

# Partitioning your hard drive(s)

If your current hard drives have already been partitioned and you don't have any spare partitions for a linux installation, don't worry! There are two free tools available which you can use to resize your existing partitions. They are **ntfsresize** (for NTFS partitions) and **fips** (for FAT partitions).

## Links:

**ntfsresize:** <http://mlf.linux.rulez.org/mlf/ezaz/ntfsresize-static-1.9.1.tgz>

**fips:** <http://www.igd.fhg.de/~aschaefer/fips/>

## Using NTFS Resize:

**BACK UP ANYTHING REALLY IMPORTANT. REALLY. THIS COULD ALL GO WRONG IN A BIG WAY.**

Boot into linux using the Red Hat or Fedora Core CD. After it asks you to test the cd's (skip this if you like) you will get into a graphical welcome screen. From here you can get into a shell prompt by typing **ctrl-alt-F2**

Put in your floppy containing ntfsresize and mount it:

```
mkdir /floppy
mount /dev/fd0 /floppy
cd /floppy
```

Work out which drive and partition your ntfs installation is on - normally the first physical hard drive = /dev/hda and the second physical drive = /dev/hdb. To view the partition information of a drive, type

```
fdisk -l <drivename>
eg. fdisk -l /dev/hda
to view the partitions on the first physical drive
```

Partitions are normally labelled numerically in the order they reside on the disk. For example, the first partition on the first physical disk will be /dev/hda1, the second partion on the first physical disk will be /dev/hda2 etc.

You can then find out where you can resize the partition

```
./ntfsresize -i <partitionname>
e.g. ./ntfsresize -i /dev/hda1
```

It will tell you where on the disk you could resize to. It checks the drive and finds the last block with any data written to it. It will then be able to change the partition size so it ends just after this block. For best results you should defragment your drive before doing this - the defragmentation process will take all your data and move it as close as possible to the start of the disk.

Lets say you have a 20GB drive and you want to split this into two 10GB partitions. If ntfsresize tells us that we can resize at 10000MB or below then it will be possible to do this. If not, defrag and try again. If you still can't get below 10000MB then you will have to consider either removing some files, settling for a smaller 2nd partition, or getting another hard drive!

To resize a partition you then type

```
./ntfsresize -n -s<partitionsizesize> <partitionname>  
e.g. ./ntfsresize -n -s10000M /dev/hda1  
will resize the first partition on physical drive 1 to 10000MB (10GB)
```

It will do a test run and if it is successful it will look something like this:

```
Current volume size: xxxxxxxxxxxx bytes (xxxxxx MB)  
New volume size    : xxxxxxxxxxxx bytes (xxxxxx MB)  
Scanning volume ...  
100.00 percent completed  
Updating $BadClust file ...  
Updating $Bitmap file ...  
Updating Boot record ...  
The read-only test run ended successfully.
```

If it is successful you can then go ahead with the resizing.

```
./ntfsresize -s<partitionsizesize> <partitionname>
```

Hopefully you should get a message that it has been successfully resized after a while. We now need to repartition the drive using fdisk.

```
fdisk <drivename>  
e.g. fdisk /dev/hda
```

You should then go through the following process (your replies in bold):

```
Command (m for help): p
```

```
Disk /dev/hda: 255 heads, 63 sectors, 2480 cylinders  
Units = cylinders of 16065 * 512 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/hda1	*	1	2479	19912536	7	

HPFS/NTFS

```
Command (m for help): d  
Partition number (1-4): 1
```

```
Command (m for help): n  
Command action  
  e  extended  
  p  primary partition (1-4)
```

```
p  
Partition number (1-4): 1  
First cylinder (1-2480, default 1): 1
```

Last cylinder or +size or +sizeM or +sizeK (1-2480, default 2480): **+<SIZE> (e.g. 10000M)**

Command (m for help): **t**  
Partition number (1-4): **1**  
Hex code (type L to list codes): **7**  
Changed system type of partition 1 to 7 (HPFS/NTFS)

Command (m for help): **a**  
Partition number (1-4): **1**

Command (m for help): **p**

Disk /dev/hda: 255 heads, 63 sectors, 2480 cylinders  
Units = cylinders of 16065 \* 512 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/hda1	*	1	1403	11269566	7	HPFS/NTFS

Command (m for help): **w**  
The partition table has been altered!

Calling ioctl() to re-read partition table.  
Syncing disks.

You are now ready to reboot and test it. If it boots into windows then happy days. If not, then even happier days since windows sucks anyway.....

### **Using fips:**

If your existing partition is a FAT partition, then you can use fips to resize it. Fips is very simple to use and a lot more user friendly. I won't go into the details here – you can find detailed instructions on the fips website – but basically you create a DOS boot disk, copy the fips executables to it, reboot your computer with it and run fips with the command **fips!** The program will then display your current partition information, ask you which partition you want to split, ask you for a size etc. When it is done it will sort everything else out for you. When you reboot the computer you will have a new partition – you don't need to worry about fdisk. One small note – fips will only work for primary partitions.

Remember – for best results defrag your disk before using fips or ntfssize