

# Bases de Datos Distribuidas

BDD=Colección de múltiples bases de datos, lógicamente interrelacionadas y distribuidas a través de una red de computadores <sup>[1]</sup>

DDBMS=sistema de software que permite la administración de una BDD y hace transparente la distribución a los usuarios <sup>[1]</sup>

$DDBS = DDBMS + BDD$

$DBE = BD + DBMS$  <sup>[2]</sup>

$DDBE = BDD + DDBMS$  <sup>[2]</sup>

# Ventajas

Estructura = Estructura Organizacional

Control sobre datos locales

Tolerancia a Fallos

Rendimiento (paralelismo, distribución, balanceo de carga)

Economía (más servidores de menor precio)

Modularidad (facilidad para agregar o quitar componentes)

Transparencia en localización de los datos

# Desventajas

Complejidad (transparencia, distribución, diseño)  
Mayor necesidad de mano de obra  
Seguridad (red, administración, dbms heterogéneos)  
Integridad (mayor dificultad, fallos)  
Recuperación (mayor dificultad ante fallos)  
Know-how escaso  
Pocos standards en BDD  
Diseño BDD más complejo y propenso a errores

# Arquitecturas SGBDD

- Repaso

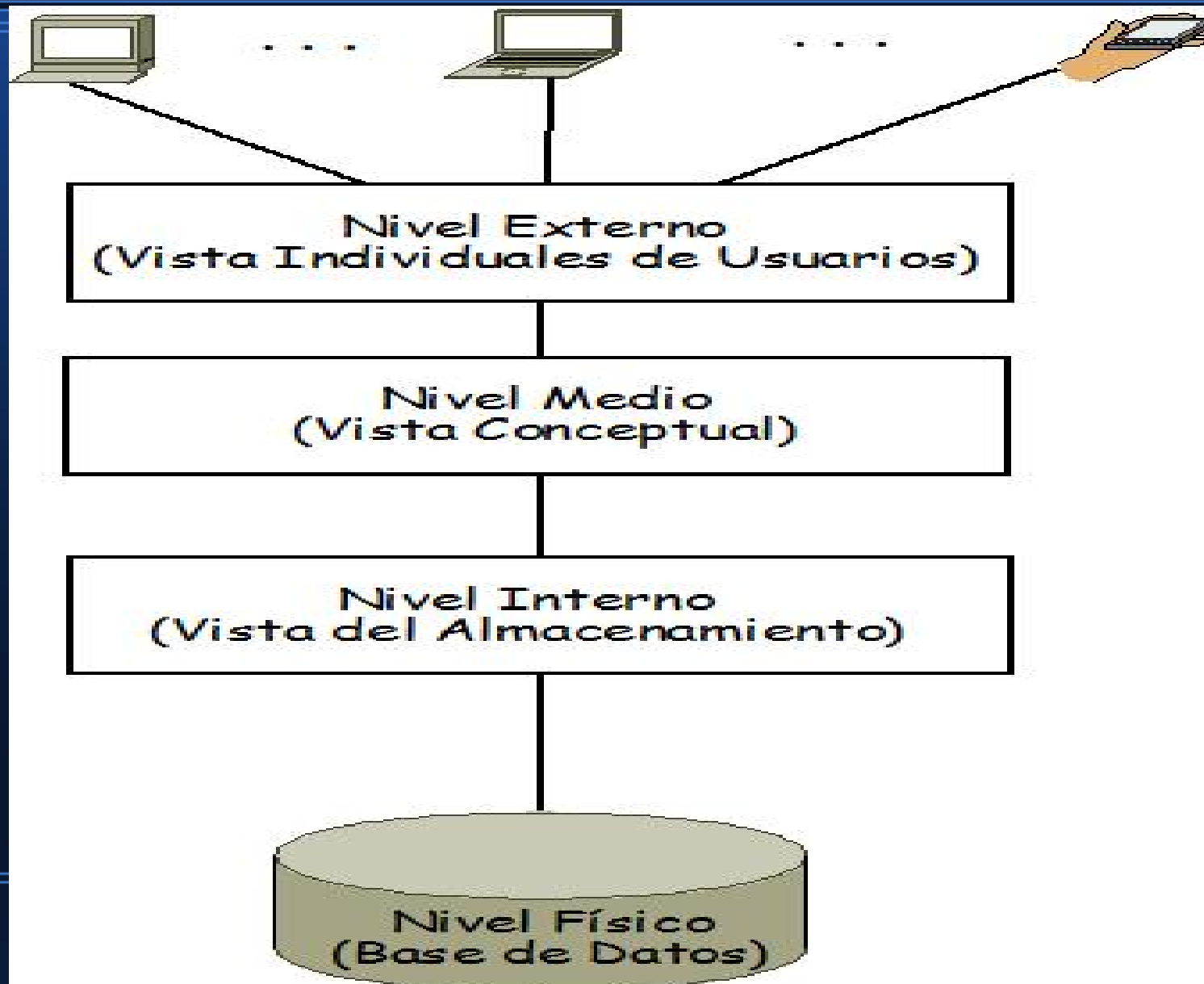
- Apunte Arquitecturas: monousuario, file-server, client-server, 3-tier, n-tier

