Chapter 4 Implementing IBRIX Fusion High Availability ................................................. 51
### Contents

- **4.1** About IBRIX Fusion High Availability ................................................................. 51  
- **4.2** About the Two IBRIX Failover Modes ..................................................................... 52  
- **4.3** What Happens During a Failover ........................................................................... 52  
- **4.4** Setting Up Automated Failover ............................................................................... 52  
- **4.5** Identifying Segment Server Standbys .................................................................... 53  
  - **4.5.1** Identifying Segment Standbys ............................................................................ 53  
  - **4.5.2** Setting Up Power Sources .................................................................................. 54  
  - **4.5.2.1** Preliminary Steps54  
  - **4.5.2.2** Identifying Power Sources54  
  - **4.5.2.3** Associating Segment Servers to Power Source Slots54  
  - **4.5.2.4** Updating the Configuration Database with Power Source Changes55  
  - **4.5.2.5** Dissociating a Segment Server from a Power Source55  
  - **4.5.2.6** Deleting Power Sources from the Configuration Database55  
  - **4.5.3** Turning Automated Failover On and Off.......................................................... 56  
- **4.6** Setting Up Manual Failover .................................................................................. 56  
- **4.7** Manually Failing Over a Segment Server .............................................................. 56  
- **4.8** Failing Back a Segment Server ............................................................................. 57  
- **4.9** Setting Up Network Interface Monitoring ............................................................ 57  
  - **4.9.1** Identifying and Deleting Standbys for a Network Interface .............................. 58  
  - **4.9.2** Identifying and Deleting Network Interface Monitors .................................... 59  
- **4.10** Setting Up HBA Monitoring ................................................................................ 59  
  - **4.10.1** Discovering HBAs .......................................................................................... 60  
  - **4.10.2** Identifying Standby-Paired HBA Ports ............................................................ 60  
  - **4.10.3** Turning HBA Monitoring On and Off .............................................................. 60  
  - **4.10.4** Deleting Standby Port Pairings ....................................................................... 61  
  - **4.10.5** Deleting HBAs from the Configuration Database ......................................... 61  
  - **4.10.6** Listing Information about HBAs ................................................................. 61  
- **4.11** Checking Your High Availability Configuration .................................................. 61  
- **4.12** Power Management for Segment Servers ............................................................ 63  

---

### Chapter 5 Managing Network Interfaces .............................................................. 65  
- **5.1** About the Cluster and User Network Interfaces ...................................................... 65  
- **5.2** Setting Up User Networks for Clients ..................................................................... 66  
- **5.3** Rules for Setting Up the Cluster Network ............................................................... 66  
- **5.4** Identifying a User Interface ................................................................................... 67  
- **5.5** Preferring Network Interfaces ................................................................................ 68  
- **5.6** Unpreferring Network Interfaces .......................................................................... 68  
- **5.7** Modifying Network Interface Options ................................................................... 69  
- **5.8** Changing IP Addresses ......................................................................................... 69  
  - **5.8.1** Changing a Linux Fusion Client IP Address ..................................................... 69  
  - **5.8.2** Changing the Cluster Interface IP Address ..................................................... 69  
- **5.9** Changing the Cluster Interface ............................................................................. 69  
- **5.10** Adding and Deleting Routing Table Entries ......................................................... 70  
- **5.11** Deleting a Network Interface ............................................................................... 70  
- **5.12** Viewing Network Interface Information .............................................................. 70  

---

### Chapter 6 NFS and CIFS Procedures ................................................................. 73  
- **6.1** Exporting and Unexporting File Systems Using NFS ........................................... 73  
- **6.2** Setting Up CIFS Management for a Cluster ........................................................... 74
Chapter 7  The Windows Fusion Client  .................................................. 77
7.1 About Windows Fusion Clients .......................................................... 77
7.2 System Requirements .......................................................................... 77
7.3 Overview of the Windows Fusion Client Setup ....................................... 78
7.4 Set up Windows Services for UNIX....................................................... 78
7.5 Configure Groups and Users on the Active Directory Server ....................... 78
7.5.1 Create Administrative User/Group .................................................... 78
7.5.2 Create Proxy User and Delegate Folder Control .................................... 79
7.5.3 “Unknown” IBRIX Windows User .......................................................... 79
7.5.4 Other Windows Fusion Client Users ..................................................... 80
7.6 Configure Active Directory Settings on Fusion Manager............................ 80
7.7 Register Windows Fusion Clients and Start Service ................................. 80
7.8 Importing UIDs/GIDs into Active Directory ............................................ 82
7.9 Using the Windows Fusion Client GUI .................................................... 82
7.10 Managing Access Control Lists ............................................................ 83
7.10.1 Windows Access Controls ................................................................. 83
7.10.2 Linux Mode Mask to Special ACEs Mapping ........................................ 83
7.10.3 User Mapping ................................................................................... 84
7.10.4 ACL to Mode Mask Mapping .............................................................. 84
7.10.5 File Creation, Ownership and Administrative Rights ............................ 85

Chapter 8  Managing File Allocation ......................................................... 87
8.1 How File Allocation Works ..................................................................... 87
8.2 How IBRIX Fusion Evaluates File Allocation Settings ............................... 88
8.3 Using Hostgroups for File Allocation Settings ........................................ 89
8.4 When File Allocation Settings Take Effect on Fusion Clients ....................... 89
8.5 Guidelines for Using File Allocation CLI Commands .................................. 89
8.6 Setting File and Directory Allocation Policies ........................................... 89
8.7 Setting Segment Preferences ................................................................. 90
8.7.1 Creating a Pool of Preferred Segments .................................................. 90
8.7.2 Preferring One Segment for a User or Group ........................................ 90
8.7.3 Restoring the Default Segment Preference .......................................... 91
8.8 Unpreferring Segments .......................................................................... 91
8.9 Tuning Allocation Policy Settings .......................................................... 92
8.10 Restoring the Default Allocation Policy ................................................ 92
8.11 Listing Allocation Policies ..................................................................... 92

Chapter 9  Servicing IBRIX Fusion ............................................................ 93
9.1 Viewing IBRIX Fusion Software Version Numbers ................................... 93
9.2 Viewing Your IBRIX Fusion License Terms ............................................ 94
9.3 Changing the IBRIX User Password ..................................................... 94
9.4 Viewing Disk Space Information for an IBRIX File System ........................ 94
9.5 Viewing Logs ......................................................................................... 95
9.6 Viewing Operating Statistics for Segment Servers .................................... 95
9.7 Starting and Stopping Processes ............................................................ 96
9.8 Managing Long-Running Jobs ................................................................. 96
9.9 Keeping Segment Server and Configuration Database Information Consistent .......... 97
9.10 Checking and Repairing IBRIX File Systems ........................................ 98
9.11 Backing Up the IBRIX Fusion Configuration ........................................... 99
9.12 Recovering the Fusion Manager ............................................................... 99
9.13 Recovering a Segment Server ................................................................. 100

Chapter 10 Troubleshooting ................................................................. 103
10.1 Problems with IBRIX Services ............................................................. 103
  10.1.1 Cannot Start Services on the Fusion Manager or a Segment Server ............... 103
  10.1.2 Cannot Start Services on a Linux Fusion Client ...................................... 103
10.2 Problems with the IBRIX File System .................................................. 104
  10.2.1 ibrix_pv -a Discovers Too Many or Too Few Devices ............................ 104
  10.2.2 Cannot Mount on a Fusion Client ...................................................... 104
  10.2.3 NFS Clients Cannot Access an Exported IBRIX File System ..................... 104
  10.2.4 User Quota Usage Data Is Not Updating ............................................ 104
10.3 Failover Problems ............................................................................... 104
  10.3.1 Cannot Failback from Failover Caused by Storage Subsystem Failure ......... 104
  10.3.2 Cannot Failback Owing to a Storage Subsystem Failure ........................... 105
  10.3.3 Fusion Client I/O Errors Following Segment Migration .......................... 105
  10.3.4 Recovering Fusion Manager High Availability When a Cable Is Disconnected .. 105
10.4 Windows Fusion Client Problems ....................................................... 106
  10.4.1 Logged in But Getting a “Permission Denied” Message ......................... 106
  10.4.2 Verify Button in the Active Directory Settings Tab Does Not Work ............ 106
  10.4.3 Mounted Drive Does Not Appear in Windows Explorer ............................ 106
  10.4.4 Mounted Drive Not Visible When Using Terminal Server ....................... 106
  10.4.5 Fusion Client Auto-Startup Interferes with Debugging ............................ 106
List of Figures

Figure 1-1 The IBRIX Segmented File System Architecture....................................................2
Figure 1-2 IBRIX File System Components .............................................................................4
Figure 3-1 Global and local settings in a hostgroup tree........................................................32
Figure 4-1 Network Interface Monitoring and Failover Scenario............................................58
Figure 5-1 Bonded and virtual interfaces ...............................................................................67
Figure 7-1 Windows Fusion Client GUI – Registration Tab ...................................................81
Figure 7-2 Windows ACL and permissions window...............................................................85
List of Tables

Table 2-1 ibrix_pv -l Physical Volumes Summary .........................................................16
Table 2-2 ibrix_vg -l and ibrix_vg -i Volume Groups Summary .................................17
Table 2-3 ibrix_lv -l Logical Volumes Summary .........................................................17
Table 2-4 ibrix_fs -l File Systems Summary ...............................................................18
Table 2-5 ibrix_fs -i File System Summary .................................................................18
Table 2-6 ibrix_fs -i Per-Segment Summary ..............................................................20
Table 3-1 Output Fields for ibrix_host -l .................................................................24
Table 4-1: Output Fields for ibrix_hba -l .................................................................61
Table 5-1 Output Fields for ibrix_nic -l .................................................................71
Table 7-1 Linux Mode Mask/Windows Special ACE Mappings ...............................83
Table 7-2 Linux/Windows Basic Permissions Mappings .........................................85
Table 8-1 File Allocation Policies ...............................................................................88
Table 9-1 Output Fields for ibrix_df ..........................................................................95
Getting Help

Contact IBRIX Customer Support with any questions or concerns about your IBRIX Fusion Server cluster:

- **Email:** support@ibrix.com
- **Telephone:** 1-800-92-IBRIX (1-800-924-2749)

Have the following information available:

- Your case number, if one is open.
- Your IBRIX Customer Support number.
- The Linux operating system and kernel version(s) used in your cluster.
- The software version number of all installed IBRIX Fusion software. Use the `ibrix_version` command to obtain the version number of your software.

Documentation Conventions

<table>
<thead>
<tr>
<th>Convention</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;ibrixhome&gt;</code></td>
<td>Home directory for IBRIX Fusion.</td>
</tr>
<tr>
<td><strong>Italic text</strong></td>
<td>Files and directories; physical or logical volume names, mountpoints, or cluster entities.</td>
</tr>
<tr>
<td><strong>Fixed-width text</strong></td>
<td>Code, commands, keywords, command output; files and directories.</td>
</tr>
<tr>
<td><strong>Italicized fixed-width text</strong></td>
<td>Variables representing context-specific values.</td>
</tr>
<tr>
<td><strong>Bold fixed-width text</strong></td>
<td>Text that you enter as shown.</td>
</tr>
<tr>
<td><strong>Square brackets:</strong> <code>[{-h HOSTLIST}]</code></td>
<td>Enclose an optional command argument.</td>
</tr>
<tr>
<td><strong>Braces:</strong> `{tcp</td>
<td>udp}`</td>
</tr>
</tbody>
</table>
Chapter 1  IBRIX Fusion Concepts

1.1 What Is IBRIX Fusion?

IBRIX Fusion™ is a highly scalable file serving solution comprised of a parallel file system for clusters, an integrated volume manager, high availability features such as multi-component automatic failover, and a centralized management interface. IBRIX Fusion is hardware, network, and protocol independent, and can be deployed in environments scaling to thousands of nodes.

Based on a patented Segmented File System architecture, IBRIX Fusion enables enterprises to integrate I/O and storage systems into a single clustered environment that can be shared across multiple applications and managed from a single central Fusion Manager workstation.

IBRIX Fusion is designed to operate with high-performance computing applications that require high I/O bandwidth, high I/Os per second (IOPS) throughput, and scalable configurations. These include Internet streaming, rich media streaming, data mining, web search, manufacturing, financial modeling, life sciences modeling, seismic processing, and others.

Significant features and benefits at a glance:

• Scalable configuration: add servers to scale performance, storage devices to scale capacity.
• Contains all directories and files in a single namespace.
• Operates in both the SAN (Storage Area Network) and DAS (Direct Attached Storage) environments.
• High availability software protects servers and the Fusion Manager.
• File Cloning: through replication, ensures redundancy for directories and files.
• Tunable for large or small-block I/O.
• Flexible: supports dynamic migration of segments for rebalancing and data tiering.
• Vendor independent: runs on off-the-shelf hardware, with proven file systems and storage methods.
1.2 The Segmented File System Architecture

IBRIX Fusion relies on a segmented architecture—that is, a clustered file system divided into storage segments containing files and directories. The operating principles of the IBRIX segmented file system architecture\(^1\) are highlighted in Figure 1-1. The steps following the figure correspond to the numbering in it.

---

1. The “file space” of the file system is a collection of segments. Segments organize data for faster access. Each segment is a repository for files and directories with no implicit namespace relationships among them (specifically, a segment need not be a complete rooted directory tree). Segments can be of any size, and different segments can be of different sizes. A file can span several segments, and multiple segments can be accessed in parallel within the same namespace, achieving tremendous throughput results.

2. The location of files and directories within particular segments is independent of their physical locations. Thus a directory may be located on one segment while the files contained in that directory are spread over other segments. The selection of segments for placement of files and directories is done dynamically at the time of file/directory creation as determined by an allocation policy. The allocation policy is set by the system administrator in accordance with the anticipated access patterns and specific criteria relevant to the installation (performance, manageability, and so forth).

3. Segment servers manage the individual segments of the file system. Each segment is assigned to a single segment server and each server can “own” multiple segments, as shown by the color coding in Figure 1-1. Segment ownership can be migrated from one server to another while the file system is actively in use.

To meet growing performance needs, additional servers can be added to the system dynamically, distributing the ownership of existing segments for proper load balancing and utilization by all servers. Conversely, additional capacity can be added to the file system while in active use without adding more servers—ownership of the new segments is distributed among existing servers.

---

1. U.S. Patent No. 6,782,389
Servers can be configured with failover protection, with other servers being designated as standby servers that automatically take control of a server’s segments if a failure occurs.

4. Clients run the applications that use the file system. Applications can also run on Segment Servers (depending on performance impact). Segment Servers and Fusion Clients running the IBRIX Fusion software and capable of reading and writing files are sometimes referred to as “engines.” Clients can access the file system either as a locally-mounted cluster file system using the IBRIX Fusion Client driver or using standard network attached storage (NAS) protocols such as NFS and Common Internet File System (CIFS). Use of the IBRIX Fusion Client has an advantage over the NAS approach—specifically, the IBRIX driver is aware of the segmented architecture of the file system and, based on the file/directory being accessed, can route requests directly to the correct segment server, yielding balanced resource utilization and high performance.

When using NAS protocols such as NFS and CIFS, a client must mount the file system from a segment server. All requests are sent to the mounting server that performs the required routing. NAS protocols offer the benefits of multi-platform support and low cost administration of client software since the client drivers for these protocols are generally available with the base operating system.

5. Segment Servers (and IBRIX Fusion Clients) are SAN-friendly. A request can be made for a file on a segment that is either owned by the server; owned by another server but accessible by this server over the SAN; or owned by another server and not accessible by this server over the SAN. In the second scenario, the server obtains the relevant metadata from the owning server and performs the I/O directly over the SAN. In the third scenario, the I/O is performed through the owning server over the IP network.

The final IBRIX file system component is not shown on the figure. Strictly speaking, you do not “need” it as the cluster can operate without it. The IBRIX Fusion Manager is the dedicated administrative management workstation. It maintains a view of the cluster configuration and IBRIX file systems, and provides access through both a graphical and a command line interface for managing and monitoring the cluster. The Fusion Manager is not involved in IBRIX file system I/O operations.
1.3 IBRIX File System Building Blocks

An IBRIX file system is created from building blocks, the first of which is the underlying physical volumes, which are combined in volume groups. Segments (logical volumes), are created from the volume groups. The built-in IBRIX volume manager handles all space allocation considerations involved in file system creation. Figure 1-2 shows the progression of building blocks in file system creation.

Physical devices can include any type of storage device, such as individual direct-attached storage (DAS), JBODs, RAID arrays, and cached disk arrays. At present, IBRIX Fusion segment formats support RAID0 striping. Depending on the particular requirements and constraints of the situation, storage devices should be protected using higher RAID mechanisms such as RAID1 or RAID5. Each segment is assigned to a Segment Server, but ownership can be migrated to a different Segment Server.

Segments can contain a mixture of files and directories or can be restricted at file system create-time to hold only directories.
1.4 IBRIX Software Components

1.4.1 Ensuring High Availability and Redundancy

The segmented architecture is also the basis for fault resilience—loss of access to one or more segments does not render the entire file system inaccessible. Individual segments can thus be taken offline temporarily for maintenance operations and then returned to the file system.

To ensure continuous data access, IBRIX Fusion High Availability (HA) provides manual and automated failover protection at various points:

• **Server:** A failed Segment Server is powered down, and its standby server assumes all of its segment management duties.

• **Segment:** Ownership of each segment on a failed Segment Server is transferred to a designated standby server.

• **Network interface:** The IP address of a failed network interface is transferred to a standby network interface until the original network interface is operational again.

• **Storage connection:** For servers with HBA-protected fibre-channel access, HBA (host bus adapter) failure triggers failover of the Segment Server to a designated standby server.

• **Fusion Manager:** Fusion Manager High Availability extends automated failover protection to the Fusion Manager and its configuration data. A failed Fusion Manager is powered down, and its standby assumes its management duties.
Chapter 2  Creating and Mounting an IBRIX File System

2.1 Overview of IBRIX File System Creation .............................................................. 7
2.2 Enabling 64-Bit Mode ......................................................................................... 8
2.3 Creating a Standard IBRIX File System in One Step ........................................ 9
2.4 Manually Creating a Standard IBRIX File System .............................................. 9
   2.4.1 Discovering Physical Volumes ................................................................. 9
   2.4.2 Creating Volume Groups .................................................................. 10
   2.4.3 Creating Logical Volumes ................................................................. 10
   2.4.4 Creating the IBRIX File System ......................................................... 11
2.5 Creating and Deleting Mountpoints ................................................................. 12
2.6 Mounting an IBRIX File System .................................................................. 13
   2.6.1 Mounting When Export Control Is Enabled ...................................... 14
2.7 Unmounting an IBRIX File System ................................................................. 15
2.8 Viewing IBRIX File System and Mountpoint Information ............................... 16
   2.8.1 Viewing Physical Volume Information ............................................. 16
   2.8.2 Viewing Volume Group Information ............................................... 16
   2.8.3 Viewing Logical Volume Information ............................................. 17
   2.8.4 Viewing IBRIX File System Information ....................................... 18
   2.8.5 Viewing Mountpoint Information ................................................... 20

2.1 Overview of IBRIX File System Creation

There are two types of IBRIX file system:

- A standard IBRIX file system in which segments can contain both directories and files.
- A data-segregated IBRIX file system, which has directory-only and file-only segments.

A standard IBRIX file system will meet the needs of most IBRIX Fusion customers. A data-segregated file system is needed only if you are planning to implement IBRIX File Replication or for special site-specific purposes.

You can choose from three ways to create either type of IBRIX file system:

- Directly from physical volumes in one step.
- Manually, by sequentially building file system components, creating the file system, and mounting it.
- Via the Fusion Manager GUI wizard.

The one-step method is the fastest and easiest method and meets the needs of most customers. The manual and Fusion Manager GUI wizard methods provide fine-grained control over the process but require more time. Familiarity with Linux LVM is helpful when using the manual or wizard method.
When naming volume groups, logical volumes, IBRIX file systems, and mountpoints, use only alphanumeric characters and the underscore (_).

### 2.2 Enabling 64-Bit Mode

If all IBRIX file system clients (NFS, CIFS, and Fusion Clients) will run only 64-bit applications, IBRIX recommends that you enable 64-bit mode when you create the file system. If you do not, the clients can run only 32-bit applications.

