



LPIC-1. Examen 101. Objectiu 102.3

LPI102.3_Gestionar llibreries compartides

Wiki:

http://acacha.org/mediawiki/index.php/LPI_102.3

Objectius

102.3. Gestionar biblioteques compartides



- **Objectiu:** Els candidats han de ser capaços de determinar les llibreries compartides de les que depenen el programari executable i instal·lar-les quan sigui necessari.
- **Pes:** 1



Àrees Clau de Coneixement:

- Identificar les llibreries compartides.
- Identificar les ubicacions típiques per a les llibreries del sistema.
- Carregar llibreries compartides.



La següent és una llista parcial de fitxers, termes i utilitats utilitzades:

- [ldd](#)
- [ldconfig](#)
- [/etc/ld.so.conf](#)
- [LD_LIBRARY_PATH](#)



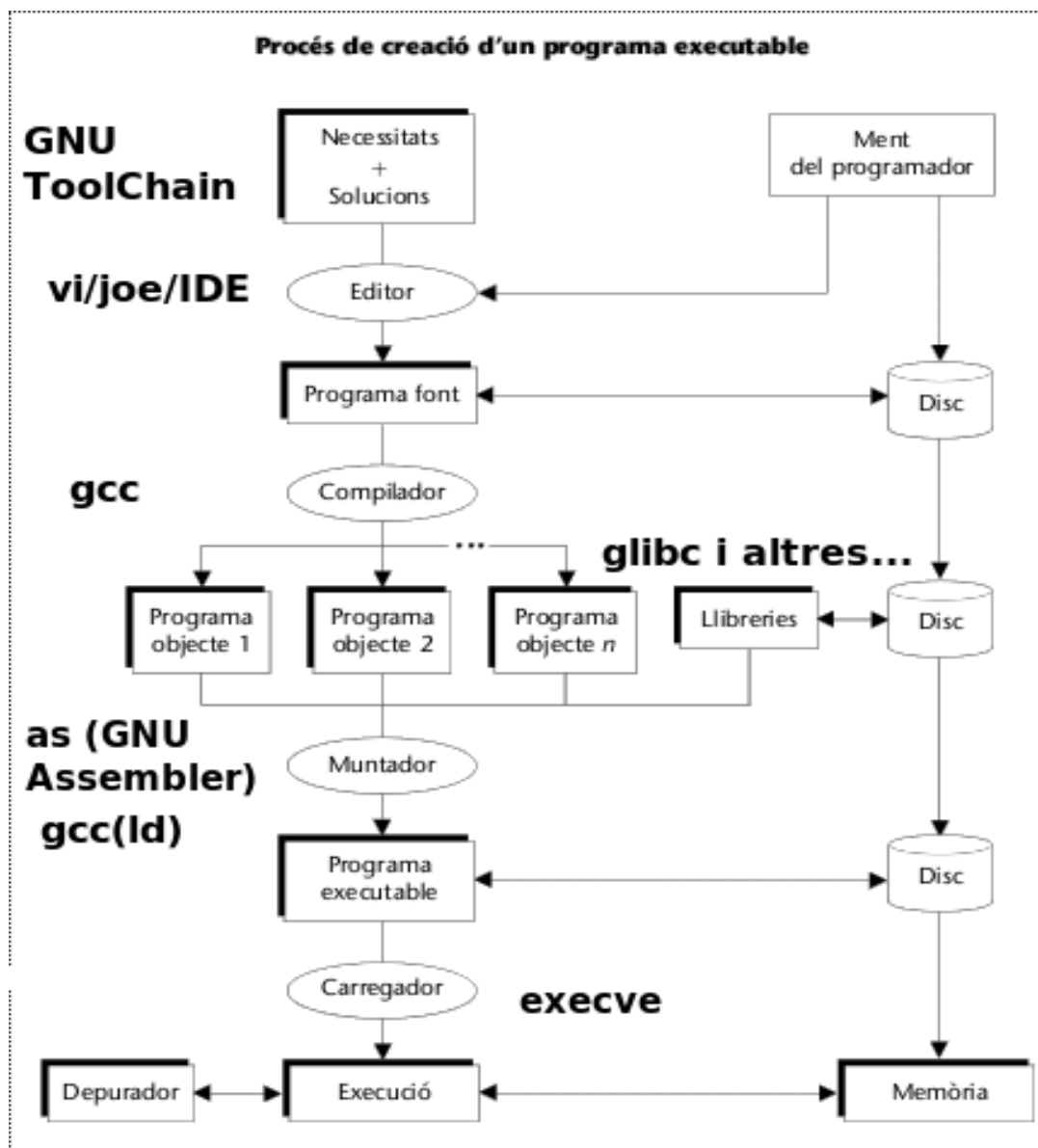
Apunts: [LPI 102.3](#). Gestionar biblioteques compartides