Arquitectura base: client-server, 2-tier

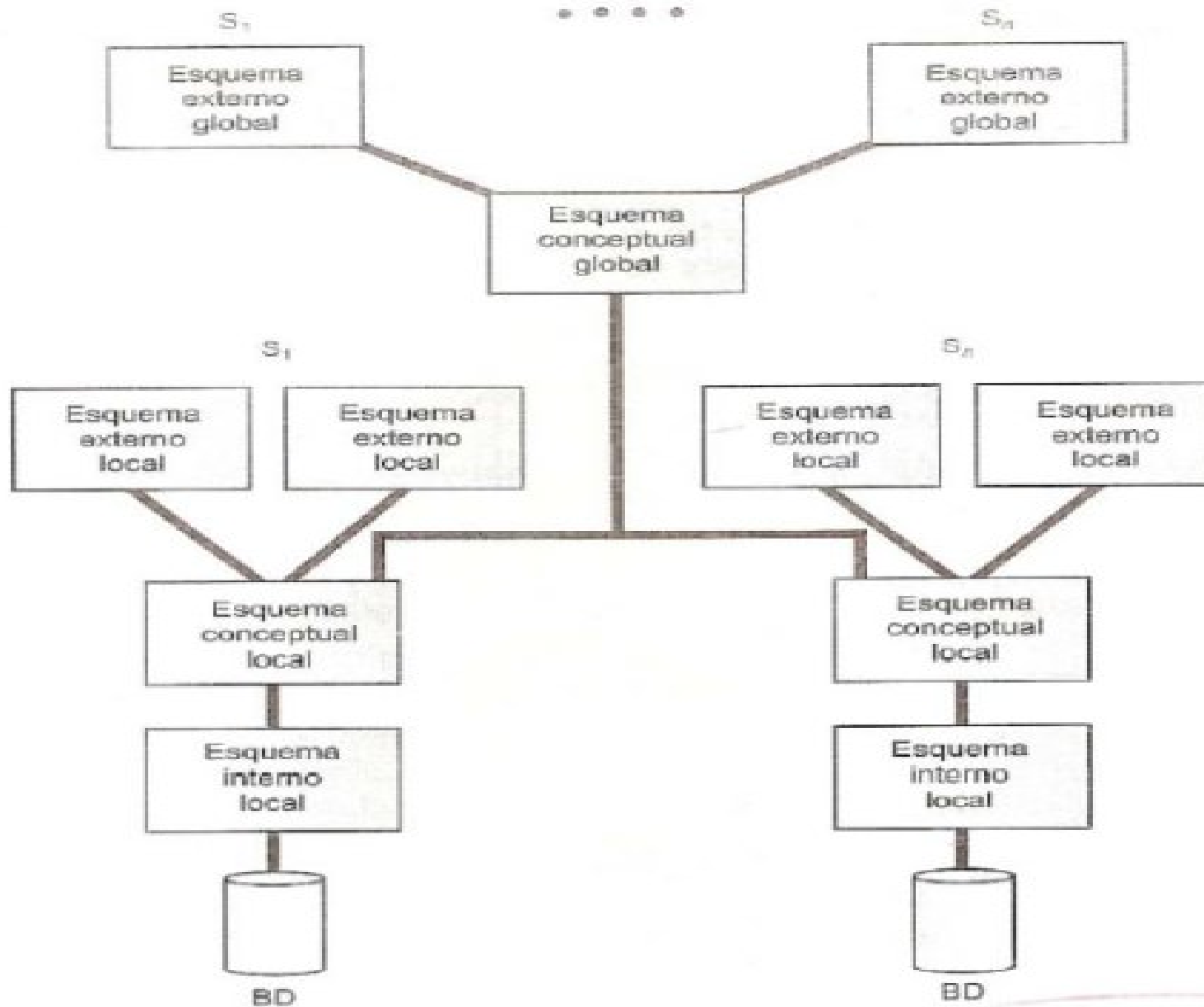
TP chat socket TCP/IP

TP productor/consumidor TCP/IP

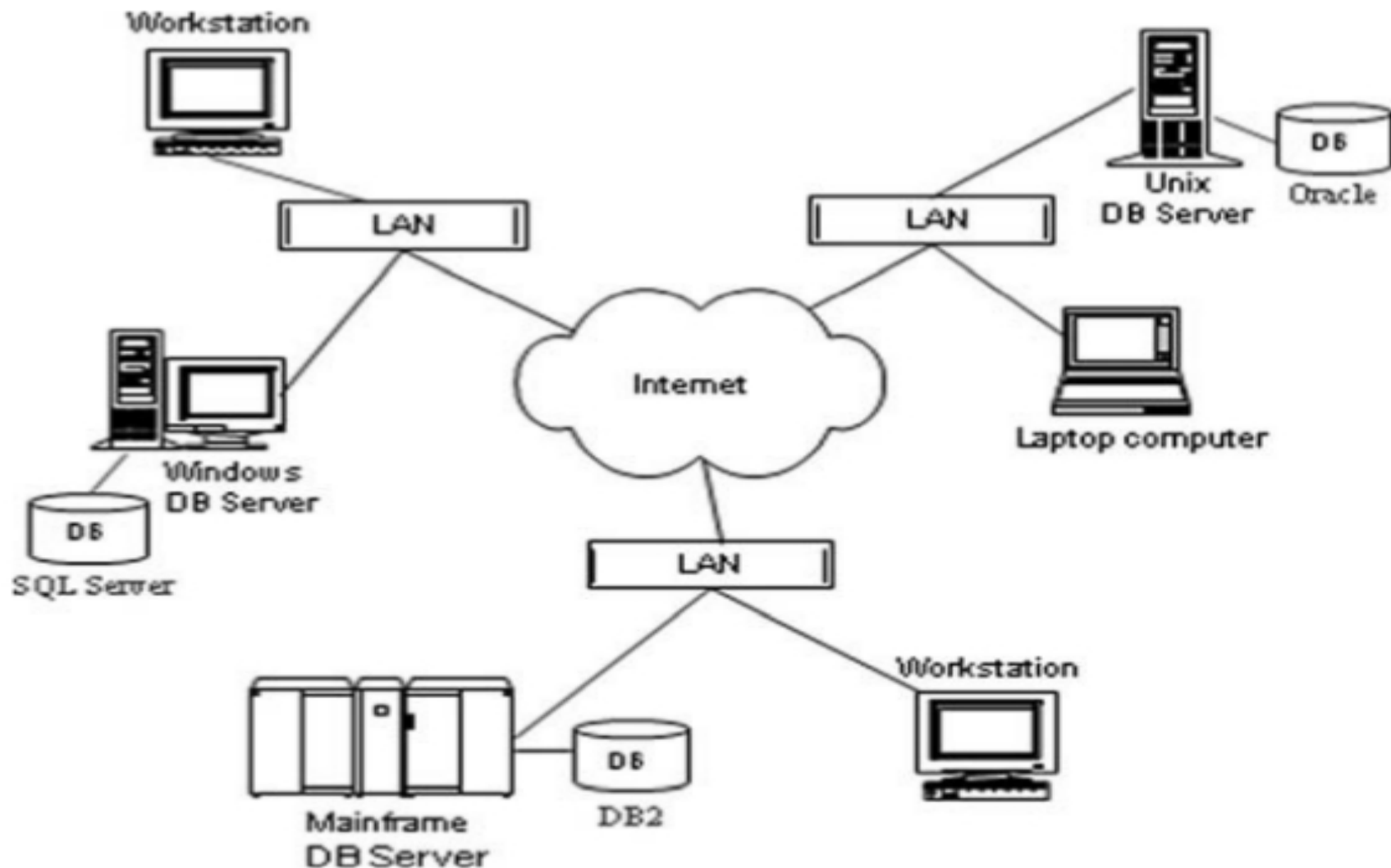
# Arquitecturas CDBE ANSI/SPARC



# Arquitecturas DDBE ANSI/SPARC



# Ejemplo DDBE



**Figure 1.7** Example of the possible deployment of an organization's CDBEs.

# Arq. Generica de Soft. DDBE

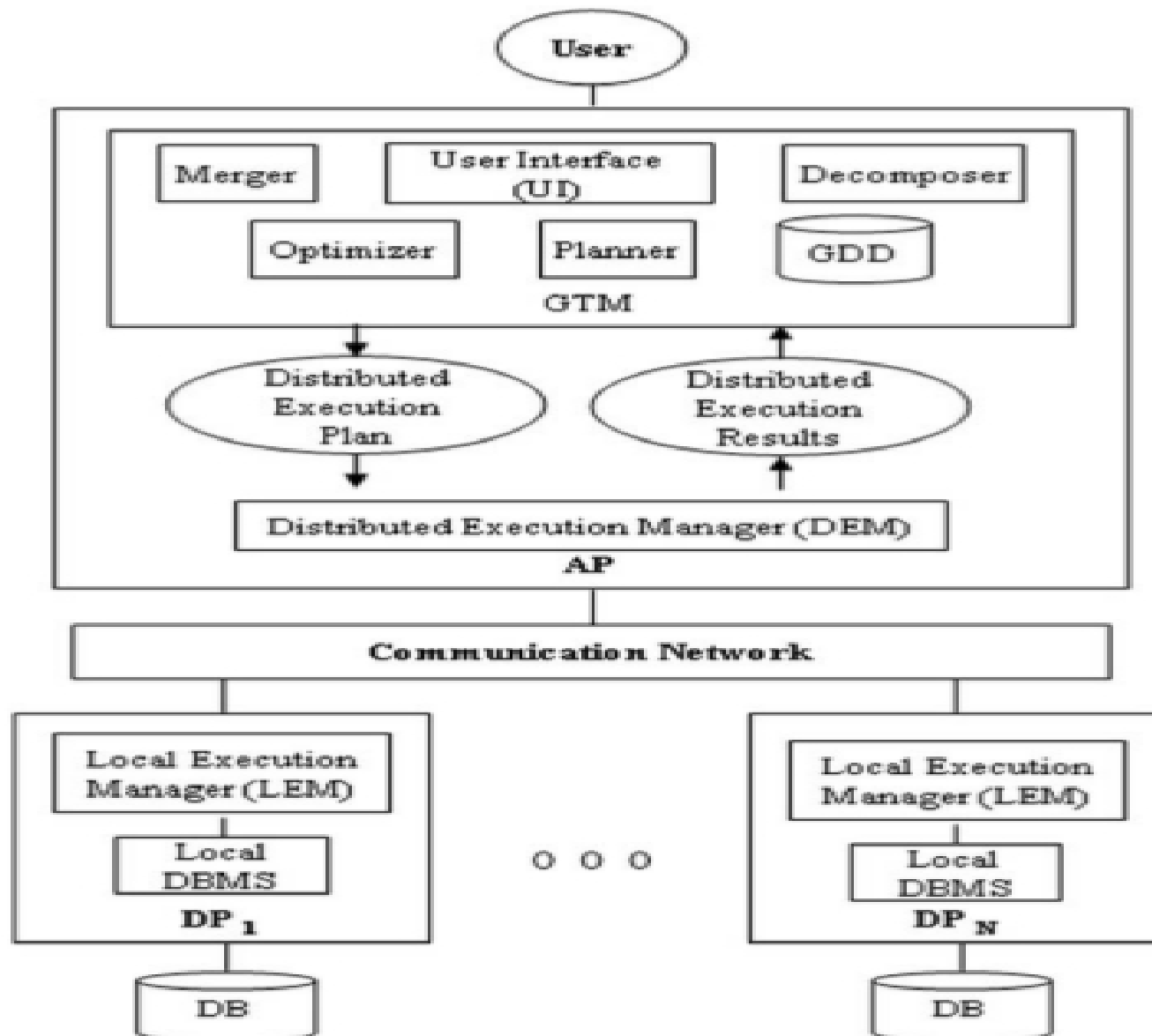


Figure 1.12 Generic DDBE software architecture.



# Transparencia

Grado de ocultamiento de la distribución al usuario final

- Localización (consulta de tablas como si fueran locales)
- Fragmentación (la tabla se percibe como completa y local)
- Replicación (la tabla se percibe como si hubiera una unica copia de la misma)

