

# Bases de datos hoy

Bruno Crespo - Trustserver SL  
Mayo 2025



# ¿Quién soy?

- Ingeniero Industrial
- Administrando UNIX desde el año 1991
- Administrando Linux desde el año 1992
- Administrando Oracle desde el año 2000
- Experiencia en distintos sectores: Investigación científica, Telecomunicaciones, Internet, Automóvil, Industria, Distribución, Editorial...
- Desarrollo: Cálculo intensivo, CAD, portales internet, vídeo bajo demanda, big data, sistemas empotrados...
- Sistemas: UNIX/Linux, redes y comunicaciones, almacenamiento, bases de datos, bases de datos, bases de datos.
- Profesional independiente desde el año 2002



# Un trabajo del cole

- Son las nueve de la noche del domingo, tu hijo te dice que necesita un libro para hacer un trabajo.
- Te conectas a la web de un megadistribuidor y le das a comprar.
- El martes a las 11:17 de la mañana un repartidor te entrega el libro.
- ¿Cómo lo han conseguido?



# Logística en acción

- La web de venta de libros te presenta los datos del libro sacados de una BBDD NoSQL MongoDB.
- Cuando le das a comprar se lanza un servicio web contra la web del banco, ejecutando la transacción en una BBDD DB/2 sobre Mainframe.
- La web de venta guarda la transacción en una BBDD PostgreSQL.
- A través de otro servicio web el pedido pasa a la BBDD Oracle de la editorial.
- El ERP de la Editorial informa al Sistema de Gestión de Almacén, sobre una base de datos SQLServer, que hay que recoger el libro y pasarlo al mensajero.
- El ERP de la Editorial, a través de otro servicio web, informa al ERP de la empresa de mensajería, sobre BBDD Oracle, que debe recoger el paquete y recibe un número de seguimiento.
- A las 3:00 am del martes el camión recoge los libros en la Editorial y la transacción queda recogida en las BBDDs Oracle del mensajero y la Editorial.
- Los ERP de la Editorial y el mensajero, sobre Oracle, transmiten la recogida a la BBDD PostgreSQL de la web de venta para que te avise.
- El paquete es seguido por su camino en camión y avión a través de media España a través de la BBDD Oracle del mensajero.
- El paquete se te entrega y el Oracle del mensajero avisa al Oracle de la Editorial y el PostgreSQL de la web de venta.



# Logística en acción

- Gracias a PostgreSQL, Oracle, SQLServer, DB2 y MongoDB tu hijo podrá hacer el trabajo.



# Nuestra vida cotidiana

- Casi cualquier actividad en nuestra vida utiliza una base de datos, y casi siempre es relacional:
  - Comprar en el supermercado.
  - Sacar dinero en un cajero.
  - Consultar los contactos en el móvil.
  - Consultar el saldo de la tarjeta del móvil.
  - Mirar nuestra cuenta de Twitter.
  - Leer lwn.net.
  - Mirar el saldo de puntos del carnet de conducir.
  - Pedir un taxi.
  - Concertar una cita médica.
  - Ver un partido de futbol en pay-per-view.
  - Pujar por algo en e-bay.
  - Recibir un paquete por mensajero.
  - Domiciliar un recibo en el banco.
  - Llamar a emergencias.
  - Beberte una cerveza.



# Bases de datos en la empresa

- En general cualquier sistema empresarial consiste en mover registros en una base de datos u obtener informes sobre estos registros.
- Mover registros: Sistema transaccional o CRUD: CReate, Update, Delete.
- Obtener informes: Data Warehouse.
- El 100% de los desarrolladores que conozco profesionalmente dedican el 100% de su tiempo a desarrollo sobre bases de datos.
- Podéis estar seguros que el 90% de vosotros trabajará toda su vida profesional administrando, instalando o desarrollando sobre bases de datos.



# Sistemas RDBMS: Transaccional

- Oracle: La mejor.
- PostgreSQL: La mejor OpenSource.
- MySQL: Un juguete.
- DB2: La más veterana.
- MS-SQL Server: Mejorando.
- MS-Access: Ni se considera.
- SQLite: La empotrada SQL más popular.
- BerkeleyDB: La empotrada más popular.
- Otras: Sybase, Informix, SAP Hana...



# Sistemas RDBMS: Data Warehouse

- Casi todas las bases de datos transaccionales tienen soporte para Data Warehouse pequeños (< 10TB y tablas con unos pocos miles de millones de filas)
- Soluciones clásicas: Teradata, PureData (Netezza), Vertica... Salvo Teradata casi todas derivadas se PostgreSQL
- Actual: Soluciones basadas en la nube como Snowflake, Google BigQuery, Azure Synapse, Amazon Redshift, IBM DB2 Warehouse, Firebolt...
- Las soluciones clásicas eran evoluciones de transaccionales, las modernas basadas en NoSQL e integración con motores estadísticos.



# Bases de datos: NoSQL

- En los últimos años se han popularizado las bases de datos NoSQL
- Estas bases de datos son rápidas y ligeras, muy útiles para ciertos tipos de cargas.
- Además escalan muy bien en cluster horizontales de tipo „share nothing“.
- A cambio eran un desastre de integridad: no soportaban transacciones, su capa de seguridad era una broma y sus características de backup y recuperación eran muy limitadas.
- Actualmente han corregido muchas de sus limitaciones a cambio de perder ligereza y rapidez.

