

# **A** **Administración de sistemas gestores de bases de datos**

Consulte nuestra página web: [www.sintesis.com](http://www.sintesis.com)  
En ella encontrará el catálogo completo y comentado



Queda prohibida, salvo excepción prevista en la ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra sin contar con autorización de los titulares de la propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (arts. 270 y sigs. Código Penal). El Centro Español de Derechos Reprográficos ([www.cedro.org](http://www.cedro.org)) vela por el respeto de los citados derechos.

# **A**ministración de sistemas gestores de bases de datos

Alex Dapena

**ASESOR EDITORIAL:**

---

Juan Carlos Moreno Pérez

© Alex Dapena

© EDITORIAL SÍNTESIS, S. A.  
Vallehermoso, 34. 28015 Madrid  
Teléfono 91 593 20 98  
<http://www.sintesis.com>

ISBN: 978-84-1357-068-6  
Depósito Legal: M-2.435-2021

Impreso en España - Printed in Spain

Reservados todos los derechos. Está prohibido, bajo las sanciones penales y el resarcimiento civil previstos en las leyes, reproducir, registrar o transmitir esta publicación, íntegra o parcialmente, por cualquier sistema de recuperación y por cualquier medio, sea mecánico, electrónico, magnético, electroóptico, por fotocopia o por cualquier otro, sin la autorización previa por escrito de Editorial Síntesis, S. A.

# Índice

<b>PRESENTACIÓN</b> .....	11
<b>1. INSTALACIÓN DEL SISTEMA GESTOR DE BASES DE DATOS</b> .....	13
Objetivos .....	13
Mapa conceptual .....	14
Glosario .....	14
<b>1.1. Introducción</b> .....	15
<b>1.2. El administrador de bases de datos</b> .....	15
<b>1.3. El sistema gestor de bases de datos</b> .....	16
1.3.1. Funciones del sistema gestor de bases de datos .....	17
1.3.2. Componentes de un sistema gestor de bases de datos .....	19
<b>1.4. Factores de elección del sistema gestor de bases de datos</b> .....	22
1.4.1. Tipos de sistemas gestores de bases de datos .....	23
1.4.2. Arquitectura del sistema gestor de base de datos .....	30
1.4.3. Tipos de licencias .....	32
1.4.4. Tipos de bases de datos .....	35
1.4.5. Requisitos necesarios .....	37
<b>1.5. Contextos de utilización de cada sistema gestor de bases de datos</b> .....	37
<b>1.6. Instalación</b> .....	39
1.6.1. Componentes de la instalación .....	39
1.6.2. Configuración inicial .....	40
1.6.3. Registro de instalación .....	41
<b>1.7. Instalación de Oracle</b> .....	41
1.7.1. Requisitos .....	42
1.7.2. Componentes de la instalación .....	42

1.7.3. El asistente de instalación OUI .....	45
1.7.4. Instalación Windows vs instalación Linux .....	47
Resumen .....	48
Ejercicios propuestos .....	49
Actividades de autoevaluación .....	51
<b>2. CONFIGURACIÓN DEL SISTEMA GESTOR DE BASES DE DATOS .....</b>	<b>53</b>
Objetivos .....	53
Mapa conceptual .....	54
Glosario .....	54
2.1. Introducción .....	55
2.2. Configuración del sistema gestor de bases de datos .....	56
2.2.1. Configuración del entorno .....	56
2.2.2. Configuración de las conexiones .....	56
2.2.3. Configuración del servidor .....	57
2.2.4. Configuración del almacenamiento .....	57
2.2.5. Configuración de las cuentas del sistema .....	59
2.3. Arranque y parada del sistema .....	59
2.4. El diccionario de datos .....	60
2.5. El cuaderno de bitácora .....	61
2.6. Documentación .....	62
2.7. Configuración de Oracle .....	63
2.7.1. Configuración del entorno .....	63
2.7.2. Configuración de las conexiones .....	64
2.7.3. Configuración de la instancia de Oracle .....	66
2.7.4. Configuración del almacenamiento .....	68
2.7.5. Cuentas de administración .....	69
2.7.6. Creación y borrado de bases de datos .....	70
2.7.7. Ficheros de LOG .....	70
2.7.8. Arranque y parada del sistema gestor de bases de datos .....	70
2.7.9. El diccionario de datos en Oracle .....	74
Resumen .....	75
Ejercicios propuestos .....	76
Actividades de autoevaluación .....	77
<b>3. GESTIÓN DE USUARIOS Y PERMISOS .....</b>	<b>79</b>
Objetivos .....	79
Mapa conceptual .....	80
Glosario .....	80
3.1. Introducción .....	81
3.2. Gestión de usuarios y permisos .....	81
3.2.1. Usuarios .....	83
3.2.2. Permisos .....	84
3.2.3. Roles .....	84
3.2.4. Perfiles (profile) .....	86
3.3. Gestión de usuarios y permisos en Oracle .....	86
3.3.1. Usuarios .....	87
3.3.2. Permisos .....	90