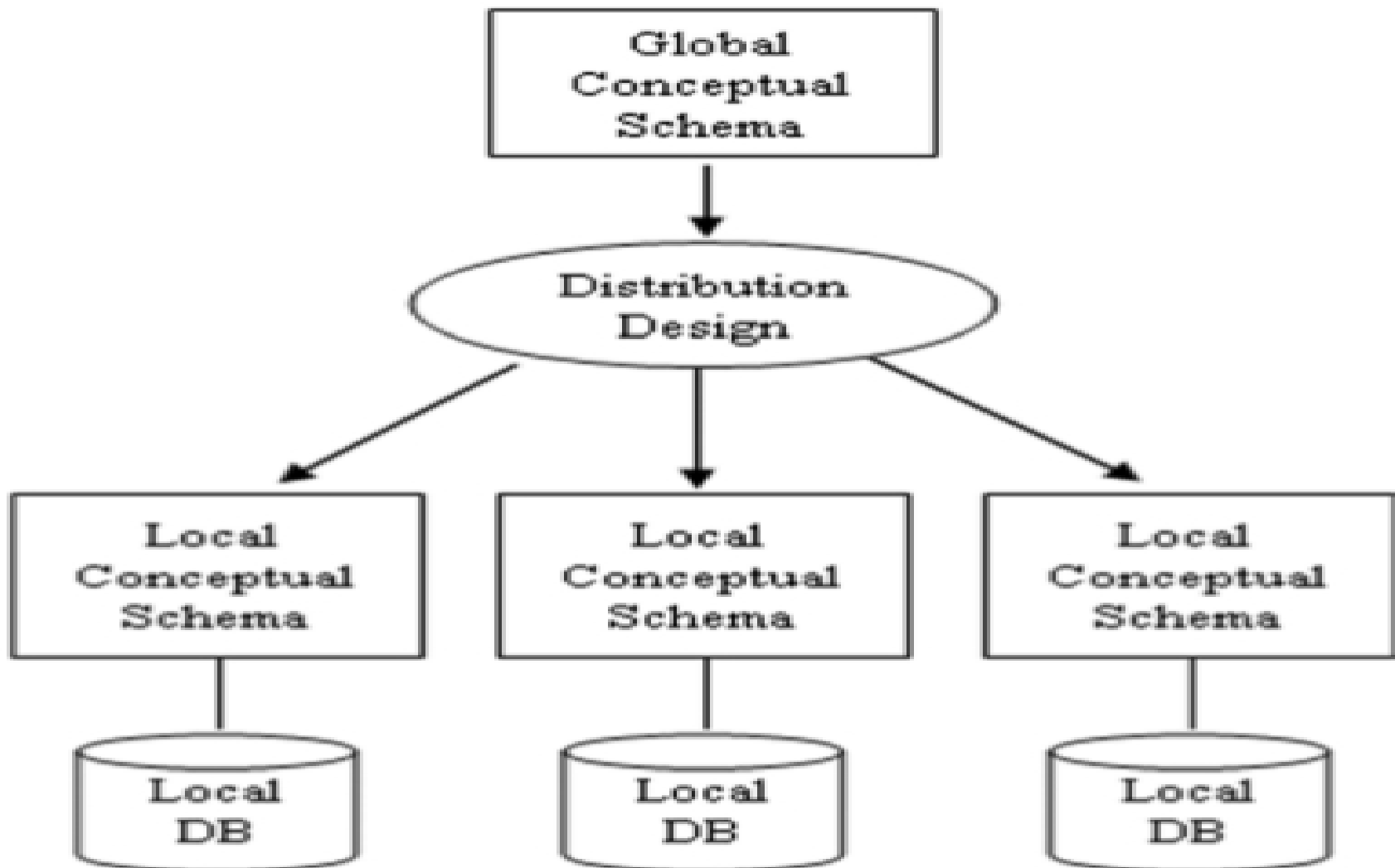
# Diseño – Enfoque Top-Down

- Enfoque Top-Down

Requerimientos->Diseño Conceptual Global->Diseño Fragmentación->Diseño