find (search for files)

This is a powerful command with lots of options. Most of the time you'll use it like so:

*find . -iname `*expression*'*

This will search the current directory (.) recursively for files whose name contains *expression*. The option *-iname* makes the search case-insensitive. If you leave out the asterisks (*) at the beginning and end of *expression*, only files will be found whose name equals *expression*.

The second most common use of *find* is to identify files that have recently changed. Try executing the following command in your home directory:

find . -mmin -15

This will find any files that have been modified within the last 15 minutes. You'll be surprised to see how many of your hidden configuration files have changed in this short time span. I just ran the command on my system and found that 12 files have been modified, among them .xession-errors and .bash_history. Try substituting *-amin* or *-cmin* for *-mmin*. The option *-amin* finds files that have been accessed recently, while *-cmin* locates files whose status has changed (see →*stat*).

To conclude, here is an advanced use of *find* that shows how much power this command really packs:

find . -type f -exec cp {} {}_backup \;

This will find all regular files in the current directory tree, ignoring any directories or special files (option *-type f*), and make a backup copy of each file by calling *cp filename filename_backup*. The curly braces *{}* are a code denoting every file found. Using the *-exec* option, you can have *find* call any Linux command. The code *\;* indicates the end of the command to be called.