

Tema 2: Arquitectura de bases de datos

Santiago Hernández Ramos

Título propio acreditado por



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH





- La arquitectura de bases de datos se centra en el **diseño**, el **desarrollo**, la **implementación** y el **mantenimiento** de Sistemas de Gestión de Bases de Datos (SGDB)
- El diseño de un SGBD depende de su arquitectura
- Las arquitecturas de bases de datos pueden clasificarse en base a **diferentes criterios**:
 - Arquitecturas mainframe, de escritorio, servidor de archivos, cliente servidor...
 - Arquitecturas de 1 capa, 2 capas, 3 capas, n capas...
 - Arquitecturas de procesamiento y almacenamiento distribuido

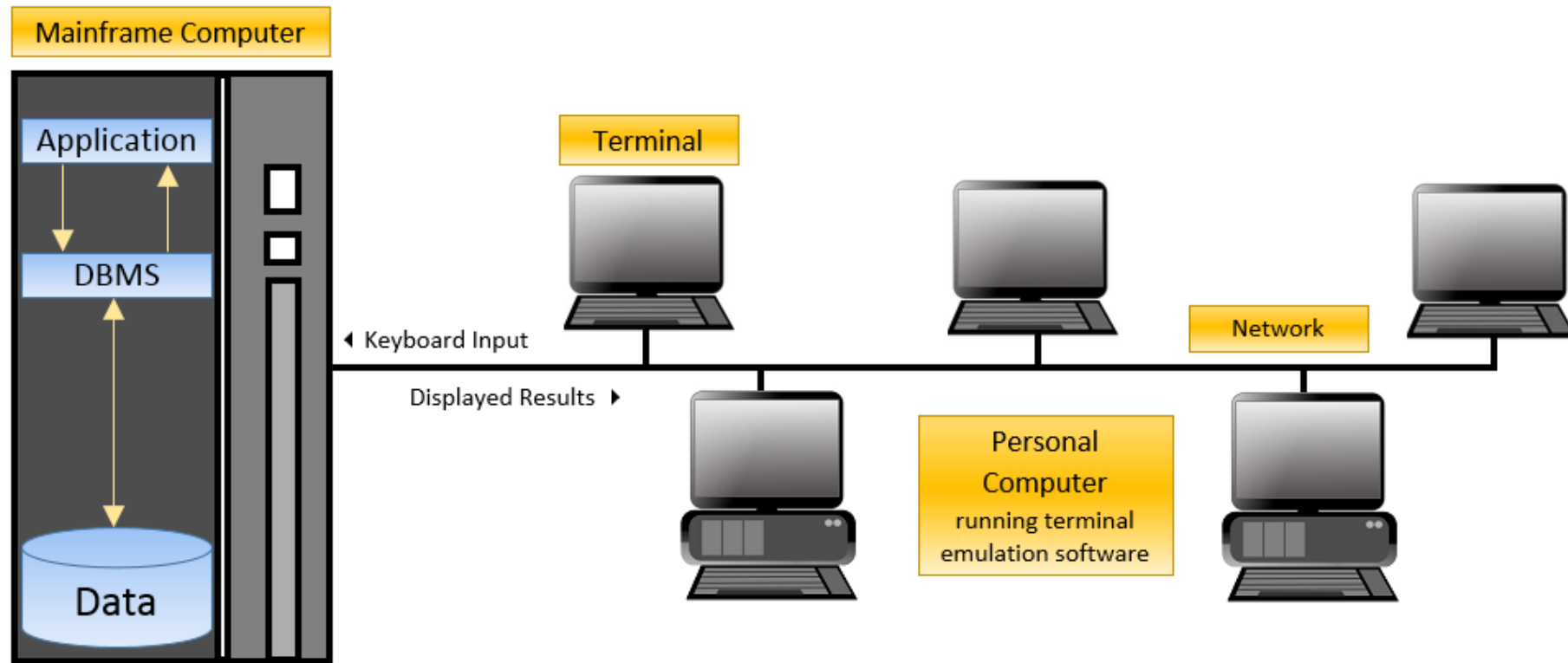
Arquitectura Mainframe





- Se corresponde con **una de las primeras implementaciones de SGDB**
- Los **datos, el SGBD y la aplicación** se encuentran **en un único ordenador central** al que los usuarios acceden a través de terminales
- Los **terminales no tienen capacidad de procesamiento**. Proporcionan una visualización de la aplicación y la entrada de teclado. Aplicación y datos se almacenan y ejecutan en el ordenador central
- **La red sólo interviene para conectar los terminales al ordenador central**. La entrada del teclado se envía desde el terminal al software de aplicación que se ejecuta en el ordenador central. Los resultados mostrados se envían desde el software de aplicación a través de la red a los terminales
- La naturaleza de esta arquitectura **permite cientos, o incluso miles, de usuarios simultáneos**
- Dado que se transfiere un mínimo de datos a través de la red, los mainframes son utilizados por grandes organizaciones para **procesar y almacenar de forma rápida, continua y segura enormes cantidades de datos**. Los bancos utilizan mainframes para gestionar todas sus transacciones

Arquitectura Mainframe





- **Ventajas:**

- Gran número de usuarios simultáneos
- Capaz de manejar un gran volumen de transacciones
- Adecuado para entornos de red lentos
- La ubicación central de la aplicación y del SGBD hace que el mantenimiento sea relativamente sencillo

- **Desventajas:**

- No aprovecha la capacidad de procesamiento del ordenador local
- El hardware es relativamente caro
- Interfaz basada en caracteres

