

ibound 2.0

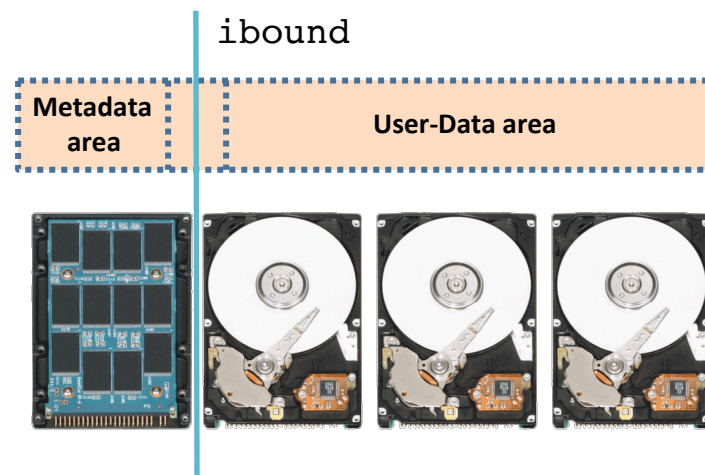
XFS for hybrid SSD/HDD filesystems

Zsolt Ferenczy/SE
zsolt@sgi.com



eXFS ibound

- XFS hybrid SSD/HDD filesystem
- XFS inodes and other metadata stored on SSD media
- User data stored on HDD media
- Available with SGI Enhanced XFS

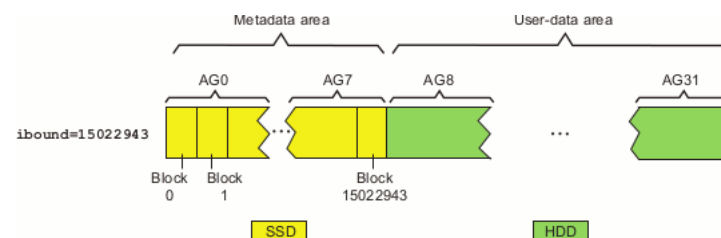


ibound 2.0

- Complete separation from inode32 code base
- Guarantees user data extents cannot be allocated in the metadata area
- Optimized user data extent allocator
- Available with ISSP 3.5

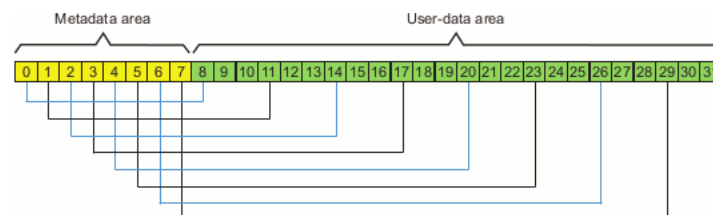
ibound Concepts

- Dedicate some number of Allocation Groups (AGs) at the beginning of the filesystem for metadata
- Metadata consists of inodes, extended attributes, and directory entries
- “ibound” mount option specifies the volume logical block address corresponding to the end of the metadata area
- The metadata area always consists of complete AGs
- There must be more AGs in the user data area than the metadata area