3.3.3. Roles .....	92
3.3.4. Perfiles .....	93
3.3.5. Diccionario de datos .....	95
3.4. Esquemas externos .....	97
3.4.1. Vistas .....	100
3.4.2. Sinónimos .....	102
3.5. Esquemas externos en Oracle .....	103
Resumen .....	104
Ejercicios propuestos .....	105
Actividades de autoevaluación .....	109
<b>4. SEGURIDAD .....</b>	<b>111</b>
Objetivos .....	111
Mapa conceptual .....	112
Glosario .....	113
4.1. Introducción .....	113
4.2. Confidencialidad .....	114
4.2.1. Encriptación .....	114
4.2.2. Auditoría .....	116
4.3. Confidencialidad en Oracle .....	117
4.3.1. Encriptación .....	117
4.3.2. Auditoría .....	121
4.4. Integridad .....	126
4.4.1. Restricciones .....	126
4.4.2. Control de concurrencia .....	128
4.4.3. Recuperación .....	132
4.4.4. Tipos de copias de seguridad .....	133
4.5. Integridad en Oracle .....	134
4.5.1. Restricciones .....	134
4.5.2. Control de concurrencia .....	135
4.5.3. Copias de seguridad .....	135
4.6. La normativa de protección de datos .....	140
4.6.1. La LOPD .....	140
4.6.2. El papel del sistema gestor en la LOPD .....	141
Resumen .....	142
Ejercicios propuestos .....	143
Actividades de autoevaluación .....	145
<b>5. MONITORIZACIÓN Y OPTIMIZACIÓN .....</b>	<b>147</b>
Objetivos .....	147
Mapa conceptual .....	148
Glosario .....	149
5.1. Introducción .....	149
5.2. Monitorización .....	149
5.2.1. Monitor de rendimiento .....	150
5.2.2. Registro de errores .....	151
5.2.3. Diccionario de datos .....	151

5.3. Monitorización en Oracle .....	152
5.3.1. Monitor de rendimiento .....	152
5.3.2. Registro de errores .....	153
5.4. Optimización .....	154
5.4.1. Optimización del entorno .....	154
5.4.2. Optimización del sistema gestor .....	155
5.4.3. Optimización de la base de datos .....	156
5.4.4. Creación de índices .....	158
5.4.5. Optimización de consultas .....	163
5.4.6. Herramientas de optimización .....	169
5.5. Optimización en Oracle .....	171
5.5.1. Optimización del sistema gestor .....	171
5.5.2. Optimización de los objetos de base de datos .....	172
5.5.3. Optimización de consultas .....	175
5.5.4. Herramientas de optimización .....	176
Resumen .....	181
Ejercicios propuestos .....	181
Actividades de autoevaluación .....	184
<b>6. AUTOMATIZACIÓN DE TAREAS .....</b>	<b>187</b>
Objetivos .....	187
Mapa conceptual .....	188
Glosario .....	189
6.1. Introducción .....	189
6.2. Rutinas de base de datos .....	190
6.3. Rutinas internas en Oracle .....	192
6.3.1. Permisos .....	192
6.3.2. Procedimientos .....	193
6.3.3. Funciones .....	195
6.3.4. Disparadores .....	196
6.3.5. Estructuras de programación .....	197
6.3.6. Llamadas a funciones y procedimientos .....	204
6.4. Automatización de tareas .....	205
6.4.1. Tareas de administración automatizables .....	206
6.4.2. Copias de seguridad .....	207
6.4.3. Particionamiento .....	207
6.4.4. Ejecución de estadísticas .....	210
6.4.5. Desfragmentación .....	210
6.4.6. Reconstrucción de índices .....	211
6.4.7. Purgado y paso a histórico .....	211
6.5. Automatización de tareas en Oracle .....	212
6.5.1. Herramientas del SO .....	212
6.5.2. DBMS Scheduler .....	212
Resumen .....	214
Ejercicios propuestos .....	215
Actividades de autoevaluación .....	216
<b>7. ALTA DISPONIBILIDAD .....</b>	<b>219</b>
Objetivos .....	219
Mapa conceptual .....	220



Glosario .....	220
7.1. Introducción .....	221
7.2. Sistemas gestores de bases de datos en la nube .....	221
7.2.1. Características .....	222
7.2.2. Ventajas e inconvenientes .....	222
7.3. Oracle en la nube .....	223
7.4. Sistema de gestión de base de datos distribuidos .....	224
7.4.1. Características .....	225
7.4.2. Ventajas e inconvenientes .....	225
7.4.3. Tipos de sistemas gestores distribuidos .....	226
7.4.4. Fragmentación y replicación .....	227
7.5. Distribución de información en Oracle .....	232
7.5.1. Fragmentación .....	232
7.5.2. Replicación .....	234
7.6. Clúster de servidores .....	235
7.6.1. Características .....	236
7.6.2. Ventajas e inconvenientes .....	237
7.7. El clúster de Oracle .....	237
Resumen .....	238
Ejercicios propuestos .....	239
Actividades de autoevaluación .....	240