After you create an IBRIX file system that is compatible with 32-bit client applications, you can convert the file system to allow clients to run 64-bit applications by disabling 32-bit mode and then extending the file system with new segments. The new segments will be large enough to accommodate the files created by clients running 64-bit applications.

To determine whether 64-bit mode is enabled on an IBRIX file system, run `ibrix_fs -i`. If the output says `Compatible? : No`, 64-bit mode is enabled.

You cannot disable 64-bit mode on an IBRIX file system. This is a one-time-only operation that cannot be reversed. If there is any chance that clients will need to run a 32-bit application, do not enable 64-bit mode.
2.3 Creating a Standard IBRIX File System in One Step

The one-step method instructs the Fusion Manager to create an IBRIX file system from a specified set of physical volumes. The Fusion Manager executes the following steps to create an IBRIX file system:

1. Create one volume group for every physical volume. Name each volume group with the prefix `ivg` followed by number of its constituent physical volume (for example, `d1` becomes `ivg1`).
2. Create even-sized segments from each volume group and assign them to Segment Servers using a round-robin distribution algorithm. The segment names consist of the prefix `ilv` and the volume group number (for example, `ilv1`).
3. Create a file system from the segments and name it.

You can specify the following options for a new standard IBRIX file system:

- Enable quotas.
- Automatically create a mountpoint with the same name as the file system and mount the file system on it.
- Enable 64-bit mode.

To create without mounting, omit the `-a` argument. To create without enabling quotas, omit the `-q` argument. To enable 64-bit mode, include `-o compat=no`.

For example, to create a standard IBRIX file system `ifs1` in one step from physical volumes `d1`, `d2`, and `d3`:

```bash
<ibrixhome>/bin/ibrix_fs -c -f ifs1 -p d[1-3]
```

2.4 Manually Creating a Standard IBRIX File System

To manually create a standard IBRIX file system, build all file system components as follows:

1. Discover physical volumes in the system.
2. Create volume groups from selected discovered physical volumes.
3. Create logical volumes (that is, segments) from volume groups.
4. Create an IBRIX file system containing the segments of a specific type (mixed for a standard file system, directory-only for a segregated file system). As with the one-step method, you can create a mountpoint, enable quotas, and enable 64-bit mode.

2.4.1 Discovering Physical Volumes

The `discovery` process identifies physical volumes to the configuration database. Physical volumes must be discovered before they can be used to create or extend an IBRIX file system.

Running the `ibrix_pv -a` command without the `-o FILTERDEVLIST` argument discovers all SCSI devices. Running with this argument discovers other types of devices, such as multipath or loop devices including ATA/IDE disks, loop devices, LVM2 multipath, EMC PowerPath, and SCSI and IDE disk partitions. `FILTERDEVLIST` takes a list of one or more device name keywords or device names. SCSI devices are not reported as being discovered with the other higher-level virtual devices such as multipath drivers. Nor are physical volumes belonging to volume groups created by programs other than IBRIX Fusion discovered.
Discover SCSI physical volumes on all Segment Servers

<ibrixhome>/bin/ibrix_pv -a

Discover SCSI physical volumes on specific Segment Servers

<ibrixhome>/bin/ibrix_pv -a -h HOSTLIST

Discover physical volumes of specific types on all Segment Servers

<ibrixhome>/bin/ibrix_pv -a -h HOSTLIST -o FILTERDEVLIST

To discover on all servers, omit the -h HOSTLIST argument.

The -o FILTERDEVLIST argument takes one or more keywords representing the types of devices to discover. Valid ibrix_pv keywords are:

- allow_partitions: SCSI and IDE disk partitions
- ata: ATA/IDE disks
- loop: Loop devices
- mpath: LVM2 multipath
- powerpath: EMC PowerPath

For example, to discover SCSI and IDE disc partitions and loop devices on all Segment Servers:

<ibrixhome>/bin/ibrix_pv -a -o allow_partitions,loop

2.4.2 Creating Volume Groups

The second step in the manual method of creating an IBRIX file system is to create volume groups from specific physical volumes. For information on viewing comprehensive information about volume groups, see page 16.

Create a volume group

<ibrixhome>/bin/ibrix_vg -c -g VGNAME -p PVLIST

For example, to create volume group ivg1 from physical volumes d1, d2, and d3:

<ibrixhome>/bin/ibrix_vg -c -g ivg1 -p d[1-3]

2.4.3 Creating Logical Volumes

The third step in the manual method of creating an IBRIX file system is to create logical volumes from a volume group. Logical volumes are the segments from which you build the file system.

Optionally you can specify the logical volume size. The minimum size is 1 GB. For logical volumes larger than 1 GB, size is adjusted to the next higher 32-MB increment. If you do not specify a size, the Fusion Manager evenly divides the whole volume group among the logical volumes. You can also specify any of the Linux lvcreate options for segments. For more information on these options, refer to the ibrix_lv command description in the IBRIX Fusion CLI Reference Guide.

Create logical volumes with automatically generated names

<ibrixhome>/bin/ibrix_lv -c -n COUNT -g VGNAME -s LVNAME
Chapter 2 Creating and Mounting an IBRIX File System

The command names logical volumes with the string `LVNAME` followed by an underscore and a numeric suffix. Numeric suffixes are assigned in sequential order beginning with 1. The maximum value of the series is set by the value of `COUNT`.

For example, to create three logical volumes from volume group `ivg1` and name them `ilv_1`, `ilv_2`, and `ilv_3`:

```
<ibrixhome>/bin/ibrix_lv -c -n 3 -g ivg1 -s ilv
```

### Create logical volumes and specify names

```
<ibrixhome>/bin/ibrix_lv -c -g VGNAME -s LVLIST
```

The `-s LVLIST` argument takes a list of logical volume names.

For example, to create two logical volumes from volume group `ivg2` named `ilv1` and `ilv2`:

```
<ibrixhome>/bin/ibrix_lv -c -g ivg2 -s ilv1,ilv2
```

#### 2.4.4 Creating the IBRIX File System

In the final step in the manual method, you create the IBRIX file system from the new logical volumes—that is, the new segments.

The Fusion Manager uses a round-robin algorithm to assign segments to Segment Servers unless you specify ownership using the CLI (you cannot specify ownership using the Fusion Manager GUI). Segment 1 is the root segment for the file system.

### Create the IBRIX file system from specific segments

```
<ibrixhome>/bin/ibrix_fs -c -f FSNAME -s LVLIST [-a] [-q] [-o OPTION1=VALUE1,OPTION2=VALUE2, ...]
```

To create without mounting, omit `-a`. To create without enabling quotas, omit `-q`. To enable 64-bit mode, include as an option `-o compat=no`.

For example, to create and automatically mount IBRIX file system `ifs1` from segments `ilv_1` to `ilv_3` (note the use of the range list):

```
<ibrixhome>/bin/ibrix_fs -c -f ifs1 -s ilv_[1-3] -a
```

### Create the IBRIX file system by assigning segments to specific Segment Servers

```
<ibrixhome>/bin/ibrix_fs -c -f FSNAME -S LV1:HOSTNAME1,LV2:HOSTNAME2... [-a] [-q] [-o OPTION1=VALUE1,OPTION2=VALUE2, ...]
```

The first segment listed will be the root segment of the file system. To create without mounting, omit `-a`. To create without enabling quotas, omit `-q`. To enable 64-bit mode, include `-o compat=no`.

For example, to create IBRIX file system `ifs2` from segments `ilv1` and `ilv2` and assign `ilv1` to `lab13-66.ibrix.com` and `ilv2` to `lab13-50.ibrix.com`:

```
<ibrixhome>/bin/ibrix_fs -c -f ifs2 -S ilv1:lab13-66,ilv2:lab13-50
```
2.5 Creating and Deleting Mountpoints

Create mountpoints before attempting to mount an IBRIX file system.

Mountpoints can be created on Segment Servers, Fusion Clients, or hostgroups.

Mountpoints are immediately created for Segment Servers. For Fusion Clients the mountpoint intention is stored in the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager for its mountpoints. If IBRIX services are already running when mountpoints are created, you can force the client to query the Fusion Manager for the mountpoint intention by executing `ibrix_client` or `ibrix_lwmount -a` on the Fusion Client, or by rebooting the Fusion Client.

### Create a mountpoint on Segment Servers and Fusion Clients

```
<ibrixhome>/bin/ibrix_mountpoint -c -h HOSTLIST -m MOUNTPOINT
```

The `-h HOSTLIST` argument takes any combination of Segment Servers and Fusion Clients.

For example, to create mountpoint `/ifs1` on Segment Servers `s1.ibrix.com` and `s2.ibrix.com`:

```
<ibrixhome>/bin/ibrix_mountpoint -c
    -h s1.ibrix.com,s2.ibrix.com -m /ifs1
```

### Create a mountpoint on a hostgroup

```
<ibrixhome>/bin/ibrix_mountpoint -c -g GROUPLIST -m MOUNTPOINT
```

For example, to create mountpoint `/ifs1` on all Fusion Clients:

```
<ibrixhome>/bin/ibrix_mountpoint -c -g clients -m /ifs1
```

### Deleting Mountpoints

You can delete mountpoints on Segment Servers, Fusion Clients, and hostgroups.

Before deleting mountpoints, verify that no IBRIX file systems are mounted on them.

Segment Servers immediately delete mountpoints, but for Fusion Clients the deletion intention is stored on the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager to learn about any mountpoints that it should delete, and then it deletes them. To force the client to query the Fusion Manager for the delete intention, execute `ibrix_client` or `ibrix_lwmount -a` on the Fusion Client, or reboot the Fusion Client.

### Delete a mountpoint from Segment Servers and Fusion Clients

```
<ibrixhome>/bin/ibrix_mountpoint -d -h HOSTLIST -m MOUNTPOINT
```

The `-h HOSTLIST` argument takes any combination of Segment Servers and Fusion Clients.

For example, to delete mountpoint `/mp01` from Segment Servers `s1.ibrix.com` and `s2.ibrix.com`:

```
<ibrixhome>/bin/ibrix_mountpoint -d
    -h s1.ibrix.com,s2.ibrix.com -m /mp01
```

### Delete a mountpoint from hostgroups

```
<ibrixhome>/bin/ibrix_mountpoint -d -g GROUPLIST -m MOUNTPOINT
```

For example, to delete mountpoint `/mp01` from all Fusion Clients:

```
<ibrixhome>/bin/ibrix_mountpoint -d -g clients -m /mp01
```
2.6 Mounting an IBRIX File System

The options that are available for `ibrix_mount` and the default file system access that Fusion Clients are allowed both depend on whether the optional Export Control feature has been enabled on an IBRIX file system (see page 34 for information on Export Control). The following discussion assumes that Export Control is not enabled (this is the default condition).

Mounts are immediately executed for Segment Servers. For Fusion Clients the mount intention is stored in the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager for the IBRIX file systems that it should mount and mounts them. If IBRIX services are already running on a Fusion Client when you set new mounts, force the client to query the Fusion Manager for its allowed mounts by executing `ibrix_client` or `ibrix_lwmount -a` on the Fusion Client, or by rebooting the Fusion Client.

You can mount an IBRIX file system as follows:

- Mount on specified Segment Servers and Fusion Clients.
- Mount on a hostgroup.
- Locally mount on a Fusion Client.

You can locally override a hostgroup mount on Linux Fusion Clients and Windows Fusion Clients.

An IBRIX file system must be mounted on the Segment Server that owns the root segment (that is, segment 1) before it can be mounted on any other host. IBRIX Fusion automatically mounts a file system on the root segment when you mount on all Segment Servers in the cluster. Note that mountpoints must be created first (see page 12).

### Mount an IBRIX file system on Segment Servers and Fusion Clients

```bash
<ibrixhome>/bin/ibrix_mount -f FSNAME [-o {RW|RO}] -h HOSTLIST -m MOUNTPOINT
```

The `HOSTLIST` argument takes any combination of Segment Servers and Fusion Clients. If Export Control is enabled, include the `-o` argument.

For example, to mount file system `ifs1` on all Segment Servers and Fusion Clients at mountpoint `/ifs1`:

```bash
<ibrixhome>/bin/ibrix_mount -f ifs1 -m /ifs1
```

### Mount an IBRIX file system on a hostgroup

```bash
<ibrixhome>/bin/ibrix_mount -f FSNAME [-o {RW|RO}] -g GROUP -m MOUNTPOINT
```

To mount a file system on all Fusion Clients, specify the `clients` hostgroup. If Export Control is enabled, include the `-o` argument.

For example, to mount IBRIX file system `ifs1` on hostgroup `finance` at mountpoint `/ifs1`:

```bash
<ibrixhome>/bin/ibrix_mount -f ifs1 -g finance -m /ifs1
```

### Mount an IBRIX file system on specific nodes

```bash
<ibrixhome>/bin/ibrix_mount -f FSNAME -h HOSTLIST -m MOUNTPOINT
```

The `HOSTLIST` argument takes any combination of Segment Servers and Fusion Clients. The first Segment Server listed must own the root segment.

For example, to mount IBRIX file system `ifs1` on Segment Servers `s1.ibrix.com` (which owns the root segment) and `s2.ibrix.com` at mountpoint `/ifs1`:

```bash
<ibrixhome>/bin/ibrix_mount -f ifs1 -h s1.ibrix.com,s2.ibrix.com -m /ifs1
```
To mount IBRIX file system **ifs1** on Fusion Clients **cl01.ibrix.com** and **cl02.ibrix.com** at mountpoint **/ifs1**:

```
<ibrixhome>/bin/ibrix_mount -f ifs1
   -h cl01.ibrix.com,cl02.ibrix.com -m /ifs1
```

- **Locally mount an IBRIX file system on a Linux Fusion Client**

```
<ibrixhome>/bin/ibrix_lwmount -f fsname -m mountpoint
```

- **Locally mount an IBRIX file system on a Windows Fusion Client**

File systems are mounted using the Windows Fusion Client GUI. Open the GUI and open the Mount tab.

Select the Cluster Name from the drop-down list (the cluster name is the Fusion Manager name). Enter the IBRIX file system name to mount. Select a drive, then click **Mount**.

If you are using Remote Desktop to access the client and the drive letter does not display, log out and log back in. This is a known limitation of Windows Terminal Services when exposing new drives.

### 2.6.1 Mounting When Export Control Is Enabled

If the optional Export Control feature has been enabled for an IBRIX file system, Fusion Clients cannot access the file system until it is mounted with **ibrix_mount** command on the Fusion Manager. An attempt to locally mount a file system on a Fusion Client via **ibrix_lwmount** will fail, for example if **ibrix_mount** is not run first.

Segment Servers continue to have RW access to the file system, because they are not affected by Export Control. For information on how to enable export control, see page 34.

When mounting a file system where Export Control is enabled, you must specify that all clients have either RO or RW access to the file system. The default is RO. In addition, the root user can be limited to read-only access on export-controlled systems in a hostgroup by adding the **root_ro** parameter to the **ibrix_mount** command.

All of the **ibrix_mount** commands described in the preceding section may be used to mount an IBRIX file system for which Export Control has been enabled.

#### Examples

To mount the file system **ifs1** on all Segment Servers and Fusion Clients at mountpoint **/ifs1**, and to give the Fusion Clients RW access:

```
<ibrixhome>/bin/ibrix_mount -f ifs1 -o RW -m /ifs1
```

To mount IBRIX file system **ifs1** in hostgroup **finance** at mountpoint **/ifs1** and to give the Fusion Clients in the hostgroup RW access:

```
<ibrixhome>/bin/ibrix_mount -f ifs1 -o RW -g finance -m /ifs1
```

To run the same command, and grant Fusion Clients in hostgroup **finance** read-write access but limit the root user to read-only access:

```
<ibrixhome>/bin/ibrix_mount -f ifs1 -o RW root_ro -g finance -m /ifs1
```

To mount IBRIX file system **ifs1** on Segment Servers **s1.ibrix.com** (which owns the root segment) and **s2.ibrix.com** at mountpoint **/ifs1**, where Export Control has not be enabled for the file system:

```
<ibrixhome>/bin/ibrix_mount -f ifs1
   -h s1.ibrix.com,s2.ibrix.com -m /ifs1
```
To mount IBRIX file system `ifs1` (for which Export Control has not been enabled) on Segment Server `s1.ibrix.com` (which owns the root segment) at mountpoint `/ifs1` and on Segment Server `s2.ibrix.com` at mountpoint `/ifs1`:

```shell
<ibrixhome>/bin/ibrix_mount -f ifs1
-M s1.ibrix.com:/ifs1,s2.ibrix.com:/ifs1
```

To temporarily unmount IBRIX file system `ifs1` (for which Export Control has not been enabled) from a single Fusion Client and then re-mount it on mountpoint `/ifs1`:

```shell
<ibrixhome>/bin/ibrix_lwumount -f ifs1 -m /ifs1
<ibrixhome>/bin/ibrix_lwmount -f ifs1 -m /ifs1
```

### 2.7 Unmounting an IBRIX File System

You can unmount an IBRIX file system from Segment Servers, Fusion Clients, and hostgroups using the Fusion Manager CLI or GUI. You can also locally unmount a file system from Fusion Clients. For information on hostgroups, see page 31.

Segment Servers immediately unmount a file system, but for Fusion Clients the unmount intention is stored on the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager for the IBRIX file systems that it should unmount, and then it unmounts them. For information on how to force Fusion Clients to unmount a file system when IBRIX services are running on them at the time that the unmount intention is stored, see the preceding section.

Unmount the root segment last. Attempting to unmount it while other segments are still mounted will result in failure. If the IBRIX file system was exported using NFS, you must unexport it before you can unmount it (see page 73).

#### Unmount an IBRIX file system from Segment Servers and Fusion Clients

Execute one of the following commands:

```shell
<ibrixhome>/bin/ibrix_umount -f FSNAME [-m MOUNTPOINT] [-h HOSTLIST]
```

```shell
<ibrixhome>/bin/ibrix_umount -m MOUNTPOINT [-f FSNAME] [-h HOSTLIST]
```

The `HOSTLIST` argument takes any combination of Segment Servers and Fusion Clients. To unmount from all Segment Servers and Fusion Clients, omit the `HOSTLIST` argument.

For example, to unmount IBRIX file system `ifs1` from mountpoint `/ifs1` on Segment Servers `s1.ibrix.com` and `s2.ibrix.com`:

```shell
<ibrixhome>/bin/ibrix_umount -f ifs1 -m /ifs1
-h s1.ibrix.com,s2.ibrix.com
```

#### Unmount an IBRIX file system from a hostgroup

Execute one of the following commands:

```shell
<ibrixhome>/bin/ibrix_umount -g GROUPLIST -f FSNAME
```

```shell
<ibrixhome>/bin/ibrix_umount -g GROUPLIST -m MOUNTPOINT
```

For example, to unmount IBRIX file system `ifs1` from all Fusion Clients:

```shell
<ibrixhome>/bin/ibrix_umount -g clients -f ifs1
```
Locally unmount an IBRIX file system from a Linux Fusion Client

Execute one of the following commands on the client:

\[
<\text{ibrixxhome}>/\text{bin}/\text{ibrix\_lwumount}\ -f\ \text{FSNAME}\ -m\ \text{MOUNTPOINT}
\]

\[
<\text{ibrixxhome}>/\text{bin}/\text{ibrix\_lwumount}\ -m\ \text{MOUNTPOINT}
\]

For example, to unmount IBRIX file system ifs1 from mountpoint /ifs1:

\[
<\text{ibrixxhome}>/\text{bin}/\text{ibrix\_lwumount}\ -f\ \text{ifs1}\ -m\ /\text{ifs1}
\]

Locally unmount an IBRIX file system from a Windows Fusion Client:

Click the Umount tab, select the IBRIX file system, then click **Umount**.

2.8 Viewing IBRIX File System and Mountpoint Information

You can use the Fusion Manager GUI to view comprehensive information about an IBRIX file system and mountpoints.

2.8.1 Viewing Physical Volume Information

Physical volume descriptors are defined in Table 2-1.