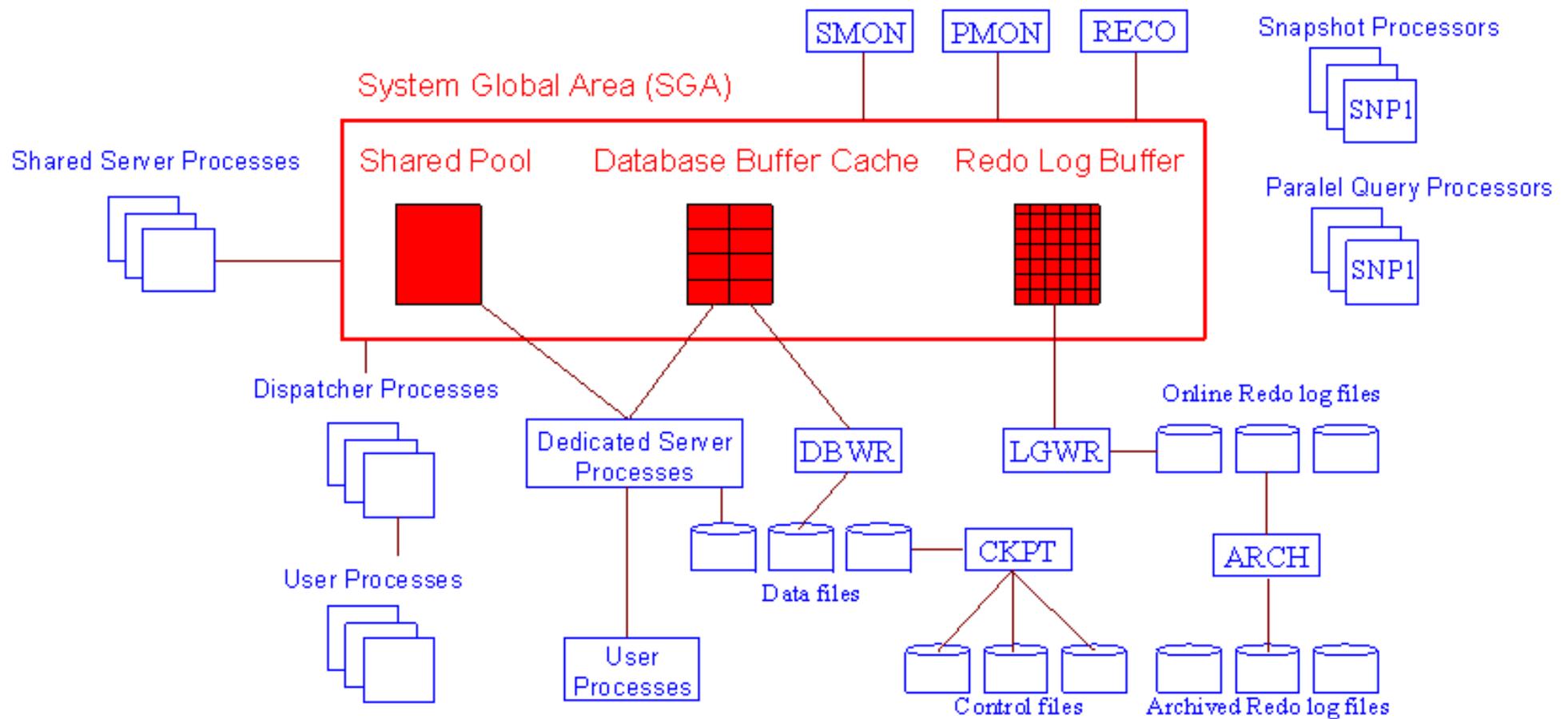


# El futuro

- Transaccionales: Bases de datos basadas en RAM:
  - Base de datos en RAM con REDO LOG en disco y backup en caliente.
  - Cuando cierras la base de datos haces una copia de la RAM a disco, si no la haces recuperas el último backup y aplicas REDO.
  - SAP Hana ha logrado mejoras en instalaciones reales frente a Oracle x100000.
- Data Warehouse: Nube.
  - Almacenamiento no estructurado (NoSQL).
  - Integración con motores estadísticos y cálculo con GPU.
  - Basados en la nube para aprovechar la elasticidad en el motor de cálculo.
  - Desafortunadamente sin soluciones estándar.



# Estructura de una base de datos



# Características habituales de las BBDD modernas

- ACID: Atomicity, Consistency, Isolation, Durability.
- Row Level Locking.
- MVCC: MultiVersion Concurrent Control.
- Hot Backup.
- Log shipping.
- Partitioning.