## PRÁCTICAS GUIADAS

- 1.1. Instalación SGBD Oracle
- 2.1. Configuración de la conexión
- 2.2. Parámetros de configuración
- 2.3. Arranque y parada y configuración del almacenamiento
- 3.1. Usuarios externos
- 4.1. Encriptación
- 4.2. Auditoría
- 4.3. Copias de seguridad exp
- 4.4. Copias de seguridad datapump
- 4.5. Copias de seguridad RMAN
- 5.1. Monitorización con EM
- 5.2. Monitorización con Developer
- 5.3. Optimización de la base de datos
- 5.4. Optimización de consultas
- 6.1. Ejemplos PLSQL
- 6.2. Procedimientos de base de datos
- 6.3. Automatización de tareas con SQL Developer
- 7.1. Base de datos en la nube
- 7.2. Replicación con Oracle Golden Gate

## RECURSOS DIGITALES

- 1.1. Base de datos transaccional vs data warehouse
- 1.2. Arquitectura de Oracle

**ACTIVIDADES PROPUESTAS**

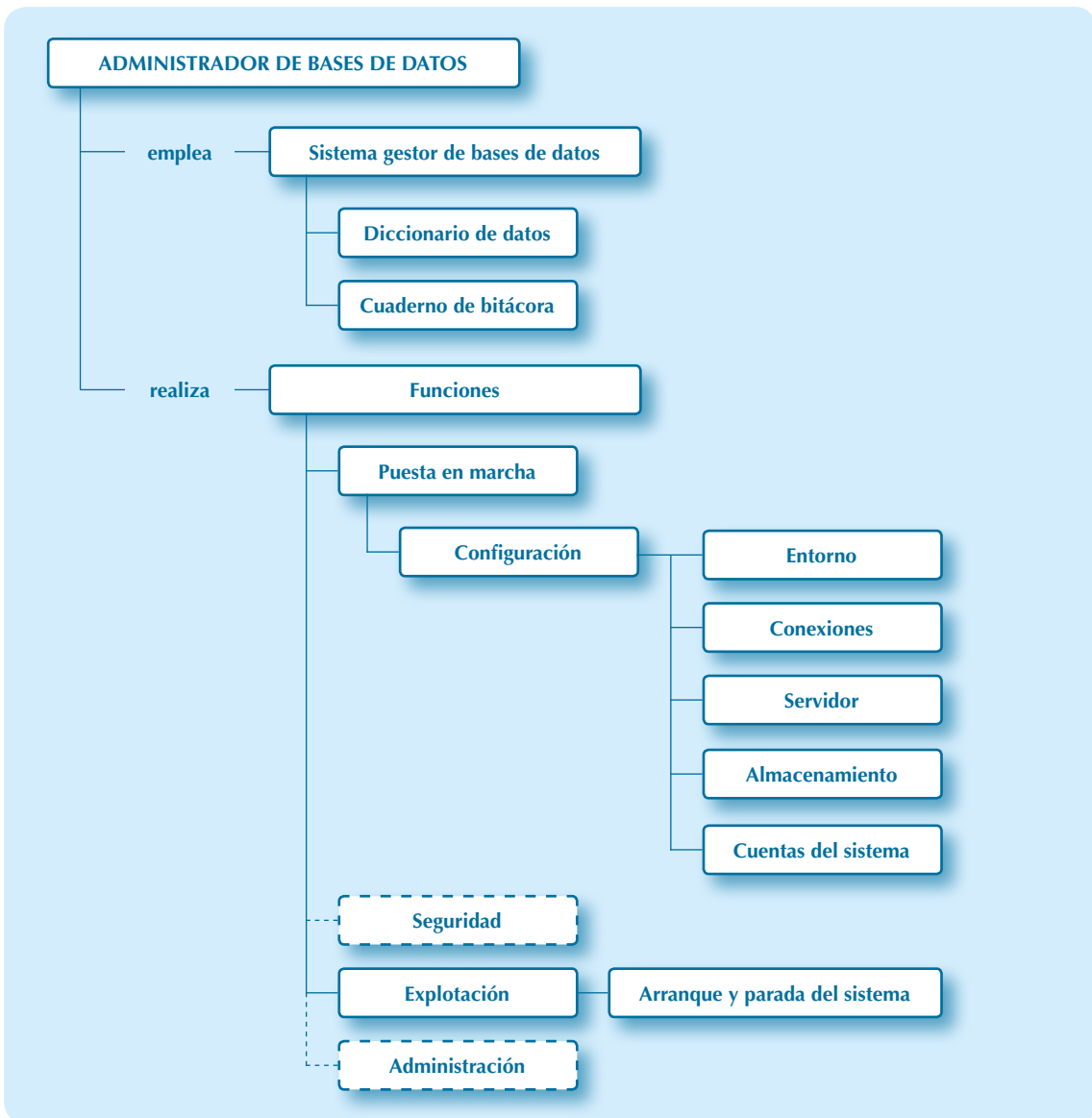
- 1.2. Elección del SGBD
- 3.1. Gestión de usuarios y permisos
- 3.2. Esquemas externos
- 4.1. Encriptación
- 5.1. Creación de índices
- 6.1. Procedimientos de administración
- 6.2. Automatización de tareas
- 7.1. Fragmentación y replicación
- 7.2. Distribución de datos en Oracle

# Configuración del sistema gestor de bases de datos

## Objetivos

- ✓ Conocer los primeros pasos que se deben llevar a cabo tras la instalación del sistema: qué elementos es necesario configurar y cómo realizar la puesta en marcha del sistema.
- ✓ Comprender la importancia y el funcionamiento de dos elementos básicos del sistema gestor de base de datos como son el diccionario de datos y el cuaderno de bitácora.
- ✓ Considerar la importancia de documentar adecuadamente toda la información referente a la instalación y configuración del sistema gestor.

