

Title: Comando Split Date: 2015-12-21 15:58 Modified: 2015-12-21 15:58 Category: Linux Tags: comandos, split, consola  
Authors: procamora Slug: comando\_split Summary: El programa split es una utilidad de los sistemas operativos tipo Unix, usada para partir un archivo en uno o más de menor tamaño. De ahí su nombre, que en inglés significa partir.  
Status: published

El programa split es una utilidad de los sistemas operativos tipo Unix, usada para partir un archivo en uno o más de menor tamaño. De ahí su nombre, que en inglés significa partir.

Sintaxis: `split [parámetros opcionales] [archivo de entrada] [archivo de salida]`

El comportamiento por defecto de split es generar archivos de salida de hasta 1000 líneas. Estos archivos se nombran añadiéndole aa, ab, ac, etcétera, a archivo de salida; si no se da el nombre del archivo de salida, se usa el nombre por defecto de x, resultando en los archivos xaa, xab, etcétera. Si se usa un guion (-) como archivo de entrada, se leen los datos de la entrada estándar.

Para unir de nuevo los archivos se usa el comando cat.

```
$ split --help
```

```
Usage: split [OPTION]... [INPUT [PREFIX]]
```

```
Output fixed-size pieces of INPUT to PREFIXaa, PREFIXab, ...; default  
size is 1000 lines, and default PREFIX is 'x'. With no INPUT, or when INPUT  
is -, read standard input.
```

Mandatory arguments to long options are mandatory for short options too.

```
-a, --suffix-length=N      utiliza sufijos de longitud N (por omisión 2)  
-b, --bytes=TAMAÑO         escribe TAMAÑO bytes en cada fichero de salida  
-C, --line-bytes=BYTES     escribe un máximo de BYTES bytes sin cortar líneas  
-d, --numeric-suffixes     utiliza sufijos numéricos en vez de alfabéticos  
-e, --elide-empty-files    do not generate empty output files with '-n'  
    --filter=COMMAND        write to shell COMMAND; file name is $FILE  
-l, --lines=NÚMERO         pone NÚMERO de líneas en cada fichero de salida  
-n, --number=CHUNKS        generate CHUNKS output files. See below  
-u, --unbuffered           immediately copy input to output with '-n r/...'  
    --verbose               print a diagnostic just before each  
                           output file is opened  
--help                     display this help and exit  
--version                  output version information and exit
```

SIZE may be (or may be an integer optionally followed by) one of following:  
KB 1000, K 1024, MB 1000\*1000, M 1024\*1024, and so on for G, T, P, E, Z, Y.

CHUNKS may be:

```
N          split into N files based on size of input  
K/N        output Kth of N to stdout  
l/N        split into N files without splitting lines  
l/K/N      output Kth of N to stdout without splitting lines  
r/N        like 'l' but use round robin distribution  
r/K/N      likewise but only output Kth of N to stdout
```

#Dividir

```
#####Partir por tamaño de fichero de 20Mb split dict.lst -b 20MB
```

Y el resultado:

```
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 xaa  
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 xab  
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 xac  
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 xad  
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 xae  
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 xaf
```

```
#####Partir por tamaño de fichero de 20Mb con nombre con letras split dict.lst -b 20MB dict_
```

Y el resultado:

```
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict_aa
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict_ab
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict_ac
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict_ad
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict_ae
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict_af
```

#####Partir por tamaño de fichero de 20Mb con nombre con números `split dict.lst -b 20MB -d dict`

Y el resultado:

```
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict00
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict01
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict02
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict03
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict04
-rw-rw-r-- 1 user user 20000 Mar 12 20:48 dict05
```

#####Partir por cada 20000 lineas `split dict.lst -l 20000`

Y el resultado:

```
-rw-rw-r-- 1 user user 2000 Mar 12 20:48 xaa
-rw-rw-r-- 1 user user 2000 Mar 12 20:48 xab
-rw-rw-r-- 1 user user 2000 Mar 12 20:48 xac
-rw-rw-r-- 1 user user 2000 Mar 12 20:48 xad
-rw-rw-r-- 1 user user 2000 Mar 12 20:48 xae
-rw-rw-r-- 1 user user 2000 Mar 12 20:48 xaf
```

## Unir

```
cat dict* > dict.lst
```

<http://hipertextual.com/archivo/2010/04/comando-linux-split/>