Conceptual Local->Diseño Lógico Local->Diseño Físico Local

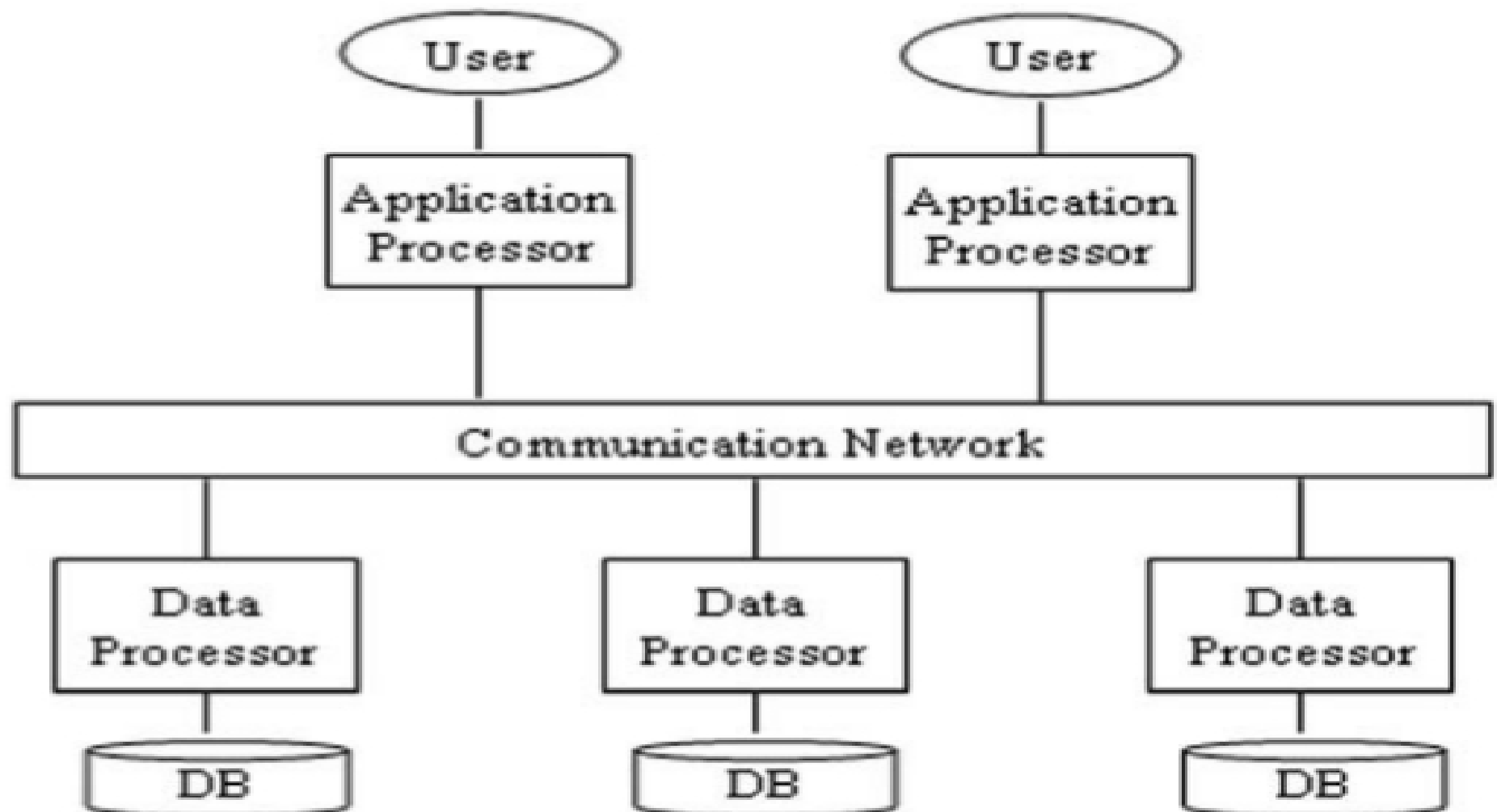
Muy usado para sistemas nuevos, BDD homogéneas

