

# Bibliotecas Compartidas

## Bibliotecas Compartidas - ¿Qué son?

Las Bibliotecas compartidas son ficheros que contienen colecciones de fragmentos de código, que nos permite el poder utilizarlos en el nuestro código o/y por los programas que tenemos instalados en nuestra máquina, sin tener que volver a programar ciertas tareas rutinarias (p.e. crear ventanas, botones, acceso a dispositivos, ... ).

Además de ahorrarnos tiempo a la hora de desarrollar aplicaciones, y posibles errores, estas bibliotecas al ser compartidas se instalan en nuestra máquina y como su nombre indica son o pueden ser compartidas por varios programas, consiguiendo así:

- Ahorrar espacio de almacenamiento
- Se cargan una sola vez en memoria, independientemente de las aplicaciones que las requieran

Cada aplicación suele utilizar una versión concreta de cada biblioteca compartida, pero esto en Linux no es ningún problema ya que Linux utiliza un sistema de numeración de versiones, que posibilita el tener instaladas varias versiones de cada biblioteca.

Las aplicaciones deben poder encontrar las bibliotecas, por lo que los cambios deben ser reflejados en unos ficheros de configuración, que son los que las aplicaciones consultan para poder localizarlas.

Si una biblioteca deja de estar accesible, las aplicaciones fallaran, o incluso lo hará el sistema completo.

Todo esto nos lleva a que debemos tener un buen control de nuestras bibliotecas, que normalmente no nos causará problemas si utilizamos el gestor de paquetes correspondiente a nuestra distribución (y no utilizamos opciones que ignoren dependencias).

Los nombres de las bibliotecas compartidas en Linux terminan en `.so` (shared object, objeto compartido).

## Rutas de los ficheros de Biblioteca

Algunas de las rutas son tenidas en cuenta por Linux aunque no las especifiquemos como `/lib` y `/usr/lib`

Existen varias formas de definir la ruta a los ficheros de biblioteca.

Editando el fichero `/etc/ld.so.conf`. En este fichero se encuentran algunas de las rutas a bibliotecas, una por línea.

En `/etc/ld.so.conf` pueden existir líneas `include` que como saben los programadores de C indica que se debe incluir el contenido de otro/s fichero/s, p.e. en Ubuntu contiene una línea `include /etc/ld.so.conf.d/*.conf` que indica que se deben incluir todos los ficheros que terminen en `.conf` dentro del directorio `/etc/ld.so.conf.d/` y estos a su vez pueden contener otras líneas `include`. En otras distribuciones se adoptan soluciones parecidas aunque con algunas pequeñas

diferencias.

Linux en una de sus tareas de arranque lee todos los ficheros que puedan estar involucrados en la definición de rutas a bibliotecas (incluidas aquellas que tiene siempre `/lib /usr/lib`). y genera un sistema de datos mas optimizado que los ficheros en texto plano que se utilizan para configurar las rutas, por eso cuando modifiquemos un fichero de en el que se encuentren rutas a bibliotecas debemos ejecutar la herramienta `ldconfig` que se encarga de leer los ficheros de configuración y volver a generar este sistema de datos optimizado con las rutas de las bibliotecas.

Normalmente no tendremos que realizar esta tarea, ya que utilizaremos el gestor de paquetes de nuestra distribución el cual, normalmente, realiza todas las acciones necesarias al instalar un nuevo paquete. Estas tareas solamente nos veremos obligados a hacerlas cuando instalemos alguna biblioteca compilada por nosotros mismos, o en circunstancias poco habituales.

## Cambiar la Ruta de Bibliotecas de Forma Temporal

Imaginemos que queremos utilizar temporalmente una biblioteca, p.e. para comprobar su efecto en el sistema, podemos guardarla en una carpeta que tenemos para estos temas quizá `~/pruebas/lib` (recordad que `~` hace referencia a la carpeta home de usuario). Ahora que ya tenemos localizada la ruta para probar nuestra librería, podemos utilizar la variable de entorno `LD_LIBRARY_PATH` las rutas añadidas de este modo serán colocadas al principio de la cadena de búsqueda interna para las librerías por lo que siempre se utilizarán antes que las que se estaban utilizando hasta el momento, si es que sustituye a alguna.

El comando a utilizar sería

```
export LD_LIBRARY_PATH=~/pruebas/lib/libreria1:~/pruebas/lib/libreria2
```

Como podemos ver se pueden poner tantas rutas como necesitemos separadas por `:"`, no tienen porque estar en la carpeta de usuario, en principio somos libres de utilizar la ruta que necesitemos, siempre y cuando tengamos permiso para acceder a ella.

Para esta tarea no es necesario ejecutar `ldconfig` para actualizar las rutas.

## Detectar y Corregir Problemas con las Rutas de las Bibliotecas

Si estamos utilizando un entorno gráfico, si algún programa no se ejecuta por problemas con las librerías, normalmente no veremos la razón del fallo. Lo podremos averiguar de varias formas pero normalmente en nuestra consola, podemos escribir el nombre el programa para ejecutarlo y si se produce un error por la falta de algunas librerías nos devolverá un mensaje indicándonos que librerías nos faltan, o que su ruta no está registrada correctamente.

Supongamos que un programa nos devuelve un error indicándonos que no encuentra la librería `"libreriaQueFalta.so"`, si normalmente utilizamos el gestor de paquetes, esto no debería suceder.

Primero buscaríamos el fichero a ver si esta instalado, p.e. podemos utilizar `find` y si no lo encontramos buscaremos el paquete al que pertenece y lo instalaremos con el gestor de paquetes.

Si el fichero de biblioteca lo tenemos instalado, probablemente el problema esté en que no se encuentra en la ruta que el programa espera, podremos averiguar cual es esta ruta que el programa espera ejecutando `ldd /ruta/al/programa/programa`, también puede suceder que tengamos instalada una revisión de la versión que no es exactamente la que el programa espera, si esta es una revisión menor p.e. tenemos la versión `libreriaQueNecesitamos.so.3` y la que tenemos es la `libreriaQueNecesitamos.so.3.4`, por norma estas

revisiones solamente tienen correcciones menores, por lo que podremos utilizarla.

Tanto si nuestro problema es que la ubicación no es la que espera el programa como si es que la versión de la que podemos disponer es una revisión menor (siempre si la que tenemos es superior a la esperada) la solución es la misma:

Crear un enlace simbólico como root y encontrándonos en el directorio donde tengamos guardada la biblioteca

```
sudo ln -s libreriaQueNecesitamos.so.3.4 libreriaQueNecesitamos.so.3
```

ó en su caso desde la carpeta donde se espera que este la biblioteca utilizando la ruta completa de la biblioteca original

```
sudo ln -s /ruta/a/la/libreria/instalada/libreriaQueNecesitamos.so  
libreriaQueNecesitamos.so
```

Tras esto debemos ejecutar `ldconfig` para que se actualicen las rutas en el sistema.

Francisco Javier López Torrijos  
Analista Sistemas Informáticos de Gestión  
Diseño y Desarrollo Web

---

**URL de origen (modified on 06/14/2012 - 21:01):** <http://www.lopeztorrijos.com/node/38>