

BASE DE DATOS

Sistemas de representación de la información. Ficheros

ÍNDICE

| | |
|---|-----------|
| / 1. Introducción y contextualización práctica | 3 |
| / 2. Sistemas de representación de la información | 4 |
| / 3. Ficheros | 5 |
| 3.1. Concepto de fichero | 5 |
| 3.2. Tipos de ficheros según su uso | 5 |
| 3.3. Tipos de ficheros según su organización | 7 |
| 3.4. Tipos de ficheros según su contenido | 8 |
| / 4. Sistemas de almacenamiento de la información | 9 |
| 4.1. Concepto | 9 |
| 4.2. Sistemas lógicos de almacenamiento | 9 |
| 4.3. Tipos de sistemas de almacenamiento | 10 |
| 4.4. Gestión de los sistemas de almacenamiento | 11 |
| / 5. Caso práctico 2: “Gestión de tienda y almacén” | 11 |
| / 6. Resumen y resolución del caso práctico de la unidad | 12 |
| / 7. Bibliografía | 13 |

OBJETIVOS

Comprender la importancia de la información y de su correcta gestión.

Conocer el concepto de «fichero» e identificar los diferentes tipos.

Entender la estructura organizativa de cualquier sistema de información.

Identificar los distintos tipos de sistemas de almacenamiento de la información.

En su conjunto, interpretar los fundamentos sobre las que se construyen las bases de datos en la actualidad.

/ 1. Introducción y contextualización práctica

Hoy en día, gran parte de lo que hacemos en nuestra vida cotidiana está intrínsecamente ligado con el uso de sistemas de almacenamiento y gestión de la información. Cuando solicitamos cita en el médico, accedemos a nuestras redes sociales, o incluso cuando hacemos una llamada telefónica con nuestro smartphone estamos utilizando este tipo de sistemas.

La información (datos) probablemente sea el elemento más valioso que puede tener una organización actualmente. Nuestro mundo actual gira en torno al dato: comunicación con otras personas, interesarnos por viajes a otros lugares, buscar restaurantes en los que comer, movernos por nuestra ciudad a través del transporte público, leer el periódico... continuamente estamos generando y demandando información, que habitualmente está contenida en bases de datos de todo tipo.

Es por tanto fundamental entender, por un lado, la importancia de una correcta gestión y almacenamiento de esta información, y, por otro lado, los pilares principales sobre los que se construyen las bases de datos.

En este tema sentaremos las bases de cualquier sistema de información, analizando diversos conceptos básicos que es necesario entender y asimilar previamente.

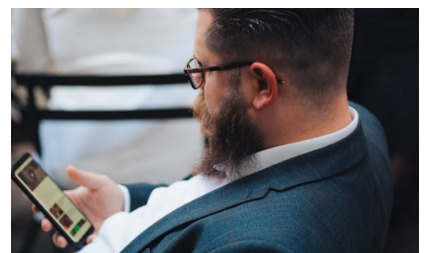


Fig. 1 El smartphone es una herramienta cotidiana que nos comunica con el mundo, permitiéndonos consultar y compartir información.



Audio intro. Manejar y gestionar la información correctamente es fundamental.
<https://bit.ly/2Uuzu4Y>





/ 2. Sistemas de representación de la información

Todos sabemos que un ordenador puede almacenar y procesar muchos tipos de datos distintos: textos, imágenes, vídeos, números, etc. Pero ¿de qué manera se manejan tantos tipos distintos? la respuesta es, a priori, sencilla: cualquier dato que se introduce en un sistema informático debe traducirse a un tipo concreto que pueda entender y manejar la máquina a bajo nivel. Este tipo concreto que es la base de cualquier sistema informático es el *bit*.

El *bit* sería la información de menor tamaño que puede manejar cualquier equipo digital, constituyendo la base sobre la que se construye el sistema.