List all discovered physical volumes

\[
<\text{ibrixxhome}>/\text{bin}/\text{ibrix\_pv}\ -l
\]

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Physical volume name. Regular physical volume names begin with the letter (d). The names of physical volumes that are part of a mirror device begin with the letter (m). Both are numbered sequentially.</td>
</tr>
<tr>
<td>Size (MB)</td>
<td>Physical volume size in MB.</td>
</tr>
<tr>
<td>VG Name</td>
<td>Name of volume group created on this physical volume, if any.</td>
</tr>
<tr>
<td>RAID type</td>
<td>RAID type 1.</td>
</tr>
<tr>
<td>RAID host</td>
<td>The import host, where the RAID device is created. Also called the primary host.</td>
</tr>
<tr>
<td>RAID device</td>
<td>The name of the RAID device, usually identified by the letter (m) as the first character in the name.</td>
</tr>
<tr>
<td>Network host</td>
<td>The export host, where the volume is copied from. Also called the secondary host.</td>
</tr>
<tr>
<td>Network port</td>
<td>The port used for mirroring transfers.</td>
</tr>
</tbody>
</table>

2.8.2 Viewing Volume Group Information

List all volume groups

To display summary information about all volume groups:

\[
<\text{ibrixxhome}>/\text{bin}/\text{ibrix\_vg}\ -l
\]
The \texttt{VG\_FREE} field indicates the amount of group space that is not allocated to any logical volume. The \texttt{VG\_USED} field reports the percentage of available space that is allocated to a logical volume.

\section*{View information about volume groups}

To display detailed information about one or more volume groups:

\begin{verbatim}
<ibrixhome>/bin/ibrix_vg -i [-g VGLIST]
\end{verbatim}

Including \texttt{VGLIST} restricts the listing to the named volume groups. Refer to Table 2-2 for descriptions of all fields.

\begin{table}[h]
\centering
\caption{ibrix\textunderscore vg -l and ibrix\textunderscore vg -i Volume Groups Summary}
\begin{tabular}{|l|l|}
\hline
Field & Definition \\
\hline Name & Volume group name. \\
Size (MB) & Volume group size in MB. \\
Free (MB) & Free (unallocated) space in MB available on this volume group. \\
Used (percentage) & Percentage of total space in the volume group allocated to logical volumes. \\
File System Name & Name of file system to which this LV belongs. \\
Physical Volume Name & Name of physical volume used to create this volume group. \\
Physical Volume Size & Size in MB of the physical volume used to create this volume group. \\
Logical Volume Name & Name of logical volumes created from this volume group. \\
Logical Volume Size & Size in MB of each logical volume created from this volume group. \\
File System Name & Name of file system created from this logical volume. \\
File System Generation & Number of times the structure of the file system has changed (for example, new segments were added) \\
Segment Number & Number of this segment (logical volume) in the file system. \\
Host Name & Segment Server that owns this logical volume. \\
State & Operational state of Segment Server. Refer to Table 3-1 for a list of all Segment Server states. \\
\hline
\end{tabular}
\end{table}

\subsection*{2.8.3 Viewing Logical Volume Information}

The \texttt{OPTIONS} field in the output for this command lists \texttt{lvcreate} options that have been set on a volume group. Table 2-3 defines the output fields.

\section*{List all logical volumes}

\begin{verbatim}
<ibrixhome>/bin/ibrix\_lv -l
\end{verbatim}

\begin{table}[h]
\centering
\caption{ibrix\_lv -l Logical Volumes Summary}
\begin{tabular}{|l|l|}
\hline
Field & Definition \\
\hline Name & Logical volume name. Regular physical volume names begin with the letter \texttt{d}. The names of physical volumes that are part of a mirror device begin with the letter \texttt{m}. Both are numbered sequentially. \\
Size (MB) & Logical volume size in MB. \\
\hline
\end{tabular}
\end{table}
2.8.4 Viewing IBRIX File System Information

If your license includes IBRIX Fusion Snap, the `ibrix_fs -i` commands will also display information about any IBRIX file system snapshots.

- **List all IBRIX file systems**

  `<ibrixhome>/bin/ibrix_fs -l`

  Refer to Table 2-4 for definitions of the output fields.

- **View information about an IBRIX file system**

  `<ibrixhome>/bin/ibrix_fs -i [-f FSLIST]`

  To view information for all IBRIX file systems, omit the `-f FSLIST` argument. See Table 2-5 for definitions of the servers output fields, and Table 2-6 for definitions of the per-segment output fields.

  For example, to view information about IBRIX file system `ifs1`:

  `<ibrixhome>/bin/ibrix_fs -i -f ifs1`

  **Table 2-3  ibrix_lv -l Logical Volumes Summary (Continued)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>File System Name</td>
<td>File system to which this LV belongs.</td>
</tr>
<tr>
<td>Segment Number</td>
<td>The import host, where the RAID device is created. Also called the primary host.</td>
</tr>
<tr>
<td>VG Name</td>
<td>Name of volume group created on this physical volume, if any.</td>
</tr>
<tr>
<td>Options</td>
<td>Linux <code>lvcreate</code> options that have been set on the volume group.</td>
</tr>
</tbody>
</table>

  **Table 2-4  ibrix_fs -i File Systems Summary**

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>File system name.</td>
</tr>
<tr>
<td>Number Segments</td>
<td>Number of file system segments.</td>
</tr>
<tr>
<td>Mounted</td>
<td>Total space available in file system.</td>
</tr>
<tr>
<td>Generation</td>
<td>Number of times the structure of the file system has changed (for example, new segments were added).</td>
</tr>
<tr>
<td>Quota Enabled</td>
<td>Yes if enabled, No if not.</td>
</tr>
<tr>
<td>Export Control Enabled</td>
<td>Yes if enabled, No if not.</td>
</tr>
<tr>
<td>Compatible</td>
<td>Yes if 32-bit compatible, No if 64-bit compatible.</td>
</tr>
</tbody>
</table>

  **Table 2-5  ibrix_fs -i File System Summary**

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Segments</td>
<td>Number of segments.</td>
</tr>
<tr>
<td>Mounted</td>
<td>Yes if this file system is mounted, No if not.</td>
</tr>
<tr>
<td>Mirrored</td>
<td>Yes if mirrored by IBRIX FileRAID, No if not.</td>
</tr>
</tbody>
</table>
### Table 2-5  
*ibrix_fs -i File System Summary (Continued)*

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compatible</td>
<td>Yes indicates the file system is 32-bit compatible, and if so, the maximum number of segments (maxsegs) allowed in this file system. No indicates a 64-bit file system.</td>
</tr>
<tr>
<td>Generation</td>
<td>Number of times the structure of the file system has changed (for example, new segments were added).</td>
</tr>
<tr>
<td>ID</td>
<td>File system ID for NFS access.</td>
</tr>
<tr>
<td>Number</td>
<td>Unique IBRIX Fusion internal file system identifier.</td>
</tr>
<tr>
<td>Export Control Enabled</td>
<td>Yes if enabled, No if not.</td>
</tr>
<tr>
<td>Quota Enabled</td>
<td>Yes if enabled, No if not.</td>
</tr>
<tr>
<td>Default Blocksize</td>
<td>Default block size (KB).</td>
</tr>
<tr>
<td>Blocks</td>
<td>Total number of 1k blocks in the file system.</td>
</tr>
<tr>
<td>Blocks Free (BFREE)</td>
<td>Number of free 1k storage blocks in this file system.</td>
</tr>
<tr>
<td>Blocks Available (BAVAIL)</td>
<td>Number of 1k storage blocks available for user files.</td>
</tr>
<tr>
<td>Percent Used</td>
<td>Percentage of total storage occupied by user files.</td>
</tr>
<tr>
<td>Files</td>
<td>Number of files that can be created in this file system.</td>
</tr>
<tr>
<td>Files Free (FFREE)</td>
<td>Number of unused file inodes available in this file system</td>
</tr>
<tr>
<td>Spillover Watermark</td>
<td>Defined as the point where the allocation policy manager no longer write partial files that spill over from another segment, defined as a percentage of available storage. The default spillover watermark is 98 percent of segment storage capacity.</td>
</tr>
<tr>
<td>Create Watermark</td>
<td>Defined as the point where the allocation policy manager no longer creates new file system objects on this segment, defined as a percentage of available storage. Writes to existing files are allowed, however. The default create watermark is 98 percent of segment storage capacity.</td>
</tr>
<tr>
<td>Write Watermark</td>
<td>The number of MB that must remain free on a segment. If the available block count of a segment falls below this value, no more data can be written to it. Attempts to append to a file on this segment spill over to another segment. If no other segment is available, the write fails. The default create watermark is 10 MB.</td>
</tr>
<tr>
<td>Prealloc(ation)</td>
<td>The number of KB an IBRIX file system preallocates to a file. Default: 256 KB.</td>
</tr>
<tr>
<td>Read Ahead</td>
<td>Specifies the number of KB Fusion will pre-fetch. The default value is 128 KB.</td>
</tr>
<tr>
<td>NFS Read Ahead</td>
<td>Specifies the number of KB Fusion pre-fetches under NFS. The default value is 128 KB.</td>
</tr>
<tr>
<td>Default Policy</td>
<td>The allocation policy assigned on this file system. Defined policies are: ROUNDROBIN, STICKY, DIRECTORY, LOCAL, RANDOM, and NONE. Refer to Section 8.1, How File Allocation Works, for information on these policies.</td>
</tr>
<tr>
<td>Default Start Segment</td>
<td>Identifies the first segment to which an allocation policy is applied in an IBRIX file system. If none is specified, starts on the segment with the most storage space available.</td>
</tr>
</tbody>
</table>
2.8.5 Viewing Mountpoint Information

You can view all mountpoints, mountpoints on given Segment Servers, or Segment Servers on which mountpoints exist.

- **List mountpoints on Segment Servers**

  ```bash
  <ibrixhome>/bin/ibrix_mountpoint -l
  ```

  This command lists all Segment Server names, their mountpoints, and the names of mounted file systems.

  To list mountpoints on specific Segment Servers:

  ```bash
  <ibrixhome>/bin/ibrix_mountpoint -l -h HOSTLIST
  ```

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment Number</td>
<td>Number of segments.</td>
</tr>
<tr>
<td>Owner</td>
<td>Yes if this file system is mounted, No if not.</td>
</tr>
<tr>
<td>LV Name</td>
<td>Yes if mirrored by IBRIX FileRAID, No if not.</td>
</tr>
<tr>
<td>State</td>
<td>File system ID for NFS access.</td>
</tr>
<tr>
<td>Blocksize</td>
<td>Default block size (KB).</td>
</tr>
<tr>
<td>Blocks</td>
<td>Total number of 1k blocks in the file system.</td>
</tr>
<tr>
<td>Blocks Free (BFREE)</td>
<td>Number of free 1k storage blocks in this file system.</td>
</tr>
<tr>
<td>Blocks Available (BAVAIL)</td>
<td>Number of 1k storage blocks available for user files.</td>
</tr>
<tr>
<td>Files</td>
<td>Number of files that can be created in this file system.</td>
</tr>
<tr>
<td>Files Free (FFREE)</td>
<td>Number of unused file inodes available in this file system.</td>
</tr>
<tr>
<td>Percent Used</td>
<td>Percentage of total storage occupied by user files.</td>
</tr>
<tr>
<td>Backup</td>
<td>Defined as the point where the allocation policy manager no longer write partial files that spill over from another segment, defined as a percentage of available storage. The default spillover watermark is 98 percent of segment storage capacity.</td>
</tr>
<tr>
<td>On Backup</td>
<td>Flag indicating whether the segment is on the backup.</td>
</tr>
<tr>
<td>In Fsck?</td>
<td>Yes if the file system is being checked by <code>ibrix_fsck</code>, No if not.</td>
</tr>
<tr>
<td>Type</td>
<td>Segment type, with respect to data segregation (mixed, directory-only, file-only).</td>
</tr>
<tr>
<td>In Replication?</td>
<td>Yes if file system is being replicated, No if not.</td>
</tr>
<tr>
<td>In Rebalance?</td>
<td>Yes if segments are being rebalanced, No if not.</td>
</tr>
<tr>
<td>Tier</td>
<td>The tier to which segment was assigned.</td>
</tr>
<tr>
<td>Host</td>
<td>Hosts that the file system is mounted on.</td>
</tr>
<tr>
<td>Mountpoint</td>
<td>Host mountpoint.</td>
</tr>
<tr>
<td>Access</td>
<td>File system access privileges, either RO or RW.</td>
</tr>
<tr>
<td>Root_RO</td>
<td>Specifies whether the root user is limited to read-only access, regardless of the Access setting.</td>
</tr>
</tbody>
</table>
List Segment Servers on which specific mountpoints have been created

<ibrixhome>/bin/ibrix_mountpoint -l -h MOUNTPOINTLIST
Chapter 3  Managing the Cluster

3.1 Monitoring Cluster Operational Status .......................... 24
   3.1.1 Monitoring Segment Server Operational Status .............. 24
   3.1.2 Monitoring Cluster Events ................................ 25
3.2 Monitoring Cluster Health .................................. 26
   3.2.1 Health Checks Performed ................................ 26
   3.2.2 Viewing Health Check Reports ............................. 26
3.3 Registering Fusion Clients .................................. 28
3.4 Extending an IBRIX File System .............................. 29
3.5 Disabling 32-Bit Mode on an IBRIX File System ............... 30
3.6 Migrating Segments ........................................ 30
3.7 Using Hostgroups ......................................... 31
   3.7.1 How Hostgroups Work ................................ 31
   3.7.2 Creating a Hostgroup Tree ............................... 32
   3.7.3 Adding a Fusion Client to a Hostgroup ................. 33
   3.7.4 Adding a Domain Rule to a Hostgroup ................ 33
   3.7.5 Performing Operations on Hostgroups ................. 33
   3.7.6 Deleting Hostgroups ................................ 33
   3.7.7 Viewing Hostgroups ................................ 34
3.8 Setting Up Quotas ........................................ 37
   3.8.1 How Quotas Work ................................ 37
   3.8.2 Enabling Quotas on an IBRIX File System .............. 37
   3.8.3 Setting User and Group Quotas .......................... 37
   3.8.4 Reinitializing Quota Usage Limits ....................... 38
   3.8.5 Deleting User and Group Quotas ....................... 38
   3.8.6 Viewing Quota Settings ................................ 39
3.9 Configuring Email Notification Settings ...................... 40
   3.9.1 Associating Events and Email Addresses ............... 40
   3.9.2 Configuring Email Notification Settings ............... 40
   3.9.3 Dissociating Events and Email Addresses ............. 41
   3.9.4 Turning Email Notifications On and Off ............... 41
   3.9.5 Testing Email Addresses ............................... 41
   3.9.6 Viewing Email Notification Settings ................... 41
3.10 Rebalancing Segments in an IBRIX File System ............... 42
   3.10.1 How Rebalancing Works ................................ 42
   3.10.2 Rebalancing All Segments ................................ 42
   3.10.3 Rebalancing Specific Segments ......................... 42
   3.10.4 Tracking the Progress of a Rebalance Job .......... 42
   3.10.5 Viewing the Status of Rebalance Jobs ............... 42
   3.10.6 Stopping Rebalance Jobs ................................ 42
   3.10.7 Cleaning Up After a Failed Rebalance Run ............ 42
3.11 Setting Up Quotas ........................................ 37
   3.11.1 How Quotas Work ................................ 37
   3.11.2 Enabling Quotas on an IBRIX File System ............ 37
   3.11.3 Setting User and Group Quotas ........................ 37
   3.11.4 Reinitializing Quota Usage Limits ..................... 38
   3.11.5 Deleting User and Group Quotas ..................... 38
   3.11.6 Viewing Quota Settings .............................. 39
3.12 Setting Up Email Notification of Cluster Events .......... 40
   3.12.1 Associating Events and Email Addresses .............. 40
   3.12.2 Configuring Email Notification Settings .............. 40
   3.12.3 Dissociating Events and Email Addresses .......... 41
   3.12.4 Turning Email Notifications On and Off ............. 41
   3.12.5 Testing Email Addresses ............................. 41
   3.12.6 Viewing Email Notification Settings ............... 41
3.13 Setting Up SNMP Notifications ................................ 42
   3.13.1 Configuring the SNMP Agent ............................ 42
3.1 Monitoring Cluster Operational Status

Instructions in this chapter describe command line interface operations. The Fusion Manager GUI provides a browser-based display of Segment Server status, IBRIX file system usage, and cluster events.

3.1.1 Monitoring Segment Server Operational Status

Information on Segment Servers is available through the `ibrix_host -l` command or the Servers panel in the Fusion Manager GUI. These summarize CPU, I/O, and network performance information and the operational state of Segment Servers.

Segment Servers can be in one of three operational states: Normal, Alert, or Error. These states are further broken down into categories, mostly related to the failover status of a host. See Table 3-1 for an explanation of the state categories.

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTO_FAILOVER</td>
<td>Indicates whether automated failover is turned on or off.</td>
</tr>
<tr>
<td>BACKUP</td>
<td>Standby server name.</td>
</tr>
<tr>
<td>HOSTNAME</td>
<td>Fusion Client that is registered with the Fusion Manager.</td>
</tr>
<tr>
<td>HOSTNAME</td>
<td>Segment Server name.</td>
</tr>
<tr>
<td>ID</td>
<td>Server internal ID.</td>
</tr>
<tr>
<td>ID</td>
<td>Fusion Client internal ID.</td>
</tr>
<tr>
<td>IPADDRESS</td>
<td>Fusion Client IP address.</td>
</tr>
<tr>
<td>STATE: Normal</td>
<td>Up: Operational.</td>
</tr>
</tbody>
</table>
**Chapter 3 Managing the Cluster**

### 3.1.2 Monitoring Cluster Events

IBRIX events fall into categories based on level of severity:

- **Alerts**: Disruptive events that can result in loss of access to file system data, for example when a segment is unavailable or a server is unreachable.
- **Warnings**: Potentially disruptive conditions in which file system access is not lost, but if the situation is not addressed it can escalate to an alert condition, for example reaching a very high server CPU utilization or nearing a quota limit.
- **Information**: Events occurring under normal or non-threatening conditions that change the cluster, for example a segment is created or a file system mounted.

The Fusion Manager GUI Events panel displays the events.

You can set up event notifications via email (Section 3.13) or SNMP traps (Section 3.14).

#### Monitor cluster events

```bash
<ibrixhome>/bin/ibrix_event -l [-n EVENTS_COUNT]
```

---

**Table 3-1  Output Fields for ibrix_host -l (Continued)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>STATE: Alerts</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Up-Alert</strong>: Server has encountered a condition that has been logged. An event will appear in the Status tab of the Fusion Manager GUI, and an email notification may be sent. <strong>Up-InFailover</strong>: Server is powered on and visible to the Fusion Manager, and the Fusion Manager is failing over the server’s segments to a standby server. <strong>Up-FailedOver</strong>: Server is powered on and visible to the Fusion Manager, but failover is complete.</td>
<td></td>
</tr>
<tr>
<td><strong>STATE: Errors</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Down-InFailover</strong>: Server is powered down or inaccessible to the Fusion Manager, and the Fusion Manager is failing over server segments to a standby server. <strong>Down-FailedOver</strong>: Server is powered down or inaccessible to the Fusion Manager, and failover is complete. <strong>Down</strong>: Server is powered down or not accessible to the Fusion Manager, and no standby server is providing access to the server’s segments</td>
<td></td>
</tr>
</tbody>
</table>

---

**Monitor Segment Server status**

```bash
<ibrixhome>/bin/ibrix_host -l
```

The command also lists Fusion Clients that are registered with the Fusion Manager.

**Sample Output**

```plaintext
HOSTNAME  BACKUP  STATE  AUTO_FAILOVER  ID
lab13-190  lab13-40  Up  No  336493cf-ec2d-4dec-bc3e-caa8ab98e683

Clients:
HOSTNAME  IPADDRESS  ID
vc02  192.168.12.157  dd7bc201-cb4c-db11-b36d-0013723f92b3
```
3.2 Monitoring Cluster Health

To monitor the functional health of Segment Servers and Fusion Clients, run the `ibrix_health` command. This command checks host performance in a variety of functional areas and provides either a summary or a detailed report of the results.