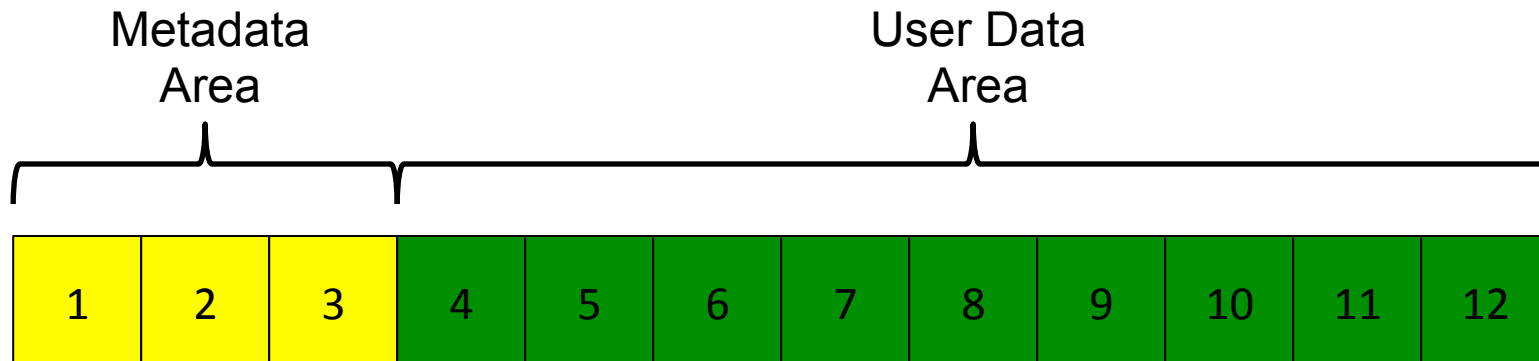


ibound 2.0 Allocation Policy

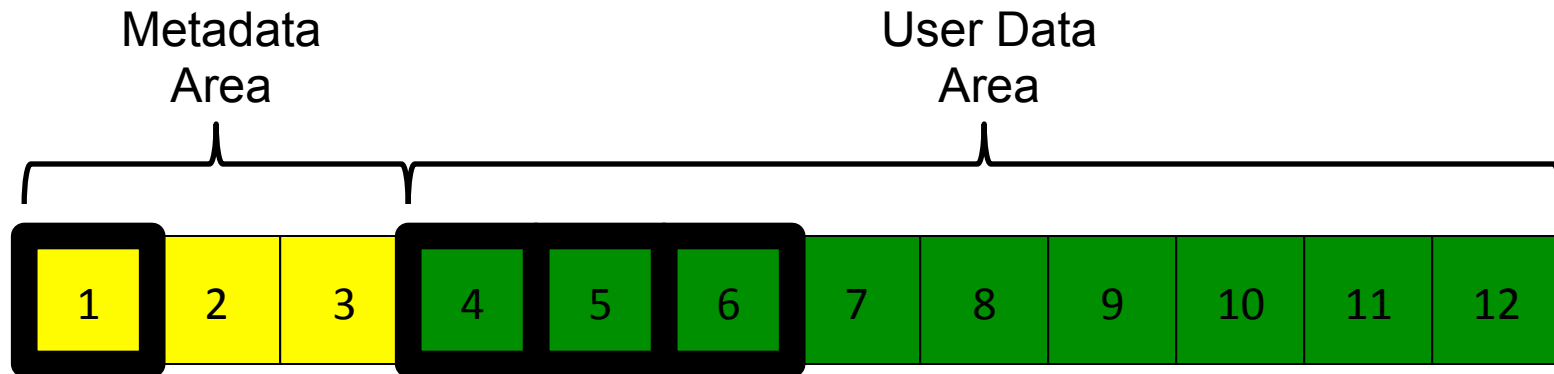
- XFS allocates the inode in an AG in the metadata region
- The start of the user data extents will be in an AG proportional to the metadata AG
- If there is insufficient space in that AG, the next AG will be used, wrapping around to the first AG in the user data area if needed
- XFS will use as many AGs in the user data area as needed to contain the entire file
- If agskip is used, it's value is used in place of the proportional indexing