Sin embargo, no se pueden representar todos los datos sólo con el bit 0 y 1, es evidente que son necesarias estructuras más grandes que permitan representar diferentes tipos de datos. Para ello, se utiliza un “patrón de bits”, que no es más que un grupo de *bits* consecutivos. Este grupo de bits será interpretado por los dispositivos o programas para representar la información adecuada.

Por tanto, para que un sistema informático pueda realizar operaciones, debe traducir (codificar) cualquier instrucción que reciba al sistema binario (formado por 0 y 1), de manera que puedan ser interpretadas por la máquina.

Para que la información pueda ser interpretada por el ser humano, existen dispositivos que “traducen” (decodifican) estas secuencias en información que podamos comprender: un texto, un sonido, vídeos, etc.

De esta forma, podemos distinguir diferentes múltiplos de bits:

| UNIDAD | SÍMBOLO | EQUIVALENTE |
|----------|---------|-------------|
| Bit | Bit | 0 ó 1 |
| Byte | b | 8 bits |
| Kilobyte | Kb | 1024 bytes |
| Megabyte | Mb | 1024 Kb |
| Gigabyte | Gb | 1024 Mb |
| Terabyte | Tb | 1024 Gb |
| Petabyte | Pb | 1024 Tb |
| Exabyte | Eb | 1024 Pb |

Tabla 1. Múltiplos de bit.

Los *bits* se agrupan para formar números, caracteres, datos... o ficheros. De esta forma, los archivos de imagen (jpg, bmp, png, etc), vídeo (mp4, mpg, mov, etc), ejecutables (exe, com, etc.) o de muchos otros tipos se conocen como ficheros binarios, es decir, son ficheros que requieren de un formato determinado para poder ser interpretados por el sistema.

De manera general, las bases de datos están formadas por ficheros de este tipo, que cuentan con un formato determinado en función del motor que utilicen en casa caso (Oracle, MySQL o Access).



/ 3. Ficheros

3.1. Concepto de fichero

En un sistema informático se pueden almacenar muchas clases de información (datos, música, vídeos, etc.), que estará contenida en los diferentes dispositivos de almacenamiento que existen. Los ficheros se utilizan para poder gestionar toda esa información, siendo estructuras que permiten almacenar datos de distintos tipos.

Un fichero se puede definir como una secuencia de bits que contiene información relacionada con un asunto concreto, de forma estructurada y organizada, y que están ubicados en dispositivos de almacenamiento. También se conocen habitualmente como “archivos”, debiendo contar con un nombre y una extensión.

Los datos que contienen los ficheros están pensados para que se pueda actuar sobre ellos de alguna manera: consultar, modificar, suprimir, etc.

Un fichero también puede entenderse como un conjunto de registros lógicos similares. Estos registros lógicos, a su vez, están formados por atributos, como se puede comprobar en la siguiente imagen:

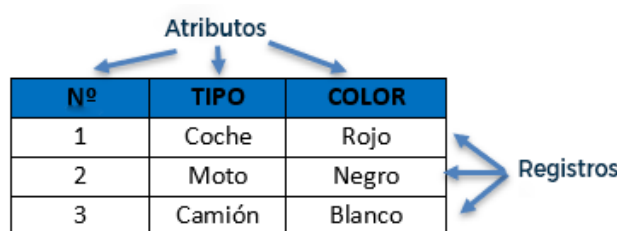


Fig. 2. Ejemplo de Fichero y campos asociados.

Podemos distinguir entre diferentes tipos de ficheros. En nuestro caso los clasificaremos según su uso, según cómo se organicen y según su contenido.

3.2. Tipos de ficheros según su uso

Si clasificásemos los ficheros en función de para qué vayan a ser utilizados, podemos distinguir dos opciones principales:



Fig. 3. Clasificación de ficheros según su uso.



3.2.1. Ficheros permanentes.

Como su propio nombre indica, están destinados a mantenerse en el sistema ya que contienen información de vital importancia para la aplicación. Dentro de ellos podemos distinguir:

- **Ficheros maestros:** Contienen información relativa al estado actual de datos concretos, que pueden ser modificados. Es decir, indican el estado de una organización o empresa en un instante determinado, en relación con algún asunto de importancia para la organización.