## Mapa conceptual



## Glosario

**Cuaderno de bitácora.** Es un fichero (o más habitualmente un conjunto de ficheros) del sistema gestor donde este registra todas las operaciones de manipulación de datos ejecutadas, así como información de gestión como las copias de seguridad realizadas o marcas de inicio y fin de transacción.

**Datafile.** Archivo de datos, estructura física donde se almacenan realmente los datos. Un tablespace puede tener uno o varios datafiles.

**Esquema.** El esquema de una base de datos define la estructura de la base de datos, y se almacena en el diccionario de datos. Está compuesto de un conjunto de objetos de bases de datos (tablas, índices, procedimientos, vistas, etc.) relacionados entre sí. Todas las tablas se deben crear dentro de un esquema, y para acceder a una tabla debemos indicar el esquema al que pertenece (por ejemplo: ventas.clientes sería la tabla “clientes” del esquema “ventas”).

**Log.** Grabación secuencial en un archivo o en una base de datos de todos los acontecimientos (eventos o acciones) que afectan a un proceso particular.

**PATH.** Variable de entorno en la que se especifican las rutas en las cuales el intérprete de comandos debe buscar los programas a ejecutar. Si a esa variable de entorno se le añade una carpeta nueva, los programas que estén en esa carpeta se podrían ejecutar en el interfaz de comandos desde cualquier ruta, ya que aunque el programa no esté en la carpeta actual, el sistema operativo irá a buscarlo en todas las carpetas incluidas en la variable PATH.

**Pluggable Database (PDB).** Es uno de los componentes de la arquitectura de Oracle: la estructura física donde se almacenan los esquemas y datos de los usuarios. De denomina “pluggable” (podría traducirse por “intercambiable”) ya que están especialmente diseñadas para que sea muy sencillo moverlas a otro servidor en caso de que fuese necesario.

**Oracle System Identifier (SID).** Es el identificador único de cada instancia del sistema gestor. En una organización con múltiples instancias instaladas, cada una de ellas tendrá un nombre único (su SID) que permitirá identificarla.

**Tablespace.** Estructura lógica de almacenamiento que engloba un conjunto de archivos donde se almacenan los datos de las tablas de las bases de datos. Permite abstraer los objetos lógicos del sistema gestor (tablas, índices, etc.) de sus detalles de almacenamiento.

**Variables de entorno.** Son variables dinámicas del sistema operativo que pueden afectar al comportamiento de los procesos en ejecución en un ordenador (un proceso en ejecución puede consultar el valor de una variable de entorno para conocer la ubicación de los archivos ejecutables del sistema operativo, o la ruta donde almacenar los ficheros temporales).

## 2.1. Introducción

Tras seleccionar el SGBD y realizar la instalación, tanto del sistema gestor como de las aplicaciones clientes que sean necesarias para poder gestionarlo, es necesario configurarlo.

Los pasos a llevar a cabo para configurar adecuadamente el servidor serán muy dependientes tanto del sistema gestor de que se trate como sobre todo de su arquitectura, pero por lo general hay varios aspectos que es necesario configurar en todos ellos. En este capítulo veremos estos aspectos generales que es necesario configurar y comprobaremos su aplicación práctica en un SGBD concreto: Oracle.

## 2.2. Configuración del sistema gestor de bases de datos

La mayoría de sistemas gestores existentes en entornos de producción son sistemas con arquitectura cliente/servidor, por lo que las tareas iniciales de configuración se centrarán en identificar al servidor en la red, permitir las conexiones desde los clientes y habilitar las cuentas necesarias para permitir dichas conexiones.

### 2.2.1. Configuración del entorno

La configuración del entorno se centra básicamente en dos aspectos:

- Asignar un nombre y dirección IP al servidor donde se alojará el sistema (que debería ser preferiblemente una dirección estática)
- Crear o modificar las variables de entorno que sean necesarias para su correcto funcionamiento. En ocasiones la actualización de las variables de entorno se realiza automáticamente durante el propio proceso de instalación, pero en otros casos es necesario hacerlo a mano. La modificación más habitual consiste en añadir al PATH del equipo (la lista de rutas o carpetas donde el sistema operativo buscará las aplicaciones que ejecutemos) la ruta donde ese encuentren tanto los ejecutables del servidor como los del cliente. Algunos sistemas gestores pueden requerir también la definición de variables propias adicionales donde almacenarán rutas a librerías que necesitan para su ejecución, la ruta base de instalación del servidor, etc.

### 2.2.2. Configuración de las conexiones

La arquitectura más común entre los sistemas gestores de bases de datos multiusuario es la arquitectura cliente/servidor en 2 o 3 capas, por lo que lo primero que será necesario configurar será la conexión entre el cliente y el servidor, para poder emplear el cliente posteriormente para realizar las modificaciones necesarias en el servidor.