3.2.1 Health Checks Performed

On Segment Servers, `ibrix_health` runs these health checks:

- Pings remote Segment Servers that share a network with the test hosts. Remote servers that are pingable may not be connected to a test host because of a Linux or IBRIX Fusion problem. Remote servers that are not pingable may be down or may have a network problem.
- If test hosts are assigned to be network interface monitors, pings their monitored interfaces to assess the health of the connection (for information on network interface monitoring, see Section 4.9, Setting Up Network Interface Monitoring).
- Determines whether specified hosts can read their physical volumes.

On Segment Servers and Fusion Clients, `ibrix_health` runs these health checks:

- Determines whether information maps on the tested hosts are consistent with the configuration database.

If you include the `-b` argument, the command also checks the health of Segment Servers high-availability standbys, if any are configured.

3.2.2 Viewing Health Check Reports

The summary report provides an overall health check result for all tested Segment Servers and Fusion Clients, followed by individual results for each. If you include the `-b` argument, the standbys for all tested Segment Servers are included when the overall result is determined.

The following results may be reported:

- **Passed:** All tested hosts and standbys passed every health check.
- **Failed:** One or more tested hosts failed a health check. The health status of Segment Server standbys is not included when this result is calculated.
- **Warning:** A suboptimal condition that may require your attention was found on one or more tested hosts or standbys.

In the case of the results for individual Segment Servers and Fusion Clients, the terms have the same meaning, but Segment Server standbys are not considered when the result is determined for the server.

The detailed report consists of the summary report plus the following additional data: a summary of the test results; host information such as operational state, performance data, and version data; non-default host tunings, and the results of the health checks. By default, the “Result Information” field in a detailed report provides data only for health checks that received a Failed or a Warned result. Optionally you can expand a detailed report to also provide data about checks that received a Passed result, as well as details about the IBRIX file system and segments.

View a summary health report

`<ibrixhome>/bin/ibrix_health -l [-h HOSTLIST] [-f] [-b]`

By default the command reports on all hosts. To view a report on specified hosts, include the `-h HOSTLIST` argument. To view results only for hosts that failed the check, include the `-f` argument. To include Segment Server standbys in the health check, include the `-b` argument.
For example, to view a summary report for Segment Server ibrix080 and Fusion Client lab13-116:

```bash
<ibrixhome>/bin/ibrix_health -l -h ibrix080,lab13-116
```

**Sample Output**

```
--------------- Overall Health Checker Results ---------------
PASSED
--------------- Host Summary Results ---------------
Host Result Type State Last Update
========= ====== ====== ===== ============================
ibrix080 PASSED Server Up Mon Apr 09 16:45:03 EDT 2007
lab13-116 PASSED Client Up Mon Apr 09 16:07:22 EDT 2007
```

**View a detailed health report**

```bash
<ibrixhome>/bin/ibrix_health -i -h HOSTLIST [-f] [-s] [-v]
```

To view results only for hosts that failed the check, include the `-f` argument. To expand the report to include information about the IBRIX file system and its segments, include the `-s` argument. To include details about checks that received a Passed result, include the `-v` argument.

For example, to view a basic detailed health report for Segment Server lab13-116:

```bash
<ibrixhome>/bin/ibrix_health -l -h lab13-116
```

**Sample Output**

```
--------------- Overall Health Checker Results ---------------
PASSED
--------------- Host Summary Results ---------------
Host Result Type State Last Update
==============  ======  ======  =====  ============================
xso1.ibrix.com PASSED  Server  Up     Wed May 09 17:24:50 EDT 2007
```

`xs01.ibrix.com Report`

```
Result Type State Module Up Time Last Update
Network Threads Protocol
PASSED Server Up Loaded 778.19 Wed May 09 17:24:50 EDT 2007
```

**Cpu(System,User,Util,Nice) Load(1,3,15 min) Network(Bps)**

```
Disk(Bps)
```

```
========
0, 0, 0, 0 0.24, 0.37, 0.29 5828 15872
```

**Mem Total Mem Free Buffers(KB) Cached(KB) SwapTotal(KB) Swap Free(KB)**

```
Mem Total Mem Free Buffers(KB) Cached(KB) SwapTotal(KB) Swap Free(KB)
```

```
========
2055828 1420088 51620 438320 2048248 2048248
```

**Fs Version IAD Version OS Kernel Version Architecture Processor**

```
========
```

```
IF001-4.3 IBRIX Fusion User Guide
```
3.3 Registering Fusion Clients

Fusion Clients must be manually registered with the Fusion Manager before they can mount an IBRIX file system. Segment Servers are registered when they are installed. Linux Fusion Clients are registered on the Fusion Manager itself, while Windows Fusion Clients must first be configured for Active Directory (refer to Section 7.6, Configure Active Directory Settings on Fusion Manager), and then registered on each client machine, through the Fusion Client GUI (refer to Section 7.7, Register Windows Fusion Clients and Start Service).

To register a Linux Fusion Client using the CLI:

```bash
<ibrixhome>/bin/ibrix_host -a -h HOST -e IPADDRESS
```

For example, to register Fusion Client client12.ibrix.com that is accessible at IP address 192.168.2.12:

```bash
<ibrixhome>/bin/ibrix_host -a -h client12.ibrix.com -e 192.168.2.12
```

3.4 Registering Multicluster Clients

A multicluster client is one that is homed to more than one Fusion Manager and is thus a member of more than one cluster. The procedure to configure such a remote client involves running programs as the root user on both the client and the Fusion Manager.

The remote Fusion Client and the new cluster must be in the same subnet and routable to each other. The remote client and the new cluster’s Fusion Manager must be running the same version of IBRIX Fusion.

The Fusion Client must be installed and registered on its first cluster before running this procedure.

1. From the Fusion Client, register the client with the new cluster:

   ```bash
   # register_client -p IPADDRESS -c IFNAME
   ```

   Where IPADDRESS is the address of the new Fusion Manager and IFNAME is the interface used for cluster communication. The new Fusion Manager entry is added to the client iadconf.xml file, and can be listed with ibrix_host on the new Fusion Manager.

2. Restart Fusion Client services:

   ```bash
   # /etc/init.d/ibrix_client restart
   ```

3. On the new cluster’s Fusion Manager, create a mountpoint for the new client and set the mount intent; in this example the file system is ifs1, the mountpoint is /mnt_ifs1, and the client’s host name is client1.net.com.

   ```bash
   # <ibrixhome>/bin/ibrix_mountpoint -c -m /mnt_ifs1 -h client1.net.com
   ```
4. On the client, mount the file system and verify.

```
# ibrix_lwmount -f cluster.net.com:ifs1 -m /mnt_ifs1
# mount
```

**Note!** Current IBRIX Fusion versions do not include a command for removing a client from a cluster. The workaround involves editing the iadconf.xml file on the client. Contact IBRIX Customer Support for details.

## 3.5 Extending an IBRIX File System

To extend an IBRIX file system, use one of the following methods to add segments or tiered segments:

- Extend in one step by allowing IBRIX Fusion to create and add new segments from specified physical volumes.
- Add specific logical volumes.

In both cases, segments are added to Segment Servers in a round-robin manner.

You can extend with directory-only or file-only segments only if the IBRIX file system was created to hold data-segregated segments.

### Extend an IBRIX file system in one step

```
<ibrixhome>/bin/ibrix_fs -e -f FSNAME -p PVLIST
```

For example, to extend IBRIX file system ifs1 with segments created on physical volumes d1 and d2:

```
<ibrixhome>/bin/ibrix_fs -e -f ifs1 -p d[1-2]
```

### Extend a file system with specific logical volumes

```
<ibrixhome>/bin/ibrix_fs -e -f FSNAME -s LVLIST
```

For example, to add logical volumes ilv1, ilv2, and ilv3 to IBRIX file system ifs1:

```
<ibrixhome>/bin/ibrix_fs -e -f ifs1 -s ilv[1-3]
```

### Extend a file system with specific logical volumes located on specific Segment Servers

```
<ibrixhome>/bin/ibrix_fs -e -f FSNAME
-S LV1:HOSTNAME1, LV2:HOSTNAME2...
```

For example, to add logical volume ilv1 (owned by s1.ibrix.com) to IBRIX file system ifs1:

```
<ibrixhome>/bin/ibrix_fs -e -f ifs1 -S ilv1:s1.ibrix.com
```

### Extend a file system with data-segregated segments

Execute one of the following commands:

```
<ibrixhome>/bin/ibrix_fs -e -f FSNAME -s LVLIST [-D]

<ibrixhome>/bin/ibrix_fs -e -f FSNAME
-S LV1:HOSTNAME1, LV2:HOSTNAME2... [-D]

<ibrixhome>/bin/ibrix_fs -e -f FSNAME -p PVLIST [-D]
```
If file system `FSNAME` was created to contain data-segregated segments, include the `-D` argument with these commands to extend directory-only segments, or omit it to extend file-only segments. If not created for data segregation, the added segments are of mixed-use type.

For example, to add file-only segments created from physical volumes `d1` and `d2` to IBRIX file system `ifs1`:

```shell
<ibrixhome>/bin/ibrix_fs -e -f ifs1 -p d[1-2]
```

### Extend a file system with tiered segments

```shell
<ibrixhome>/bin/ibrix_fs -e -f FSNAME -s LVLIST -t TIERNM
```

This extends the IBRIX file system `FSNAME` with the segments in `LVLIST` assigned to `TIERNM`. As tiering cannot be used with directory segments, the `-D` argument is ignored in any attempt to expand a file system with both the `-t` and `-D` arguments. If tiering rules are already defined for this file system, `-t` is required.

Avoid the situation where a tiering job runs at the same time as a file system expansion. The expansion takes priority and the tiering job is killed.

For example, to extend file system `ifs1` with logical volumes `ilv1`, `ilv2`, and `ilv3` on tier2:

```shell
<ibrixhome>/bin/ibrix_fs -e -f ifs1 -s ilv[1-3] -t tier2
```

A similar command exists for extending with physical volumes:

Refer to the IBRIX Data Tiering Reference Guide for additional syntax examples for expanding a file system with data tiers.

### 3.6 Disabling 32-Bit Mode on an IBRIX File System

If your cluster clients are converting from using 32-bit to using 64-bit applications, you must disable 32-bit mode on the IBRIX file system, which enables 64-bit mode. For information on 64-bit mode, see page 8.

**Note!** You cannot disable 64-bit mode on an IBRIX file system. This is a one-time-only operation that cannot be reversed. If there is a chance that clients will ever need to run a 32-bit application, do not enable 64-bit mode.

To disable 32-bit mode using the CLI:

1. Unmount the IBRIX file system (see page 15).
2. To disable 32-bit mode on the named file system, run:
   ```shell
   <ibrixhome>/bin/ibrix_fs -w -f FSNAME
   ```
3. Re-mount the file system (see page 13).

### 3.7 Migrating Segments

Segment ownership can be transferred from one host to another to improve cluster performance through segment migration. The segment migration functionality is licensed separately and must be activated on the Fusion Manager prior to use.

Segment migration does not move segments from their physical locations in networked storage systems. It only transfers segment ownership in the configuration database. This ownership data is part of the metadata that the Fusion Manager distributes to Segment Servers and Fusion Clients so that they can locate segments.
3.8 Using Hostgroups

A hostgroup is a named set of Fusion Clients. Hostgroups provide a convenient way to centrally manage Fusion Clients using the Fusion Manager. You can put different sets of Fusion Clients into hostgroups and then perform the following operations on all members of the group:

- Create and delete mountpoints.
- Mount IBRIX file systems.
- Prefer a network interface.
- Tune host parameters.
- Set allocation policies.

Hostgroups are optional. If you do not choose to set them up, you can mount IBRIX file systems on Fusion Clients and tune host settings and allocation policies on an individual level.

3.8.1 How Hostgroups Work

In the simplest case, the hostgroups functionality allows you to perform an allowed operation on all Fusion Clients by executing a Fusion Manager command on the default clients hostgroup via either the CLI or the GUI. The command intention is stored on the Fusion Manager until the next time the Fusion Clients contact the Fusion Manager, which happens when you restart IBRIX services on them, reboot them, or execute ibrix_lwmount -a or ibrix_lwhost --a. When contacted, the Fusion Manager informs the Fusion Clients about commands that were executed on hostgroups to which they belong. The Fusion Clients then use this information to perform the operation.

The simple scenario allows you to perform the same operation on all Fusion Clients. If you want to do different things to different sets of Fusion Clients, you need to assign them to different hostgroups. This involves creating a hostgroup tree whose root element is the clients hostgroup. Each hostgroup in a tree can have exactly one parent, and a parent can have multiple children (see Figure 3-1).
Fusion Clients must be explicitly assigned to non-default hostgroups. You can do this manually or have the Fusion Manager automatically perform the assignment when you register a Fusion Client. To set up automatic hostgroup assignments, define a domain rule for hostgroups. A domain rule restricts hostgroup membership to Fusion Clients on a particular cluster subnet. The Fusion Manager uses the IP address that you specify for Fusion Clients when you register them to perform a subnet match and sort the Fusion Clients into hostgroups based on the domain rules. Setting domain rules on hostgroups thus provides a convenient way to centrally manage mounting, tuning, allocation policies, and preferred networks on different subnets of Fusion Clients.

In a hostgroup tree, operations performed on lower-level nodes take precedence over operations performed on higher-level nodes. This means that you can effectively establish global Fusion Client settings, which you can override for specific Fusion Clients.

For example, suppose that you want all Fusion Clients to be able to mount IBRIX file system `ifs1` and to implement a set of host tunings denoted as Tuning 1, but you want to override these global settings for certain hostgroups (see Figure 3-1). You would mount `ifs1` on the `clients` hostgroup, `ifs2` on hostgroup A, `ifs3` on hostgroup C, and `ifs4` on hostgroup D, in any order. You would set Tuning 1 on the `clients` hostgroup and Tuning 2 on hostgroup B. The end result is that all the Fusion Clients in hostgroup B will mount `ifs1` and implement Tuning 2. The Fusion Clients in hostgroup A will mount `ifs2` and implement Tuning 1. The Fusion Clients in hostgroups C and D will respectively mount `ifs3` and `ifs4` and implement Tuning 1.

![Figure 3-1 Global and local settings in a hostgroup tree](image)

### 3.8.2 Creating a Hostgroup Tree

To set up one level of hostgroups beneath the root, simply create the new hostgroups. You do not need to declare that the root node is the parent. To set up lower levels of hostgroups, declare a parent element for hostgroups.

Optionally you can specify a domain rule for a hostgroup (see above).

Use only alphanumeric characters and the underscore character (_) in hostgroup names.

Do not use a hostname as a groupname.

To create a hostgroup tree with the CLI:

1. Create the first level of the tree, and optionally declare a domain rule for it:
   ```plaintext
   <ibrixhome>/bin/ibrix_hostgroup -c -g GROUPNAME [-D DOMAIN]
   ```

2. Create all other levels by specifying a parent for the group and optionally a domain rule:
   ```plaintext
   <ibrixhome>/bin/ibrix_hostgroup -c -g GROUPNAME [-D DOMAIN] -p PARENT
   ```
3.8.3 Adding a Fusion Client to a Hostgroup

You can add a Fusion Client to a hostgroup or move one to a different hostgroup. All Fusion Clients belong to the default clients hostgroup.

Add a host to a hostgroup

To add or move a host to a hostgroup using the CLI:

```
<ibrixhome>/bin/ibrix_hostgroup -m -g GROUP -h MEMBER
```

For example, to add the named host to the finance group:

```
<ibrixhome>/bin/ibrix_hostgroup -m -g finance -h cl01.ibrix.com
```

3.8.4 Adding a Domain Rule to a Hostgroup

A domain rule is a subnet IP address that corresponds to a client network. Adding a domain rule to a hostgroup restricts its members to Fusion Clients that are on the specified subnet. You can add a domain rule at any time.

Add a domain rule to a hostgroup

To add a domain rule to a hostgroup using the CLI:

```
<ibrixhome>/bin/ibrix_hostgroup -a -g GROUPNAME -D DOMAIN
```

For example, to add the domain rule 192.168 to the finance group:

```
<ibrixhome>/bin/ibrix_hostgroup -a -g finance -D 192.168
```

3.8.5 Performing Operations on Hostgroups

Instructions for performing the following operations on hostgroups are provided at the locations noted:

- Creating a mountpoint (see page 12).
- Deleting a mountpoint (see page 12).
- Mounting an IBRIX file system (see page 13).
- Unmounting an IBRIX file system (see page 15).
- Changing host tuning parameters (see page 47).
- Preferring a network interface (see page 68).
- Setting allocation policy (see Chapter 8).

3.8.6 Deleting Hostgroups

When you delete a hostgroup, its members are assigned to the parent of the deleted group.

To force the moved Fusion Clients to implement the mounts, tunings, network interface preferences, and allocation policies that have been set on their new hostgroup, either restarting IBRIX services on the Fusion Clients (see page 96) or locally execute the following commands:

- `ibrix_lwmount -a` to force the Fusion Client to pick up mounts or allocation policies.
- `ibrix_lwhost --a` to force the Fusion Client to pick up host tunings.

To delete a hostgroup using the CLI:

```
<ibrixhome>/bin/ibrix_hostgroup -d -g GROUPNAME
```
3.8.7 Viewing Hostgroups

You can view a list of all hostgroups or list one hostgroup.

To view hostgroups using the CLI:

<ibrixhome>/bin/ibrix_hostgroup -l [ -g GROUP]

3.9 Enabling the Export Control Feature

When the Export Control feature is enabled on an IBRIX file system, Fusion Clients have no access to the file system by default. Fusion Clients have access to an export-controlled file system only after a systems administrator grants them access by executing the `ibrix_mount` command on the Fusion Manager. For information on mounting file systems when Export Control is enabled, see page 14.

Enabling Export Control does not affect Segment Server access to an IBRIX file system (and thereby, NFS/CIFS client access). Segment Servers always have RW access.

To determine whether Export Control is enabled, run `ibrix_fs -i` or `ibrix_fs -l`. The output contains a field stating whether Export Control is enabled. To view sample output for these commands, see page 18.

- Enable Export Control

<ibrixhome>/bin/ibrix_fs -C -E -f FSNAME

- Disable Export Control

<ibrixhome>/bin/ibrix_fs -C -D -f FSNAME

3.10 Rebalancing Segments in an IBRIX File System

Segment rebalancing involves redistributing files among segments in an IBRIX file system to balance segment utilization and server workload. Usually you will want to rebalance all segments, possibly as a `cron` job. In special situations, you may want to rebalance specific segments. The segment rebalancing functionality is licensed separately and must be activated on the Fusion Manager prior to use.

You can only rebalance files-only segments and mixed segments (that is, segments holding both files and directories). Directory-only segments and segments marked as bad (that is, segments that cannot be activated for some reason) are not candidates for rebalancing. For more information on data-segregated segments, see Chapter 2.

An IBRIX file system must be mounted prior to rebalancing its segments.

3.10.1 How Rebalancing Works

During a rebalancing run on an IBRIX file system, files are moved from source segments to destination segments. IBRIX Fusion calculates the average aggregate utilization of all the source segments, then moves files from sources to destinations to bring each candidate source segment as close as possible to the calculated utilization threshold. The final absolute percent usage in the segments depends on the average file size for the target file system.

If you do not specify any sources or destinations for a run, candidate segments are sorted into sources and destinations and then rebalanced as evenly as possible.
If you specify sources, all other candidate segments in the file system are tagged as destinations, and vice versa if you specify destinations. Following the general rule, IBRIX Fusion will calculate the utilization threshold from the sources and then bring the sources as close as possible to this value by evenly distributing their excess files among all the destinations. If you specified sources, the end result is that only those segments are rebalanced, and the overflow is distributed among all remaining candidate segments. If you specified destinations, the end result is that all segments except the specified destinations are rebalanced, and the overflow is distributed only to the destinations.

If you specify both sources and destinations, only the specified sources are rebalanced, and the overflow is distributed only among the specified destinations.

If there is not enough aggregate room in destination segments to hold the files that must be moved from source segments in order to balance the sources, IBRIX Fusion issues an error message and does not move any files. The more restricted the number of destinations, the higher the likelihood of this error.

