SILICON GRAPHICS | The Source of Innovation and Discovery™

Silicon Graphics, Inc.

XFS Overview & Internals
05 - Mount

Presented by:

November 2006

Mounting XFS Filesystems

- This section describes some XFS-specific mount options and how they affect the behaviour of XFS

Mount Options

- Described in
  - In linux tree:
    - `Documentation/filesystems/xfs.txt`
    - `xfs/fs/xfs_vfsops.c`
  - `mount(8)`

- What options are specific to XFS?

- What common Linux mount options does XFS not support?

Mount Options – Log & Realtime Devices

- Use an external log (metadata journal) device:

```
mount -o logdev=log_device device mountpoint
```

- Use an external log (metadata journal) and real-time device:

```
mount -o logdev=log_device,rtdev=rt_device device mountpoint
```

Mount Options – 64bit Inodes

- By default XFS uses 32bit inodes
  - The inode’s number roughly equates to its location on disk
    - Combination of allocation group, cluster and block
  - Inode on Linux is 32bit on 32bit machines
    - May change in future kernels
  - Allocator will place 32bit inodes in the first terabyte
    - Using a larger inode size means less inodes per cluster allowing 32bit inodes to be located beyond the first terabyte

- inode64 option on 64bit machines allows inodes to span the entire filesystem
  - Allocator will try to put file extents in same allocation group as inode
  - Not all backup tools support 64bit inodes
    - Inode number used to identify file between backups

Mount Options - Stripes

- Specify the stripe unit and width for a RAID device or a stripe volume.
- Values must be specified in 512-byte block units.
- For example, to use a stripe unit of 1MB and a stripe width of 8MB:

```
mount -o sunit=2048,swidth=16384 device mountpoint
```

- `swalloc` option
  - data allocations will be rounded up to stripe width boundaries when the current end of file is being extended and the file size is larger than the stripe width size.

Mount Options – Large I/O

- `largeio`
  - A filesystem that has a `swidth` specified will return the `swidth` value (in bytes) in st_blksize
  - If the filesystem does not have a `swidth` specified but does specify an `allocsize` then `allocsize` (in bytes) will be returned instead.
  - If neither of these two options are specified, then filesystem will behave as if `nolargeio` was specified.

- `nolargeio`
  - The optimal I/O reported in `st_blksize` by `stat(2)` will be as small as possible to allow user applications to avoid inefficient read/modify/write I/O.

Mount Options – Performance Tweaks

- `noalign`
  - Data allocations will not be aligned at stripe unit boundaries.

- `allocsize`
  - Sets the buffered I/O end-of-file preallocation size when doing delayed allocation writeout (default size is 64KB).

- `noatime`
  - Access timestamps are not updated when a file is read.

Mount Options – Performance Tweaks

- `osyncisdsync`
  - Writes to files opened with the O_SYNC flag set will behave as if the O_DSYNC flag had been used instead.
  - This can result in better performance without compromising data safety.
  - However timestamp updates from O_SYNC writes can be lost if the system crashes. Use osyncisosync to disable this setting.

- `ihashsize`
  - Sets the number of hash buckets available for hashing the in-memory inodes of the specified mount point.

- `ikeep`
  - When inode clusters are emptied of inodes, keep them around on the disk.

Mount Options – Extended Attributes

- `attr2`
  - Enable the use of version two of the extended attribute inline allocation policy.
- `noattr2`
  - Force the use of version one of the extended attribute inline allocation policy.
- When the new form is used for the first time (by setting or removing extended attributes) the on-disk superblock feature bit field will be updated to reflect this format being in use.

```
mount -o attr2 device mountpoint
```

Mount Options – Group Ids

- These options define what group ID a newly created file gets.
- `grpid/bsdgroups`
  - Take the group ID of the directory in which it is created
- `nogrpid/sysvgroups`
  - Take the fsgid of the current process, unless the directory has the setgid bit set, in which case it takes the gid from the parent directory, and also gets the setgid bit set if it is a directory itself.

Mount Options - Recovery

- An XFS filesystem can be mounted without running log recovery.
- If the filesystem was not cleanly unmounted, it is likely to be inconsistent when mounted in `norecovery` mode. Some files or directories may not be accessible because of this.
- Filesystems mounted `norecovery` must be mounted read-only or the mount will fail.

```
mount -o ro,norecovery device mountpoint
```

Mount Options - Barriers

- Write cache on disk can result in filesystem corruption since XFS is told the log transaction has reached the disk when in fact it is still in the disk cache.
  - A journal log assumes that the log transaction is updated on disk before the metadata is updated, but this may not be true with write caching
- XFS is able to issue write barriers if the device supports it
  - Ensures that the log entry is written before any other data
- Write barriers are enabled by default in SLES10
  - Filesystem will attempt to determine is barriers are supported and will issue a warning if they are not
  - The `nobarrier` option disables write barriers
- See
  - http://oss.sgi.com/projects/xfs/faq.html#wcache

Mount Options – User/Group/Project Quotas

- User disk quota accounting enabled, and limits (optionally) enforced.

```
mount -o uquota device mountpoint
```

- Group disk quota accounting enabled, and limits (optionally) enforced.

```
mount -o grpquota device mountpoint
```

- Project quota accounting enabled, and limits (optionally) enforced.

```
mount -o prjquota device mountpoint
```

- Can optionally specify `uqnoenforce`, `gqnoenforce` and `pqnoenforce` to use soft limits.

Mount Options – DMAPI

- `dmapi`
  - Enable the DMAPI (Data Management API) event callouts.

- `mtpt`
  - Specify a value to be included in the DMAPI mount event, and should be the path of the actual mountpoint that is used.

```
mount -o dmapi,mtpt=mountpoint device mountpoint
```

sgi®