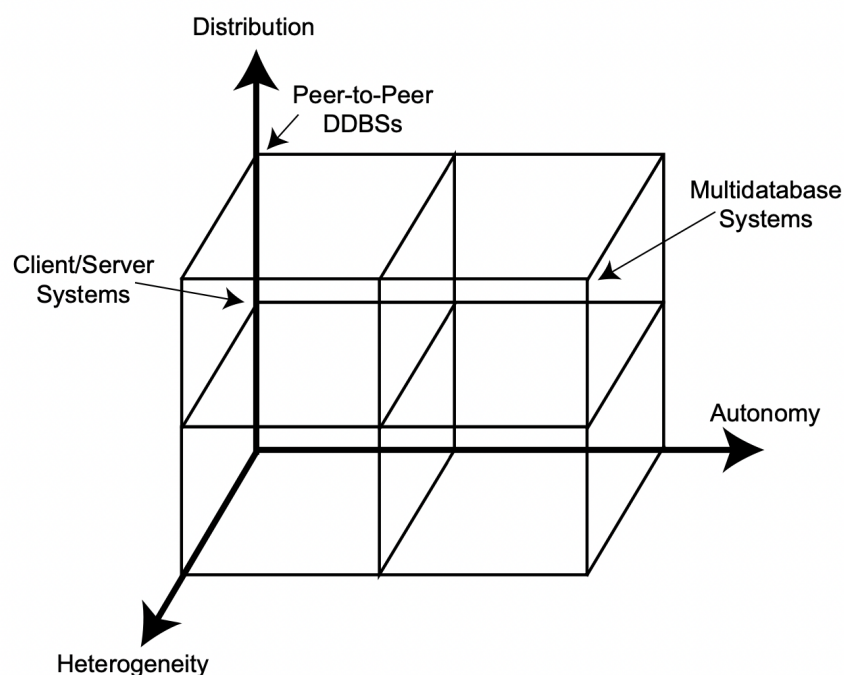
La **capa de ejecución** dirige la ejecución de los planes de acceso generados en la capa anterior.

La **capa de acceso a los datos** maneja las estructuras de datos y el caché que generan las consultas frecuentes.

Por último, la **capa de consistencia** maneja el control de la concurrencia y el registro de solicitudes de actualización.

- Modelos de arquitectura para los manejadores de bases de datos distribuídas.

Ahora que sabemos que tenemos diferentes maneras de estructurar la arquitectura de nuestro DBMS, usamos una clasificación que organiza el sistema con respecto de la autonomía de los sistemas locales, su distribución y su heterogeneidad.



## **Conclusiones de la lectura**

Hoy en día nos resulta tan común acceder a información almacenada en una base de datos que no dimensionamos la cantidad de movimientos y arquitecturas que tuvieron que ser creadas para hacer esto posible, una de ellas es la arquitectura ANSI/SPARC, la cuál nos ayuda a mantener la independencia de nuestros datos para brindar un nivel extra de protección mediante una serie de procesos en cascada que ayudan a concretar una consulta.

## **Bibliografia**

[1] M. T. Özsu y P. Valduriez, Principles of Distributed Database Systems. Springer, 2019.

[2] Bases de datos avanzadas. Castelló de la Plana, Spain: Universitat Jaume I, 2013.