♦ 2 tipus de biblioteques

- ♦ **Estàtiques:** un arxiu amb un conjunt de rutines que es copien a una aplicació durant el **procés de compilació** a través del enllaçador (ld), també conegut com a **muntador**. Cada biblioteca produeix un **fitxer .o**
Construcció estàtica de l'aplicació objectiu.
- ♦ **Dinàmiques o compartides:** Enllaç dinàmic significa que **les rutines d'una biblioteca són carregades en un programa en temps d'execució**. El fitxer són **.so (shared objects)**. En aquest cassos l'enllaçador realitza una mínima quantitat de treball en temps de compilació i la major part de la labor d'enllaçat es realitza en el moment en què l'aplicació es carrega (temps de càrrega o loadtime) o durant l'execució (temps d'execució o runtime).

NOTA: Tingueu en compte que library és un típic **false friend** que sovint es tradueix per llibreria. Realment la traducció és biblioteca. De totes maneres és un error tant comú que “gairebé” es considera correcte. **Recull de termes de softcatalà**

Procés de compilació





Biblioteques compartides

♦ Shared Libraries a Linux

- ♦ La majoria d'aplicacions Linux tenen fortes dependències amb biblioteques compartides
- ♦ També són conegudes com a biblioteques dinàmiques.
- ♦ Al instal·lar una aplicació que depèn de biblioteques compartides, cal tenir instal·lada la biblioteca compartida i, sovint, a més, la versió exacte de la biblioteca adequada.
- ♦ L'objectiu de les biblioteques es **facilitar la tasca dels programadors** proveint mòduls, funcions o fragments de codi reutilitzables.
- ♦ Una de les biblioteques més importants és **libc**, la **biblioteca de C**

Biblioteques compartides

- ◆ Altres biblioteques típiques són les associades a entorns gràfics d'usuari (GUI) que proporcionen a les aplicacions el suport per a treballar amb finestres i ginyes (widgets) gràfics com desplegable, caixes de selecció, etc. Llibreries GTK+ (Gnome) i Qt (KDE)
- ◆ Cal tenir en compte que les biblioteques són seleccionades pels programadors de les aplicacions i no pas pels usuaris i per tant els usuaris no podem escollir en quina biblioteca volem treballar.
- ◆ Les biblioteques compartides tenen la extensió **.so** (shared object)

NOTA: El concepte de biblioteca compartida a Linux és similar al concepte de DLL a Windows



Biblioteques compartides

- ♦ Les rutines proporcionades per una biblioteca compartida, poden ser utilitzades en una aplicació com qualsevol altre biblioteca, però això té certs problemes:
 - Si cada aplicació conté una còpia de la biblioteca, estem malgastant espai i **augmenta la mida de les aplicacions**
 - No només **augmenta** la mida que ocupa l'aplicació, sinó també **l'ús de RAM**. Si varies aplicacions executen diversos cops el mateix tros de codi estem malgastant la RAM.
 - El programes no poden obtenir els **avantatges d'una actualització** de la biblioteca compartida sense recompilar (o relinking).

Per aquesta raó, la majoria de programes utilitzen biblioteques compartides, de forma que l'aplicació no inclou la majoria de funcions rutinàries, sinó que són les biblioteques compartides les que realitzen aquestes tasques, de forma que s'optimitza l'ús del

Biblioteques compartides

♦ Inconvenients

- ♦ El primer programa en utilitzar la biblioteca compartida és una mica més lent ja que ha de carregar aquesta biblioteca.
- ♦ Poden provocar complicacions en la gestió del programari (instal·lacions, actualitzacions, etc...)
 - Quan hi ha un canvi en una biblioteca compartida, aquest canvi pot provocar que algunes aplicacions que l'utilitzen deixin de funcionar per incompatibilitats.
 - Linux utilitza un **esquema de numeració** que permet als usuaris mantenir instal·lades al mateix temps múltiples versions d'una mateixa biblioteca compartida.
- ♦ Quan hi ha una actualització de la biblioteca que es sap que no té incompatibilitats, es modifica la biblioteca instal·lada al sistema.