# Las diferencias

- Cluster
  - Almacenamiento distribuído: muy útil para data warehouse (DB2 UDB).
  - Almacenamiento compartido: rendimiento en transaccional (Oracle, DB2 Mainframe).
- Oracle ASM: Hace un análisis de las zonas calientes del disco y distribuye los datos de forma óptima.
- Parallel query: Oracle EE, DB2, SQL Server, PostgreSQL.
- Integración con R: Oracle EE, PostgreSQL, SQLServer.
- JVM en la base de datos: Oracle, DB2, PostgreSQL, SQLServer.
- Auditoría y permisos por fila: Gran diferencia entre versiones „Enterprise“ y „Standard“.



# Licenciamiento

- Las bases de datos son muy caras.
- Oracle es muuuy caro.
- Cluster dos nodos Intel (8 cores), cabina de discos 50 TB Flash nvme:
  - Hardware: 90K€
  - Oracle EE + RAC + partitioning: 960K€
  - No se incluyen los pack de administración, si los quieres +400K€
- El esfuerzo de entender y asegurar el cumplimiento de la licencia supone un coste extra significativo.



# Licenciamiento

- El software que utilizas terminas pagándolo, si no quieres pagarlo, no lo utilices.
- No instales o desactives todo aquello para lo que no tienes licencia, si no alguien terminará utilizándolo y, al final, habrá que pagarlo.
- Cuidado con el licenciamiento traidor!
- Estúdiate los contratos, licencias y documentación técnica sobre licenciamiento, aprovecha cada resquicio.
- Nunca mientas en una auditoría.



# Comparar bases de datos

- Asistir a demostraciones, presentaciones comerciales o leer documentación comercial provoca daños cerebrales irreversibles. No lo hagas.
- Para evaluar un producto de software lee los manuales técnicos, no la documentación comercial.
- Evaluar un producto de software necesita trabajo y esfuerzo, normalmente de meses.
- Una licencia de Oracle puede costar millones de euros, dedica unos cuantos miles a asegurarte que lo necesitas.
- Sólo hay dos tipos de documento en que los fabricantes no mienten: a) los formularios de la SEC y b) el „Full Disclosure Report“ del Transaction Processing Council: familiarízate con ellos y aprende a entenderlos