### 3.10.2 Rebalancing All Segments

The most common reason to rebalance all segments is to evenly redistribute files among all segments after adding new segments to an IBRIX file system.

**Rebalance all segments**

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f FSNAME
```

### 3.10.3 Rebalancing Specific Segments

To move files out of certain over-used segments, specify source segments.
To move files into certain under-used segments, specify destination segments.
To move files out of certain segments and place them in certain destinations, specify both source and destination segments.

**Rebalance by specifying source segments**

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f FSNAME
   [-s SRCSEGMENTLIST] [-S SRCLVLIST]
```

For example, to rebalance segments 2 and 3 only, and to specify them by segment name:

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f ifs1 -s 2,3
```

To rebalance segments 1 and 2 only, and to specify them by their logical volume name:

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f ifs1 -S ilv1,ilv2
```

**Rebalance by specifying destination segments**

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f FSNAME
   [-d DESTSEGMENTLIST] [-D DESTLVLIST]
```

For example, to rebalance segments 3 and 4 only, and to specify them by segment name:

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f ifs1 -d 3,4
```

To rebalance segments 3 and 4 only, and to specify them by their logical volume name:

```bash
<ibrixhome>/bin/ibrix_rebalance -r -f ifs1 -D ilv3,ilv4
```
3.10.4 Tracking the Progress of a Rebalance Job

To track the progress of a rebalance job, periodically use the CLI or the Fusion Manager GUI to view status information for the IBRIX file system and check the percent usage for each segment. As a run progresses, usage approaches an average value across segments, except on those segments that are not candidates for rebalancing (directory-only segments, bad segments), or those segments containing files that are in heavy use during the run.

- **Track rebalance job progress**

  `<ibrixhome>/bin/ibrix_fs -i`

  The output lists detailed information about the IBRIX file system. The USED% field shows usage per segments.

- **Track rebalance job progress using the Fusion Manager GUI**

  Click the Configuration tab, then the File System tab. Select a file system, click on its Segments, and view the Used (%) field.

3.10.5 Viewing the Status of Rebalance Jobs

You can determine whether a rebalance job is still running or has completed.

- **View the status of rebalance jobs**

  Execute one of the following commands, to either view status for jobs on all file systems, or jobs on the file systems in `FSLIST`:

  `<ibrixhome>/bin/ibrix_fs -i`

  `<ibrixhome>/bin/ibrix_rebalance -i [-f FSLIST]`

  The `-i` report lists jobs by job id, file system, and indicates if the job is running or stopped. Jobs that are in the analysis (“Coordinator”) phase are listed separately from those in the implementation (“Worker”) phase.

3.10.6 Stopping Rebalance Jobs

You can stop running or stalled rebalance jobs. If the Fusion Manager cannot stop the job for some reason, you can force the job to stop.

Stopping a job poses no risks for the IBRIX file system. The Fusion Manager completes any file migrations that are in process when you issue the stop command. Depending on when you stop a job, segments may contain more or fewer files than before the run began.

- **Stop rebalance jobs**

  `<ibrixhome>/bin/ibrix_rebalance -k -j JOBLIST [-F]`

  To force the listed jobs to stop, include the `-F` argument.

3.10.7 Cleaning Up After a Failed Rebalance Run

If a rebalance run does not successfully complete, you will need to clean up the temporary files that were created during the failed run before beginning a new rebalance run. The command that performs this clean-up runs in two phases. The first time that you issue the command, specify phase 1. When the command finishes phase 1, it automatically begins phase 2. If for some reason phase 2 does not complete, the command will stop running and a job.failed event
Clean up after a failed rebalance run

1. Execute the following command:
   \[<ibrixhome>/bin/ibrix_rebalance -C -f FSNAME -p 1\]
2. If the command does not complete successfully, execute the following command:
   \[<ibrixhome>/bin/ibrix_rebalance -C -f FSNAME -p 2\]

3.11 Setting Up Quotas

You can set quotas on the amount of storage or the number of files that a user or a group may use in an IBRIX file system.

It is best to set up quotas when you create an IBRIX file system, but you can set them up at any time.

3.11.1 How Quotas Work

A quota is delimited by hard and soft storage limits for both megabytes of storage and number of files allotted to a user or a group. The hard limit specifies the maximum allotted storage, in terms of file size and number of files. The soft limit specifies the number of megabytes or files that, when reached, causes the Segment Server to start a countdown timer.

The timer runs until either the hard storage limit is reached or seven days elapse. When the timer stops, the user or group for whom the quota was set cannot store any more data, and the system issues `Disk quota exceeded` messages at each write attempt by the user or group.

**Note!** User quota statistics are updated on a regular basis (at 1-minute intervals). At each update, the file and storage usage for each quota-enabled user or group is queried, and the result is distributed to all Segment Servers. Users or groups can temporarily exceed their quota if the allocation policy in effect for an IBRIX file system causes their data to be written to different Segment Servers during the statistics update interval. In this situation it is possible for the storage usage visible to each Segment Server to be beneath or at the quota limit while the aggregate storage use exceeds the limit.

There is a delay of several minutes between the time a command to update quotas is executed and when the results are displayed by the `ibrix_edquota -l` command. This is normal behavior.

3.11.2 Enabling Quotas on an IBRIX File System

Quotas must be enabled before you can set quota limits for users or groups. The IBRIX file system must be unmounted when you enable quotas.

**Enable quotas**

\[<ibrixhome>/bin/ibrix_fs -q -E -f FSNAME\]

3.11.3 Setting User and Group Quotas

Before you can set up quotas, quotas must be enabled on the IBRIX file system and the file system must be mounted (see above).

For the purposes of setting quotas, no UID or GID can exceed 2000000000.
Setting Up Quotas

- **Set a quota for multiple users**
  
  `<ibrixhome>/bin/ibrix_edquota -s -u -S startID -E endID -f FSNAME -M SOFT_MEGABYTES -m HARD_MEGABYTES -i SOFT_FILES -i HARD_FILES`

- **Set a quota for multiple groups**
  
  `<ibrixhome>/bin/ibrix_edquota -s -g -S startID -E endID -f FSNAME -M SOFT_MEGABYTES -m HARD_MEGABYTES -i SOFT_FILES -i HARD_FILES`

- **Set a quota for a single user**
  
  `<ibrixhome>/bin/ibrix_edquota -s -u USER -f FSNAME -M SOFT_MEGABYTES -m HARD_MEGABYTES -i SOFT_FILES -i HARD_FILES`

- **Set a quota for a single group**
  
  `<ibrixhome>/bin/ibrix_edquota -s -g GROUP -f FSNAME -M SOFT_MEGABYTES -m HARD_MEGABYTES -i SOFT_FILES -i HARD_FILES`

### 3.11.4 Reinitializing Quota Usage Limits

You must reinitialize the storage quota process with `ibrix_edquota -r` in the following circumstances:

- You turned quotas off for a user, the user continued to store data in an IBRIX file system, and now you want to turn quotas back on for this user.
- You are setting up quotas for the first time for a user who has previously stored data in an IBRIX file system.

In both cases, reinitializing the process updates the configuration database with information about storage that should be credited to the user’s quota account.

### 3.11.5 Deleting User and Group Quotas

You can delete user and group quotas at any time.

- **Delete quotas for a user**
  
  `<ibrixhome>/bin/ibrix_edquota -D -u UID [-f FSNAME]`

- **Delete quotas for users within a specified range of UIDs**
  
  `<ibrixhome>/bin/ibrix_edquota -D -u -S startID -E endID [-f FSNAME]`

- **Delete quotas for a group**
  
  `<ibrixhome>/bin/ibrix_edquota -D -g GID [-f FSNAME]`

- **Delete quotas for groups within a specified range of UIDs**
  
  `<ibrixhome>/bin/ibrix_edquota -D -g -S startID -E endID [-f FSNAME]`
3.11.6 Viewing Quota Settings

You can list the quotas that have been set for users or groups on an IBRIX file system. For more information about viewing quota settings, refer to the `ibrix_edquota` command in the CLI Reference.

- **List quota parameters for a user**
  
  `<ibrixhome>/bin/ibrix_edquota -l -u UID`

- **List quota parameters for a group**
  
  `<ibrixhome>/bin/ibrix_edquota -l -g GID`

- **List quota parameters for all users**
  
  `<ibrixhome>/bin/ibrix_edquota -l -U`

- **List quota parameters for all groups**
  
  `<ibrixhome>/bin/ibrix_edquota -l -G`

3.12 Listing Cluster Events

IBRIX events are categorized by type, based on level of severity:

- **Alerts**: Disruptive events that can result in loss of access to file system data, for example when a segment is unavailable or a server is unreachable.
- **Warnings**: Potentially disruptive conditions in which file system access is not lost, but if the situation is not addressed it can escalate to an alert condition, for example reaching a very high server CPU utilization or nearing a quota limit.
- **Information**: Events occurring under normal or non-threatening conditions that change the cluster, for example a segment is created or a file system mounted.

Events are written to the configuration database as they are generated. This table can be pruned to keep its size manageable, and we recommend you do this regularly.

The list of all defined, or a list of all events by type of event, is available as an option to the `ibrix_event` command.

- **Display the list of defined events by type**
  
  `<ibrixhome>/bin/ibrix_event -q [-e ALERT|WARN|INFO]`

- **List events**

  Generated events are listed on a last-in, first-out basis.

  `<ibrixhome>/bin/ibrix_event -l`

- **List a designated number of events**

  The `ibrix_event` command displays the most recent 100 messages by default. This can be increased or decreased with the `-n EVENTS_COUNT` option.

  `<ibrixhome>/bin/ibrix_event -l [-n EVENTS_COUNT]`

  For example, to display the most recent 25 events:

  `<ibrixhome>/bin/ibrix_event -l -n 25`
Remove events from the events database table

Events can be removed from the events table, oldest events first, with the \(-p\) argument. The default is to remove the oldest seven days of messages. To change the number of days, include the \(-o DAYS_COUNT\) argument.

\(<ibrixhome>/bin/ibrix_event -p \[-o DAYS_COUNT]\)
For example, to configure email settings to use the mail.ibrix.com SMTP server and to turn notifications on:

```
<ibrixhome>/bin/ibrix_event -m on -s mail.ibrix.com -f FM@ibrix.com -r MIS@ibrix.com -t Cluster1 Notification
```

3.13.3 Dissociating Events and Email Addresses

- **Dissociate events and email addresses**

```
<ibrixhome>/bin/ibrix_event -d [-e ALERT|WARN|INFO|EVENTLIST] -m EMAILLIST
```

For example, to dissociate event notifications for admin@ibrix.com:

```
<ibrixhome>/bin/ibrix_event -d -m admin@ibrix.com
```

To turn off all Alert notifications for admin@ibrix.com:

```
<ibrixhome>/bin/ibrix_event -d -e ALERT -m admin@ibrix.com
```

To turn off the server.registered and filesystem.created notifications for admin1@ibrix.com and admin2@ibrix.com:

```
<ibrixhome>/bin/ibrix_event -d -e server.registered,filesystem.created -m admin1@ibrix.com,admin2@ibrix.com
```

3.13.4 Turning Email Notifications On and Off

After configuration is complete, turn email notifications on.

- **Turn email notifications on and off**

```
<ibrixhome>/bin/ibrix_event -m on|off -s SMTP -f from
```

3.13.5 Testing Email Addresses

To test an email address with a test message, notifications must be turned on. If the address is valid, the command signals success and sends an email containing the settings to the recipient. If the address is not valid, the command returns an address failed exception.

- **Test an email address**

```
<ibrixhome>/bin/ibrix_event -u -n EMAILADDRESS
```

3.13.6 Viewing Email Notification Settings

The `ibrix_event` command provides comprehensive information about both email settings and configured notifications.

- **List email settings**

```
<ibrixhome>/bin/ibrix_event -L
```

**Sample Output**

```
State : on
SMTP : mail.ibrix.com
From : sysop@ibrix.com
ReplyTo : admin@ibrix.com
Subject : lab13-20 notification:
```


### 3.14 Setting Up SNMP Notifications

Previous versions of IBRIX Fusion included support for SNMP (Simple Network Management Protocol) versions 1 and 2 for purpose of event (trap) notification and device monitoring. SNMP configuration was accomplished through the use of a single CLI command or its equivalent in the FusionManager GUI.

**Note!** IBRIX Fusion now supports SNMPv3. All aspects of v1/v2 are still supported. Users of IBRIX Fusion versions older than 4.1 should be aware that the single `ibrix_snmp` command of previous versions has been replaced by two commands, `ibrix_snmpagent` and `ibrix_snmptrap`. If you wrote scripts that include `ibrix_snmp`, remember to edit the scripts as necessary to include the correct commands.

Whereas SNMPv2 security was enforced by use of community password strings, v3 introduces the USM (User Security Model) and the VACM (View Access Control Model). Discussion of these models is beyond the scope of this document. Readers are referred to RFCs 3414 and 3415 available at [http://www.ietf.org](http://www.ietf.org).

- In the SNMPv3 environment, every message contains a user name, and the function of the USM is to authenticate users and ensure message privacy through message encryption and decryption. Both authentication and privacy, and their passwords, are optional and will use default settings where security is less of a concern.
- With users validated, the VACM determines what managed objects these users are allowed to access. The VACM includes an access scheme to control user access to managed objects; context matching to define which objects can be accessed; and MIB views, defined by subsets of IOD subtree and associated bit mask entries, which define what in the MIB is accessible to a given user.

Steps to setting up SNMP include:

- Agent configuration (all SNMP versions)
- Trapsink configuration (all SNMP versions)
- Associating event notifications with trapsinks (all SNMP versions)
- View definition (v3 only)
- Group and user configuration (v3 only)

IBRIX Fusion implements an SNMP agent on the Fusion Manager that supports the IBRIX private MIB. The agent can be polled and can send SNMP traps to configured trapsinks. Setting up SNMP notifications is similar to setting up email notifications. You must associate events to trapsinks and configure SNMP settings for each trapsink, in order for the agent to send a trap when an event occurs.

#### 3.14.1 Configuring the SNMP Agent

The SNMP agent is created automatically as part of FusionManager installation, and is configured initially as an SNMPv2 agent. It is configured to be off by default when the FusionManager is installed.

Some SNMP parameters and the SNMP default port are the same regardless of SNMP version. The agent port is 161 by default. `SYSCONTACT`, `SYSNAME` and `SYSLOCATION` are optional MIB-II agent parameters that have no default values.
The -c and -s options are also common to all SNMP versions. The -c option turns the encryption of community names and passwords on or off. There is no encryption by default. Using the -s option toggles the agent on and off; it turns the agent on by starting a listener on the SNMP port, and turns it off by shutting off the listener. The default is off.

The format for a v1 or v2 update command is:

```bash
ibrix_snmpagent -u -v {1|2} [-p PORT] [-r READCOMMUNITY] [-w WRITECOMMUNITY] [-t SYSCONTACT] [-n SYSNAME] [-o SYSLOCATION] [-c {yes|no}] [-s {on|off}]
```

The update command for SNMPv1 and v2 takes optional community names. By convention, the default READCOMMUNITY name used for read-only access and assigned to the agent is `public`. No default WRITECOMMUNITY name is set for read-write access (although the name `private` is often used).

For example, to update a v2 agent with the write community name `private`, the agent’s system name, and that system’s physical location:

```bash
ibrix_snmpagent -u -v 2 -w private -n agenthost.domain.com -o DevLab-B3-U6
```

The SNMPv3 format adds an optional engine id, which overrides the default value of the agent’s host name, and provides the -y and -z options to set whether a v3 agent can process v1/v2 read and write requests from the management system. The format is:

```bash
ibrix_snmpagent -u -v 3 [-e engineId] [-p PORT] [-r READCOMMUNITY] [-w WRITECOMMUNITY] [-t SYSCONTACT] [-n SYSNAME] [-o SYSLOCATION] [-y {yes|no}] [-z {yes|no}] [-c {yes|no}] [-s {on|off}]
```

For detailed descriptions of the agent commands, refer to the description of the `ibrix_snmpagent` command in the `IBRIX Fusion CLI Reference Guide`.

### 3.14.2 Configuring Trapsink Settings

A **trapsink** is the host destination where agents send **traps**, asynchronous notifications sent by the agent to the management station. A trapsink is specified either by name or IP address. IBRIX supports multiple trapsinks; you can define any number of trapsinks of any SNMP version, but you can define only one trapsink per host regardless of version.

At a minimum, trapsink configuration requires a destination host and SNMP version. All other parameters are optional and many are passed a default value if not specified. Trapsink configuration for SNMPv3 is more detailed than for earlier versions. The main version differences involve the additional security parameters added by SNMPv3.

The format for creating a v1/v2 trapsink is:

```bash
ibrix_snmptrap -c -h HOSTNAME -v {1|2} [-p PORT] [-m COMMUNITY] [-s {on|off}]
```

If a port is not specified, the command defaults to port 162. If a community is not specified, the command defaults to the community name `public`. The -s option toggles agent trap transmission on and off. The default is on. For example, to create a v2 trapsink with a new community name, enter:

```bash
ibrix_snmptrap -c -h lab13-116 -v 2 -m private
```

Additional options for a v3 trapsink define security settings. **USERNAME** is a v3 user defined on the trapsink host and is required. The security level associated with the trap message depends on which passwords are specified—the authentication password, both authentication and privacy passwords, or no passwords. The **CONTEXT_NAME** is required if the trap receiver has defined subsets of managed objects. The format is:

```bash
ibrix_snmptrap -c -h HOSTNAME -v 3 [-p PORT]-n USERNAME [-j {MD5|SHA}] [-k AUTHORIZATION_PASSWORD] [-y {DES|AES}]
```
[-z PRIVACY_PASSWORD] [-x CONTEXT_NAME] [-s {on|off}]

For example, to create a v3 trapsink with a named user and passwords applied to the default algorithms. If specified, passwords must contain at least eight characters.

ibrix_snmptrap -c -h lab13-114 -v 3 -n trapsender -k auth-passwd
-z priv-passwd

For detailed descriptions of the trapsink commands, refer to the description of the ibrix_snmptrap command in the IBRIX Fusion CLI Reference Guide.

### 3.14.3 Associating Events and Trapsinks

Associating events with trapsinks is similar to associating events with email recipients except that you specify the host name or IP address of the trapsink instead of an email address.

Use the ibrix_event command to associate SNMP events with trapsinks. The format is:

```
<ibrixhome>/bin/ibrix_event -c -y SNMP [-e ALERT|INFO|EVENTLIST]
[-t THRESHOLD] -m TRAPSINK
```

Refer to the ibrix_event command description in the IBRIX Fusion CLI Reference Guide for information on specifying events and a threshold.

For example, to associate all Alert events and two specific Info events with a trapsink at IP 192.168.2.32, enter:

```
<ibrixhome>/bin/ibrix_event -c -y SNMP
-e ALERT,server.registered,filesystem.created -m 192.168.2.32
```

Use ibrix_event -d to dissociate events and trapsinks. The format is:

```
<ibrixhome>/bin/ibrix_event -d -y SNMP [-e ALERT|INFO|EVENTLIST]
-m TRAPSINK
```

### 3.14.4 Defining Views

A MIB view is a collection of paired OID subtrees and associated bitmasks that identify which sub-identifiers are significant to the view’s definition. Through use of the bitmasks, individual OID subtrees can be included in or excluded from the view.

An instance of a managed object belongs to a view if:

- The OID of the instance has at least as many sub-identifiers as the OID subtree in the view
- Each sub-identifier in the instance and the subtree match when the bit mask of the corresponding sub-identifier is non-zero.