# Diseño – Enfoque Top-Down



**Figure 12.1** Top-down information architecture design.

# Diseño Soft. – Enfoque Top-Down



**Figure 1.10** Top-down DDBE software architecture.

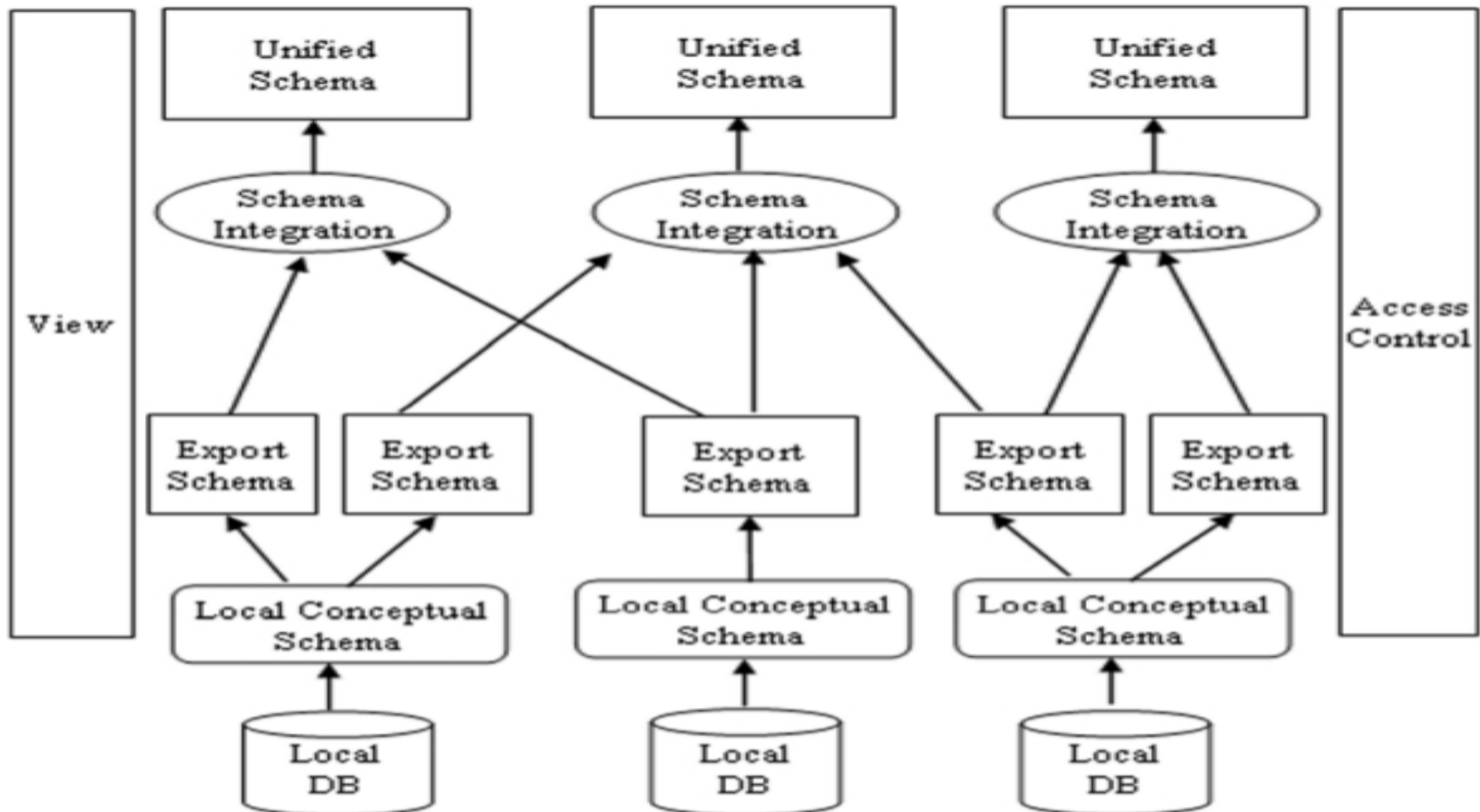
# Diseño – Enfoque Bottom-Up

- Enfoque Bottom-Up

Integrar DBMS's existentes dentro de DDBE.