Los parámetros básicos que suele ser necesario especificar para establecer la conexión son los siguientes:

a) En el servidor:

- El puerto donde escuchará en espera de conexiones de los clientes.
- El protocolo de comunicación que se utilizará para gestionar la comunicación.
- Aspectos de configuración adicionales que afecten a la conexión (que dependerán mucho del SGBD utilizado): el modo en que se autenticará a los usuarios (si la realizará el propio SGBD, con LDAP, con radius, etc.), parámetros necesarios para establecer una conexión cifrada...

b) En el cliente:

- El nombre o dirección IP del servidor.
- El puerto del servidor al que deberá conectarse el cliente.
- El protocolo de comunicación que se utilizará.

- Las credenciales (usuario y password) que empleará para identificarse una vez establecida la conexión.

### 2.2.3. Configuración del servidor

Normalmente el servidor dispone de un fichero de configuración donde se guardan todos los parámetros de configuración del mismo. Cuando se arranca el servidor, este lee dicho fichero y asigna a cada parámetro el valor que le corresponda. Los parámetros que se almacenan en el fichero de configuración dependerán especialmente de la complejidad del SGBD (cuanto más complejo sea, más parámetros tendrá y más específicos serán).

Hay, sin embargo, algunos parámetros que suelen ser comunes en todos los sistemas:

- Ruta de los ficheros de control.
- Ruta de los ficheros de datos (donde se almacenan los datos de las tablas).
- Ruta de los ficheros de LOG del sistema.
- Aspectos relativos al control de las conexiones de los clientes: tiempo de timeout, número máximo de conexiones soportadas, máximo número de conexiones por usuario, etc.



Existen parámetros cuya modificación puede tener efecto inmediato, pero hay otros que requieren el reinicio del sistema gestor para pasar a ser efectivos, ya que al reiniciar el gestor este cargaría de nuevo el fichero de parámetros con los nuevos valores que se hayan especificado.

### 2.2.4. Configuración del almacenamiento

Para abstraer los detalles físicos de almacenamiento de los objetos del SGBD y así facilitar el mantenimiento de las tablas y controlar el espacio ocupado por cada esquema, los SGBD suelen incorporar algún tipo de elemento lógico que facilite la gestión de los distintos ficheros de cada esquema. Estas estructuras lógicas se suelen denominar espacios de tablas o tablespaces (es así en Oracle, PostgreSQL o MySQL), pero pueden tener otros nombres (como los Filegroup de SQL Server).

Lo habitual es dedicar un tablespace a cada esquema, pero esto no tiene por qué ser así (pueden incluirse datos de múltiples esquemas en un mismo tablespace). El uso de tablespaces tiene varias ventajas:

- Si el almacenamiento trabajase a nivel físico y tuviésemos toda la información en un mismo fichero, en caso de quedarnos sin espacio y tener que ampliar su capacidad nos obligaría a detener el sistema mientras se realiza la ampliación o copia del fichero a otro disco, dejando la base de datos inaccesible durante el tiempo que durase esa operación. Utilizando tablespaces, para incrementar el espacio del tablespace solo habrá que crear

un fichero nuevo (puede estar en otro disco) con el tamaño que queramos ampliar y añadirlo al tablespace, sin afectar en absoluto a la disponibilidad del sistema.

- Permiten gestionar conjuntamente todos los ficheros físicos de un esquema, lo que es muy útil para hacer copias de seguridad, copiar datos, controlar el espacio ocupado por cada esquema, etc.

## Ejemplo

La funcionalidad que aporta el incorporar un elemento lógico que permita abstraer los detalles de almacenamiento para una tabla podría equipararse a la funcionalidad de un call center de atención al cliente, donde cada operador del call center sería el equivalente a un archivo de datos y el propio call center sería la estructura lógica equivalente al tablespace:

- Si los clientes llamasen directamente a operadores concretos, esto implicaría que tendrían que conocer con qué operador quieren hablar y en caso de que este estuviese ocupado o de vacaciones no podrían acceder al servicio. Del mismo modo, si los datos de una tabla se guardasen directamente en un fichero, en caso de que este estuviese lleno o no estuviese disponible, no sería posible acceder a los datos de la tabla.
- El uso del call center permite introducir una estructura “abstracta” para la gestión de llamadas de los clientes: si hubiese muchas llamadas y se saturase el servicio, solamente habría que contratar a más operadores para poder seguir dando el servicio correctamente. De la misma forma, si los archivos de datos de un tablespace estuviesen en un disco lleno, solamente habría que añadir un nuevo archivo de datos ubicado en otro disco al tablespace y podrían insertarse nuevos datos sin problema en la tabla.
- De cara a la organización interna de la empresa, también sería más sencillo gestionar a nivel de call center que hacerlo individualmente operador a operador: si queremos derivar los clientes a los que atiende un call center a otro situado en otro país, solamente sería necesario redirigir las llamadas dirigidas al call center, en lugar de tener que desviar teléfono a teléfono. Del mismo modo, si queremos hacer una copia de seguridad de varias tablas incluidas en un tablespace, es más sencillo copiar el tablespace que tener que buscar todos los archivos de datos en lo que están las tablas y copiarlas uno a uno.