The FusionManager automatically creates the excludeAll view that blocks access to all OIDs. This view cannot be deleted. It is the default read and write view if one is not specified for a group with the ibrix_snmpgroup command. Its catch-all OID and mask are:

```
OID  = .1
Mask = .1
```

Consider these examples:

```
OID  = .1.3.6.1.4.1.18997
Mask = .1.1.1.1.1.1.1

OID  = .1.3.6.1.2.1
Mask = .1.1.0.1.0.1
```

Where instance .1.3.6.1.2.1.1 matches, instance .1.3.6.1.4.1 matches, and instance .1.2.6.1.2.1 does not match.
Chapter 3 Managing the Cluster

The format to add a pairing of an OID subtree value and a mask value to a new or existing view is:

```
# ibrix_snmpview -a -v VIEWNAME [-t {include|exclude}]
-o OID_SUBTREE [-m MASK_BITS]
```

The subtree is added in the named view. For example, to add the IBRIX private MIB to the view named `ibrix`, enter:

```
ibrix_snmpview -a -v ibrix -o .1.3.6.1.4.1.18997 -m .1.1.1.1.1.1
```

3.14.5 Configuring Groups and Users

A group defines the access control policy on managed objects for one or more users. All users must belong to a group. Groups and users exist only in SNMPv3. Groups are assigned a security level, which enforces use of authentication and privacy, and specify read and write views to identify which managed objects group members can read and write.

The command to create a group assigns its SNMPv3 security level, read and write views, and context name. A context is a collection of managed objects that can be accessed by an SNMP entity. A related option, `-m`, determines how the context is matched. The format is:

```
ibrix_snmpgroup -c -g GROUPNAME
 [-s {noAuthNoPriv|authNoPriv|authPriv}] [-r READVIEW]
 [-w WRITEVIEW] [-x CONTEXT_NAME] [-m {exact|prefix}]
```

For example, to create the group `group2` to require authorization but no encryption, and read access to the `ibrix` view, enter:

```
ibrix_snmpgroup -c -g group2 -s authNoPriv -r ibrix
```

The format to create a user and add that user to a group is:

```
ibrix_snmpuser -c -n USERNAME -g GROUPNAME [-j {MD5|SHA}]
 [-k AUTHORIZATION_PASSWORD] [-y {DES|AES}] [-z PRIVACY_PASSWORD]
```

Authentication and privacy settings are optional. An authentication password is required if the group has a security level of either authNoPriv or authPriv. The privacy password is required if the group has a security level of authPriv. If unspecified, MD5 is used as the authentication algorithm and DES as the privacy algorithm, with no passwords assigned.

For example, to create user3 and add it to group2, specifying an authorization password for authorization and no encryption, enter:

```
ibrix_snmpuser -c -n user3 -g group2 -k auth-passwd -s authNoPriv
```

Refer to the description of the `ibrix_snmpgroup` and `ibrix_snmpuser` commands in the IBRIX Fusion CLI Reference Guide for further information.

3.14.6 Deleting Elements of the SNMP Configuration

All the SNMP commands employ the same syntax for delete operations, using the `-d` flag to indicate the object following is to be deleted. For example to delete a list of hosts as trapsinks, enter:

```
ibrix_snmptrap -d
 -h lab15-12.domain.com,lab15-13.domain.com,lab15-14.domain.com
```

There are two restrictions on SNMP object deletions:

- A view cannot be deleted if it is referenced by a group.
- A group cannot be deleted if it is referenced by a user.

Refer to the IBRIX Fusion CLI Reference Guide for the specific syntax used with each delete command.
3.14.7 Listing SNMP Configuration Information

All the SNMP commands employ the same syntax for list operations, using the -l flag. For example:

```bash
# ibrix_snmpgroup -l
```

This command lists the defined groups settings for all SNMP groups. Specifying an optional group name lists the defined settings for that single group.

Refer to the IBRIX Fusion CLI Reference Guide for the specific syntax used with each list command.

3.15 Deleting IBRIX File Systems and File System Components

3.15.1 Deleting an IBRIX File System

Before deleting an IBRIX file system, unmount it from all Fusion Clients and Segment Servers (see page 15).

**Caution!** When an IBRIX file system is deleted from the configuration database, its data becomes inaccessible. Make sure that you have specified the correct file system, to avoid unintended service interruptions.

- **Delete an IBRIX file system**

  ```bash
  <ibrixhome>/bin/ibrix_fs -d -f FSLIST
  ```

  For example, to delete IBRIX file systems ifs1 and ifs2:

  ```bash
  <ibrixhome>/bin/ibrix_fs -d -f ifs1,ifs2
  ```

3.15.2 Deleting Segments, Volume Groups, and Physical Volumes

Three rules about deletions to remember:

- A segment cannot be deleted until the IBRIX file system to which it belongs is deleted.
- A volume group cannot be deleted until all segments that were created on it are deleted.
- A physical volume cannot be deleted until all volume groups created on it are deleted.

If you delete physical volumes but do not remove the physical storage from the network, the volumes may be re-discovered when you next perform a discovery scan on the cluster.

- **Delete segments**

  ```bash
  <ibrixhome>/bin/ibrix_lv -d -s LVLIST
  ```

  For example, to delete segments ilv1 and ilv2:

  ```bash
  <ibrixhome>/bin/ibrix_lv -d -s ilv1,ilv2
  ```

- **Delete volume groups**

  ```bash
  <ibrixhome>/bin/ibrix_vg -d -g VGLIST
  ```

  For example, to delete volume groups ivg1 and ivg2:

  ```bash
  <ibrixhome>/bin/ibrix_vg -d -g ivg1,ivg2
  ```

- **Delete physical volumes**

  ```bash
  <ibrixhome>/bin/ibrix_pv -d -p PVLIST [-h HOSTLIST]
  ```
For example, to delete physical volumes d1, d2, and d3:

```
<ibrixhome>/bin/ibrix_pv -d -p d[1-3]
```

### 3.15.3 Deleting Segment Servers and Fusion Clients

Before deleting a Segment Server, unmount all IBRIX file systems from it, migrate any segments that it owns to a different server, and ensure that it is neither serving as a failover standby, nor involved in network interface monitoring.

- **Delete Segment Servers or Fusion Clients**

  `<ibrixhome>/bin/ibrix_host -d -h HOSTLIST`

  For example, to delete Segment Servers s1.ibrix.com and s2.ibrix.com:

  `<ibrixhome>/bin/ibrix_host -d -h s1.ibrix.com,s2.ibrix.com`

### 3.16 Tuning Segment Servers and Fusion Clients

The default host tuning settings are adequate for most cluster environments. However, your IBRIX Customer Support representative may recommend that you change certain Segment Server or Fusion Client tuning settings, to improve performance.

Host tuning changes are immediately executed for Segment Servers. For Fusion Clients a tuning intention is stored in the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager for the host tunings that it should use, and then it implements them. If IBRIX services are already running on a Fusion Client when you set new host tunings, you can force the client to query the Fusion Manager for the tuning intention by executing `ibrix_client` or `ibrix_lwhost --a` on the Fusion Client, or by rebooting the Fusion Client.

You can locally override host tunings that have been set on Fusion Clients via `ibrix_lwhost`.

All of the Fusion Manager commands for tuning hosts include a `-h HOSTLIST` argument that can take one or more hostgroups. Setting host tunings on a hostgroup is a convenient way to tune a set of Fusion Clients all at once. To set the same host tunings on all Fusion Clients, specify the `clients` hostgroup.

**Caution!** Changing host tuning settings will alter IBRIX file system performance. Contact IBRIX Customer Support before changing any host tuning settings.

- **List default values and valid ranges for all permitted host tunings**

  `<ibrixhome>/bin/ibrix_host_tune -L`

- **Tune host parameters on nodes or hostgroups**

  `<ibrixhome>/bin/ibrix_host_tune -S {-h HOSTLIST|-g GROUPLIST} -o OPTIONLIST`

  An IBRIX Customer Support representative will tell you the values to enter for `OPTIONLIST`. List options as a comma-separated list of `option=value` pairs.

  To set host tunings on all Fusion Clients, use `-g clients`.

- **Reset host parameters to their default values on nodes or hostgroups**

  `<ibrixhome>/bin/ibrix_host_tune -U {-h HOSTLIST|-g GROUPLIST} [-n OPTIONS]`
To reset all options on all Segment Servers, hostgroups, and Fusion Clients, omit the `-h HOSTLIST` and `-n OPTIONS` arguments. To reset host tunings on all Fusion Clients, use `-g clients`.

The values that are restored depends on what is supplied to the `HOSTLIST` argument:

- **Segment Servers**: The default Segment Server host tunings are restored.
- **Fusion Clients**: The host tunings that are in effect for the default `clients` hostgroup are restored.
- **Hostgroups**: The host tunings that are in effect for the parent of the specified hostgroups are restored.

### List host tuning settings on Segment Servers, Fusion Clients, and hostgroups

```
<ibrixhome>/bin/ibrix_host_tune -l [-h HOSTLIST] [-n OPTIONS]
```

To list all tunings on all Segment Servers, hostgroups, and Fusion Clients, omit the `-h HOSTLIST` and `-n OPTIONS` arguments.

### Set the communications protocol on nodes and hostgroups

```
<ibrixhome>/bin/ibrix_host_tune -p {UDP|TCP} {-h HOSTLIST} {-g GROUPLIST}
```

To set a protocol on all Fusion Clients, use `-g clients`.

### Set server threads on Segment Servers, hostgroups, and Fusion Clients:

```
<ibrixhome>/bin/ibrix_host_tune -t THREADCOUNT {-h HOSTLIST} {-g GROUPLIST}
```

To set server threads on all Fusion Clients, use `-g clients`.

### Set admin threads on Segment Servers, hostgroups, and Fusion Clients:

```
<ibrixhome>/bin/ibrix_host_tune -a THREADCOUNT {-h HOSTLIST} {-g GROUPLIST}
```

To set admin threads for all Fusion Clients, use `-g clients`. 
3.17 Tuning Fusion Clients Locally

**Linux.** To tune host parameters on a Linux Fusion Client, use `ibrix_lwhost`. For example, to set the communications protocol, enter:

```
<ibrixhome>/bin/ibrix_lwhost --protocol -p {tcp|udp}
```

To list host tuning parameters that have been changed from their defaults, enter:

```
<ibrixhome>/bin/ibrix_lwhost --list
```

Refer to the `ibrix_lwhost` command description in the *IBRIX Fusion CLI Reference Guide* for other available options.

**Windows.** Windows Fusion Clients are tuned on the client’s Tune Host tab. Tunable parameters include the NIC to prefer (The client uses the Cluster interface by default unless a different network interface is preferred for it.), the communications protocol (UDP or TCP), and the number of server threads to use. Refer to the client’s online help if necessary.
Chapter 4  Implementing IBRIX Fusion High Availability

4.1 About IBRIX Fusion High Availability

IBRIX Fusion HA (high availability) keeps your data accessible at all times. Failover protection can be configured for Segment Servers, network interfaces, individual segments, and HBAs (see Figure 4-1). With IBRIX Fusion Manager High Availability, failover protection can be extended to the Fusion Manager. Through physical and logical configuration policies, you can set up a flexible and scalable high availability solution. Clients experience no changes in service and are unaware of the failover events.

IBRIX Fusion High Availability, including IBRIX Fusion Manager High Availability, is licensed separately and must be activated on the Fusion Manager prior to use.
4.2 About the Two IBRIX Failover Modes

IBRIX Fusion High Availability has two failover modes, the default manual failover and the optional automated failover.

A manual failover uses the `ibrix_server` command or the Fusion Manager GUI to fail over a Segment Server to its standby. The server can be powered down or remain up during the procedure. Manual failover also includes failover of any network interfaces for NFS and SAMBA with defined standbys. You can perform a manual failover at any time, regardless of whether automated failover is in effect.

Automated failover allows the Fusion Manager to initiate failover when it detects that standby-protected components have failed. A basic automated failover setup protects all Segment Servers. A comprehensive setup also includes network interface monitoring to protect User interfaces, and HBA monitoring to protect Segment Server access to storage via a Host Bus Adapter (HBA).

When automated failover is on, the Fusion Manager listens for heartbeat messages that the Segment Servers broadcast at one-minute intervals. The Fusion Manager automatically initiates failover when it fails to receive five consecutive heartbeats or, if HBA monitoring is enabled, when a heartbeat message indicates that a monitored HBA or pair of HBAs has failed.

If network interface monitoring is enabled, automated failover occurs when the Fusion Manager receives a heartbeat message indicating that a monitored network may be down; and when the Fusion Manager cannot subsequently reach that interface.

Following any Segment Server failover, you will need to manually fail back the server.

4.3 What Happens During a Failover

The following events occur during both automated and manual Segment Server failover of a primary to its standby:

1. The Fusion Manager verifies that the standby is powered on and accessible.
2. The Fusion Manager migrates ownership of the primary’s segments to the standby and notifies all Segment Servers and Fusion Clients about the migration. This is a persistent change.
3. If network interface monitoring has been set up, the Fusion Manager activates the standby User interface and transfers the IP address of the primary’s User interface to it.

To determine the progress of a failover, view the Fusion Manager GUI Status tab or execute `ibrix_server -l`. While the Fusion Manager is migrating segment ownership, the operational status of the primary is Up-InFailover or Down-InFailover, depending on whether the primary was powered up or down when failover was initiated. When failover completes, the operational status changes to Up-FailedOver or Down-FailedOver. For more information on monitoring operational status, see page 24. For complete definitions of failover-related operational states, see Table 3-1.

Both automated and manual failovers trigger an event that is reported in the Fusion Manager GUI Status tab.

4.4 Setting Up Automated Failover

This is the suggested minimum setup for automated failover protection.

1. Identify standbys for Segment Servers or for specific segments. You must implement either server-level or segment-level standby protection. You cannot implement both.
2. Identify Segment Server power sources. For APC power sources, associate Segment Servers to power source slots.
3. Turn on automated failover.

If your cluster includes one or more User interfaces carrying NFS/CIFS client traffic, IBRIX recommends that you identify standby network interfaces and set up network interface monitoring.

If your Segment Servers are connected to storage via HBAs, IBRIX recommends that you set up HBA monitoring.

4.5 Identifying Segment Server Standbys

Segment Servers can be configured to provide standby service for one another in the following configurations:

- **1 x 1**: Set up standby pairs, where each server in a pair is the standby for the other.
- **1 x N**: Assign the same standby to some number of primaries.
- **N x M**: Set up a variety of topologies, such as a round-robin topology where B is the standby for A, C is the standby for B, and so forth.

For recommendations based on your specific environment, contact IBRIX Customer Support.

The following constraints apply to all types of standby configurations:

- The Fusion Manager must have access to both the primary segment server and its standby.
- The same IBRIX file system must be mounted on both a primary and its standby.
- A server identified as a standby must be able to see all segments that might fail over to it. In the A-to-B, B-to-C, C-to-A case, all segments must be visible to all three servers for complete standby protection.
- In a SAN environment a primary and its standby must use the same storage infrastructure to access a segment’s physical volumes (for example, a multi-ported RAID array).

#### Identify a standby Segment Server

```
<ibrixhome>/bin/ibrix_server -b -h HOSTNAME1,HOSTNAME2
```

For example, to identify Segment Server s2.ibrix.com as the standby for all segments on Segment Server s1.ibrix.com:

```
<ibrixhome>/bin/ibrix_server -b -h s1.ibrix.com,s2.ibrix.com
```

4.5.1 Identifying Segment Standbys

Rather than fail all segments over to a single Segment Server, it may be preferable for performance reasons to fail specific segments over to specific Segment Servers.

#### Identify a segment standby

```
<ibrixhome>/bin/ibrix_fs -b -f FSNAME -s LVLIST -h HOSTNAME
```

For example, to identify Segment Server s1.ibrix.com as the standby for segments ilv_1, ilv_2, and ilv_3 in IBRIX file system ifs1:

```
<ibrixhome>/bin/ibrix_fs -b -f ifs1 -s ilv_1,ilv_2,ilv_3 -h s1.ibrix.com
```
4.5.2 Setting Up Power Sources

To implement automated failover, to perform a forced manual failover, or to remotely power a
Segment Server up or down, you must set up programmable power sources for Segment
Servers and their standbys. Using programmable power sources prevents a “split-brain
scenario” between a failing Segment Server and its standby, by allowing the failing server to
be centrally powered down by the Fusion Manager in the case of automated failover and
manually in the case of a forced manual failover.

IBRIX Fusion works with APC power sources and the following integrated power sources:
Intelligent Platform Management Interface (IPMI), OpenIPMI, OpenIPMI, and Integrated
Lights Out (ILO).

4.5.2.1 Preliminary Steps

When setting up power sources, be sure to perform these steps:

• All types: If you will be implementing automated failover, make sure that the Fusion
Manager has LAN access to the power sources.

• Integrated power sources: Install the environment and any drivers and utilities as
specified by the vendor documentation. If you will be protecting access to the power
sources, set up the UID and password to be used.

• APC: Enable SNMP access. Set the Community Name to ibrix and the Access Type to
write+. If write+ does not work with your configuration, set the Access Type to write.

4.5.2.2 Identifying Power Sources

All power sources must be identified to the configuration database before they can be used. In
the case of APC power sources, you must associate Segment Servers to power source slots (see
Section 4.5.2.3 below).

Identify an integrated power source

<ibrixhome>/bin/ibrix_powersrc -a -t {ipmi|openipmi|openipmi2|ilo} 
-h HOSTNAME -I IPADDR [-u USERNAME -p PASSWORD]

For example, to identify an OpenIPMI-managed power source at IP address 192.168.3.170 for
Segment Server ss01:

<ibrixhome>/bin/ibrix_powersrc -a -t openipmi -h ss01 -I 192.168.3.170

Identify an APC power source

<ibrixhome>/bin/ibrix_powersrc -a -t {apc|apc_msp} -h POWERSRCNAME 
-n NUMSLOTS -I IPADDR

For example, to identify an eight-port APC power source named ps1 at IP address
192.168.3.150:

<ibrixhome>/bin/ibrix_powersrc -a -t apc -h ps1 -n 8 
-I 192.168.3.150

4.5.2.3 Associating Segment Servers to Power Source Slots

After identifying APC power sources, you must declare Segment Server slot associations.

It is not necessary to declare a slot association for integrated power sources because Segment
Servers are connected by default to slot 1 on these types of power sources.
Chapter 4 Implementing IBRIX Fusion High Availability

4.5.2.4 Updating the Configuration Database with Power Source Changes

If you move a Segment Server to a different power source slot, unplug it from a power source slot, or change its IP address or password, you must update the configuration database with the changes.

- Associate a Segment Server to a power source slot

  `<ibrixhome>/bin(ibrix_hostpower -a -i SLOTID -s POWERSOURCE
  -h HOSTNAME

For example, to identify that Segment Server s1.ibrix.com is connected to slot 1 on APC power source ps1:

  `<ibrixhome>/bin/ibrix_hostpower -a -i 1 -s ps1 -h s1.ibrix.com

- Dissociate a Segment Server from a power source slot

  `<ibrixhome>/bin/ibrix_hostpower -m -i FROM_SLOT_ID,TO_SLOT_ID
  -s POWERSOURCE -h HOSTNAME

For example, to identify that Segment Server s1.ibrix.com has been moved from slot 3 to slot 4 on APC power source ps1:

  `<ibrixhome>/bin/ibrix_hostpower -m -i 3,4 -s ps1 -h s1.ibrix.com

- Change the slot association for a Segment Server

  `<ibrixhome>/bin/ibrix_hostpower -m -i FROM_SLOT_ID,TO_SLOT_ID
  -s POWERSOURCE -h HOSTNAME

For example, to dissociate Segment Server s1.ibrix.com from slot 3 on APC power source ps1:

  `<ibrixhome>/bin/ibrix_hostpower -d -s ps1 -i 3 -h s1.ibrix.com

- modify power source attributes

  `<ibrixhome>/bin/ibrix_powersrc -m [-I IPADDR] [-u USERNAME]
  [-p PASSWORD] [-s] -h POWERSRCLIST

The username and password options are available only for remotely managed power sources. Include the –s argument to have the FusionManager skip Baseboard Management Configuration (BMC).

For example, to change the IP address for powersource ps1:

  `<ibrixhome>/bin/ibrix_powersrc -m -I 192.168.3.153 -h ps1

4.5.2.5 Dissociating a Segment Server from a Power Source

You can dissociate a server from an integrated power source by dissociating it from slot 1 (its default association) on the power source (see page 55).

- Dissociate a Segment Server from a power source

  `<ibrixhome>/bin/ibrix_hostpower -d -s POWERSOURCE -h HOSTNAME

4.5.2.6 Deleting Power Sources from the Configuration Database

Delete power sources that are no longer in use from the configuration database, to conserve storage.
4.5.3 Turning Automated Failover On and Off

Automated failover is turned off by default. Turning automated failover on causes the Fusion Manager to begin monitoring heartbeat messages from Segment Servers.