Arquitectura de Escritorio

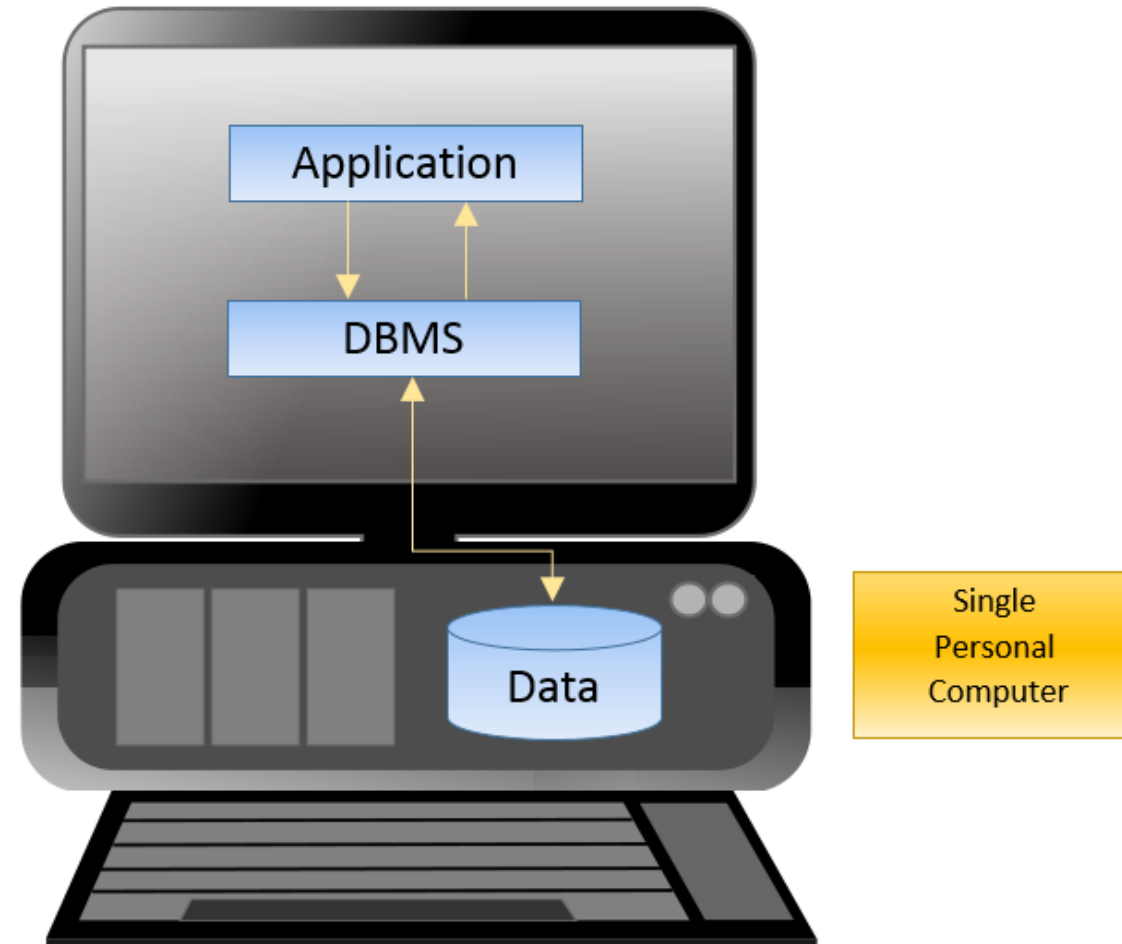




- Es similar a la arquitectura de mainframe en el sentido de que la **aplicación, el SGBD y los datos residen en un único ordenador**
- La principal diferencia es que la arquitectura de escritorio **se limita a un solo usuario** y, por lo tanto, no requiere una red
- Los productos de bases de datos más conocidos que funcionan con esta arquitectura son Microsoft Access, FileMaker Pro, OpenOffice Base y Paradox



- Access almacena los datos en un único archivo con extensión ".accdb"
- Si se crea una base de datos y se le da el nombre de "productos", el archivo se llamará "productos.accdb" en el sistema de archivos
- Todo lo que se ve cuando se trabaja con Access es su software de aplicación. Este software de aplicación se amplía a medida que un desarrollador crea formularios e informes y los aumenta con código personalizado en forma de macros o módulos
- El SGBD en el caso de Access es un programa llamado Microsoft JET Engine
- Microsoft JET Engine se instala como parte de Access y no es visual
- Es posible utilizar las herramientas de aplicación de Access con otros motores de bases de datos. Normalmente, cuando se hace esto se obtiene una arquitectura diferente





- **Ventajas:**

- Económico de adquirir
- Fácil de configurar
- Requiere pocos recursos para funcionar

- **Desventajas:**

- Sólo disponible para un único usuario a la vez
- Sólo disponible en un único ordenador
- Puede imponer límites a la cantidad de datos que se pueden almacenar

Arquitectura de 1 capa





Arquitectura de 1 capa

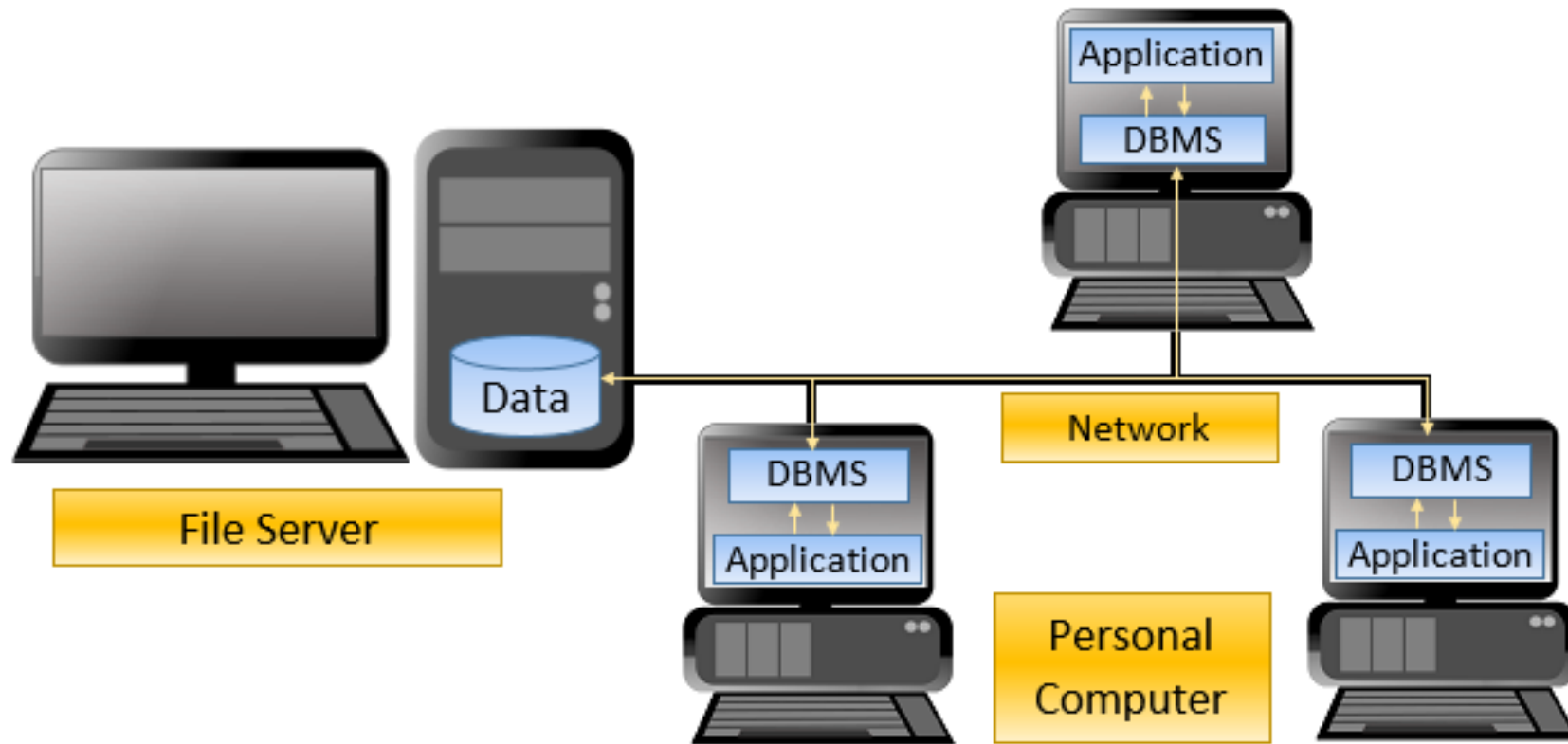
- Tanto la arquitectura de mainframe como de escritorio suelen referenciarse como arquitecturas de 1 capa
- Las arquitecturas de 1 capa mantienen todos los elementos de una aplicación, incluyendo la **interfaz**, el **middleware** y los **datos en un solo lugar**