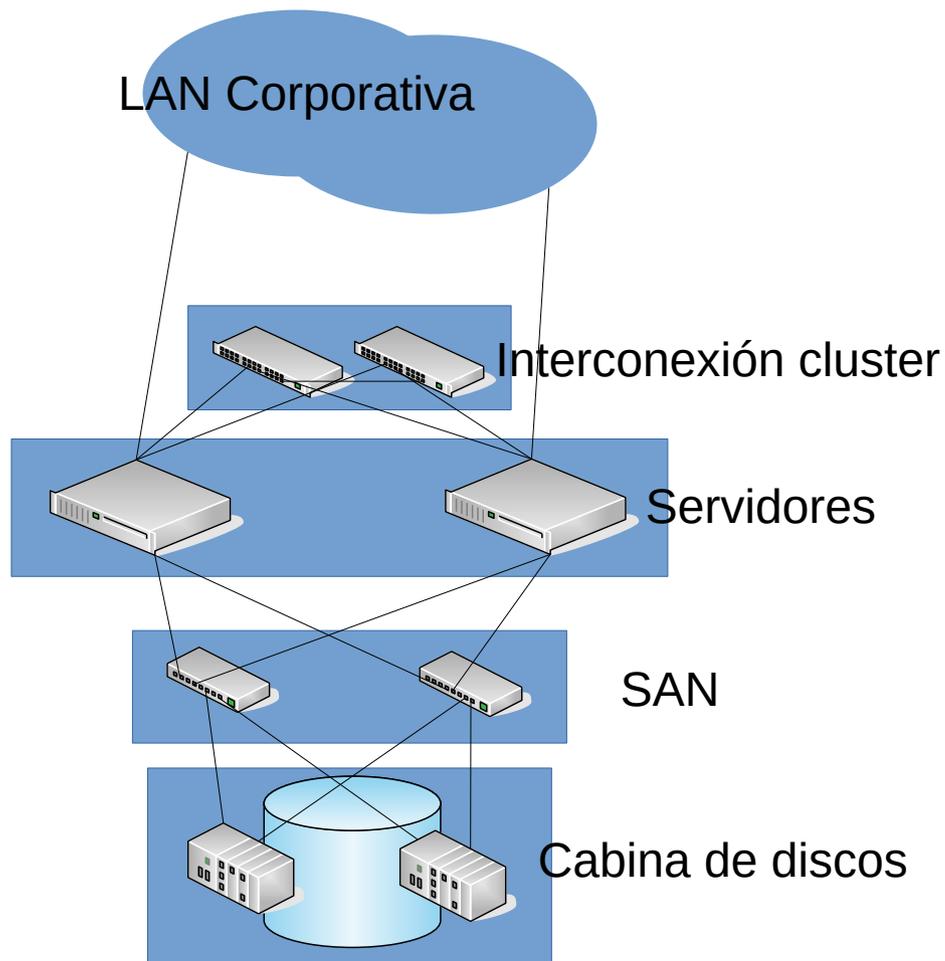


# Hierro

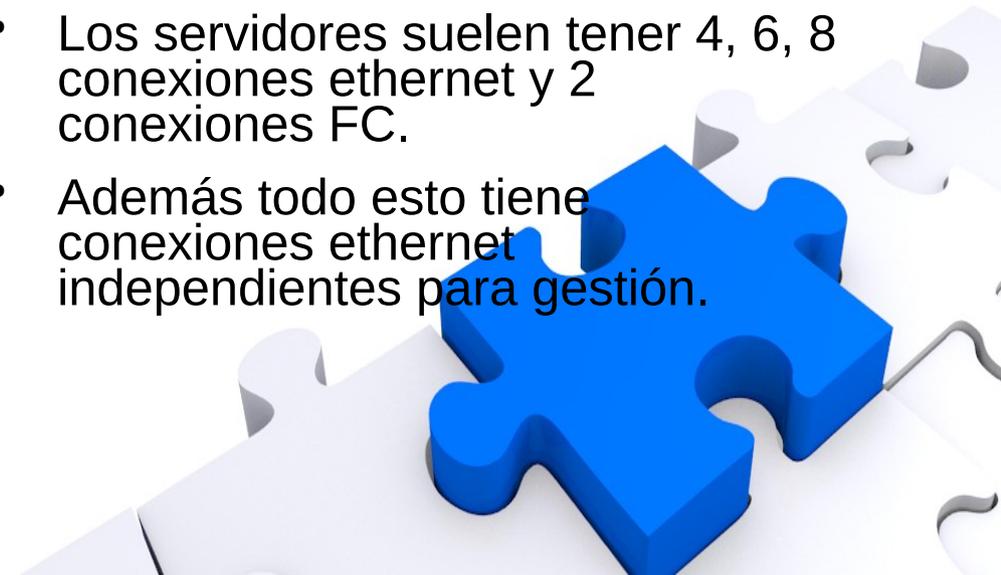
- Una instalación moderna de base de datos rara vez es un equipo con discos internos.
- Las instalaciones de BBDD son críticas para las operaciones del negocio, así que tienen que ser resistentes a fallos.
- Todo está duplicado, se eliminan puntos simples de fallo.
- Para instalaciones críticas sólo se utiliza hardware „Enterprise Level“, nada de material de uso doméstico.
- Todo el hardware es gestionable: switches, servidores, cabinas, librerías de cintas...



# Todo se duplica



- Todo está duplicado
- Dos servidores
- Dos switches ethernet para interconexión
- Dos switches FC en la SAN
- Cabina de discos con doble controladora
- Los discos se organizan en RAID 1, 5, 6, 10, 50 o 60
- Los servidores suelen tener 4, 6, 8 conexiones ethernet y 2 conexiones FC.
- Además todo esto tiene conexiones ethernet independientes para gestión.



# Un armario de verdad



- Armario con dos servidores VMware y dos servidores Oracle, cabina de discos y dos switch ethernet.
- En este caso los servidores son un cluster activo-pasivo.
- Cabina de discos con 20 discos. La conexión de la cabina es SAS y no hacen falta switch FC.
- 200 metros de cable.
- Da servicio a 4 plantas industriales y más de 2000 usuarios interactivos.



# Hierro: Mainframes

- IBM zSeries y predecesores: s/390, 3090, s/360.
- Virtualmente indestructibles: son capaces de sobrevivir a la destrucción de una CPU SIN que el proceso que está corriendo en ella cuando falla llegue a saberlo.
- Capacidad para decenas de miles de dispositivos conectados de forma simultánea.
- Rendimiento pobre en data warehouse, pero sobresaliente en transaccional.
- Caro no, carísimo
- Prácticamente ya sólo se utilizan para el transaccional de banca minorista.



# Hierro: UNIX

- De las más de 35 versiones distintas a mediados de los 80, tras las devastadoras „Guerras UNIX“ y el efecto Linux sólo quedan 3 de uso corporativo: AIX (IBM), HP-UX (HP) y Solaris (Oracle).
- El hardware actual escala desde equipos similares a un servidor Intel hasta servidores bastante parecidos a los Mainframe.
- Todos tienen integradas soluciones de virtualización y/o particionamiento hardware.
- Son difíciles de ver en empresas con menos de 1000 M€ de ventas anuales.