Biblioteques compartides

- ♦ Quan es tracta d'una actualització major, amb canvis en el funcionament de la biblioteca, aleshores s'instal·la la nova versió mantenint també l'antiga versió.
- ♦ Això minimitza els problemes però encara i així, de tant en tant us podeu trobar errors amb biblioteques compartides.

```
$ ls -l /lib/libacl* | more
lrwxrwxrwx 1 root root      17 2009-11-15 11:33 /lib/libacl.a ->
/usr/lib/libacl.a
lrwxrwxrwx 1 root root      18 2009-11-15 11:33 /lib/libacl.la ->
/usr/lib/libacl.la
lrwxrwxrwx 1 root root      11 2009-11-15 11:33 /lib/libacl.so -> libacl.so.1
lrwxrwxrwx 1 root root      15 2009-07-27 09:31 /lib/libacl.so.1 ->
libacl.so.1.1.0
-rw-r--r-- 1 root root 26156 2008-05-03 02:01 /lib/libacl.so.1.1.0
```

Exemple de biblioteca compartida

- ♦ **Biblioteques per a treballar amb sistemes de fitxers ext:**

```
$ dpkg -L e2fslibs
/.
/lib
/lib/libext2fs.so.2.4
/lib/libe2p.so.2.3
/usr
/usr/share
/usr/share/doc
/usr/share/doc/e2fslibs
/usr/share/doc/e2fslibs/copyright
/usr/share/doc/e2fslibs/changelog.Debian.gz
```

- ♦ Al gestor de paquets Synaptic hi ha un apartat dedicat a les biblioteques

♦ Procés de localització

- ♦ Les aplicacions han de ser capaces de localitzar les biblioteques compartides de les que depenen. Aquesta tasca es realitza modificant **fitxers de configuració i variables d'entorn**.
- ♦ Els programes tenen dos opcions alhora de referir-se a una biblioteca compartida:
 - Utilitzar el nom de la biblioteca (p. ex. libc.so.6)
 - Utilitzar el camí (path) complet (p. ex. /lib/libc.so.6)
- ♦ El programa encarregat de carregar les biblioteques dinàmiques en temps d'execució és **ld.so**.

♦ Al manual de ld.so:

- ♦ 1. Utilitzar les variables d'entorn **LD_LIBRARY_PATH** o **LD_AOUT_LIBRARY_PATH** (està última només per a fitxers a.out). Si l'executable és un executable setuid/setgid aleshores aquesta variable s'ignora.
- ♦ 2. Es busca la biblioteca a la cache. La cache es troba a: **/etc/ld.so.cache**
- ♦ 3. Finalment es busca a les carpetes **/lib** i **/usr/lib** (que són les carpetes on s'han de trobar les biblioteques segons FHS)

ld.so

- ♦ **És l'aplicació que carrega les aplicacions i les seves biblioteques dinàmiques**
 - ♦ Realment és una crida de sistema. Secció 8 del manual

```
$ man ld.so
```
 - ♦ Hi ha dos:
 - /lib/ld.so.*: format a.out
 - /lib/ld-linux.so.* : format ELF

Exemple bàsic de compilació en C

♦ **Hola mon!**

♦ Codi font

```
$ sudo vi holamon.c:
#include <stdio.h>
int main (void) {
    printf ("Hello, world!\n");
    return 0;
}
```

♦ Compilació:

```
$ gcc -Wall holamon.c -o holamon
```

♦ Biblioteques dinàmiques

```
$ ldd hola
linux-gate.so.1 => (0x00e53000)
libc.so.6 => /lib/tls/i686/cmov/libc.so.6 (0x009ff000)
/lib/ld-linux.so.2 (0x00617000)
```

♦ Fins i tot les aplicacions més simple en Linux utilitzen llibreries dinàmiques

Ordre file

♦ Pot identificar les llibreries compartides

```
$ file /lib/libext2fs.so.2.4  
/lib/libext2fs.so.2.4: ELF 32-bit LSB shared object, Intel  
80386, version 1 (SYSV), dynamically linked, stripped
```