Arquitectura de servidor de archivos





- La arquitectura de servidor de archivos es una consecuencia de un sistema de base de datos que utiliza la arquitectura de escritorio y necesita ampliarse para dar cabida a múltiples usuarios concurrentes
- La arquitectura de servidor de archivos se caracteriza por **separar el SGBD de los datos mediante la red**
- **El SGBD debe gestionar todos los aspectos de los datos a través de la red.** Las velocidades de transferencia de datos a través de la red son sustancialmente inferiores a las velocidades de transferencia de datos en el disco local, esto introduce una **importante limitación de rendimiento**
- Si otros usuarios que comparten segmentos de red con el usuario de la aplicación de base de datos realizan actividades de uso intensivo de la red, el rendimiento de la aplicación de base de datos será aún más lento
- Cada usuario de la bases de datos tiene que añadir una instancia del SGBD para controlar los datos. **Los SGBD de cada ordenador deben coordinarse entre sí** para mantener la integridad y la seguridad de los datos
- Una instancia del SGBD puede ser incapaz de alertar a otras instancias de que su usuario ha terminado con un recurso de datos concreto por problemas de conectividad de la red o por otros problemas del sistema operativo





- **Ventajas:**

- Económico de adquirir
- Fácil de configurar
- Opción de bajo coste para la concurrencia multiusuario

- **Desventajas:**

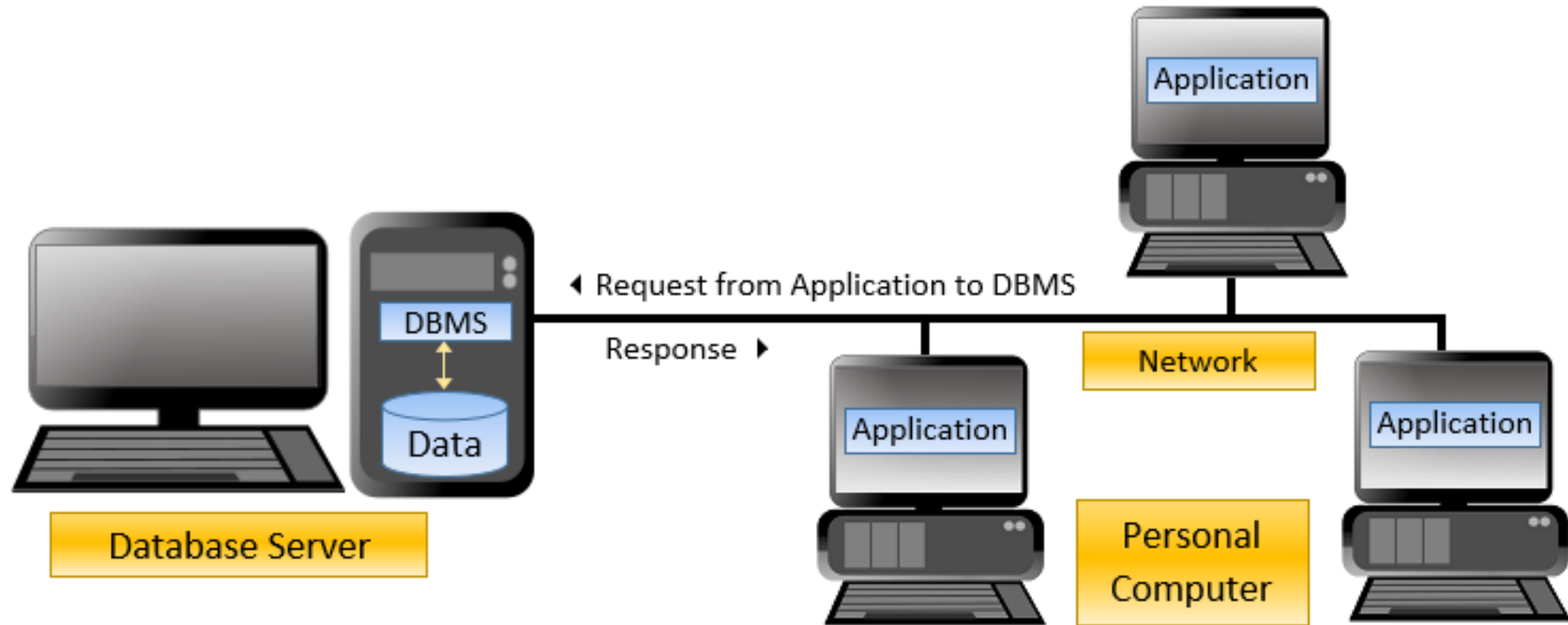
- Sólo es viable para un número limitado de usuarios, quizás 10 o menos
- El rendimiento de las consultas disminuye rápidamente a medida que aumenta la cantidad de datos
- La instalación múltiple del software de aplicación y del SGBD puede provocar problemas de control de versiones.

Arquitectura Cliente-Servidor





- La arquitectura cliente/servidor reúne las mejores características de las arquitecturas anteriores
- Se caracteriza por **ejecutar el SGBD en una máquina con acceso directo a los datos**. Este ordenador se denomina servidor de la base de datos. Varias copias de **la aplicación se ejecutan en ordenadores personales distintos** que se denominan clientes
- Los principales inconvenientes de la arquitectura de servidor de archivos se derivan de tener múltiples instancias del SGBD separadas de los datos por la red. La arquitectura cliente/servidor evita estos problemas al desplegar una única instancia del SGBD con acceso directo a los datos
- Se **aprovecha la potencia de procesamiento del ordenador cliente para ejecutar el software de aplicación**
- La parte más intensiva en recursos de esta arquitectura es el servidor de la base de datos. Si requiere muchos recursos computacionales, suele ser un clúster de máquinas físicas





- **Ventajas:**

- Aprovecha la capacidad de procesamiento de varios ordenadores
- Maneja grandes cantidades de datos y transacciones
- Puede soportar miles de usuarios simultáneos

- **Desventajas:**

- El software de aplicación está distribuido, lo que puede dar lugar a problemas de control de versiones
- El software DBMS es relativamente caro en comparación con el enfoque del servidor de archivos