You can turn automated failover on and off for all Segment Servers or selected ones.

- **Turn on automated failover**
  
  `<ibrixhome>/bin/ibrix_host -m [-h HOSTNAME]
  
  To turn on automated failover for a single Segment Server, include the `-h HOSTNAME` argument.

- **Turn off automated failover**
  
  `<ibrixhome>/bin/ibrix_host -m -U [-h HOSTNAME]
  
  To turn off automated failover for a single Segment Server, include the `-h HOSTNAME` argument.

4.6 Setting Up Manual Failover

To set up a cluster for manual failover, identify server-level or segment-level standbys for Segment Servers as described in Section 4.5 and Section 4.5.1, respectively.

Manual failover does not require the use of programmable power supplies. However, if you have installed and identified power supplies for Segment Servers, you can power down a server prior to manually failing it over.

4.7 Manually Failing Over a Segment Server

You can manually failover a Segment Server even when automated failover is turned on.

- **Manually fail over a Segment Server**

  1. Run `ibrix_server -f` to fail over the Segment Server, the specified `HOSTNAME`. Optionally, include the `-p` argument to power down the Segment Server prior to segment migration:

     `<ibrixhome>/bin/ibrix_server -f [-p] -h HOSTNAME`

  2. Check whether the failover was successful:

     `<ibrixhome>/bin/ibrix_server -l`

The contents of the STATE field indicate the status of the failover. If the field persistently shows Down-InFailover or Up-InFailover, the failover did not complete, and you should contact your IBRIX Customer Support representative for assistance. For information on the values that may appear in the STATE field, see page 52.
4.8 Failing Back a Segment Server

Following both automated and manual failover of a Segment Server, you must manually failback the server, which restores ownership of the failed-over segments and network interfaces to the server.

Before manually failing back a Segment Server, confirm that the primary can see all of its storage resources and networks. The segments owned by the primary server will not be accessible if it cannot see its storage.

After failing back a Segment Server, be sure to check whether the failback fully completes, as described in the following procedure. If it does not, contact your IBRIX Customer Support representative for assistance.

![Note!](image)

A failback might not succeed if the time period between the failover and the failback is too short, and the primary server has not fully recovered. We recommend making sure both servers are up and running, then waiting for 30 seconds before starting the failback.

### Fail back a Segment Server

```bash
<ibrixhome>/bin/ibrix_server -f -U -h HOSTNAME
```

The `HOSTNAME` argument takes the name of the failed-over Segment Server.

4.9 Setting Up Network Interface Monitoring

To set up network interface monitoring, you must (1) identify a standby for the network interface that you want to protect on each Segment Server and then (2) identify a network interface monitor for every Segment Server. A network interface monitor is a Segment Server that is monitoring the health of another Segment Server over a given network interface.

Unlike Fusion Clients, NFS and CIFS clients cannot re-route file requests to a standby if the Segment Server where they are mounted fails. To ensure continuous client access to files, IBRIX recommends that you put NFS/CIFS traffic on a User network interface (see page 68) and then implement network interface monitoring for it.

In network interface monitoring, one Segment Server monitors another Segment Server over a designated network interface. If the monitoring server loses contact with its destination server over the interface, it notifies the Fusion Manager. If the Fusion Manager also cannot contact the destination server over that interface, it fails over both the destination server and the network interface to their standbys. Clients that were mounted on the failed-over server do not experience any service interruption and are unaware that they are now mounting the IBRIX file system on a different server.

Comprehensive protection of NFS/CIFS traffic also involves setting up network interface monitoring for the Cluster interface. Although the Fusion Manager will eventually detect interruption of a Segment Server’s connection to the Cluster interface and (if automated failover is turned on) initiate segment failover, failover will occur much faster if the interruption is detected via network interface monitoring. (If automated failover is not turned on, you will begin to see file access problems if the Cluster interface fails.) There is no difference in the way that monitoring is set up for the Cluster interface and a User interface. In both cases, you set up Segment Servers to monitor each other over the interface.

Figure 4-1 illustrates a monitoring and failover scenario for a cluster in which 1:1 standby relationship is configured, and each standby pair is also a network interface monitoring pair. When SS1 loses its connection to the User interface (eth1), as shown by the dotted line, SS2 can no longer contact SS1 (A). SS2 notifies the Fusion Manager, which then tests its own connection with SS1 over eth1 (B). This is shown by the dashed line. The Fusion Manager
cannot contact SS1 on eth1 either, so it initiates failover of SS1’s segments (C) and User interface (D).

![Network Interface Monitoring and Failover Scenario](image)

Figure 4-1   Network Interface Monitoring and Failover Scenario

### 4.9.1 Identifying and Deleting Standbys for a Network Interface

You must identify a standby for the network interface that you want to protect, on each Segment Server that connects to it.

The following constraints apply when identifying a standby network interface:

- The standby must be unconfigured and connected to the same switch (network) as the primary interface.
- The Segment Server that supports the standby network interface must have access to the IBRIX file system that the clients on that interface will mount.

Virtual interfaces (VIFs) are highly recommended for handling User interface failovers. If a VIF User interface is teamed/bonded, failover occurs only if all the teamed interfaces fail. Otherwise, traffic switches to the surviving teamed network interfaces. For information about creating a VIF, see page 67.

#### Identify standbys for a network interface

Execute this command once for every Segment Server. `IFNAME1` is the network interface that you want to protect, and `IFNAME2` is the standby interface.

```
<ibrixhome>/bin/ibrix_nic -b -H HOSTNAME1/IFNAME1, HOSTNAME2/IFNAME2
```

For example, to identify virtual interface eth2:02 on Segment Server s2.ibrix.com as the standby interface for interface eth2 on Segment Server s1.ibrix.com:

```
<ibrixhome>/bin/ibrix_nic -b -H s1.ibrix.com/eth2, s2.ibrix.com/eth2:2
```

#### Delete a network interface standby

```
<ibrixhome>/bin/ibrix_nic -b -U -H HOSTNAME1/IFNAME1
```
For example, to delete the standby that was assigned to interface eth2 on Segment Server s1.ibrix.com:

```bash
<ibrixhome>/bin/ibrix_nic -b -U -H s1.ibrix.com/eth2
```

### 4.9.2 Identifying and Deleting Network Interface Monitors

It is convenient to identify Segment Server failover pairs as network interface monitors for each other. This is a two-pass process for each failover pair, because the monitoring must be declared in both directions.

- **Set up a network interface monitor**

  ```bash
  <ibrixhome>/bin/ibrix_nic -m -h MONHOST -A DESTHOST/IFNAME
  ```

  For example, to set up Segment Server s2.ibrix.com to monitor Segment Server s1.ibrix.com over User interface eth1:

  ```bash
  <ibrixhome>/bin/ibrix_nic -m -h s2.ibrix.com -A s1.ibrix.com/eth1
  ```

- **delete network interface monitoring**

  ```bash
  <ibrixhome>/bin/ibrix_nic -m -h MONHOST -D DESTHOST/IFNAME
  ```

### 4.10 Setting Up HBA Monitoring

IBRIX Fusion High Availability can be configured to initiate automated failover upon detection of a failed HBA. HBA monitoring can be set up for single-port HBAs, whether standalone or paired for standby switching via software such as EMC PowerPath, and dual-port HBAs with built-in standby switching. IBRIX Fusion does not play any role in vendor- or software-mediated HBA failover—traffic moves to the remaining functional port without any Fusion Manager involvement.

HBAs use worldwide names (WWN) for some parameter values. These are either worldwide node names (WWNN) or worldwide port names (WWPN) The WWPN is the name an HBA presents when logging into a SAN fabric. Worldwide names consist of 16 hexadecimal digits grouped in pairs. In IBRIX Fusion, these are written as dot-separated pairs, for example 21.00.00.e0.8b.05.05.04.

To set up HBA monitoring, discover HBAs and then perform the procedure that matches your HBA hardware:

- For single-port HBAs without built-in switching: Turn on HBA monitoring for all ports that you want to monitor for failure (refer to Section 4.10.3).

- For dual-port HBAs with built-in standby switching and single-port HBAs that have been set up as standby pairs via software: Identify the standby pairs of ports to the configuration database (refer to Section 4.10.2), then turn on HBA monitoring for all paired ports (Section 4.10.3). If monitoring is turned on for just one port in a standby pair, and if that port fails, the Fusion Manager will failover the server even though the HBA has automatically switched traffic to the surviving port. When monitoring is turned on for both, the Fusion Manager initiates failover only when both ports in a pair fail.

When both HBA monitoring and automated failover for Segment Servers are set up, the Fusion Manager will fail over a server in two situations:

- Both ports in a monitored set of standby-paired ports fail. The Fusion Manager knows that it should do nothing if only one port in a standby pair fails, because as part of setting up HBA monitoring you identify all standby pairs to the configuration database, and therefore the Fusion Manager knows that failover is only required when both ports fail.
• A monitored single-port HBA fails. Because no standby has been identified for the failed port, the Fusion Manager knows to initiate failover immediately.

4.10.1 Discovering HBAs

You must discover HBAs before you can set up HBA monitoring, and also when you replace an HBA or add a new one to the cluster.

Discovery only informs the configuration database about a port’s WWPN. You must identify ports that are teamed as standby pairs.

Discover HBAs

<ibrixhome>/bin/ibrix_hba -a -h HOSTLIST

4.10.2 Identifying Standby-Paired HBA Ports

Identifying standby-paired ports to the configuration database allows the Fusion Manager to apply the following logic when they fail:

• If one port in a pair fails, do nothing, because traffic will automatically switch to the surviving port, as provided by the vendor or by software.
• If both ports in a pair fail, failover the server’s segments to the server standby.

Identify two HBA ports as a standby pair

<ibrixhome>/bin/ibrix_hba -b -P WWPN1:WWPN2 -h HOSTNAME

Enter the WWPN as decimal-delimited pairs of hex digits. For example, to identify that port 20.00.12.34.56.78.9a.bc is the standby for port 42.00.12.34.56.78.9a.bc for the HBA on Segment Server s1.ibrix.com:

<ibrixhome>/bin/ibrix_hba -b
- P 20.00.12.34.56.78.9a.bc:42.00.12.34.56.78.9a.bc
- h s1.ibrix.com

4.10.3 Turning HBA Monitoring On and Off

If your cluster uses single-port HBAs, turn on monitoring for all the ports to set up automated failover in the event of HBA failure.

Turn on HBA monitoring for an HBA port

<ibrixhome>/bin/ibrix_hba -m -h HOSTNAME -p PORT

For example, to turn on IBRIX Fusion monitoring for port 20.00.12.34.56.78.9a.bc on Segment Server s1.ibrix.com:

<ibrixhome>/bin/ibrix_hba -m -h s1.ibrix.com
- p 20.00.12.34.56.78.9a.bc

Turn off HBA monitoring for an HBA port

<ibrixhome>/bin/ibrix_hba -m -U -h HOSTNAME -p PORT

For example, to turn off monitoring for port 20.00.12.34.56.78.9a.bc on Segment Server s1.ibrix.com:

<ibrixhome>/bin/ibrix_hba -m -U -h s1.ibrix.com
- p 20.00.12.34.56.78.9a.bc
4.10.4 Deleting Standby Port Pairings

Deleting information about the pairing from the configuration database does not remove the standby pairing of the ports. The standby pairing is either built in by the vendor or implemented by software.

- Delete a set of standby-paired HBA ports from the configuration database

```bash
<ibrixhome>/bin/ibrix_hba -b -U -P WWPN1:WWPN2 -h HOSTNAME
```

For example, to delete the pairing of ports 20.00.12.34.56.78.9a.bc and 42.00.12.34.56.78.9a.bc on Segment Server s1.ibrix.com:

```bash
<ibrixhome>/bin/ibrix_hba -b -U -P 20.00.12.34.56.78.9a.bc:42.00.12.34.56.78.9a.bc -h s1.ibrix.com
```

4.10.5 Deleting HBAs from the Configuration Database

Before switching an HBA card to a different machine, delete the HBA from the configuration database.

- Delete an HBA

```bash
<ibrixhome>/bin/ibrix_hba -d -h HOSTNAME -w WWN
```

4.10.6 Listing Information about HBAs

To list information for all hosts, omit the `-h HOSTLIST` argument. See Table 4-1 for definitions of the output fields.

- List information about HBAs

```bash
<ibrixhome>/bin/ibrix_hba -l [-h HOSTLIST]
```

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Server where the HBA is installed.</td>
</tr>
<tr>
<td>Node WWN</td>
<td>This HBA’s worldwide node name (WWNN).</td>
</tr>
<tr>
<td>Port WWN</td>
<td>This HBA’s worldwide port name (WWPN).</td>
</tr>
<tr>
<td>Port State</td>
<td>Operational state of the port.</td>
</tr>
<tr>
<td>Backup Port WWN</td>
<td>(Standby-paired HBAs only) WWPN of the standby port for this port.</td>
</tr>
<tr>
<td>Monitoring</td>
<td>Whether HBA monitoring is enabled for this port.</td>
</tr>
</tbody>
</table>

4.11 Checking Your High Availability Configuration

Use the `ibrix haconfig` command to check whether IBRIX Fusion High Availability high availability features have been configured for particular Segment Servers. The command
checks whether the following high availability features are configured for Segment Servers and provides either a summary or a detailed report of the results:

- Programmable power source
- Standby server or standby segments
- Cluster and User network interface monitors
- Standby network interface for each User network interfaces
- HBA port monitoring
- Automated failover turned on

The summary report returns one of the following result statuses for each high availability feature, for each tested Segment Server and optionally for their standbys:

- **Passed:** The feature has been configured.
- **Warning:** The feature has not been configured, but the significance of the finding is not clear. For example, the absence of discovered HBAs may indicate either that the HBA monitoring feature was not configured or that HBAs are not physically present on the tested servers.
- **Failed:** The feature has not been configured.

The detailed report includes an overall result status for all tested Segment Servers and describes details about the checks performed on each high availability feature. By default the report includes details only about checks that received a **Failed** or a **Warned** result. You can expand the report to also include details about checks that received a **Passed** result.

### View a summary high availability configuration report

Running the command with no arguments returns a summary of all Segment Servers. To check specific Segment Servers, include the `-h HOSTLIST` argument. To check standbys, include the `-b` argument. To view results only for Segment Servers that failed a check, include the `-f` argument.

```bash
<ibrixhome>/bin/ibrix_haconfig -l [-h HOSTLIST] [-f] [-b]
```

For example, to view a summary report for Segment Servers `xs01.ibrix.com` and `xs02.ibrix.com`:

```bash
<ibrixhome>/bin/ibrix_haconfig -l -h xs01.ibrix.com,xs02.ibrix.com
```

<table>
<thead>
<tr>
<th>Host</th>
<th>HA Configuration</th>
<th>Power Sources</th>
<th>Backup Servers</th>
<th>Auto Failover</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xs01.ibrix.com</td>
<td>FAILED</td>
<td>PASSED</td>
<td>PASSED</td>
<td>PASSED</td>
</tr>
<tr>
<td>xs02.ibrix.com</td>
<td>FAILED</td>
<td>FAILED</td>
<td>FAILED</td>
<td>FAILED</td>
</tr>
</tbody>
</table>

### View a detailed high availability configuration report

```bash
<ibrixhome>/bin/ibrix_haconfig -i [-h HOSTLIST] [-f] [-b] [-s] [-v]
```

See the preceding command for information on the `-h HOSTLIST`, `-f`, and `-b` arguments. To expand the report to include information about the IBRIX file system and its segments, include the `-s` argument. To expand the report to include detailed information about configuration checks that received a **Passed** result, include the `-v` argument.

For example, to view a basic detailed report for Segment Servers `xs01.ibrix.com`:

```bash
<ibrixhome>/bin/ibrix_haconfig -i -h xs01.ibrix.com
```
4.12 Power Management for Segment Servers

Segment Servers can be powered up or down, or reset remotely, as long as the servers are connected to properly configured power sources. To prevent interruption of service, set up standbys for the servers (see page 53) and then manually fail them over before powering them down (see page 56). Remotely powering down a Segment Server does not trigger failover.

<ibrixhome>/bin/ibrix_server -P {on|reset|off} -h HOSTNAME
Chapter 5 Managing Network Interfaces

5.1 About the Cluster and User Network Interfaces

IBRIX Fusion supports two types of logical network interface:

- A Cluster interface that carries Fusion Manager traffic and, by default, traffic between Segment Servers and clients. A cluster may have only one Cluster interface.
- A User interface that carries traffic between Segment Servers and clients. Multiple User interfaces are permitted.

Dedicated network interfaces are not required for these logical interfaces.

IBRIX Fusion supports both link aggregation of network resources and virtual interfaces (VIFs). Link aggregation allows you to combine physical resources into a single VIF. VIFs allow you to provide many named paths within the larger physical resource, each of which can be managed and routed independently (see Figure 5-1).

Link aggregation of network interfaces for the Cluster interface must be done before you install a Segment Server or Fusion Client. Link aggregation to create a User interface may be done at any time. An IBRIX Fusion cluster must adhere to the rules or restrictions that a network interface vendor requires regarding link aggregation, as spelled out in product documentation.

**Note!** All Cluster network interfaces must be defined (including link aggregation, if applicable) and operational before installing IBRIX Fusion.
5.2 Setting Up User Networks for Clients

By default, all Segment Servers, Fusion Clients, NFS clients, CIFS clients, and the Fusion Manager use the Cluster network. This arrangement may be adequate if your environment includes only Fusion Clients or only NFS or CIFS clients. However, if your installation must accommodate both NFS/CIFS clients and Fusion Clients, or if you need to segregate client traffic to different networks, you will need to identify one or more User networks (see page 67). If the User interface is for Fusion Client traffic, you will additionally need to prefer the network for the Fusion Clients that should use the network (see page 68).

**Note!** Fusion Clients and NFS/CIFS clients cannot share a User interface.

Additionally, it is inadvisable to use the Cluster interface for NFS traffic because this interface cannot be failed over. If the Cluster interface on a Segment Server fails and the server’s segments are automatically or manually failed over to a standby, any NFS clients that were using the failed interface to access a mounted IBRIX file system will lose contact with the file system because they have no knowledge of the cluster and cannot re-route requests to the standby.

To prevent interruptions to NFS traffic, IBRIX recommends that you put it on a dedicated User network and then set up automated failover for the network, to ensure continuous client access to files (see page 52).

You may find the IBRIX Fusion Autoconnect feature helpful in managing NFS clients. Autoconnect simplifies mounting and balances client load among the Segment Servers. For information on NFS and Autoconnect, refer to page 75.

5.3 Rules for Setting Up the Cluster Network

Follow these rules when setting up the cluster network:

1. **The Fusion Manager be connected to all machines in the cluster, including standby servers, using the Cluster network interface.**
   
   Each Segment Server and Fusion Client must be connected to the Fusion Manager by the same Cluster network interface. Each machine can only support one Cluster network interface. All IBRIX Fusion traffic defaults to the Cluster interface unless you prefer a User network interface for it.
   
   You must identify the Cluster network interface and the IP address (optionally, a hostname) of the Fusion Manager in all of the installations you perform.
   
   You must use a Gigabit Ethernet port for the Cluster interface, especially if it supports traffic between Segment Servers and Fusion Clients as well as Fusion Manager traffic.

2. **Segment Servers that need to communicate for file system coverage or for failover must be on the same network.**

   All Segment Servers must be on the Cluster interface to enable communication with each other and the Fusion Manager regarding the IBRIX file system. In addition, Segment Servers that are set up as a failover pair must be connected to the same User interface. IBRIX recommends that you use a Gigabit Ethernet port for User networks.

3. **Fusion Clients must have network connectivity to the Segment Servers that manage their data and to those servers’ standbys.**

   This traffic can use the Cluster network interface or a User interface.
### 5.4 Identifying a User Interface

User interfaces that are intended for Fusion Client traffic must be identified to the configuration database as described below and then preferred for the Fusion Clients as described on page 68.

It is not necessary to identify a User interface that is intended for NFS traffic, but IBRIX recommends that you do this so that you can implement automated failover for the interface. It is not necessary to prefer a User interface for NFS traffic because NFS clients use client-side information to select a User interface when they mount an IBRIX file system.