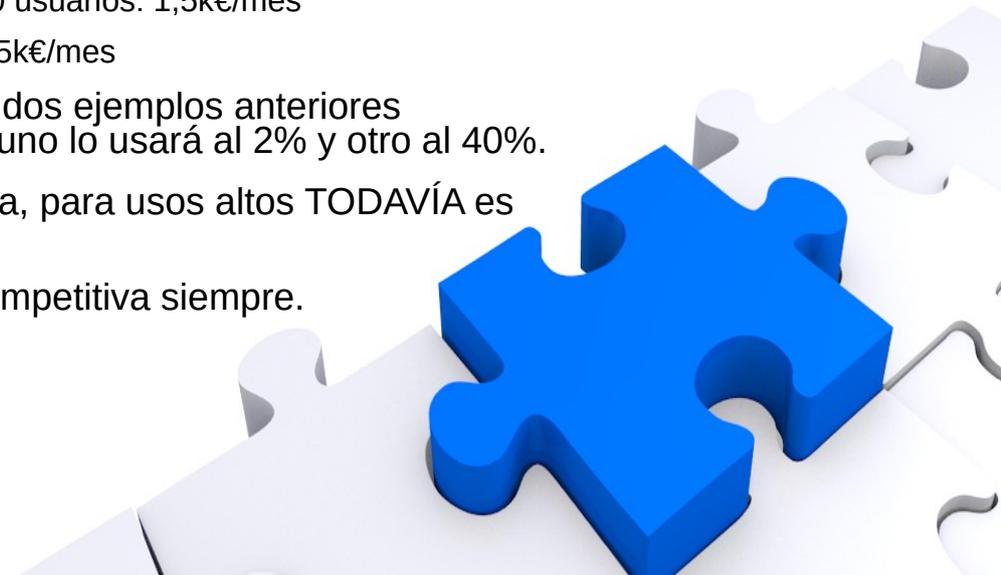
# Hierro: Servidores Intel

- Servidores con procesadores Intel o AMD.
- Arquitectura de 64 bit (x86\_64).
- Sistemas operativos Windows o Linux.
- Para cargas grandes sólo Linux es viable.
- Caso real: Procesador Intel 1CPUx6core, 64GB RAM soporta 2000 usuarios en interactivo con uso del 15% de la CPU. Coste aprox.: 20K€.
- En la actualidad no hay ninguna carga transaccional que no pueda basarse en un servidor Linux con 2 socket.
- Para cargas pequeñas muchos servidores están virtualizados.



# Hierro: La nube

- Visión de Greenberger („The computers of tomorrow“): La capacidad de proceso como la red eléctrica.
- En lugar de comprar una instalación, alquilas máquinas virtuales en la nube.
- Los costes dependen mucho de la política de licenciamiento.
- Por el lado del hardware, en casa:
  - Instalación con dos servidores, cabina de discos y librería de cintas, switches, armarios, UPS: 40k€-100k€ / 5 años.
  - Comunicaciones, electricidad: 1k€-2k€ / mes.
  - Mantenimiento hardware: 3k€ - 5k€ / año.
  - Esto es la configuración MÍNIMA para una instalación crítica.
- En la nube una instalación depende del uso, ejemplos Amazon Web Services:
  - Instalación para ERP, filial empresa sector alimentación, 20 usuarios: 1,5k€/mes
  - Instalación para ERP, empresa industrial, 2000 usuarios: 15k€/mes
- El secreto está en el grado de uso de tu instalación, los dos ejemplos anteriores necesitan el mismo hardware para instalación en casa, uno lo usará al 2% y otro al 40%.
- Si utilizas muy poco tu capacidad, la nube es muy barata, para usos altos TODAVÍA es cara.
- Uso especial: replicación de datacenter, donde ya es competitiva siempre.