Arquitectura de 2 capas





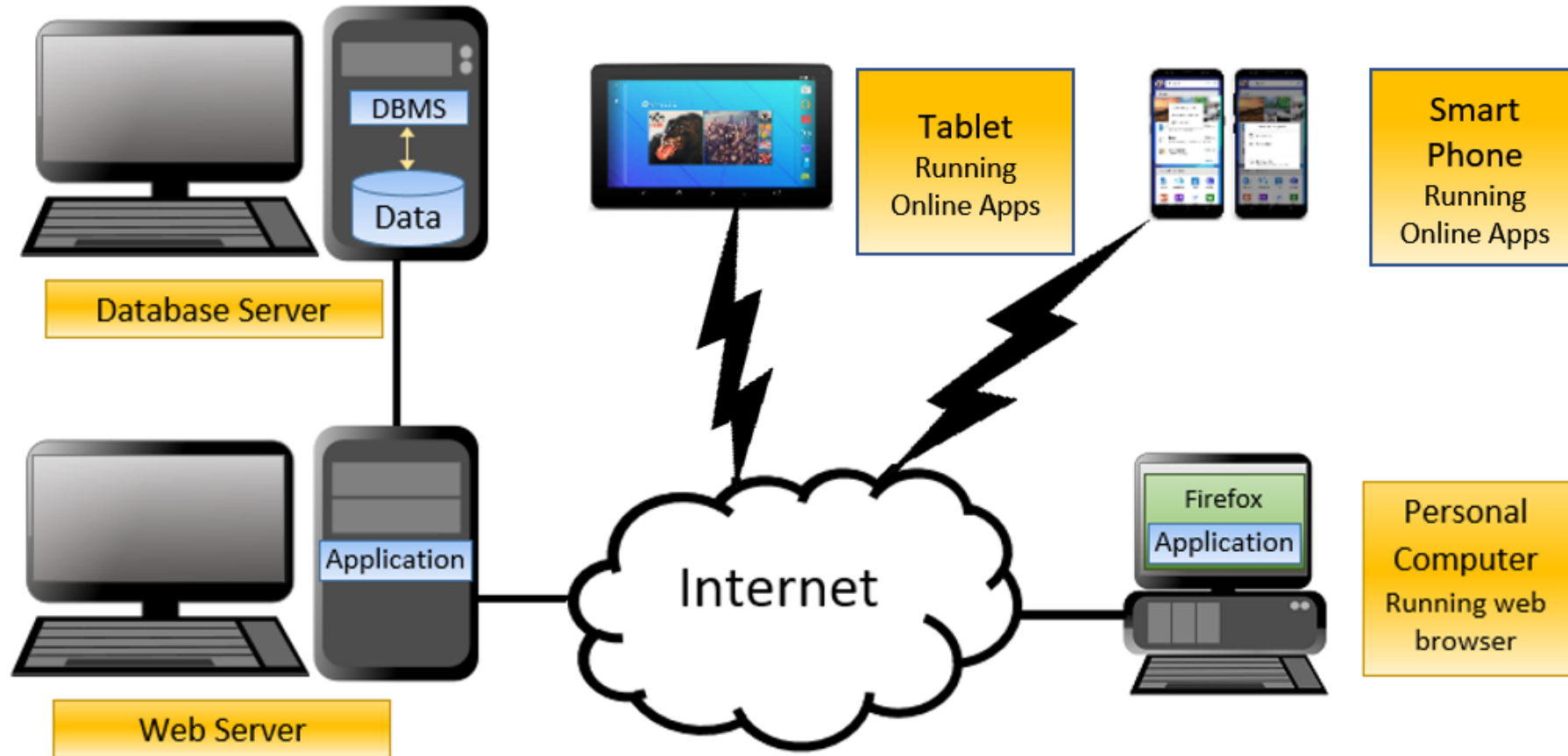
- La **arquitectura cliente-servidor** suele referenciarse de manera común como arquitectura de 2 capas
- Este tipo de arquitecturas se caracterizan porque la **capa de presentación se ejecuta en un cliente** (PC, móvil, tableta, etc.), y **el SGBD y los datos se almacenan en un servidor** que representa la capa 2
- La arquitectura de dos niveles proporciona una **mayor seguridad al SGBD**, ya que no está expuesto al usuario final directamente

Arquitectura Web





- La arquitectura web es una extensión de la arquitectura cliente-servidor. Es la **arquitectura más utilizada en la actualidad**
- El **software de aplicación se ejecuta en un servidor web** que se conecta al SGBD a través de una conexión de red (el servidor de base de datos y el servidor web pueden estar en la misma máquina física).
- Lo que se muestra al usuario final (cliente) se genera en el servidor web y se envía a través de Internet en el formato adecuado para la aplicación receptora
- **La aplicación receptora del cliente suele ser un navegador web.** Antiguamente, el servidor web solía generar una simple página web estática para su visualización. En la actualidad se delega más trabajo en el navegador con páginas web dinámicas y asíncronas, y menos en el servidor web
- Se suelen utilizar técnicas de balanceo de carga para permitir que la aplicación se ejecute en varios servidores web, cada uno de ellos conectado al servidor de base de datos. Esto evita que un solo servidor web se convierta en el cuello de botella en el proceso
- La parte de la aplicación que se ejecuta en el navegador se envía a la máquina del usuario cada vez que éste actualiza la página. Esto hace que el versionado de la aplicación sea más manejable que con las implementaciones tradicionales de cliente-servidor





- **Ventajas:**

- Todas las ventajas de la arquitectura cliente/servidor
- Mayor facilidad para el control de las versiones de las aplicaciones
- Muy buena gestión del uso intermitente de un gran número de usuarios (millones)

- **Desventajas:**

- Añade otra capa de complejidad (servidor web)
- Varias aplicaciones de software deben mantenerse sincronizadas con el software de la base de datos
- Las actualizaciones de los navegadores pueden requerir cambios en la aplicación

Arquitectura de 3 capas





Arquitectura de 3 capas

- La arquitectura web suele referenciarse de manera común como arquitectura de 3 capas
- Se corresponde con una arquitectura cliente-servidor en la que **el desarrollo y el mantenimiento de los procesos funcionales, la lógica, el acceso a los datos, el almacenamiento de datos y la interfaz de usuario son módulos separados**
- Una arquitectura de 3 niveles tiene las siguientes capas:
 - **Capa de base de datos (datos)** - En este nivel reside la base de datos junto con sus lenguajes de procesamiento de consultas
 - **Capa de aplicación (lógica de negocio)** - En este nivel residen el servidor de aplicación y los programas que acceden a la base de datos. La capa de aplicación actúa como mediador entre el usuario final y la base de datos, también procesa la lógica funcional, las restricciones y las reglas antes de pasar los datos al usuario o al SGBD
 - **Capa de usuario (presentación)** - Los usuarios finales operan en este nivel y no conocen la existencia de la base de datos más allá de este nivel. En esta capa, la aplicación puede proporcionar múltiples vistas de la base de datos
- El objetivo de la arquitectura de 3 capas es:
 - Separar las aplicaciones del usuario y la base de datos física
 - Independencia de los datos y del programa
 - Soportar múltiples vistas de los datos

DBMS Distribuido (DDBMS)





Sistema de Gestión de Base de Datos Distribuido (DDBMS)

- Un sistema DDBMS gobierna el almacenamiento y procesamiento de datos en sistemas informáticos interconectados en los que tanto **los datos como el procesamiento están distribuidos entre varios lugares**
- Durante la década de 1970, las empresas implementaron DBMS tradicionales para satisfacer sus necesidades de información estructurada
- El uso de una base de datos centralizada requería que los datos corporativos se almacenaran en un único lugar, normalmente un ordenador central
- Las **dos últimas décadas** han dado lugar cambios sociales y tecnológicos cruciales que han afectado a la naturaleza de los datos y a las necesidades de acceso a los mismos:
 - El **acceso rápido a los datos** se ha convertido en algo crucial en el entorno de toma de decisiones de respuesta rápida
 - El acceso a los datos distribuidos es necesario para dar **soporte a unidades de negocio geográficamente dispersas**



- La **gestión centralizada** de las bases de datos está sujeta a problemas como:
 - **Degradación del rendimiento** debido a un número creciente de ubicaciones remotas a mayores distancias
 - **Altos costes asociados al mantenimiento** y funcionamiento de grandes sistemas de bases de datos centrales (mainframe)
 - **Problemas de fiabilidad** creados por la dependencia de un sitio central (síndrome del punto único de fallo) y la necesidad de replicar los datos
 - **Problemas de escalabilidad** asociados a los límites físicos impuestos por una única ubicación, como el espacio físico, el acondicionamiento de la temperatura y el consumo de energía
- La **rigidez organizativa** impuesta por la base de datos, lo que significa que podría no soportar la flexibilidad y agilidad que requieren las organizaciones globales modernas

