

Una de las grandes ventajas de linux es la ejecución de procesos en background. Por ejemplo si sabemos que la copia de un fichero puede llevar mucho tiempo y no queremos estar esperando a que termine para seguir trabajando entonces podemos lanzarlo en background. Para ello añadimos **&** al comando que queramos lanzar por ejemplo:

```
cp archivo1 archivo2 &
```

El proceso lo podemos pasar también a background con CTRL+z, pero se queda parado o volverlo a continuar en foreground con fg.

Un proceso que este parado en background (al pasarlo presionando CTRL +z) podemos ponerlo a continuar en background ejecutando la orden **bg**

Con el comando **jobs**, podemos ver los procesos que están corriendo y sus posiciones

Un proceso que este parado en background (al pasarlo presionando CTRL +z) podemos ponerlo a continuar en background de la siguiente forma:

ejecutamos

```
jobs -l
```

para averiguar el pid del proceso ahora ejecutamos

```
kill -CONT pid_del_proceso
```

Pongamos un ejemplo de lo anterior

Supongamos que acabo de ejecutar el comando visudo para editar el fichero de permisos de sudo, pero en un momento dado quiero pasarlo a background pulsando CTRL+Z con lo cual vuelvo a la consola. Para volverlo a llamar me basta con poner fg y el número de proceso

Lanzar procesos con nohup

Esto es útil si queremos lanzar procesos que no se terminen al matar el shell padre, por ejemplo para crear scripts de inicio

```
nohup proceso &
```

From:

<https://intrusos.info/> - LCWIKI

Permanent link:

<https://intrusos.info/doku.php?id=linux:procesos&rev=1519819356>

Last update: **2023/01/18 13:55**