Por ejemplo, podría ser el fichero con los usuarios de una determinada plataforma, que sufrirá modificaciones debido a las altas y bajas de usuarios.

- **Ficheros históricos:** Suelen construirse a partir de los maestros, ya que contienen información antigua, con todos los movimientos que han podido ocurrir sobre el asunto concreto para el que se utilice el fichero.

Un ejemplo de fichero histórico podría ser los libros de contabilidad de una compañía, que contendrán información sobre su facturación desde la creación de la misma.

- **Fichero de constantes:** Como su propio nombre indica, la información que contienen varía muy poco a lo largo del tiempo.

Por ejemplo, un fichero de constantes podría ser el que contiene los datos personales de los trabajadores de una empresa determinada.

| Provincia | Población | Código Postal | lat | lon |
|-------------|--------------------------|---------------|----------------|----------------|
| Ciudad Real | Arroba de los Montes | 13193 | -4.543.405.160 | 39.153.483.150 |
| Ciudad Real | Ballesteros de Calatrava | 13432 | -3.944.291.080 | 38.832.773.180 |
| Ciudad Real | Bolaños de Calatrava | 13260 | -3.666.163.430 | 38.905.878.650 |
| Ciudad Real | Brazatortas | 13449 | -4.292.527.750 | 38.658.227.580 |
| Ciudad Real | Brazatortas | 13450 | -4.292.527.750 | 38.658.227.580 |
| Ciudad Real | Cabezarados | 13192 | -4.297.622.270 | 38.844.702.910 |
| Ciudad Real | Cabezarrubias del Puerto | 13591 | -4.183.781.480 | 38.614.771.350 |
| Ciudad Real | Calzada de Calatrava | 13370 | -3.776.998.700 | 38.704.690.550 |
| Ciudad Real | Calzada de Calatrava | 13739 | -3.776.998.700 | 38.704.690.550 |
| Ciudad Real | Calzada de Calatrava | 13779 | -3.776.998.700 | 38.704.690.550 |
| Ciudad Real | Campo de Criptana | 13610 | -3.126.142.790 | 39.405.568.800 |

Fig. 4. Ejemplo de fichero de constantes.

3.2.2. Ficheros temporales.

Se generan a partir de los permanentes, y su durabilidad es más reducida, ya que guardan información necesaria durante un periodo de tiempo, que posteriormente puede ser desechada.

- **Ficheros de movimientos:** De manera habitual contienen información sobre las modificaciones realizadas en el fichero maestro, ya sea modificar registros existentes, crear nuevos o eliminar alguno.

Por tanto, cuando se actualiza el fichero maestro con la información que contiene el fichero de movimientos, este último ya puede desecharse.

- **Ficheros de trabajo o de maniobra:** Estos archivos contienen información sobre datos que no pueden almacenarse en memoria por falta de espacio. Es decir, se utilizan como ficheros intermedios, para guardar datos de manera temporal en sistemas o aplicaciones que disponen de poca memoria.

3.2.3. Caso práctico 1: “Gestión de tienda y almacén”.

Planteamiento: Un amigo de la infancia ha abierto una tienda de productos de informática con la que está muy ilusionado. Se le da muy bien la parte comercial, pero no tiene mucho conocimiento sobre aplicaciones informáticas. Sabes que somos unos grandes profesionales en este campo, por lo que nos ha pedido ayuda para la gestión de su negocio con el fin de controlar el stock del que dispone en su almacén.



Nudo: Una solución rápida (mientras construimos una base de datos potente y robusta) sería construir un fichero maestro que contenga todo el stock del almacén. Además, generaremos dos ficheros más de movimientos: uno para el personal de tienda (que contendrá las ventas del día) y otro para el de almacén (que contendrá las entradas de nuevo material). Cada tipo de producto tendrá un ID distinto para poder distinguirlo del resto. Debemos ejemplificarlo con un caso para poder presentárselo a nuestro amigo y que conozca cómo debe utilizar estos ficheros.