♦ 2 formats de fitxers:

- **ELF (Executable al linkable format):** és el més utilitzat actualment
- **a.out:** format antic dels executables

♦ Mostra les dependències de biblioteques compartides d'un fitxer

♦ Sintaxi és:

```
$ sudo ldd --help  
Usage: ldd [OPTION]... FILE...
```

♦ Opcions:

- **--version**: Mostra el número de versió de [[ldd]]
- **-v --verbose**: Mostra tota la informació
- **-d --data-relocs**: recol·loca fitxers i reporta objectes perduts (només ELF)
- **-r --function-relocs**: recol·loca fitxers tant per a objectes de dades com funcions, i reporta objectes perduts
- **-u --unused**: Mostra dependències directes que no s'estan utilitzant.

ldconfig

◆ Configura la memòria cau

- ◆ El que fa és configurar els enllaços simbòlics correctes per a les biblioteques compartides i torna a crear la cache. Les tasques que realitza són crear, actualitzar i/o eliminar els enllaços simbòlics necessaris per al correcte funcionament del enllaçador en temps d'execució (ld.so).
- ◆ Els enllaçadors dinàmics de Linux (ld.so i ld-linux.so) no llegeixen el fitxer /etc/ld.so.conf cada cop que un programa s'executa. El sistema utilitza una cache que es troba al fitxer:

```
/etc/ld.so.cache
```



ldconfig

- ◆ Es tracta d'una llista (en forma binari) que es molt més eficient que utilitzar una llista de text. El problema però es que si feu cap modificació en els camins de les biblioteques compartides aquesta cache s'ha de tornar a generar. Aquesta tasca la fa l'ordre ldconfig.
- ◆ Les biblioteques es busquen a:
 - A les carpetes especificades mitjançant opcions de l'ordre ldconfig
 - Als fitxers especificats a /etc/ld.so.conf
 - A les carpetes /usr/lib i /lib
- ◆ S'ha d'executar com a superusuari:
 - **\$ sudo ldconfig**
 - No news good news! Si voleu més info utilitzeu l'opció verbose



Fitxers de configuració

♦ **/etc/ld.so.conf**

- ♦ Estableix els paths principals (a més dels per defecte /lib i /usr/lib)

```
$ cat /etc/ld.so.conf
include /etc/ld.so.conf.d/*.conf
```

- ♦ Diversos fitxers:

```
$ ls /etc/ld.so.conf.d/*.conf
/etc/ld.so.conf.d/i486-linux-gnu.conf
/etc/ld.so.conf.d/libasound2.conf
/etc/ld.so.conf.d/libc.conf
```

♦ Canviar temporalment el PATH

```
$ export  
LD_LIBRARY_PATH=/usr/local/testlib:/opt/newlib
```

- ♦ Els canvis permanents es fan al fitxer **/etc/ld.so.conf**

♦ Corregir un problema de versions:

```
$ gimp  
gimp: error while loading shared libraries:  
libXinerama.so.1: cannot open shared object file:  
No such file or directory  
$ whereis gimp  
gimp: /usr/bin/gimp ...  
$ ldd /usr/bin/gimp  
$ sudo ln -s biglib.so.5.2 biglib.so.5
```




Reconeixement 3.0 Unported

Sou lliure de:



copiar, distribuir i comunicar públicament l'obra



fer-ne obres derivades

Amb les condicions següents:



Reconeixement. Heu de reconèixer els crèdits de l'obra de la manera especificada per l'autor o el llicenciador (però no d'una manera que suggereixi que us donen suport o rebeu suport per l'ús que feu l'obra).

- Quan reutilitzeu o distribuïu l'obra, heu de deixar ben clar els termes de la llicència de l'obra.
- Alguna d'aquestes condicions pot no aplicar-se si obteniu el permís del titular dels drets d'autor.
- No hi ha res en aquesta llicència que menyscabi o restringeixi els drets morals de l'autor.

Advertiment

Els drets derivats d'usos legítims o altres limitacions reconegudes per llei no queden afectats per l'anterior
Això és un resum fàcilment llegible del **text legal** (la llicència completa).

<http://creativecommons.org/licenses/by/3.0/deed.ca>