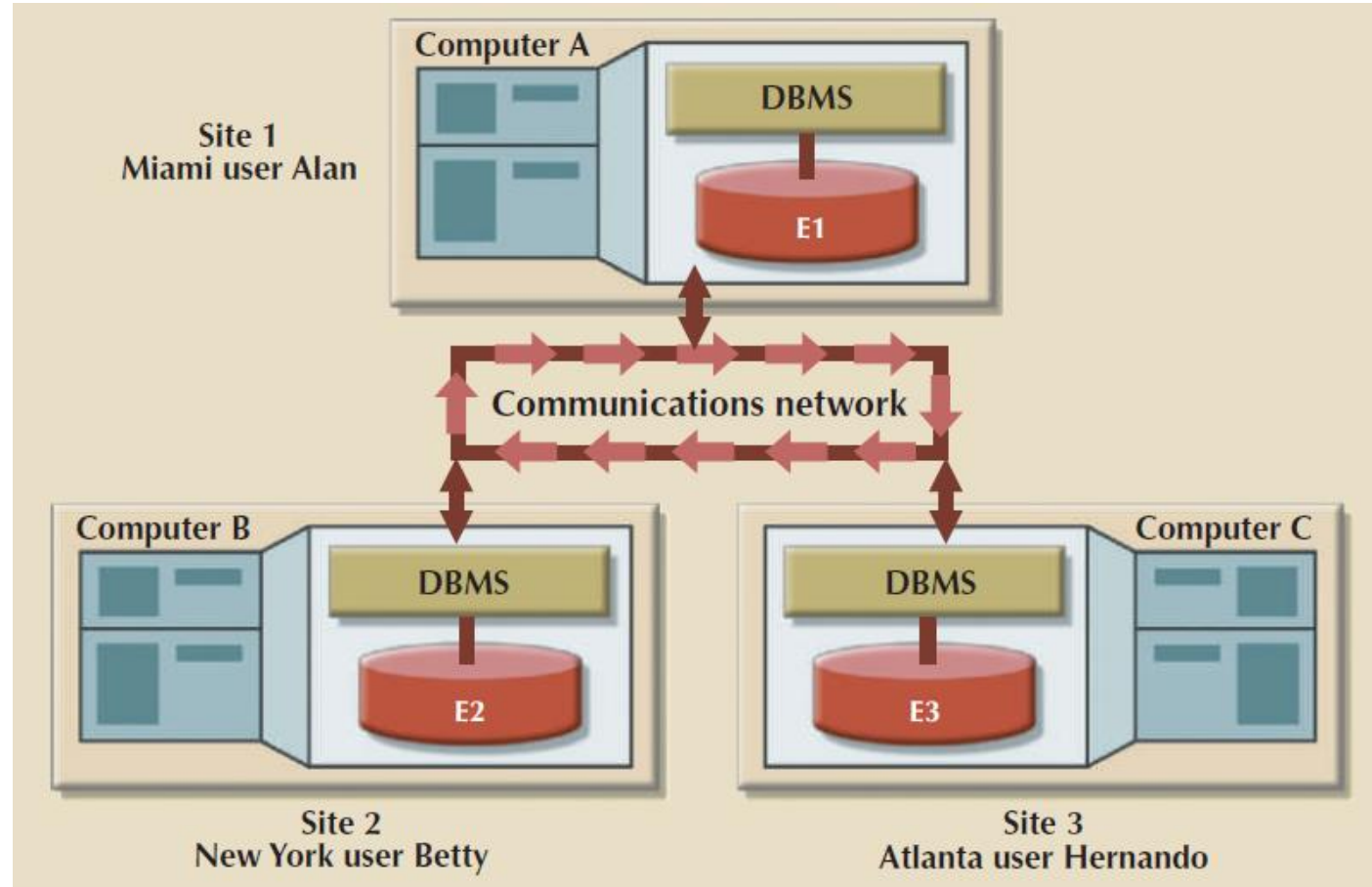


- En el procesamiento distribuido, el **procesamiento lógico de una base de datos se comparte entre dos o más sitios físicamente independientes** que están conectados a través de una red
- La entrada/salida de datos (I/O), la selección de datos y la validación de datos pueden realizarse en un sistema informático y un informe basado en esos datos puede crearse en otro sistema diferente
- Aunque la base de datos reside en un solo lugar, cada sistema distribuido puede acceder a los datos y actualizar la base de datos
- La base de datos se encuentra en un ordenador de red conocido como servidor de base de datos



- Un sistema de procesamiento distribuido utiliza sólo una base de datos en un único lugar pero comparte las tareas de procesamiento entre varios sitios
- En un sistema de base de datos **distribuida comparte una base de datos entre varios lugares**. Cada una de las partes se denomina **fragmento de base de datos**
- Los fragmentos de la base de datos **están ubicados en diferentes sitios** y pueden ser replicados entre ellos.
- Cada fragmento de base de datos es **gestionado por su proceso de base de datos local**
- El procesamiento distribuido no requiere una base de datos distribuida, pero una base de datos distribuida requiere un procesamiento distribuido.

Base de datos distribuida



Componentes de una base de datos distribuida



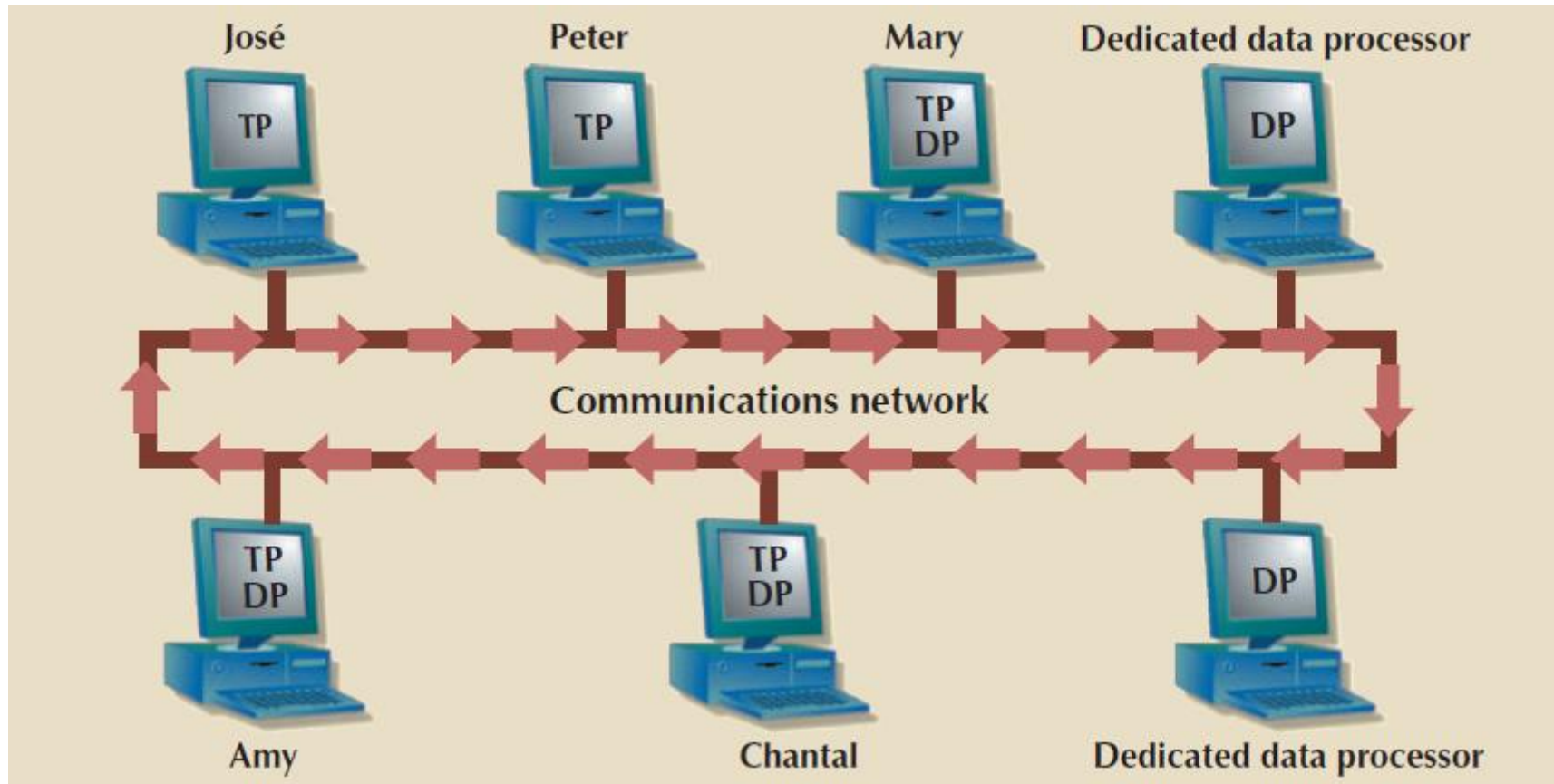
- El DDBMS debe incluir al menos los siguientes componentes:
- **Estaciones de trabajo** o dispositivos remotos (sitios o nodos) que forman la red
- **Componentes de hardware y software de red** que residen en cada estación de trabajo o dispositivo. Estos componentes permiten que todos los nodos interactúen e intercambien datos
- **Medios de comunicación** que permiten intercambiar datos entre los nodos. El DDBMS debe ser independiente de los medios de comunicación
- **El procesador de transacciones (TP)** es el componente de software que se encuentra en cada ordenador o dispositivo que solicita datos. El procesador de transacciones **recibe y procesa las solicitudes de datos remotas y locales de la aplicación**. El TP también se conoce como procesador de aplicaciones (AP) o gestor de transacciones (TM).
- El **procesador de datos (DP)** es el componente de software que reside en cada ordenador o dispositivo que **almacena y recupera los datos ubicados en ese lugar**. El DP también se conoce como gestor de datos (DM). Un procesador de datos puede ser incluso un SGBD centralizado.