Desenlace: Una posible propuesta de diseño es la siguiente. En ella, se ha generado un fichero de stock existente en el almacén. Al final del día, se ha obtenido el fichero de movimientos con las ventas realizadas y el de almacén con las nuevas entradas. Como resultado, se obtiene el fichero maestro de stock modificado.

| ID PRODUCTO | DESCRIPCIÓN | UDs |
|-------------|---------------------|------|
| 1 | Cable CAT6A | 5000 |
| 2 | Conector RJ45 CAT6A | 100 |
| 3 | Panel de 24p RJ45 | 10 |

Tabla 2. Archivo maestro de Stock existente.

| ID PRODUCTO | DESCRIPCIÓN | UDs |
|-------------|--------------|------|
| 4 | Fibra óptica | 1000 |

Tabla 3. Archivo de movimientos del día (entradas a almacén).

| ID PRODUCTO | DESCRIPCIÓN | UDs |
|-------------|---------------------|------|
| 1 | Cable CAT6A | 1000 |
| 2 | Conector RJ45 CAT6A | 30 |
| 3 | Panel de 24p RJ45 | 2 |

Tabla 4. Archivo de movimientos del día (ventas realizadas).

| ID PRODUCTO | DESCRIPCIÓN | UDs |
|-------------|---------------------|------|
| 1 | Cable CAT6A | 4000 |
| 2 | Conector RJ45 CAT6A | 70 |
| 3 | Panel de 24p RJ45 | 8 |
| 4 | Fibra óptica | 1000 |

Tabla 5. Archivo maestro de Stock resultante.

3.3. Tipos de ficheros según su organización

En este caso, distinguimos diferentes tipos de ficheros según la forma en la que se accede a su información.

3.3.1. Ficheros secuenciales.

La información en estos ficheros está dispuesta de manera ordenada y consecutiva. Por tanto, para poder acceder a unos datos concretos es necesario pasar por los todos los anteriores. Cuentan con una marca o hito al final del mismo para indicar que no hay más contenido. Un ejemplo de fichero de este tipo se muestra a continuación:

```
1111AJT // BMW320D // Madrid // Antonio // García // Jiménez //
2222BJT // RENAULT MEGANE // Ciudad Real // Manuel // Jiménez //
Del Río // 3333CZD // SMART FOR FOUR // Toledo // Laura //
Martínez // Aparicio // 3872DAD // Zaragoza // Ana // López //
Fernández //
```

Fig.5. Ejemplo de fichero secuencial

3.3.2. Ficheros de acceso directo.

En este tipo de ficheros los datos están organizados con unos tamaños fijos, de forma que se puede acceder directamente. Al fijarse un tamaño determinado, a veces no se ocupa de manera completa, con lo que se desperdicia espacio en disco. A continuación, se muestra un ejemplo de este tipo de fichero:

```
1111AJT BMW320D      Madrid      Antonio García  Jiménez
2222BJT RENAULT MEGANE Ciudad Real Manuela Jiménez Del Río
3333CZD SMART FOR FOUR Toledo      Laura      Martínez Aparicio
3872DAD VOLVO V49     Zaragoza    Ana        López      Fernández
```

Fig. 6. Ejemplo de fichero de acceso directo

3.3.3. Ficheros indexados.

En este caso, los datos que contienen los ficheros están organizados en base a un índice, que se utiliza para poder acceder a los mismos. Obviamente, el acceso a la información en este tipo de ficheros es más rápida, pero también más compleja de implementar.

```
1111AJT → BMW320D Madrid Antonio García Jiménez
2222BJT → RENAULT MEGANE Ciudad Real Manuela Jiménez Del Río
3872DAD ↗ SMART FOR FOUR Toledo Laura Martínez Aparicio
4333CZD ↘ VOLVO V49 Zaragoza Ana López Fernández
```

Fig. 7. Ejemplo de fichero indexado

3.4. Tipos de ficheros según su contenido

Desde el punto de vista de las bases de datos, y debido a las características de las de nueva generación, cada vez se atiende más a clasificar los ficheros según su uso o según el tipo del que sean. En este caso, podemos distinguir entre ficheros planos y ficheros binarios:

3.4.1. Ficheros planos o de texto.

Se caracterizan por ser entendibles directamente por nosotros, sin necesidad de ser interpretados por parte de ningún software.

