How to Create, Mount and Extend xfs Filesystem

XFS is a high-performance 64-bit journaling file system created by SGI in 1993. It was introduced in the Linux kernel in 2001, XFS is supported by most Linux distributions, some of which use it as the default file system (RHEL/CentOS 7.0).

XFS excels in the execution of parallel input/output (I/O) operations due to its design, which is based on allocation groups, because of this, XFS enables extreme scalability of I/O threads, file system bandwidth, and size of files and of the file system itself when spanning multiple physical storage devices. A disadvantage of the XFS file system is that it cannot be shrunk, also metadata operations in have historically been slower than with other file systems, resulting in, for example, poor performance with operations such as deletions of large numbers of files. FAQ of xfs.org is a good place (http://xfs.org/index.php/XFS_FAQ) to read before you start implementing this filesystem.

Creating a new XFS partition

To create a new XFS file system you will first need a partition to format. You can use fdisk to create a new partition, like in the example below, you first need to invoke fdisk with the name of the harddisk you wish to create the partition on and then use "n" command inside fdisk for a new parttion, after you have set the size like in the example below you will need to use the "w" command to write the new table to disk.

```
[root@centos ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x74fb586c.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p):
Using default response p
Partition number (1-4, default 1):
First sector (2048-62914559, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-62914559, default 62914559): +15G
Partition 1 of type Linux and of size 15 GiB is set

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

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

(https://linoxide.com/wp-content/uploads/2014/08/xfs-fdisk-new-partition.gif)

You can find more info about fdisk here (https://linoxide.com/how-tos/how-to-create-new-partition-filesystem-on-linux/).

If the partition you want to format as xfs already exists on the system you must make sure it's not mounted by using the command umount command like this:

```
# umount /dev/sdb1
```

/dev/sdb1 should of course be replaced by the name of the partition you want to use.

Now that your partition is ready you can create a xfs filesystem by using the mkfs.xfs command, with the name of the partition you created like this:

```
# mkfs.xfs /dev/sdb1
```

```
[root@centos ~]# mkfs.xfs /dev/sdb1
meta-data=/dev/sdb1              isize=256    agcount=4, agsize=983040 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=0
data     =                       bsize=4096   blocks=3932160, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=0
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

(https://linoxide.com/wp-content/uploads/2014/08/xfs-mkfs.gif)

Mounting the xfs filesystem

To mount the newly created partition you will have to first create a directory to be a mount point with the mkdir command, in our example we will use /mnt/db. Next you can mount the xfs parttion using the mount command as you would with any partition. Afterwards you can use the mount command to check if the partition was correctly mounted.

```
# mkdir /mnt/db
# mount /dev/sdb1 /mnt/db
# mount | grep /dev/sdb1
```

```
[root@centos ~]# mkdir /mnt/db
[root@centos ~]# mount /dev/sdb1 /mnt/db
[root@centos ~]# mount | grep /dev/sdb1
/dev/sdb1 on /mnt/db type xfs (rw,relatime,seclabel,attr2,inode64,noquota)
```

(https://linoxide.com/wp-content/uploads/2014/08/xfs-mount.gif)

If you have environment with filesystem above 2 TB , you could try benchmark with mounting with **inode64 option**.

```
# mount -o inode64 /dev/device /mount/point
```

XFS enable write barriers to ensure file system integrity which preserves it across power failure, interface resets, system crashes by default. If your hardware have a healthy write cache feature then its recommended to disable write barriers otherwise performance would be negatively affected. You can disable write barrier using mount option below.

```
# mount -o nobarrier /dev/device /mount/point
```

Extending a xfs filesystem

You can see the size of the current mounted partitions using the following command:

 # df -h

```
[root@centos ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/centos-root    18G  5.5G   12G  32% /
devtmpfs                  909M     0  909M   0% /dev
tmpfs                     918M  140K  917M   1% /dev/shm
tmpfs                     918M  8.9M  909M   1% /run
tmpfs                     918M     0  918M   0% /sys/fs/cgroup
/dev/sda1                 497M  160M  338M  33% /boot
/dev/sdb1                  15G   33M   15G   1% /mnt/db
```

(https://linoxide.com/wp-content/uploads/2014/08/xfs-df-h.gif)

To extend a xfs filesystem you will first have to modify the partition table to the new size, you can do this similar to the way you created the partition using fdisk, first use "d" command to remove the partition (be careful to select the correct partition if you have more then one on the device) then use the "n" to create a partition of a bigger size, in our case 20GB.

```
[root@centos ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p):
Using default response p
Partition number (1-4, default 1):
First sector (2048-62914559, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-62914559, default 62914559): +20G
Partition 1 of type Linux and of size 20 GiB is set

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

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

(https://linoxide.com/wp-content/uploads/2014/08/xfs-extend-fdisk.gif)

Next all you have to do is run the xfs_growfs command with the -d switch (to grow the data part of the file system) and the filesystem will be grown to the new size of the partition.

```
# xfs_growfs -d /mnt/db
```

```
[root@centos ~]# xfs_growfs -d /mnt/db
meta-data=/dev/sdb1              isize=256    agcount=4, agsize=983040 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=0
data     =                       bsize=4096   blocks=3932160, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=0
log      =internal               bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 3932160 to 5242880
[root@centos ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   18G  5.5G   12G  32% /
devtmpfs                 909M     0  909M   0% /dev
tmpfs                    918M  140K  917M   1% /dev/shm
tmpfs                    918M  8.9M  909M   1% /run
tmpfs                    918M     0  918M   0% /sys/fs/cgroup
/dev/sda1                497M  160M  338M  33% /boot
/dev/sdb1                 20G   33M   20G   1% /mnt/db
```

(https://linoxide.com/wp-content/uploads/2014/08/xfs_growfs.gif)

Note : Filesystem must be mounted for you to be able to use xfs_growfs and also that **you can never shrink a xfs filesystem**.