Componentes de una base de datos distribuida



- Los protocolos determinan el funcionamiento del sistema de base de datos distribuida:
 - Interactuar con la red para transportar datos y comandos entre los DP y los TP
 - Sincronizar todos los datos recibidos de los DP (lado TP) y dirigir los datos recuperados a los TPs apropiados (lado DP)
 - Garantizar las funciones comunes de la base de datos en un sistema distribuido. Estas funciones incluyen la seguridad de los datos, la gestión de las transacciones y el control de la concurrencia y sincronización, y copia de seguridad y recuperación de datos
- Los DP y TP deben añadirse al sistema de forma transparente sin afectar a su funcionamiento
- Un TP y un DP pueden residir en el mismo ordenador, permitiendo al usuario final acceder tanto a los datos locales como a los datos remotos de forma transparente
- En teoría, una DP puede ser un SGBD centralizado independiente con interfaces adecuadas para soportar el acceso remoto desde otros SGBD independientes en la red

Componentes de una base de datos distribuida



Clasificación de los sistemas de almacenamiento y procesamiento distribuido

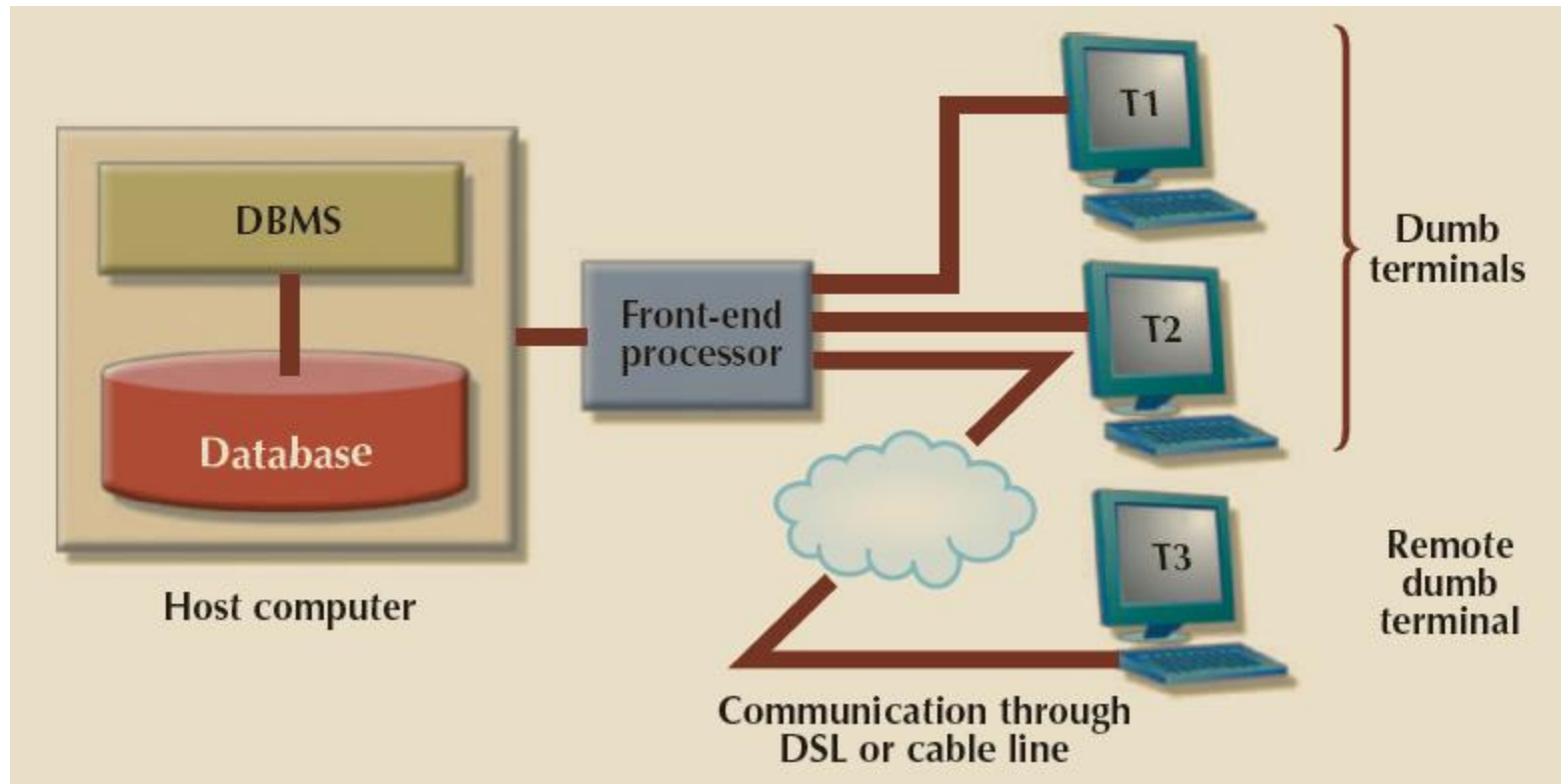
10





- En el escenario de **procesamiento en un solo sitio, datos en un solo sitio (SPSD)**, todo el procesamiento se realiza en un único ordenador central, y todos los datos se almacenan en el sistema de disco local del ordenador central
- El procesamiento no puede realizarse en el lado del usuario final del sistema. Este escenario es típico de la mayoría de los SGBD que implementan arquitectura mainframe
- El SGBD está en el ordenador central al que se accede mediante terminales conectados a él