Suelen estar contruidos mediante código ASCII, que es un estándar de representación basado en asociar un número a cada carácter que se quiera representar, según una tabla de códigos normalizada.

El hecho de que no necesiten ser procesados no implica que no tengan una extensión determinada, con el fin de que el usuario sepa de antemano qué tipo de texto contiene el fichero. De esta forma, existen archivos de texto de diferentes tipos:

- **Ficheros web:** html, php, xml
- **Ficheros de código:** java, js, sql
- **Ficheros de configuración:** ini, conf

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Esta es nuestra ayuda JavaHelp de prueba</title>
</head>
<body>

<h1>Esta es una ayuda creada con JavaHelp</h1>

<h2>Esta ayuda la hemos creado a modo de prueba</h2>

<p>Para ello podemos utilizar un generador de código HTML cualquiera</p>
<p>Podemos también añadir imágenes o incluso referenciar
entre sí las diferentes páginas de la ayuda</p>



<p></p>

Este es el enlace al

<a href="tema1.html">Tema 1</a> de ayuda

<p><a href="tema2.html">Tema 2</a> de ayuda

</body>
```

Fig.8. Ejemplo de fichero de texto en HTML



Video 1. "Diferencias entre ficheros
planos de texto y ficheros binarios"
<https://bit.ly/3cSv7qC>





3.4.2. Ficheros binarios.

Según se indicó en el punto 2, los datos que contienen típicamente las bases de datos están basados en este tipo de ficheros.

Se caracterizan por estar codificados en binario, por lo que deben ser procesados por alguna aplicación concreta, según su extensión.

Algunos ejemplos de este tipo de ficheros son los siguientes:

- **Ficheros de vídeo:** mp4, avi
- **Ficheros ejecutables:** exe
- **Ficheros de imágenes:** jpeg, png, gif
- **Ficheros de aplicaciones concretas:** xls, pdf

/ 4. Sistemas de almacenamiento de la información

4.1. Concepto

Los sistemas de almacenamiento de la información consisten en dispositivos de diferentes características y construcciones que permiten el almacenamiento y gestión de datos de diferentes tipos.

Algunos ejemplos de dispositivos de almacenamiento se pueden observar en la siguiente imagen: disco duro, dispositivos de memoria flash, DVD, NAS, o incluso el almacenamiento en la nube, que cada día cobra mayor importancia.

En el aspecto concreto de las bases de datos, todos los ficheros que las forman deben encontrarse almacenados de alguna manera a través de uno de estos sistemas, mediante los cuales, se podrá acceder a la información que se necesite en un momento determinado.



Fig.9. Diferentes sistemas de almacenamiento de información.

<https://comofriki.com/que-es-un-dispositivo-de-almacenamiento/>



Audio 1. "Operaciones básicas de cualquier sistema de almacenamiento de la información".

<https://bit.ly/37jH1ZK>



4.2. Sistemas lógicos de almacenamiento

Desde el punto de vista de las bases de datos, es importante tener claro el concepto de sistema de almacenamiento lógico.

Como se puede comprobar en la imagen anterior, un sistema de almacenamiento lógico abarca uno o varios medios físicos, ocupando particiones de los mismos. Desde el punto de vista lógico, tendremos un único sistema de almacenamiento, que físicamente se encuentra distribuido.

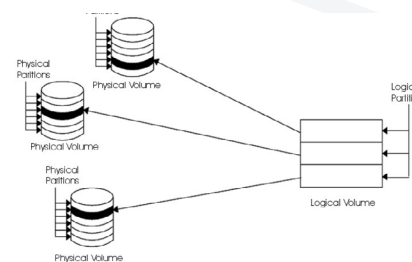


Fig. 10. Almacenamiento lógico

4.3. Tipos de sistemas de almacenamiento

Los sistemas de almacenamiento pueden clasificarse atendiendo a diferentes criterios:

- Según su construcción (ópticos y magnéticos)
- Según su capacidad
- En base a su velocidad