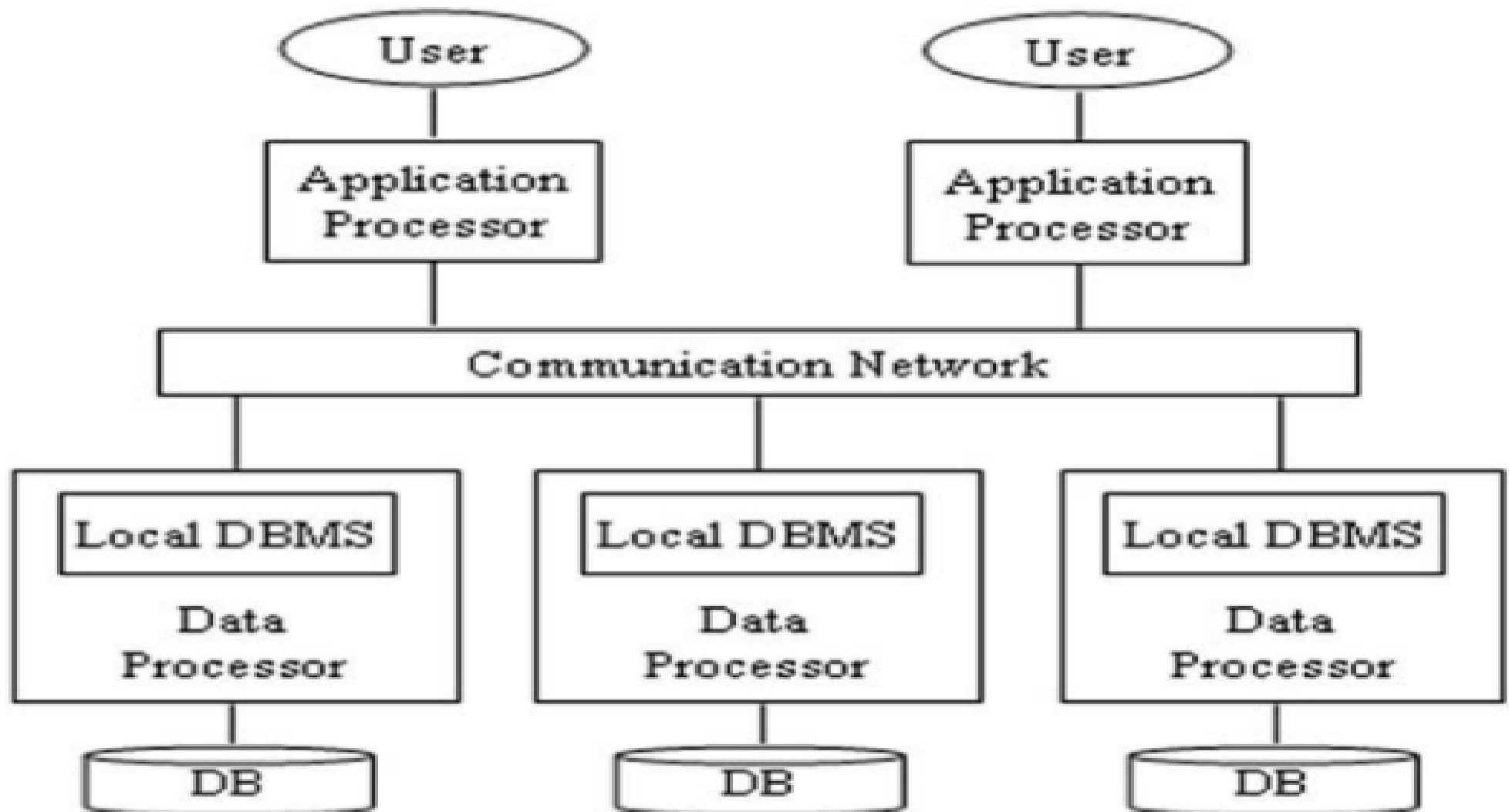
Muy usado para sistemas existentes, BDD heterogéneas, con necesidad de integración.