Existen varios tipos de tablespaces:

- De sistema:* todos los SGBD tienen al menos un tablespace SYS o SYSTEM para almacenar toda la información del sistema (el diccionario de datos).
- Temporal:* cuando realizamos consultas pesadas sobre el SGBD, cuando se queda sin espacio en memoria para almacenar los resultados parciales de la consulta necesita guardar esta información en disco, para lo que usará el tablespace temporal. Es importante dimensionar adecuadamente este tablespace, ya que si los resultados de una consulta no caben en el tablespace temporal, la consulta no se podrá resolver. Cuando finaliza la consulta, el espacio que ocupaba se libera, por lo que en las tareas de revisión de espacio no se debe tener en cuenta este tablespace.
- De registro o de UNDO (de “deshacer”):* algunos SGBD disponen de un tablespace de UNDO para facilitar las operaciones de rollback. Cuando iniciamos una transacción,



al modificar los datos se almacenan en el tablespace de UNDO los datos previos a la modificación, de forma que si en algún momento se realiza un rollback, solamente es necesario recuperar dicha información para deshacer todas las operaciones que haya llevado a cabo la transacción (lo cual es inmediato).

- d) *De usuario*: son los tablespaces que se crean para almacenar la información de los esquemas. Cada SGBD maneja estos tablespaces a su manera: MySQL crea por defecto un tablespace para cada base de datos, Oracle y SQL Server guardan por defecto la información de todos los usuarios en un tablespace de usuario o principal, etc.

### 2.2.5. Configuración de las cuentas del sistema

Cuando se instala el SGBD, se crea automáticamente un usuario con permisos de (llámese) root, system, etc. Este usuario podrá realizar cualquier operación sobre el sistema gestor, por lo que será el usuario que tendremos que utilizar para conectarnos por primera vez al sistema gestor de bases de datos, configurarlo y crear el resto de usuarios que podrán trabajar con el sistema (algunos sistemas ya permiten crear usuarios durante el propio proceso de instalación).

Una peculiaridad relevante de algunos sistemas gestores es que permiten establecer conexiones con el usuario administrador desde el propio servidor sin necesidad de contraseña, de forma que si perdistes la contraseña del administrador podrías conectarte al servidor para recuperarla (siempre y cuando lo hagas desde el propio servidor).

## 2.3. Arranque y parada del sistema

Actualmente la mayoría de sistemas gestores permiten realizar su instalación como servicios, de modo que no es necesario realizar ninguna configuración adicional para que estos se arranquen automáticamente.

Para parar o arrancar normalmente el sistema gestor solamente será necesario reiniciar el servicio, aunque en ocasiones será necesario arrancarlo a mano para poder realizar algunas tareas de administración específicas que necesitan que el sistema esté activo para poder hacer los cambios necesarios, pero que no esté abierto y aceptando conexiones de los usuarios ya que eso impediría realizar dichos cambios. Esto sucede por ejemplo si queremos mover los ficheros donde se almacenan los datos de una tabla: si el sistema está abierto estará usando el fichero, por lo que el sistema operativo no nos permitirá moverlo. Para poder mover el fichero tendríamos que detener el sistema gestor, arrancarlo en un estado intermedio en que no acceda a los ficheros de datos, mover el fichero en el sistema operativo y actualizar la configuración del sistema gestor para indicarle la nueva ruta donde tendrá que buscar el archivo cuando arranque normalmente.



---

Antiguamente la instalación del sistema gestor se llevaba a cabo como una aplicación particular, de modo que era necesario configurar explícitamente el inicio del servidor cuando se arrancase el equipo.

---

## 2.4. El diccionario de datos

El diccionario de datos es el repositorio de información donde se almacenan los metadatos del SGBD (toda la información referente a los objetos que contienen las bases de datos y su estructura).

El diccionario de datos suele implementarse por medio de una base de datos del sistema propiedad de un usuario con rol de DBA, de modo que únicamente ese usuario o aquellos a los que asigna permisos explícitamente puedan acceder a la información del sistema. En esta base de datos interna se guardarán todas las tablas, columnas, vistas, usuarios, permisos, índices, restricciones, procedimientos, etc. que hayamos creado en nuestras bases de datos.

El diccionario de datos es una de las herramientas más importantes del DBA a la hora de administrar el SGBD, ya que el hecho de disponer de toda la información del sistema gestor en una base de datos interna permite realizar consultas que monitoricen el funcionamiento del sistema, que busquen objetos del sistema que puedan estar en mal estado para repararlos, ficheros de almacenamiento con poco espacio libre para reservar más espacio, etc. Prácticamente todas las herramientas utilizadas para la gestión, monitorización y administración del SGBD internamente lo que hacen es realizar consultas sobre el diccionario de datos para obtener la información necesaria para cada tarea.

Es importante resaltar que la información del diccionario de datos no debe modificarse directamente en ningún caso, ya que ello podría provocar inconsistencias en la base de datos. Si por ejemplo quiero borrar una columna de una tabla, no puedo borrarla directamente de la tabla de COLUMNAS, ya que esa columna puede estar en otras tablas del diccionario de datos (en la tabla de CLAVES\_FORANEAS, la de VISTAS, etc.) que quedarían inconsistentes.