- Según la organización de los datos (secuenciales o direccionables)
- Según el tipo de datos que se dese almacenar y en función de la cantidad de los mismos, se podrá determinar la elección de un sistema u otro.

A continuación, se presentan las características de algunos de los sistemas de almacenamiento más conocidos y utilizados.

4.3.1. Discos duros.

Los discos duros pueden ser magnéticos o de estado sólido (SSD), que aportan una mayor velocidad y fiabilidad, aunque también son más caros.

La capacidad de estos medios se incrementa conforme avanza la tecnología. Hoy en día, permiten almacenar cientos de TB de datos para almacenamiento.

4.3.2. NAS (Network Attached Storage) y SAN (Storage Area Network).

Se trata de sistemas de almacenamiento a través de la red, con diferencias importantes entre sí. Mientras que un NAS es un solo dispositivo de almacenamiento centralizado (por ejemplo, un disco duro conectado a una red LAN), SAN es en sí mismo una red de almacenamiento dedicada, que permite intercambiar datos mediante la utilización de un software especial.

¿Cuál es preferible utilizar? Como en muchos casos, esto dependerá del caso concreto. Un NAS es un sistema más sencillo y fácil de mantener, además de eficiente.

Sin embargo, una SAN es una opción más potente, que ofrece mayor rendimiento, pero también más cara y compleja.

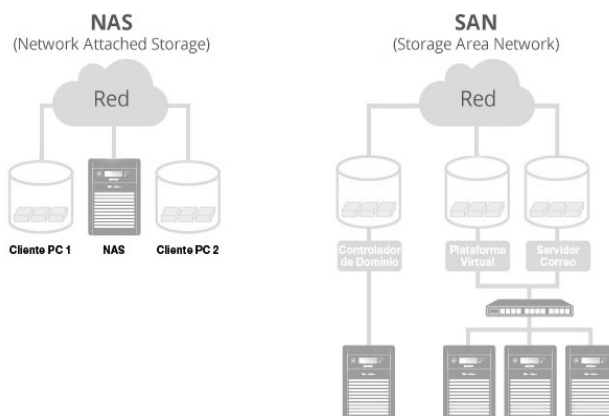


Fig. 11. NAS y SAN

<https://www.infordisa.com/es/comparativa-san-vs-nas/>

4.3.3. Almacenamiento RAID.

RAID (Redundant Array of Independent Disks) es una configuración basada en la utilización de varios discos independientes entre sí, de forma que estos se comporten como si fuesen un único dispositivo.

Este hecho permite reducir el tiempo necesario para acceder a la información, mientras que posibilita utilizar redundancia en los discos, con el fin de reducir las posibilidades de pérdida de información.



Existen varias construcciones RAID, con diferentes características según su configuración. De esta manera, se habla de niveles RAID, algunos de los más comúnmente utilizados son los siguientes:

| Nivel | Confiabilidad | Rendimiento | Disponibilidad |
|--------|---|---|--|
| RAID 0 | • No proporciona tolerancia a fallos. | • Mejora la tasa de transferencia y el tiempo de acceso a los datos. | • El sistema deja de funcionar si hay una unidad de disco en falla. |
| RAID 1 | • Protege la información en caso de falla. | • Mejora la lectura de los datos. | • Evita interrupciones por fallos en las unidades. |
| RAID 2 | • El uso del código Hamming permite detectar y corregir errores. | • Mejora la operación de aplicaciones con alta tasa de transferencia. | • Usa múltiples discos dedicados que permiten redundancia de datos. |
| RAID 3 | • El disco de paridad permite reconstruir la información. | • Elevada tasa de transferencias secuenciales. | • Si falla un disco el sistema puede seguir funcionando. |
| RAID 4 | • Es ideal para almacenar ficheros de gran tamaño. | • Durante las operaciones de lectura-escritura las unidades de disco son accedidas de forma individual. | • Es tolerante a fallos ya que se puede recuperar los datos de un disco averiado en tiempo real. |
| RAID 5 | • Distribuye los datos de paridad entre todas las unidades de disco. | • La velocidad de transferencia de datos es alta. | • Es tolerante a fallos con una unidad de disco averiada. |