Single-Site Processing, Single-Site Data

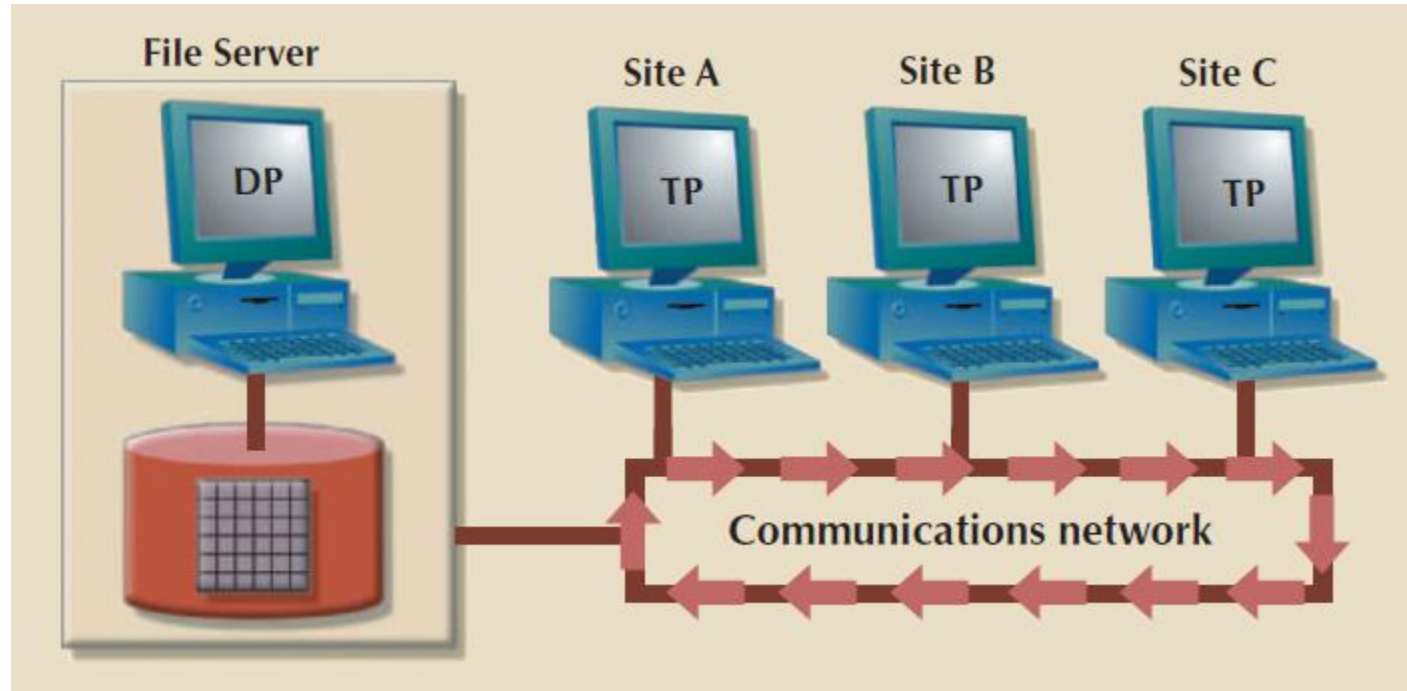


Multiple-Site Processing, Single-Site Data



- En el escenario de **procesamiento en varios sitios, datos en un solo sitio (MPSD)**, varios procesos se ejecutan en diferentes ordenadores que comparten un único repositorio de datos
- Normalmente, el escenario MPSD requiere un servidor de archivos en red
- El TP en cada estación de trabajo redirige las solicitudes de datos al servidor de archivos
- El usuario final ve el servidor de archivos como un disco duro más porque solo maneja la entrada/salida de datos
- El usuario final debe hacer una referencia directa al servidor de archivos para acceder a los datos remotos
- Todas las actividades de bloqueo de registros y archivos se realizan en la ubicación del usuario final
- Todas las funciones de selección, búsqueda y actualización de datos tienen lugar en la estación de trabajo, por lo que los archivos completos viajan a través de la red para ser procesados en la estación de trabajo
- Este requisito aumenta el tráfico de la red, ralentiza el tiempo de respuesta y aumenta los costes de comunicación