### Ejemplo

Cuando creamos una tabla como la siguiente en la base de datos:

```
CREATE TABLE ventas.clientes(
    idcliente number primary key,
    nombre varchar(100) not null,
    apellidos varchar(200) not null,
    telefono char(9) unique
);
```

Internamente, el sistema gestor creará un nuevo registro en cada una de las tablas del diccionario de datos donde se almacena la información referente a todos los objetos que estamos creando:

- En la tabla TABLAS se insertará un registro correspondiente a la tabla nueva.

idtabla	tabla	esquema	num_filas
1	CLIENTE	VENTAS	0

**Figura 2.1**  
Ejemplo tabla "TABLAS" del diccionario de datos.

- En la tabla COLUMNAS se insertará un registro por cada una de las columnas de la tabla.

idcolumna	idtabla	columna	tipo_datos	PK	NULL
1	1	IDCLIENTE	NUMBER	Y	N
2	1	NOMBRE	VARCHAR	N	N
3	1	APELLIDOS	VARCHAR	N	N
4	1	TELEFONO	CHAR	N	Y

**Figura 2.2**  
Ejemplo tabla  
"COLUMNAS"  
del diccionario  
de datos.

- Se insertará en la tabla de RESTRICCIONES un registro por cada una de las restricciones existentes en la tabla.

idrestric	idcolumna	restriccion
1	4	UNIQUE

**Figura 2.3**  
Ejemplo tabla "RESTRICCIONES"  
del diccionario de datos.

- Si el sistema gestor crea automáticamente índices en las claves primarias, por ejemplo, crea el índice y guardará un registro en la tabla INDICES correspondiente a ese índice.

idindice	idtabla	idcolumna	unique
1	1	1	Y

**Figura 2.4**  
Ejemplo tabla "INDICES" del diccionario de datos.

## 2.5. El cuaderno de bitácora

El cuaderno de bitácora es un log donde se registran todas las operaciones de modificación de datos ejecutadas sobre la base de datos. Recibe su nombre de los antiguos cuadernos de bitácora donde los capitanes del barco registraban todos los acontecimientos relevantes sucedidos durante una travesía.

Cuando se ejecuta una transacción sobre el sistema gestor, este inserta en el cuaderno de bitácora:

- un registro de inicio de transacción;
- una entrada por cada operación de la transacción que modifique los datos existentes;
- una señal de fin de transacción si esta finalizó correctamente.

Se registran también en el cuaderno de bitácora todas las copias de seguridad y restauraciones de datos ejecutadas.

Este fichero de log es especialmente importante porque se utiliza en algunas tareas muy importantes, como:

- *Gestión de transacciones*: todas las operaciones realizadas por una transacción se registran en este fichero, de modo que si ejecutamos un rollback y el sistema gestor tiene que deshacer los cambios realizados hasta ese momento, consultará en el cuaderno de bitácora las operaciones que tiene que deshacer.
- *Recuperación del sistema ante caídas*: si se cae el sistema y tenemos que recuperar la última copia de seguridad, tendremos que volver a realizar todas las operaciones ejecutadas desde el momento en que se hizo la copia de seguridad para volver al estado anterior a la caída del sistema.
- *Replicación del SGBD*: para replicar una base de datos, normalmente lo que hace el sistema gestor es trasladar al sistema gestor de la réplica todas las operaciones que se registren en el cuaderno de bitácora, de forma que cada operación que se ejecute en el servidor principal, se ejecutará también en la réplica. Si una réplica está temporalmente fuera de servicio, cuando estuviese de nuevo online se le enviarían todas las instrucciones del log que tiene pendiente de ejecutar y de ese modo se actualizaría automáticamente.

## Ejemplo

< copia seguridad YYYY-MM-DD hh:mm:ss >
TX0023 - inicio transaccion
TX0023 - update impuestos set iva=0,21 where iva=0,16
TX0023 - update producto set importe_iva=pvp*0,21
TX0023 - update producto set pvp_iva=pvp+importe_iva
TX0023 - confirmación transacción

**Figura 2.5**  
Ejemplo cuaderno de bitácora.

## 2.6. Documentación

Un administrador de sistemas debe almacenar información detallada y actualizada de todos los servidores de la organización y de los servicios que proporcionan.

En el caso de un servidor de bases de datos, además de guardar los datos correspondientes al hardware (capacidad de disco, memoria principal y extendida, placa base, etc.) y al software de este (sistema operativo y programas en ejecución), se debe guardar también información de:

1. Versión y todas las actualizaciones realizadas de todos los programas del servidor (incluido el SGBD). Es muy importante mantener esta información actualizada porque será imprescindible tenerla a mano en caso de que sea necesario contactar con el soporte del sistema gestor.
2. Información de las bases de datos que estén instaladas en el servidor: modelo Entidad-Relación, esquema relacional y descripción de los objetos de la base de datos (tablas, columnas de cada tabla, restricciones, funciones, disparadores).
3. Usuarios de la base de datos y permisos de cada uno de ellos.