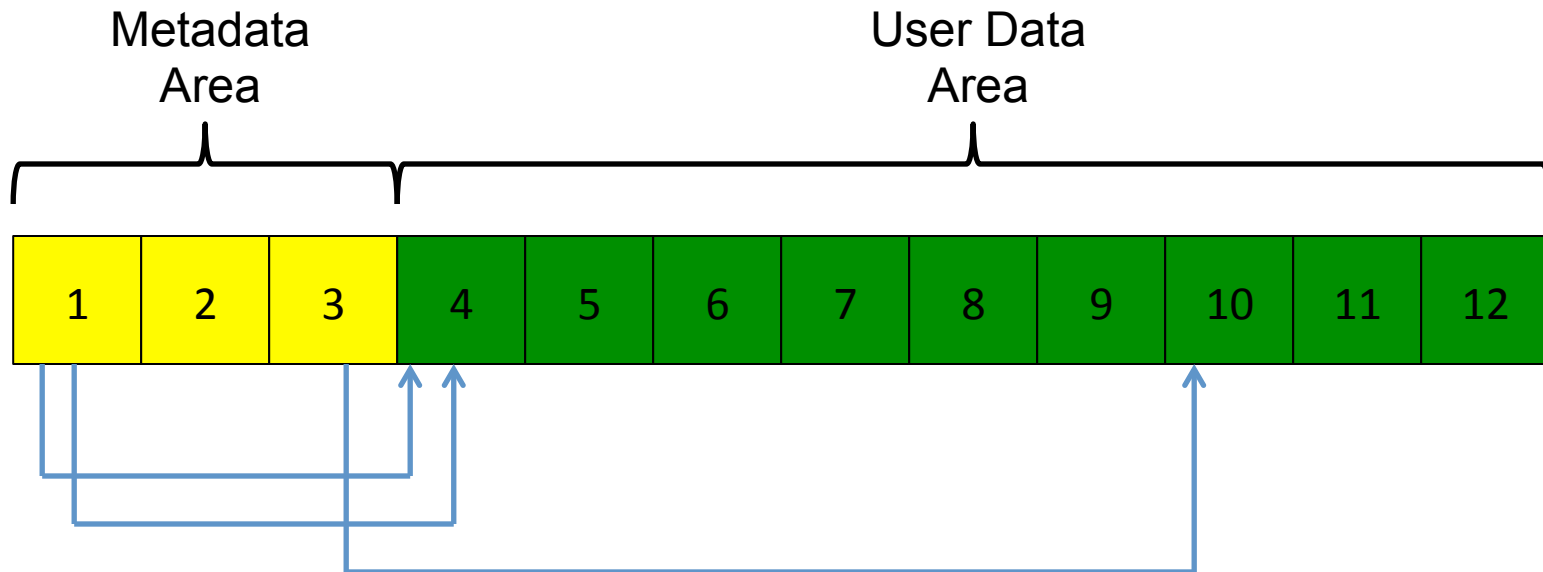
ibound 2.0 Allocation



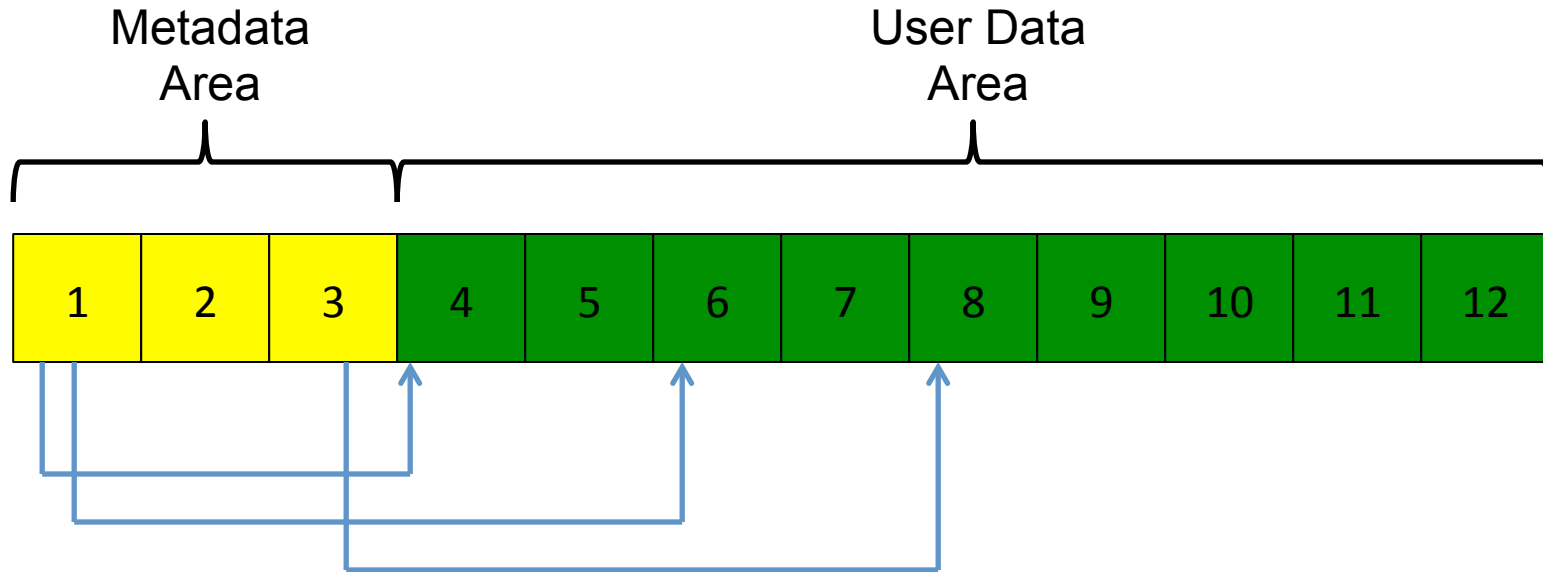
ibound 2.0 Allocation



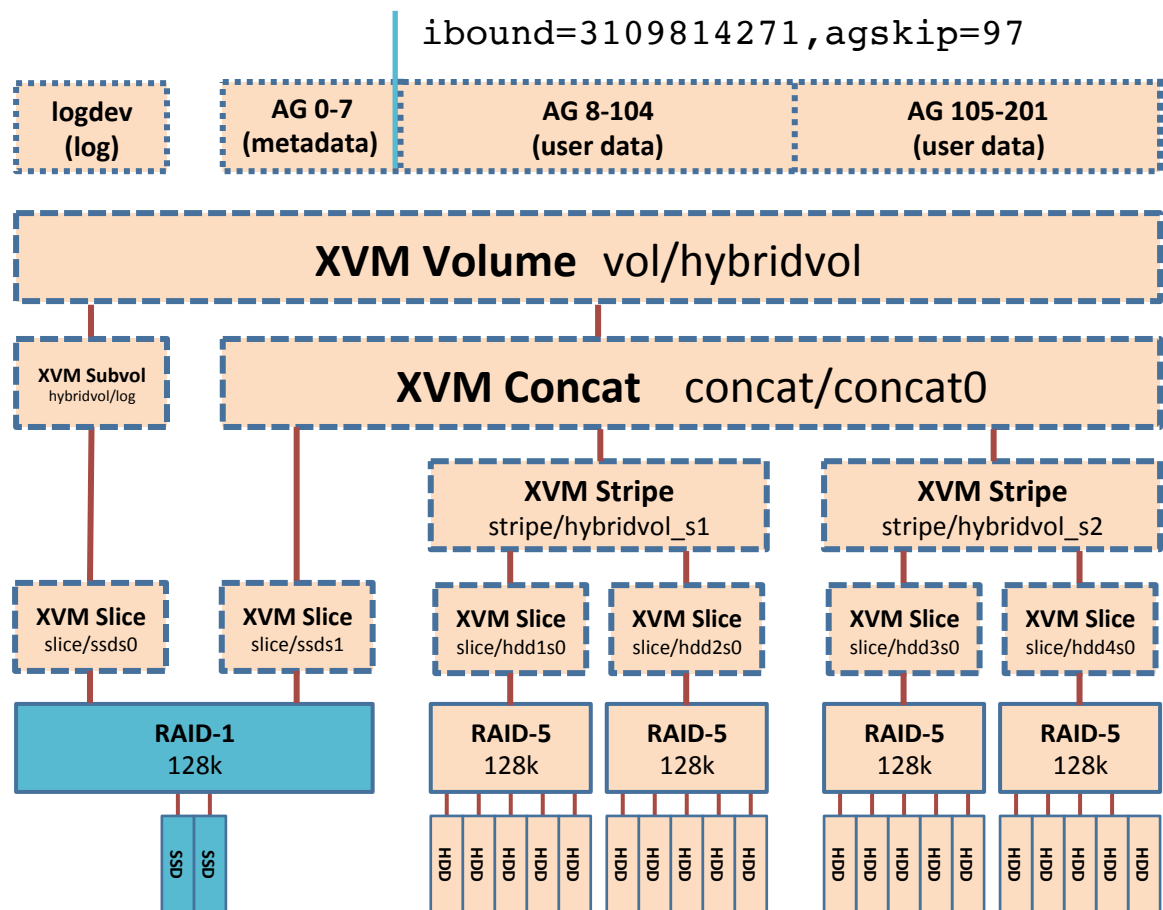
ibound 2.0 Allocation



ibound 2.0 Allocation w/ agskip=2



ibound Volume Example



ibound Best Practices

- Must be least 8 AGs in the metadata area
- Keep the filesystem less than 90% full
 - ibound filesystems tend to have a large number of AGs, XFS will consume more CPU time to find free blocks on a very full filesystem
- Configure the most SSD space that is possible
 - Larger ibound area will result in larger AGs and thus fewer number of AGs in the filesystem

sgi®