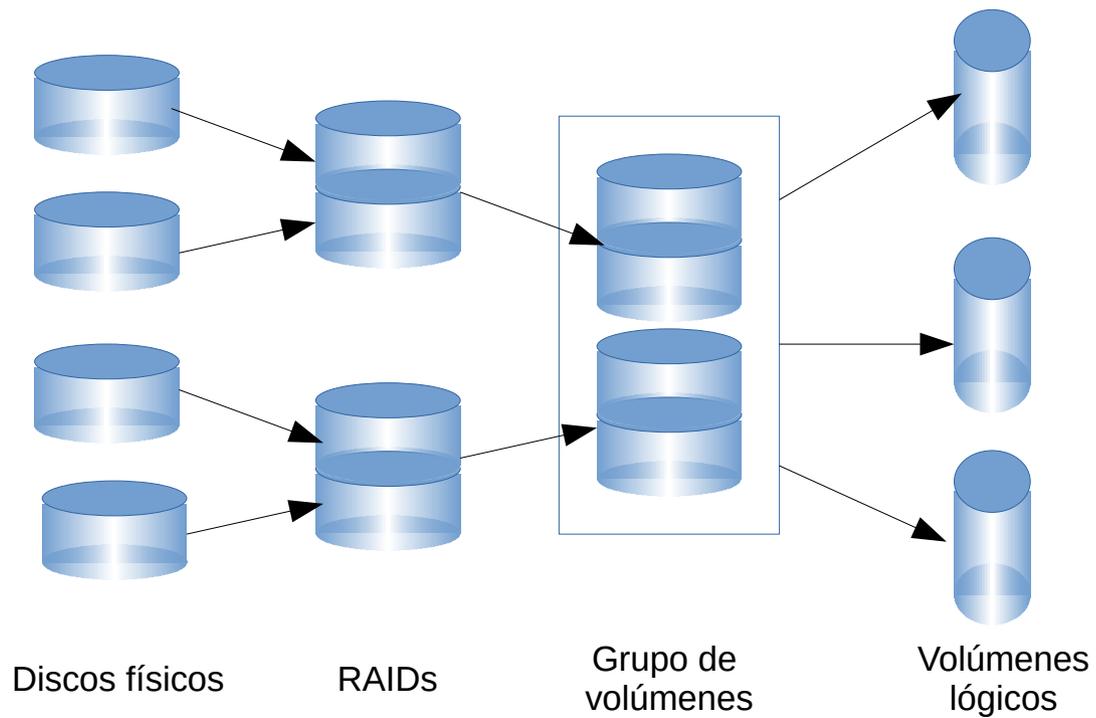


# Cabinas de discos

- Caja para discos que permite cambio en caliente con dos controladoras.
- Las controladoras forman un cluster activo/pasivo (gama baja) o activo/activo (gama media y alta).
- Las controladoras tienen memoria caché protegida por baterías o por supercondensadores con backup en flash, esto permite que el cache write-back sea seguro si se pierde la alimentación eléctrica.
- Cada controladora puede tener conexiones Fibre Channel (FC), Serial Attached SCSI (SAS) o iSCSI (SCSI sobre TCP/IP). En muchos casos tienen las tres.
- Las cabinas juntan los discos en RAID, varios RAID se unen en un grupo de volúmenes y cada grupo de volúmenes se puede partir en volúmenes lógicos, cada volumen lógico se presenta a cada servidor como una LUN SCSI.



# Cabinas de discos



# Cabinas de discos: Lo que hacen

- Cache: entre 16 y 128GB de RAM por controladora
- Snapshots: La cabina hace una copia instantánea de un volumen lógico y presenta la copia a los servidores como una nueva LUN, muy útil para backup.
- Replicación: Una cabina puede replicar las operaciones sobre un volumen lógico en otro volumen lógico de la misma cabina o de otra distinta que puede estar a miles de km.
- Virtualización: Una cabina puede controlar otras cabinas esclavas y se presenta a los servidores como una sólo cabina unificada: útil para migración de datos y ampliaciones.
- Dynamic provisioning: Crea copias de volúmenes lógicos virtuales bajo demanda a partir de plantillas. Útil para escritorios virtuales.



# Cabinas: Multitier

- La cabina cuenta con discos de distintas tecnologías y distribuye los datos de forma automática dependiendo de lo „calientes“ que están y de como se acceden.
- Normalmente una cabina tiene 3 niveles: SSD, mecánicos rápidos (15k rpm) y mecánicos lentos (7k2 rpm).
- Los datos muy calientes a SSD, los muy fríos a 7k2 y los intermedios o aquellos que se acceden siempre en secuencial a los 15k.
- La cabina divide la información en bloques de aprox. 1MB y guarda estadísticas de su acceso:
  - Cuantas veces se acceden.
  - Si se accede a todo el bloque o a trozos pequeños.
  - Estadística separada para lectura y escritura.
- El administrador de la cabina puede marcar prioridades y parámetros de calidad de servicio por LUN.
- Periódicamente (normalmente 24 horas) la cabina genera la distribución óptima de los bloques en disco y planifica trabajos de migración de datos entre los distintos discos.
- El trabajo de análisis y planificación en ocasiones lo hacen las controladoras de la cabina (IBM Storwize, por ejemplo) o un servidor externo (Fujitsu Eternus).
- La distribución típica para bases de datos: 10-20% SSD y resto mitad 15k y mitad 7k2.
- En este momento no se entiende un despliegue de base de datos sin utilizar multitier, es todo más sencillo.



# Cabinas: Todo Flash

- Las cabinas „todo Flash“ con todos los discos SSD o NVME son cada vez más populares.
- Ejemplo: Huawei Dorado 3000 v6: Doble controladora con 48 cores ARM y 64GB de RAM cada una. Doble fuente de alimentación, 4 conexiones iSCSI 1Gb/s y 4 conexiones a elegir: FC o iSCSI a 10Gb/s, 25Gb/s o 40Gb/s
- Proporciona SAN (FC, iSCSI), NAS (NFS, CIFS), Replicación, Snapshots...
- Con 24 discos NVME (70TB) y mantenimiento ilimitado durante 5 años: 50k€
- Rendimiento sostenido de 500-800GB/s con tiempo de acceso menor de 1ms

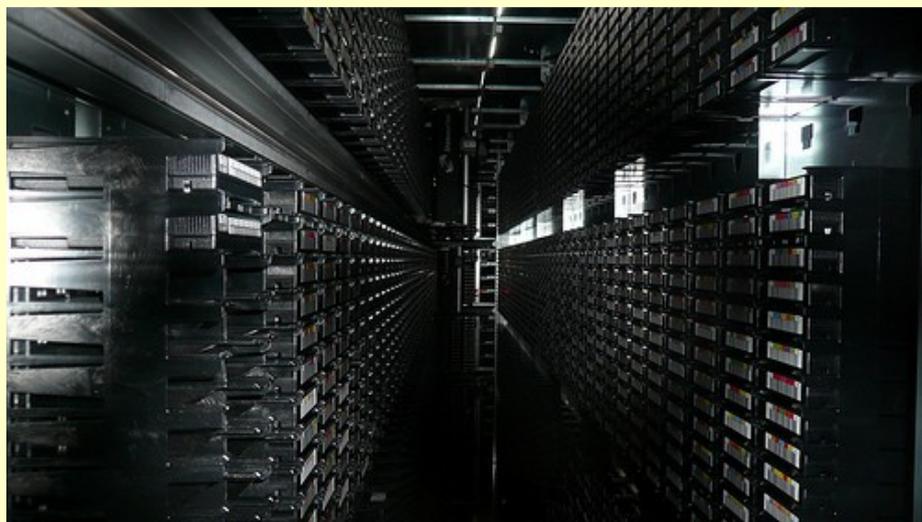


# Librerías de cintas

- Los sistemas grandes siguen haciendo el backup en cinta.
- Las cintas ofrecen mayor capacidad de almacenamiento que los discos para el mismo volumen, son más fiables tras almacenamientos prolongados, son más baratas para datos fríos y son obligatorias para ciertos entornos legalmente regulados.
- Todos los avances en ciencia básica que permiten abaratar y aumentar la capacidad de los discos permiten los mismos avances en las cintas.
- Para poder gestionar el contenido de las cintas es necesario un software de backup adecuado.
- Los sistemas de cintas de uso corporativo consisten en librerías de cintas: una librería tiene uno o más lectores, capacidad para varias cintas (entre 10 y varios miles) y un brazo robot para trasladar las cintas.
- Coste: una TS-4300 de IBM con 1 lector (2 máximo), brazo robot y capacidad para 24 LTO-9, aprox. 13K€. Con 2 lectores: 20K€. Una cinta LTO-9: 100-150€.
- LTO-9: Capacidad por cinta: 18TB (45TB comprimidos), velocidad de transferencia: 300MB/s (sin comprimir).
- Puedes quedarte con la generación anterior (LTO-8), mitad de capacidad, misma velocidad, una tercera parte del precio.



# Librerías de cintas



# El trabajo de DBA

- „Un DBA es un hombre que viste, a la vez, cinturón y tirantes“
- Los DBA son los trabajadores más felices del mundo: Si tienes mucho stress no estás haciendo bien tu trabajo y duras poco.
- Ser DBA es una cosa muy polifacética:
  - Mantener la base de datos en buen estado.
  - Ser mejor desarrollador que tu equipo de desarrollo.
  - Ser mejor en administración de sistemas que tu equipo de sistemas.
  - Saber más de almacenamiento que tu equipo de almacenamiento.
  - Saber más de red que tu equipo de redes.
  - Recomendar inversiones.
  - Etc, etc, etc....



# La parte más dura

- Es un trabajo tecnológico y eso quiere decir:
  - La gente a tu alrededor se juega su forma de vida.
  - Los accionistas, inversores, trabajadores, proveedores o clientes se juegan su dinero e ingresos.
  - Los usuarios pueden llegar a jugarse la vida.
  - El juez último de tus acciones no es ni tu jefe, ni tus compañeros, ni siquiera tus usuarios, el juez último es la naturaleza y es un juez implacable, intolerante y que no negocia.
  - Como consecuencia todo tu entorno te va a juzgar exclusivamente por la calidad de tu trabajo.
- Si eres competente la vida es buena, si eres un inútil no esperes comprensión ni consuelo.
- En particular nadie va a preocuparse de tus sentimientos. Si no eres capaz de soportar un entorno basado en el reconocimiento del trabajo, el esfuerzo, el talento y el mérito, en la crítica implacable de los errores y en la absoluta intolerancia a la incapacidad, mejor cambia de carrera.



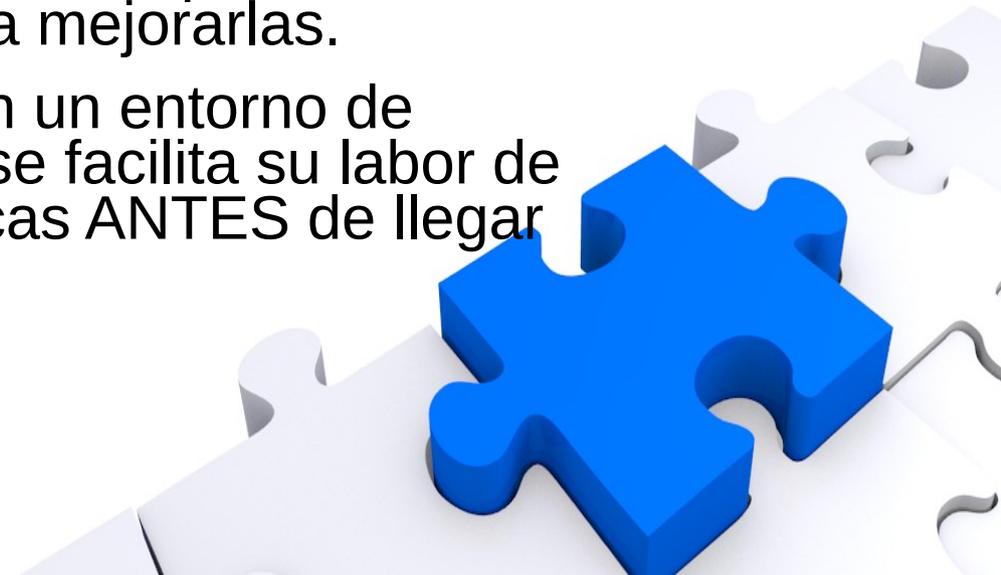
# Rendimiento de la base de datos

- Al DBA se le exige hacer magia, y no siempre es posible.
- Problemas de rendimiento:
  - Hardware inadecuado.
  - Almacenamiento inadecuado.
  - Mala parametrización.
  - Distribución de I/O no óptima.
  - Consultas no óptimas.
- El DBA sólo puede arreglar la mala parametrización y, a veces, la distribución de I/O, pero un buen DBA puede ayudar a la organización a arreglar las demás.
- Cuando el usuario tiene problemas con el tiempo de respuesta de la aplicación la culpa será siempre del DBA hasta que demuestre lo contrario.



# Los desarrolladores

- Los desarrolladores suelen tener pocos conocimientos de optimización de consultas.
- Los desarrolladores están muy presionados para añadir nuevas características a un sistema y no tienen tiempo de optimizar las consultas.
- Al final las consultas mal escritas acaban en el entorno de producción y todos pagan las consecuencias.
- El DBA puede a) Localizar las consultas problemáticas y b) ayudar a los desarrolladores a mejorarlas.
- Si los desarrolladores cuentan con un entorno de desarrollo parecido a producción se facilita su labor de localizar las consultas problemáticas ANTES de llegar al entorno productivo.



# Los sistemas

- Una base de datos no se ejecuta en el vacío: se ejecuta sobre un sistema operativo, utiliza la red, almacenamiento.
- Cualquier problema en uno de los componentes provocará un problema de rendimiento en la base de datos, pero la responsabilidad de arreglarlo siempre recae en el DBA.
- Es trabajo del DBA localizar el problema y, cuando es un elemento externo a la base de datos, convencer al responsable de que tiene un problema y ayudarlo a arreglarlo.
- Denunciar ante la organización que el responsable de red o de almacenamiento está haciendo un mal trabajo no suele mejorar su ánimo de colaboración, y los usuarios seguirán quejándose.
- Los responsables de almacenamiento, cuando los hay, son especialmente talibanes en sus creencias y refractarios a cualquier razonamiento en contra de ellas, la única forma de razonar con ellos son los informes de rendimiento.



# El hardware

- En las organizaciones las decisiones se toman con criterios de negocio y financiero.
- Las personas que van a tomar la decisión sobre comprar un nuevo hardware no tienen capacidad ni conocimiento para entender los problemas técnicos.
- Las personas que van a tomar la decisión sobre comprar un nuevo hardware planifican los gastos de la empresa con 12-18 meses de antelación como mínimo y no les gustan las sorpresas.
- El DBA debe tener siempre bajo control el nivel de uso del hardware y una previsión del nivel de uso a 18-24 meses basados en la evolución de uso del sistema en el pasado y las previsiones de evolución del negocio en el futuro.
- Cuando se detecta que el hardware va a necesitar ampliación en 18-24 meses, es necesario transmitir la información a gerencia de forma técnicamente sencilla y en un lenguaje que puedan entender. Es imprescindible justificar la inversión desde un punto de vista de negocio y financiero.
- Cuando se toman decisiones de negocio que van a impactar en el grado de uso de las bases de datos es preciso implicarse en el proceso y advertir de las inversiones en hardware y licencias que serán necesarias ANTES de que la organización tome la decisión.



# Sistemas de monitorización

- Larry Wall, las virtudes del programador: vagancia, impaciencia y soberbia.
- No hagas tú lo que puede hacer una máquina por tí.
- Instala y afina un sistema de monitorización.
- Cuando tienes 200 sistemas bajo tu responsabilidad es imposible que los tengas siempre en perfecto estado: monta un sistema que vigile por tí.
- Un buen DBA con un buen sistema de monitorización puede mantener una base de datos típica (1000 usuarios, transaccional) con menos de media hora de dedicación al mes.



# Backup

- La tarea más importante de un DBA es el backup... y la segunda... y la tercera.
- No sirve de nada el rendimiento de la base de datos, ni la calidad de estos ni lo bonita que es la aplicación si cuando hace falta recurrir al backup éste falla.
- El punto anterior dice „CUANDO hace falta“, no dice „SI hace falta“.... tarde o temprano hará falta.
- Los fallos en la recuperación de un backup del sistema financiero suelen suponer la quiebra de la compañía en 6 meses.
- Los procedimientos de backup y recuperación deben estar probados y automatizados, recuperar un backup es una tarea realizada bajo mucha presión y cualquier posibilidad de error será aprovechada por el humano para liarla.
- Las pruebas de recuperación de backup periódicas son imprescindibles para garantizar la calidad del backup.



# Backup

- Backup significa „backup físico“, los backup lógicos (export, dump...) no sirven para nada (salvo en DB2).
- Los DBA de verdad hacen los backup en caliente, siempre.
- Las bases de datos SIEMPRE deben estar en modo archivado del log.
- El archivado se debe evacuar al sistema de backup con la mayor frecuencia posible, eso reduce la pérdida de datos en caso de catástrofe completa.
- El backup y los datos nunca se deben guardar en los mismos discos.
- Cuanto más lejos estén físicamente el backup y los datos, mejor.
- Las catástrofes ocurren.
- „Sólo los paranoicos sobreviven“ (Andrew S. Groove)