| RAID 6 | • Cada dato de paridad es redundante y distribuido en dos unidades de disco diferentes. | • Las operaciones de escritura resultan más lentas que las de lectura de datos. | • Es tolerante a fallos con dos unidades de discos averiadas. |

Fig. 12. Diferentes tipos de RAID

<https://tecnovirtualization.wordpress.com/2015/10/11/raid/>



Video 1. "Sistemas de almacenamiento RAID anidados"

<https://bit.ly/37qz2wp>



4.4. Gestión de los sistemas de almacenamiento

En cualquier organización es de vital importancia la correcta gestión de la información, ya que como se indicó anteriormente, es el elemento que más valor tiene y que es más importante mantener.

Para ello, es muy importante, entre otras cosas, realizar una gestión y un mantenimiento adecuados de los sistemas de almacenamiento de manera que:

- Se minimice la pérdida de datos en caso de fallo crítico.
- Se alargue la vida útil de los dispositivos utilizados.
- Se optimice la utilización de los mismo.
- Se garantice en todo momento la seguridad e integridad de la información existente.

De esta manera, hay cuatro aspectos fundamentales en la gestión de cualquier sistema de almacenamiento que se deben tener siempre presentes en la gestión del mismo: **capacidad, recuperabilidad, rendimiento y fiabilidad.**

/ 5. Caso práctico 2: "Gestión de tienda y almacén"

Planteamiento: Un amigo que ha abierto una tienda de productos de informática nos ha pedido ayudar para la gestión de su negocio, ya que no tiene claro cómo puede controlar el stock del que dispone en su almacén. En una de sus comprobaciones, necesita que, en cada test unitario, se realicen pruebas matemáticas con 2 números. Para ello, al ejecutar cada test, Miguel deberá cargar automáticamente en memoria una lista con 2 números antes de empezar el código de cada prueba unitaria.

Nudo: Ya le hemos presentado varios ficheros que le podrán ayudar de momento a gestionar su tienda, mientras que pensamos en una solución más robusta y versátil.

En este punto, creemos necesario que nuestro amigo haga una pequeña inversión para adquirir un sistema de almacenamiento de la información más potente, ya que actualmente utiliza el mismo disco duro externo en el que guarda las películas.

Dado que nuestra idea es ayudarlo en un proceso de renovación de su gestión de información, estudiamos varias posibilidades para verificar qué opción de las vistas en este tema podría resultarle más atractiva, y se adapta mejor a sus necesidades.

Hay que tener presente que, en principio, nuestro amigo no piensa ampliar su negocio, que consiste en una pequeña tienda de informática con un pequeño almacén anexo, en la que trabajan él y su pareja. Aunque se trata de un negocio pequeño en el que no se manejarán grandes cantidades de información (al menos de momento), sí le gustaría tener cierta tranquilidad de que no perderá los datos ante cualquier fallo del equipo.

Desenlace: Básicamente, nuestro amigo requiere para su tienda seguridad en la información y asegurar su disponibilidad (redundancia).

Das las características de la tienda, y en base a lo visto a lo largo el tema, no parece que la opción del disco duro en un PC vaya a resultar la más óptima, aunque podría ser viable siempre y cuando se aplique redundancia.

Tampoco parece que implementar un sistema SAN sea lo mejor, dado que son caros y complejos.

Por tanto, para este entorno quizá la mejor opción sería un sistema NAS que aplique RAID, de forma que se asegure la redundancia del sistema. Una posible opción sería utilizar el configurador de Synolgy para construir el RAID que queramos, y poder comparar diferentes equipos que lo implementen. [Pruébalo aquí.](#)