Single-Site Processing, Single-Site Data

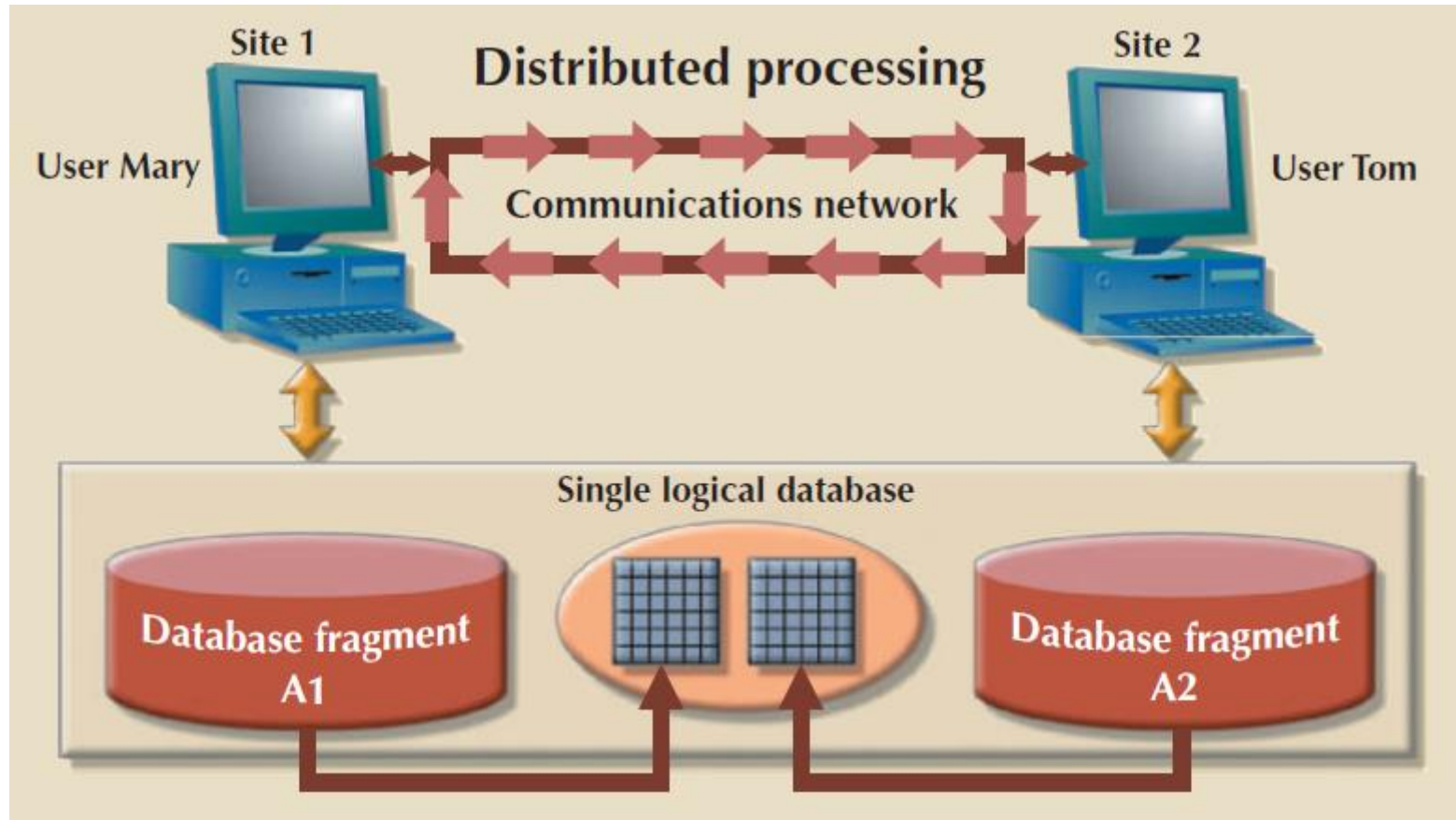




Multiple-Site Processing, Multiple-Site Data

- El escenario **de procesamiento en múltiples sitios, datos en múltiples sitios (MPMD)** describe un DBMS completamente distribuido con soporte para múltiples procesadores de datos y procesadores de transacciones en múltiples sitios
- Dependiendo del nivel de soporte para varios tipos de bases de datos, los SGBD se clasifican en homogéneos o heterogéneos
- Los DDBMS homogéneos integran múltiples instancias del mismo SGBD a través de una red (por ejemplo, varias instancias de Oracle 11g que se ejecutan en diferentes plataformas)
- Los DDBMS heterogéneos integran diferentes tipos de DBMS en una red, pero todos soportan el mismo modelo de datos. Un DDBMS totalmente heterogéneo soportará diferentes SGBD, cada uno de los cuales soportará un modelo de datos diferente, ejecutándose en diferentes sistemas informáticos
- Las implementaciones de bases de datos distribuidas se entienden mejor como una capa de abstracción sobre un SGBD. Esta capa de abstracción proporciona una funcionalidad adicional que permite soporte para diferentes características como replicación, fragmentación avanzada de datos, sincronización e integración

Single-Site Processing, Single-Site Data



Fragmentación vs Replicación vs Asignación

11





Fragmentación vs Replicación vs Asignación

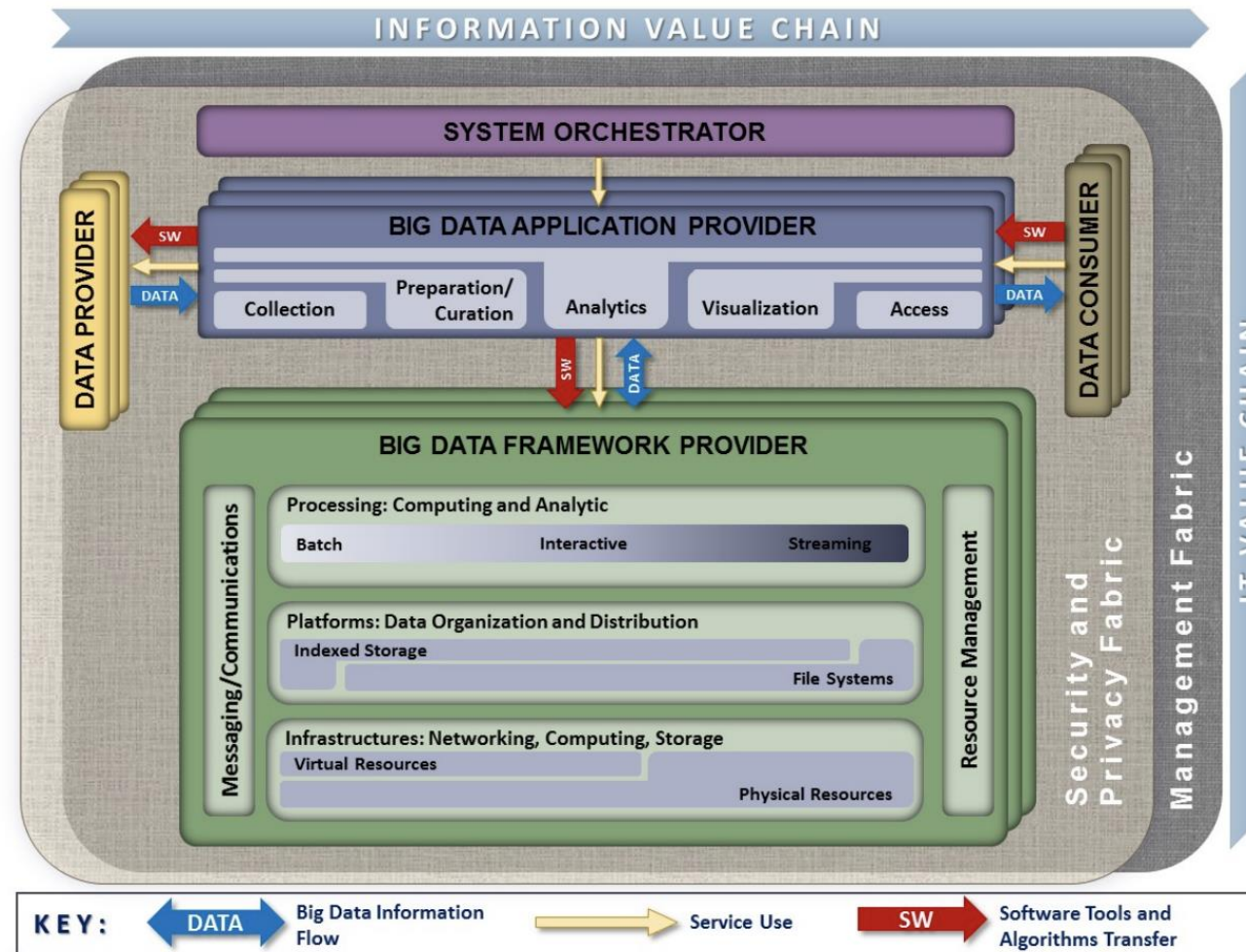
- La **fragmentación de datos** permite **dividir un único objeto en dos o más segmentos o fragmentos**. El objeto puede ser una base de datos, una tabla o un registro. Cada fragmento puede almacenarse en cualquier sitio de una red informática. La información sobre la fragmentación de los datos se almacena en el catálogo de datos distribuido (DDC), que es accedido por el TP para procesar las solicitudes de los usuarios.
- La **replicación de datos** se refiere al **almacenamiento de diferentes copias del mismo objeto en diferentes sitios de una red informática**. Permite mejorar la disponibilidad de los datos y el tiempo de respuesta en el acceso a los mismo.
- La **asignación de datos describe el proceso de decidir dónde ubicar los datos**. Las estrategias de asignación de datos son las siguientes:
 - Asignación de datos centralizada: toda la base de datos se almacena en un sitio.
 - Asignación de datos particionados: la base de datos se divide en dos o más fragmentos y se almacena en dos o más sitios.
 - Asignación de datos replicada: las copias de uno o más fragmentos de la base de datos se almacenan en varios sitios.

Big Data: Arquitectura de Referencia

12



Big Data: Arquitectura de Referencia



<https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1500-6r2.pdf>



Título propio acreditado por



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH