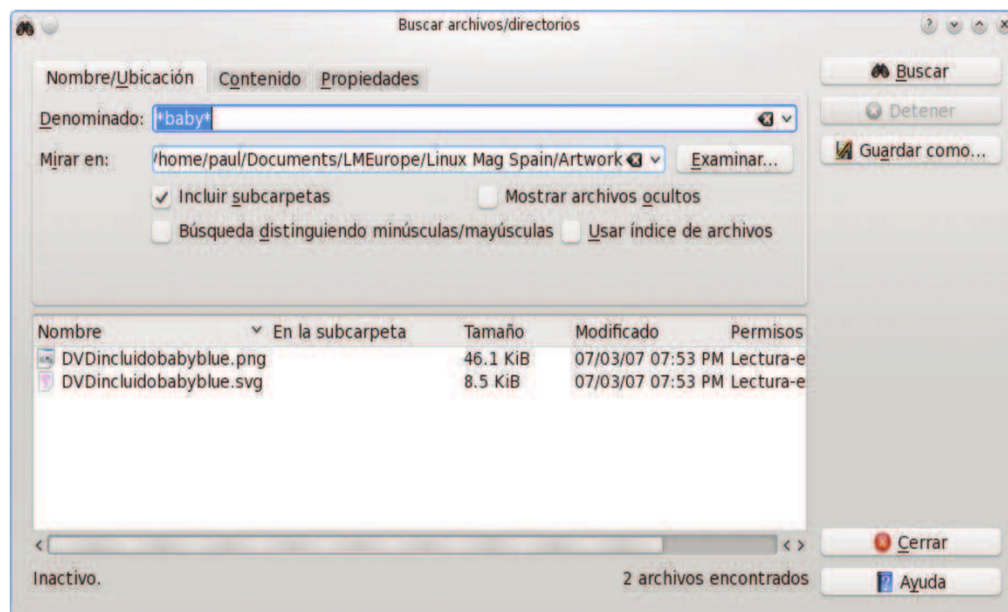




Find i Locate

Si estem buscant fitxers en Linux, la línia d'ordres és la millor opció per a uns resultats ràpids i fiables. Les interfícies gràfiques d'usuari com KFind de KDE (Figura 1), estan molt bé, però no tenen certes funcionalitats, flexibilitat ni velocitat comparables a les de la consola.



La comanda més important per a la recerca de fitxers és **find**. Sense especificar cap opció, l'eina es limitarà a buscar tots els fitxers sota l'actual directori. Si es vol buscar en qualsevol altre directori, simplement cal especificar el nom del directori com a primer argument. Per exemple, **find /home** mostrarà una llista de tots els fitxers i directoris sota **/home**.

Però find és realment útil quan comença a utilitzar-se amb opcions. L'argument **-name** permet als usuaris dir-li els fitxers a buscar que coincideixin amb el patró donat. Per exemple,

```
find ~ -name "*firefox*"
```

busca tots els fitxers al directori home de l'usuari (Recorda que ~ equival al teu directori d'usuari) que incloguin la cadena firefox en els seus noms.

Quan s'utilitzen comodins es recomanable emprar cometes per a tancar les expressions.

Opcions, Proves i Accions

find reconeix tres tipus d'arguments: **opcions**, **proves** i **accions**.

- Les "**opcions**" influeixen en el comportament de find en si mateix, per exemple, restringint el nombre de nivells de subdirectoris a buscar o restringint la cerca a una única partició.
- Les "**proves**" permeten a l'usuari restringir la cerca a fitxers específics, per exemple,

- o fitxers que tenen menys d'una setmana,
find -mtime -8,
- o fitxers que tenen més d'una setmana,
find -mtime +7.

Poden, també, restringir els comandaments a un tipus específic:

- o **-type f** (per buscar només fitxers),
- o **-type d** (per buscar només directoris) o
- o **-type l** (per buscar només enllaços).

Un administrador també ha de buscar fitxers que pertanyen a usuaris o grups inexistents per procedir a la seva neteja. El següent comanda s'ocuparà de realitzar aquesta tasca:

find -nouser -o -nogroup -o potser ser -or

L'opció **-o** és l'operador lògic OR, que enllaça les dues expressions, **-nouser** i **-nogroup**. Un altre truc molt útil per als administradors és la recerca de fitxers per l'UID o GID usant

- Les **accions** influeixen en la forma que es mostren els resultats. L'acció **-exec ordre**, per exemple, passaria els noms dels fitxers trobats al programa comanda. Les accions també suporten operadors lògics, permetent als administradors especificar un comportament de sortides complex.

Exemples

find -perm +6000 -ls	El -ls li diu a aquesta comanda que executi ls -lsid per a cada fitxer.
find / -user alumne	Arxius que pertanyin a l'usuari que s'indiqui a continuació de -user
find / -group adm	Arxius que pertanyin al grup adm
find -mtime n	Arxius modificats fa n dies
find /home/usuari -mtime +n	Arxius modificats fa més de n dies
find / -mtime -n	Arxius modificats fa menys de n dies
find / -size -m find / -size +m	De mida menor a m blocs o major (bloc=512 bytes)
find \$HOME -size +1500 -mtime -5	Més gran de 1500 blocs i modificat fa menys de 5 dies. Fa la recerca a partir del directori \$HOME
find \$HOME -type f -a -user \$LOGNAME	Fitxers de l'usuari que ha iniciat sessió -and

Execució d'ordres

Com esmentem més amunt, **-exec** permet utilitzar noms de fitxers com arguments i fins i tot crear petits guions. Però cal anar amb compte amb els perills que això suposa: usar aquesta funcionalitat com a root en directoris públics (com /tmp o el directori /home) no és molt bona idea. Hi ha un retard entre la troballa dels arxius que coincideixen amb el paràmetre de recerca i l'execució de l'ordre indicada, i un atacant maliciós podria explotar aquest buit. En el passat hi ha hagut nombrosos atacs en instal·lacions Unix que utilitzaven la comanda find per netejar el directori /tmp.