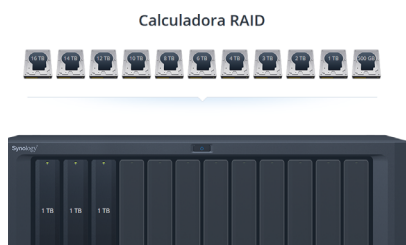


Fig. 13. Ejemplo de configuración RAID de Synolgy.

/ 6. Resumen y resolución del caso práctico de la unidad

A lo largo del tema se han desarrollado varios conceptos básicos pero de gran importancia para poder iniciarse en el mundo de las bases de datos.

De esta manera, comenzamos haciendo énfasis en **la importancia que tiene para cualquier organización una buena gestión de la información**, lo que implica la necesidad de una correcta infraestructura que lo permita, tanto física como lógica. Ahí es dónde el mundo de las bases de datos demuestra su gran importancia y utilidad, especialmente en la actualidad, dónde cada día hay más aplicaciones que dependen del uso del dato.

Posteriormente se presentó **el concepto de fichero, haciendo hincapié en los diferentes tipos de ficheros que hay**. Al fin y al cabo, las primeras formas de tratamiento y gestión de la información se llevaron a cabo gracias a los ficheros, antes de la llegada de las bases de datos tal y como las conocemos hoy en día.

A continuación, se detallaron **las principales características de los sistemas de almacenamiento de la información** más extendidos y conocidos en la actualidad, haciendo énfasis en las fortalezas y debilidades de cada uno de ellos.

Este asunto es importante, dado que en función del caso que nos ocupe será preferible la utilización de uno u otro, según los objetivos que queramos cumplir. Cerramos el tema recordando la **importancia de la redundancia en estos sistemas**, y de una correcta gestión de los mismos.



Resolución del caso práctico de la unidad

En el caso inicial planteábamos algunas cuestiones para ayudar a nuestro amigo con su tienda. Todas ellas pueden resolverse con el contenido que hemos visto a lo largo del tema, e incluso los casos prácticos 1 y 2 nos ayudan tanto a ello, como a tener una mayor perspectiva de lo que necesitamos hacer para poder echar una mano a nuestro amigo. Por tanto, una posible respuesta a las preguntas planteadas al principio puede ser la siguiente:

- **¿En qué manera le ayudará en su gestión diaria informatizar sus procesos?**

Se ha comprobado la importancia de contar con un sistema de gestión de la información robusto. El principal activo de una compañía es la información que maneja.

- **¿Cómo podemos ayudarle?**

Podemos ayudarle a implementar un sistema de base de datos que le ayude en su día a día.

- **¿Qué información de partida necesitaremos para ello?**

Básicamente, necesitamos conocer la problemática de su día a día y sus necesidades, para poder diseñar algo a su medida: ni demasiado sencillo ni demasiado potente (y por tanto costoso).

Es importante saber para qué lo quiere y qué quiere hacer con él.

- **¿Habrá alguna solución rápida que podamos aplicar?**

Mientras diseñamos un sistema más potente, podemos realizar la gestión mediante ficheros informáticos, que siempre serán más potentes que el papel. Al fin y al cabo, así es cómo se hacía antes de la aparición de las bases de datos modernas.

/ 7. Bibliografía

Oppel, A. (2009): *Databases: A Beginner's Guide*. Madrid, España: McGraw-Hill.

Elmasri, R.; Navathe, S. (2007): *Fundamentos de Bases de Datos* (5.a. ed.). Madrid, España: Pearson Addison-Wesley

López, I.; Castellano, M.J. y Ospino, J. (2011): *Bases de datos*. Madrid, España: Garceta

Cabrera, G. (2011): *Sistemas gestores de bases de datos*. Madrid, España: Paraninfo.