# Diseño – Bottom-Up



**Figure 12.2** Bottom-up information architecture design.

# Diseño Soft. – Bottom-Up



**Figure 1.11** Bottom-up DDBE software architecture.

# Arquitectura BD Federada (BDF)

DDBE formado por N sub-DBE's

Cada sub-DBE es:

- Autónomo
- Soporta acceso por fuera del DBE
- Puede ser Relacional, Jerárquico, Red, Objetos, etc



# Arquitectura BD No-Federada (BDNF)

DDBE formado por N sub-DBE's

Cada sub-DBE es:

- NO Autónomo
- NO Soporta acceso por fuera del DBE
- Puede ser Relacional, Jerárquico, Red, Objetos, etc

# Arquitectura Multi-Database (SMDB)

DDBE formado por N sub-DBE's

Limitados a ser un DBE centralizado (CDBE)

Pueden ser tanto Federados como No Federados

Ejemplo: BD Ms-Access linkeando tablas de servidor Oracle, SQL Server

# Homogéneo - Heterogéneo

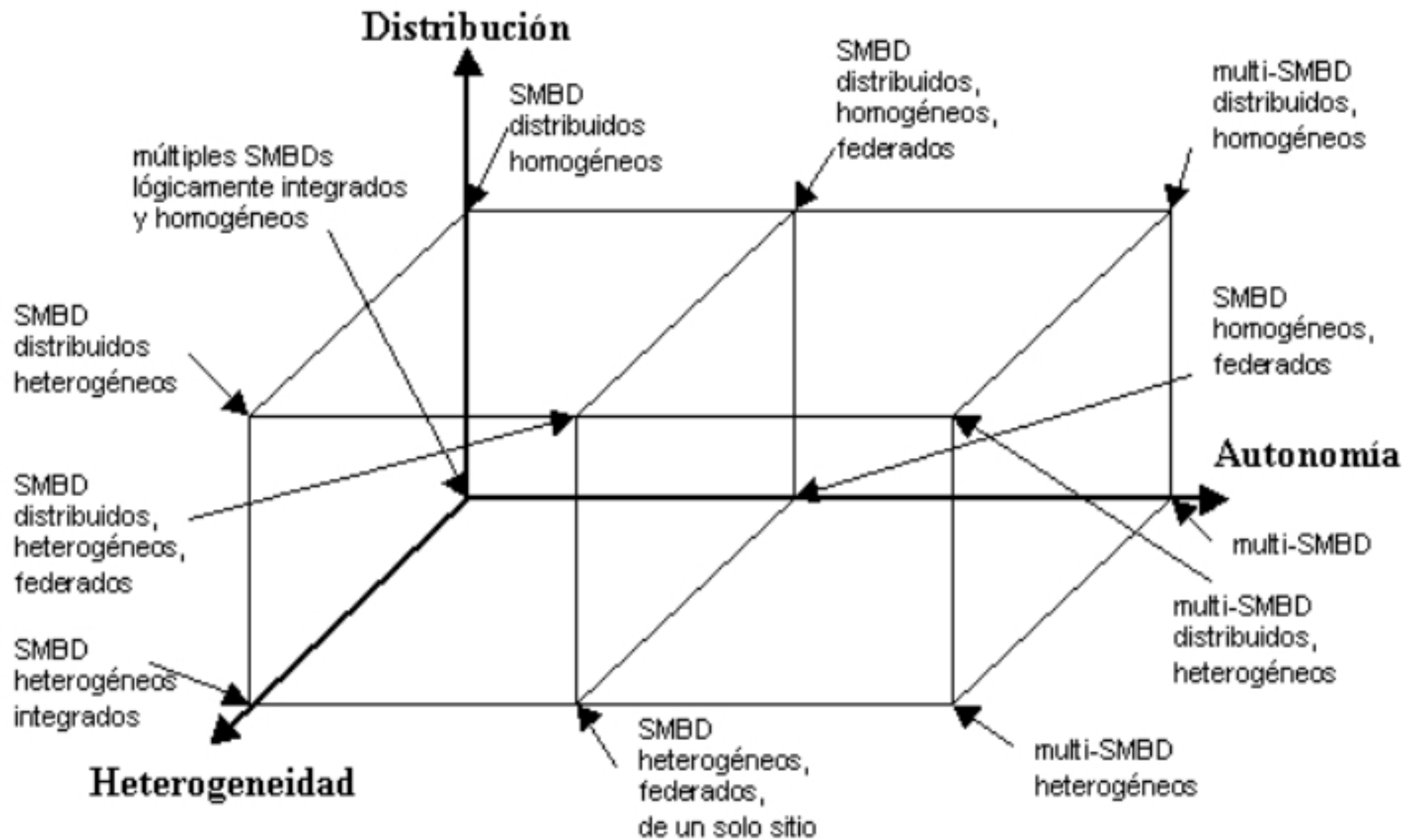
## Homogéneo

- Todos los DBE que componen el DDBE son del mismo tipo: mismo modelo de datos (ML), administrados por el mismo SGBD, mismos protocolos de commit y recuperación.

## Heterogéneo

- Todos los DBE que componen el DDBE son de distinto tipo: distintos modelos de datos (ML), administrados por distintos SGBD, distintos protocolos de commit y recuperación.

# Clasificación SGBDD



# Bibliografía

- [1] M. Tamer Özsu, Patrick Valduriez, “Principles of Distributed Database Systems”, 3rd Ed, Springer, ISBN 978-1-4419-8833-1, 2011
- [2] Saeed K. Rahimi, Frank S. Haug, "Distributed Database Management Systems: A Practical Approach", Wiley-IEEE Computer Society Press., 2010
-