## TOMA NOTA



Hay programas específicos que ayudan a realizar la documentación de las bases de datos (como Mogwai, ERWIN o el propio MySQL Workbench), ya que permiten generar automáticamente los modelos lógico y relacional de las bases de datos, así como documentar la información de todos los objetos de la base de datos.

Aparte de la documentación propia para el mantenimiento del sistema gestor, es responsabilidad del DBA definir o al menos participar en la definición de una serie de procedimientos que garanticen que se cumplen las políticas de la organización y faciliten la administración de los sistemas (especialmente en organizaciones grandes):

- a) Procedimientos de copia de seguridad y restauración: es muy importante que existan estos documentos porque un fallo del sistema gestor suele generar situaciones de mucho estrés en las que es mucho más adecuado seguir una guía de pasos a ejecutar para la restauración de los datos que tener que pensar en cada momento qué paso debo ejecutar a continuación y cómo debo hacerlo.
- b) Procedimientos de registro de datos: para garantizar el cumplimiento de la Ley Orgánica de Protección de Datos (LOPD) es necesario llevar a cabo una serie de tareas cada vez que se vayan a registrar datos de carácter personal. El procedimiento para llevar a cabo estas tareas debería estar documentado para agilizarlo.
- c) Definición y organización de simulacros: cuando existen sistemas de réplica y respaldo es necesario planificar y realizar periódicamente simulacros para garantizar que en caso de desastre la recuperación sea lo más rápida posible. Esta tarea suele ser responsabilidad del administrador de sistemas, pero el DBA tendrá también un rol importante.
- d) Procedimiento de alta de usuarios y permisos: para facilitar el mantenimiento del sistema de usuarios y permisos, el alta de usuarios y la asignación de permisos deberían estar procedimentados para facilitar su mantenimiento.
- e) Guía de estilo para la creación de bases de datos: en una organización que disponga de múltiples esquemas de bases de datos, si estos son muy heterogéneos (cada cual con su diseño y nomenclatura particular), complica mucho su mantenimiento, por lo que debería definirse un procedimiento para que todos los esquemas, tablas, tablespaces, usuarios, etc., se ajusten a una nomenclatura común que sea significativa.

## 2.7. Configuración de Oracle

El servidor Oracle tiene una arquitectura cliente/servidor, por lo que, tal como se comentó anteriormente, será necesario realizar una configuración inicial del servidor y permitir la conexión desde las aplicaciones clientes.

### 2.7.1. Configuración del entorno

El asistente de instalación de Oracle para entornos Windows ya incluye, como un paso más del proceso de instalación, la configuración del entorno, por lo que no sería necesario realizar ninguna intervención adicional en este sentido.

En entornos Linux/Unix, tendríamos que configurar manualmente el entorno de ejecución del servidor Oracle, lo que implica:

- Definir las variables de entorno: es necesario definir tres nuevas variables (ORACLE\_HOME, LD\_LIBRARY\_PATH y ORACLE\_SID) y modificar el PATH del sistema:

```
export ORACLE_HOME=(ruta hasta la carpeta dbhome)
export LD_LIBRARY_PATH="$ORACLE_HOME/lib:$ORACLE_HOME"
export ORACLE_SID=(nombre de la instancia, por defecto orcl)
export PATH="$ORACLE_HOME/bin:$PATH"
```

- Tendremos que dar permisos de lectura y ejecución a todos los usuarios (o al grupo de usuarios de instalación) en los directorios \$ORACLE\_HOME/lib, \$ORACLE\_HOME/bin, \$ORACLE\_HOME/oracore y \$ORACLE\_HOME/sqlplus.

## 2.7.2. Configuración de las conexiones

Oracle utiliza una arquitectura cliente/servidor en 2 capas, por lo que, para poder establecer conexiones, tendremos que configurar tanto el cliente como el servidor.

Oracle dispone de una aplicación que permite realizar la configuración de red por medio de un interfaz gráfico: el asistente de configuración de red de Oracle. Esta aplicación está escrita en Java, por lo que se puede utilizar tanto en un entorno Windows como Linux, y consisten en un asistente que facilita la modificación del contenido de los ficheros de configuración de red de Oracle: el tnsnames.ora, el sqlnet.ora y el listener.ora. Todos estos ficheros de configuración de red se almacenan en la carpeta “ORACLE\_HOME/network/admin”.



### TEN EN CUENTA

El formato de las entradas de los ficheros de configuración de red de Oracle es MUY delicado: es muy habitual tener problemas de conexión debidos a la presencia de espacios o tabuladores donde no corresponde, que son muy difíciles de detectar. Para añadir nuevas entradas en estos ficheros, se recomienda para evitar problemas partir siempre una entrada existente y copiarla y pegarla modificando únicamente los valores necesarios para la nueva conexión (o usar el asistente de configuración de red, que ya nos garantiza que no habrá errores de este tipo en las entradas que defina automáticamente).

## A) Configuración del servidor

Para que los clientes puedan conectarse al servidor de Oracle, será necesario configurar dos aspectos en el servidor:

### 1. Dónde se deben conectar los clientes

El *listener* es el programa que se ejecuta en el servidor y se encarga de escuchar en un puerto concreto para atender las peticiones de los clientes que llegan a ese puerto. Este programa se