El paràmetre **-exec** va seguit d'una ordre i unes claus (**{}**). El programa substitueix les claus pel nom del fitxer que trobi en la recerca. Un punt i coma finalitza la comanda. Necessitem separar el punt i coma amb una barra per impedir que la shell ho interpreti (com un separador d'ordres).

La següent comanda és una variant del comandament DOS ren, que canvia el nom fitxers amb l'extensió *.txt a l'extensió *.txt.bak:

```
$ find . -name "*.txt" -exec mv {} {}.bak \;
```

Si els requisits del que volem fer són més complexos, és preferible utilitzar la sortida de find en un script de l'interpret d'ordres. La comanda **-printf** pot ajudar. La següent comanda crea un fitxer com el mostrat en la Llista 1:

```
$ find /home/mas -type f -name "*.txt" -printf "mkdir -p /export/backup/%h\ncp -p %p /export/backup/%h/ %f.copy\n"
```

%h i %f són valors que depenen del fitxer trobat. %h és la ruta del fitxer trobat sense el nom i %f és el nom del fitxer sense la ruta.

```
01 mkdir -p /export/backup/  
   /home/mas  
02 cp -p /home/mas/.emacs  
   /export/backup/  
   /home/mas/.emacs.copy  
03  
04 mkdir -p  
   /export/backup//home/mas  
05 cp -p /home/mas/.fetchmailrc/  
   export/backup/  
   /home/mas/.fetchmailrc.copy
```

Locate

A més de les característiques esmentades aquí, find té un ampli ventall d'opcions que no podem incloure per qüestions de temps. Malgrat això, el programa té un gran inconvenient: és lent. find analitza els directoris de cerca d'un en un i també ha d'analitzar els inodes dels arxius per a

realitzar les proves que se li especifiquen. Si només necessita buscar una subsecció del directori /home amb uns quants centenars de fitxers, no notaràs el retard. Però si s'està buscant un fitxer que està ocult en qualsevol part del sistema, find necessitarà buscar entre els milers i milers d'inodes. Naturalment, això pot portar algun temps.

Locate resol aquest problema mitjançant la creació d'un índex dels fitxers recopilats en l'equip i emmagatzemant els seus noms en una base de dades centralitzada. L'eina no necessita llegir els inodes per fer la cerca, en comptes d'això buscarà en una base de dades. Aquesta recerca normalment retornarà els resultats en una fracció de segon, eliminant la necessitat d'esperar minuts i evitant també qualsevol possible impacte en el rendiment per a usuaris i serveis.

Locate té algunes limitacions, així que en molts casos se segueix utilitzant find. En comptes de la completa sintaxi de find a la línia d'ordres, locate només suporta recerques bàsiques d'elements per noms de fitxers (de fet, això és tot el que s'emmagatzema a la base de dades). L'eina locate permet comodins, com la interrogació per a un sol caràcter i l'asterisc per uns quants caràcters. Aquests comodins també poden representar el caràcter barra (/), a diferència del que passa amb els comodins de la shell:

```
$ locate /home/mas/*mail*  
  
/home/mas/.fetchmailrc  
/home/mas/.procmaildata-bulk  
/home/mas/.procmaildata-inbox  
/home/mas/.procmailrc
```

Hi ha dos requisits bàsics per a la utilització de locate, que poden ser bastant molestos. El primer és que no totes les distribucions de Linux instal·len aquesta utilitat per defecte. En segon lloc, la base de dades normalment s'actualitza mitjançant cron, i si s'apaga l'equip durant la nit i si la distribució no utilitza anacron o una eina semblant per actualitzar les tasques del cron no s'han executat, els usuaris necessitaran actualitzar la base de dades manualment amb l'ordre updatedb.

Annex

Traducció de l'ajuda

Valors numèrics
+n més gran que n
-n menor que n,
n exactament n

-amin n

Fitxers accedits fa n minuts exactes

-amin +n	Fitxers accedits fa + de n minuts
-amin -n	Fitxers accedits fa - de n minuts
	accedit vol dir llegit, modificat, etc...
-anewer file	Fitxer accedit més recentment que la darrera modificació del fitxer file
-cmin n (o +n o -n)	Fitxers que han canviat el seu estat fa n minuts (o més de n minuts o menys de n minuts)
-atime n (+n, -n)	Similar a amin però expressat en dies (sense fraccions, o sigui -atime +1 correspondria a dos o més dies)
-cnewer file	Com anewer però fent referència a arxius que han canviat el seu status, i no accedits, abans que s'hagi modificat el fitxer indicat.
-ctime n (+n , -n)	Similar a cmin però expressat en dies (sense fraccions, o sigui -ctime +1 correspondria a dos o més dies)
-mtime / -mmin -newer	Com els anterior, però modificant l'arxiu.
-perm /u=r,g=r	Que els permisos de l'usuari O el del grup sigui de lectura. És a dir, que al meys alguns dels permisos es compleixin
-perm -u=r,g=r	Que els permisos de l'usuari I el del grup sigui de lectura. Es a dir, que es compleixin tots.
-perm u=r,g=r	Que els permisos siguin exactament aquests,