Use only alphanumerics and underscores in a User interface name. To name a VIF, add the VIF suffix (:nnnn) to the physical interface name.

After you identify a network interface for a Segment Server, the Fusion Manager queries the server for its IP address, netmask, and MAC address and imports these values into the configuration database. You can modify these values later as necessary.

Note that in the case of VIFs, the Fusion Manager does not automatically query the Segment Server for this information. If you are planning to use the VIF only as a standby network interface in an automated failover setup, the Fusion Manager will query the server the first time a network is failed over to the VIF. Otherwise, you must manually enter the VIF’s IP address and netmask into the database (see page 69; it is not necessary to enter a MAC address because the Fusion Manager does not need this information for a VIF).

#### Identify a User interface

```
<ibrixhome>/bin/ibrix_nic -a -n IFNAME -h HOSTLIST
```

For example, to identify physical network interface eth1 on Segment Servers s1.ibrix.com and s2.ibrix.com:

```
<ibrixhome>/bin/ibrix_nic -a -n eth1 -h s1.ibrix.com,s2.ibrix.com
```

To identify virtual interface eth1:1 to physical network interface eth1 on Segment Servers s1.ibrix.com and s2.ibrix.com:

```
<ibrixhome>/bin/ibrix_nic -a -n eth1:1 -h s1.ibrix.com,s2.ibrix.com
```
5.5 Preferring Network Interfaces

Fusion Clients and Segment Servers use the Cluster interface by default unless assigned, or preferred, to a User network interface. It is not necessary to prefer a network interface for NFS or CIFS clients because they can select the correct User interface at mount-time.

When you prefer a User interface for traffic from a source host to a destination host, traffic in the reverse direction remains defaulted to the Cluster interface.

To return Fusion Clients to the Cluster interface, locally re-prefer the Cluster interface for them. To return Segment Server traffic to the Cluster interface, unprefer the network interface that has been preferred for them.

You can prefer an interface on multiple hosts at one time by specifying a hostgroup as the source host. To prefer a User interface for all Fusion Clients, specify the clients hostgroup. After preferring a network interface for a hostgroup, you can locally override the preference on individual Fusion Clients with the command `ibrix_lwhost`.

A network interface preference is immediately executed on Segment Servers. For Fusion Clients the preference intention is stored in the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager for the network interface that has been preferred for it, and then it begins to use that interface.

If IBRIX services are already running on Fusion Clients when you prefer a network interface, you can force clients to query the Fusion Manager for the preference intention by executing `ibrix_client` or `ibrix_lwhost --a` on the Fusion Client, or rebooting the Fusion Client.

- **Prefer a network interface for a Segment Server or Fusion Client**

  ```bash
  <ibrixhome>/bin/ibrix_host -n -h SRCHOST -A DESTHOST/IFNAME
  ```

  Execute this command once for each destination host that the specified Segment Server or Fusion Client should contact using the specified network interface (`IFNAME`).

  For example, to prefer network interface `eth3` for traffic from Segment Server `s1.ibrix.com` to Segment Server `s2.ibrix.com`:

  ```bash
  <ibrixhome>/bin/ibrix_host -n -h s1.ibrix.com -A s2.ibrix.com/eth3
  ```

- **Prefer a network interface for a hostgroup**

  ```bash
  <ibrixhome>/bin/ibrix_hostgroup -n -g HOSTGROUP -A DESTHOST/IFNAME
  ```

  The destination host (`DESTHOST`) cannot be a hostgroup.

  For example, to prefer network interface `eth3` for traffic from all Fusion Clients to Segment Server `s2.ibrix.com`:

  ```bash
  <ibrixhome>/bin/ibrix_hostgroup -n -g clients -A s2.ibrix.com/eth3
  ```

5.6 Unpreferring Network Interfaces

To return Segment Servers or Fusion Clients to the Cluster interface, unprefer their preferred network interface.

- **Unprefer a network interface for a Segment Server or Fusion Client**

  ```bash
  <ibrixhome>/bin/ibrix_host -n -h SRCHOST -D DESTHOST
  ```

- **Unprefer a network interface for a hostgroup**

  ```bash
  <ibrixhome>/bin/ibrix_host -n -g HOSTGROUP
  ```
5.7 Modifying Network Interface Options

You can modify selected ifconfig options for a network. Also, to make a VIF usable you must modify its network options to configure an IP address and a netmask for it.

Modify network interface options

```
<ibrixhome>/bin/ibrix_nic -c -n IFNAME -h HOSTNAME [-I IPADDR]
   [-M NETMASK] [-B BCASTADDR] [-T MTU]
```

For example, to set netmask 255.255.0.0 and broadcast address 10.0.0.4 for interface eth3 on Segment Server s4.ibrix.com:

```
<ibrixhome>/bin/ibrix_nic -c -n eth3 -h s4.ibrix.com
   -M 255.255.0.0 -B 10.0.0.4
```

5.8 Changing IP Addresses

5.8.1 Changing a Linux Fusion Client IP Address

If you change a Linux Fusion Client’s IP address, use the following procedure to update the configuration database with the new information, or the Fusion Manager will be unable to communicate with the client.

To change a Linux Fusion Client IP address:

1. Unmount the IBRIX file system from the Fusion Client (see page 15).
2. Change the Fusion Client’s IP address.
3. Reboot the Fusion Client or restart the network interface card.
4. Re-register the Fusion Client with the Fusion Manager:
   ```
   <ibrixhome>/bin/register_client -x
   ```
5. Delete the old IP address from the configuration database:
   ```
   <ibrixhome>/bin/ibrix_host -d -h CLIENT
   ```
6. Enter the new IP address:
   ```
   <ibrixhome>/bin/ibrix_host -a -h CLIENT -e IPADDRESS
   ```
7. Remount the IBRIX file system on the Fusion Client (see page 13).

5.8.2 Changing the Cluster Interface IP Address

The address must be changed on both Segment Servers and the Fusion Manager. The usual case is to change the Cluster IP address on all Segment Servers.

1. If high availability is enabled, disable it if necessary with `ibrix_host -m -U`.
2. Unmount the IBRIX file system from all Segment Servers (see page 15).
3. Locally change the IP address of the Cluster interface on each Segment Server.
4. Change the Cluster interface IP address for each Segment Server on the Fusion Manager:
   ```
   <ibrixhome>/bin/ibrix_nic -c -n IFNAME -h HOSTNAME [-I IPADDR]
   ```
5. Re-mount the IBRIX file system (see page 13).
6. Re-enable Fusion High Availability if necessary with `ibrix_host -m`.

5.9 Changing the Cluster Interface

If you restructure your networks, you may need to change the cluster interface.
5.10 Adding and Deleting Routing Table Entries

IBRIX Fusion supports one route for each network interface in the system routing table. Entering a new route for an interface overwrites the existing routing table entry for that interface.

Add a routing table entry

\[
\text{<ibrixhome>/bin/ibrix_nic -r -n IFNAME -h HOSTNAME -A -R ROUTE}
\]

For example, to enter a route for virtual interface `eth2:232` on Segment Server `s2.ibrix.com`, sending all traffic through gateway `gw.ibrix.com`:

\[
\text{<ibrixhome>/bin/ibrix_nic -r -n eth2:232 -h s2.ibrix.com -A -R gw.ibrix.com}
\]

Delete a routing table entry

If you delete a routing table entry, it is not replaced with a default entry. A new replacement route must be manually added. To delete a route:

\[
\text{<ibrixhome>/bin/ibrix_nic -r -n IFNAME -h HOSTNAME -D}
\]

For example, to delete all routing table entries for virtual interface `eth0:1` on Segment Server `s2.ibrix.com`:

\[
\text{<ibrixhome>/bin/ibrix_nic -r -n eth0:1 -h s2.ibrix.com -D}
\]

5.11 Deleting a Network Interface

If you delete the Cluster interface from a Segment Server, be sure to identify a new one so that the machine can contact the Fusion Manager.

Delete a network interface

\[
\text{<ibrixhome>/bin/ibrix_nic -d -n IFNAME -h HOSTLIST}
\]

For example, to delete interface `eth3` from Segment Servers `s1.ibrix.com` and `s2.ibrix.com`:

\[
\text{<ibrixhome>/bin/ibrix_nic -d -n eth3 -h s1.ibrix.com,s2.ibrix.com}
\]

5.12 Viewing Network Interface Information

See Table 5-1 definitions of the output fields.

List network interfaces

Running the command with no arguments lists all interfaces on all Segment Servers. Use `-h` to list interfaces on specific hosts.

\[
\text{<ibrixhome>/bin/ibrix_nic -l -h HOSTLIST}
\]

For example, to list network interfaces on Segment Server `lab13-41` on the local network:

\[
\text{<ibrixhome>/bin/ibrix_nic -l -h lab13-41}
\]
<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKUP HOST</td>
<td>Segment Server for the standby network interface.</td>
</tr>
<tr>
<td>BACKUP-IF</td>
<td>Standby network interface.</td>
</tr>
<tr>
<td>HOST</td>
<td>Segment Server. An asterisk (*) denotes the Fusion Manager.</td>
</tr>
<tr>
<td>IFNAME</td>
<td>Network interface on this Segment Server.</td>
</tr>
<tr>
<td>IP_ADDRESS</td>
<td>IP address of this NIC.</td>
</tr>
<tr>
<td>MAC_ADDR</td>
<td>MAC address of the network interface card.</td>
</tr>
<tr>
<td>ROUTE</td>
<td>IP address in routing table used by this NIC.</td>
</tr>
<tr>
<td>STATE</td>
<td>Network interface state.</td>
</tr>
<tr>
<td>TYPE</td>
<td>IBRIX Fusion network type (Cluster or User).</td>
</tr>
</tbody>
</table>
Chapter 6  NFS and CIFS Procedures

6.1 Exporting and Unexporting File Systems Using NFS

The Fusion Manager manages the table of exported IBRIX file systems and distributes the information to Segment Server /etc/exports files. All entries are automatically re-exported to NFS clients and to the Segment Server standbys unless you specify otherwise.

An IBRIX file system must be mounted before it can be exported.

On the exporting Segment Server, configure the number of NFS server threads based on the expected workload. The default is 8 threads. It can be increased to 16 or 64 if the Segment Server will service many clients. To configure server threads, change the default value of RPCNFSDCOUNT in the /etc/init.d/nfs file from 8 to either 16 or 64.

```
# Number of servers to be started by default
[ -z "$RPCNFSDCOUNT" ] && RPCNFSDCOUNT=8
```

Export an IBRIX file system using NFS

```
<ibrixhome>/bin/ibrix_exportfs -f FSNAME -h HOSTNAME
    -p CLIENT;PATHNAME [-o "OPTIONS"] [-b]
```

The CLIENT argument takes a Segment Server name, a Segment Server IP address, a Segment Server name with a wildcard, or a netmask. To provide world access, omit the CLIENT specification, but include the colon, as shown in the second example below. The command uses the default Linux exportfs options unless specific options are provided. If you do not want to export the file system to the server’s standby, include the -b argument.

For example, to provide NFS clients *.ibrix.com with read-only access to IBRIX file system ifs1 at directory /usr/src on Segment Server s1.ibrix.com:

```
<ibrixhome>/bin/ibrix_exportfs -f ifs1 -h sl.ibrix.com
    -p *.ibrix.com:/usr/src -o "ro"
```

To provide world read-only access to IBRIX file system ifs1 located at /usr/src on Segment Server s1.ibrix.com:

```
<ibrixhome>/bin/ibrix_exportfs -f ifs1 -h sl.ibrix.com
    -p :/usr/src -o "ro"
```
Unexport an IBRIX file system

A file system should be unexported before it is unmounted.

```
<ibrixhome>/bin/ibrix_exportfs -f FSNAME -U -h HOSTNAME
   -p CLIENT:PATHNAME [-b]
```

6.2 Setting Up CIFS Management for a Cluster

To enable CIFS clients to access an IBRIX file system, configure CIFS settings on Segment Servers that mount the file system. Users may then add CIFS shares to the file system. Optionally, you can use RPC authentication instead of ADS authentication (the default) and enable NetBIOS. If you enable NetBIOS, you may wish to specify a WINS server to improve performance.

In addition to the settings that you explicitly configure, IBRIX Fusion also implements certain default CIFS settings and share settings. You may modify the default values of some of these settings.

The minimum level of Samba that the `ibrix_cifs` command supports is 3.0.24.

Configure CIFS settings using the Fusion Manager CLI

```
<ibrixhome>/bin/ibrix_cifs -n DOMAIN_NAME -c DOMAIN_CONTROLLER
   -A AUTH_PROXY_USER_NAME -P AUTH_PROXY_PASSWORD
   [-w WINSSERVER] [-E] [-R] [-h HOSTLIST] [-S SETTINGLIST]
```

To configure settings for specific Segment Servers, include the `-h HOSTLIST` argument.

To enable RPC authentication, include the `-R` argument. To enable NetBIOS, include the `-E` argument. To specify a Windows Internet Naming Server (WINS) server, include the `-w WINSSERVER` argument.

To modify CIFS settings, include the `-S SETTINGLIST` argument. Enter settings as a list of comma-separated configuration-value pairs, with each pair separated by an equals sign (for example, `socket options=TCP_NODELAY`).

Add a CIFS share to Segment Servers

```
<ibrixhome>/bin/ibrix_cifs -a -s SHARENAME -p SHAREPATH
   [-S SETTINGLIST] [-h HOSTLIST]
```

To add shares to specific Segment Servers, include the `-h HOSTLIST` argument.

To modify default share settings, include the `-S SETTINGLIST` argument. Enter share settings as a list of configuration-value pairs, with each pair separated by an equals sign (for example, `read only=no`).

Modify a CIFS share

```
<ibrixhome>/bin/ibrix_cifs -m -s SHARENAME [-S SETTINGLIST]
   [-h HOSTLIST]
```

Delete a CIFS share

```
<ibrixhome>/bin/ibrix_cifs -d -s SHARENAME [-h HOSTNAME]
```

View CIFS shares on Segment Servers

```
<ibrixhome>/bin/ibrix_cifs -l [-h HOSTLIST]
```
View modified CIFS settings and share settings

<ibrixhome>/bin/ibrix_cifs -L

6.3 Autoconnecting NFS Clients

An IBRIX Fusion autoconnection enables NFS clients to mount IBRIX file systems automatically whenever they are accessed. At the same time, Autoconnect manages how these connections are distributed among Segment Servers. Autoconnect uses the Linux automount daemon, and working familiarity with automount is recommended.

Autoconnect accesses a user-edited script that directs NFS client file requests to the Fusion Manager, where they are checked against the database and matched to a mount string. The mount string and any mount options are returned to the client along with the Segment Server that the client should use for the mount.

Mount points are stored in the Autoconnect table in the IBRIX database. Each mount point is described by a user-defined identifier, or key; the IBRIX file system to mount; and any assigned mount options. The ibrix_autoconnect -l command displays any current autoconnect entries.

6.3.1 Adding Mount Points to the Autoconnect Table

An exported file system must be available in order to add mount points.

Add an autoconnect entry

<ibrixhome>/bin/ibrix_autoconnect -A -k KEY -f FSNAME [-o OPTIONS]

Where KEY is a user-defined key, FSNAME is the file system name, and OPTIONS identify NFS mount options (listed in the Linux mount command’s man page).

For example, to add an entry for the user-defined key name ifs1_rw to mount the file system ifs1 with options that identify the file system type as NFS and the allowed user operations as read and write, enter:

<ibrixhome>/bin/ibrix_autoconnect -A -k ifs1_rw -f ifs1
-o -fstype=nfs,rw

6.3.2 Deleting Mount Points from the Autoconnect Table

Mount points are deleted by specifying a key entry:

<ibrixhome>/bin/ibrix_autoconnect -D -k KEY

For example, to delete the key ibrix_ro:

<ibrixhome>/bin/ibrix_autoconnect -D -k ibrix_ro

6.3.3 Setting Up NFS Clients

With mount points defined in the Fusion Manager, client setup entails editing an IBRIX-supplied Autoconnect script to point to the Fusion Manager and the correct port, editing the /etc/auto.master file on the client, and restarting services.

As a preparatory check on every client,

• Verify that automount is installed by running /etc/init.d/autofs status.
• Make sure that wget or curl is installed if you plan to install a modified auto.curl or auto.wget script. Use a utility like find to search if necessary.
Every client must have two files placed in its /etc directory. The necessary files are on the Fusion Manager, so edit them there and copy them to each client. For every client:

1. Make a copy of the appropriate script (auto.curl, auto.sh, auto.wget) found in `<ibrix-home>/examples/autoconnect`. Edit the copy to set fusionmanager to the IP address of your Fusion Manager, and verify port is set to 9009.
   If you choose, you can write a custom script that provides the same functionality as one of the IBRIX-supplied scripts. You can also create multiple scripts, each mapping to a different set of primary keys.

2. Set permissions on the script file to make it executable, for example for a curl script:
   ```bash
   chmod +x /etc/auto.curl
   ```

3. Edit the /etc/auto.master file to map a base automount mount point to the script edited in step 1. For example, to map the base mount point /ibrix to a curl script, enter:
   ```bash
   /ibrix /etc/auto.curl --timeout=60
   ```
   Where timeout indicates the number of seconds that a connection remains idle before it is automatically disconnected.
   The mount point name could be the same as a key name, but this is not recommended.

4. Copy the edited files to the clients’ /etc directory.

5. On each client, restart autofs and enable it for future use:
   ```bash
   /etc/init.d/autofs restart
   chkconfig autofs on
   ```

6. Confirm that autofs recognizes the primary keys that you entered in auto.master:
   ```bash
   /etc/init.d/autofs status
   ```

7. Confirm that files are visible at the IBRIX mount point by listing the directory by base mount point and key name. For example, for the key name ifs1_RW:
   ```bash
   ls -l /ibrix/ifs1_RW
   ```
Chapter 7  The Windows Fusion Client

7.1 About Windows Fusion Clients

The Windows Fusion Client allows applications running on Windows machines to access and update an IBRIX file system. Like Linux Fusion Clients, Windows clients can read, write and delete files by sending requests to Segment Servers. Cross-platform access is achieved by mapping Linux UIDs and GIDs to Windows users and storing the information on the Active Directory server.

The Windows Fusion Client software is usually installed when the IBRIX Fusion cluster is set up. If it was not, refer to the IBRIX Fusion Installation and Upgrade Guide for information on installation prerequisites and procedures.

7.2 System Requirements

The Windows Fusion Client requires Microsoft’s .NET Framework Version 2.0 be installed. Client computers must be members of a security domain managed by Active Directory.

The Windows Fusion Client requires a connection to a Windows Active Directory server to lookup mappings between Windows and Linux users. Two versions of Active Directory servers are supported:

- Windows Server 2003 SP2. This server requires that you install Microsoft’s Services for UNIX 3.5 (SFU) software, which is part of the server distribution but not installed by default.
- Windows Server 2003 R2. SFU is built into R2 and requires no separate installation.
Copy the Windows Fusion Client installer to all client machines and run it. If you did not register the clients when they were installed, do so following the instructions in Section 7.7, Register Windows Fusion Clients and Start Service.

7.3 Overview of the Windows Fusion Client Setup

Setting up the Windows Fusion Client entails work on the Active Directory Server, the FusionManager, and the Windows client itself.

1. Set up Services for UNIX 3.5 on the Active Directory global catalog server.
2. Define an administrative group with a GID of zero on the Active Directory Server.
3. Create a default Windows user on the Active Directory Server.
4. Create an Active Directory proxy user with permission to read only UID/GID information and delegate control of user folders to this proxy user.
5. Configure Active Directory settings on the Fusion Manager to enable Fusion Client look-ups on the Active Directory server.
6. Set up and register each Windows Fusion Client on the Fusion Manager.

The following procedures were tested on the Active Directory Users and Computers component of the Microsoft Management Console shipped with Windows Server 2003 SP2 with SFU 3.5. Prompts, window names, etc. may vary slightly on other configurations.

7.4 Set up Windows Services for UNIX

The setup procedure differs on the supported versions of Windows Server 2003.

Services for UNIX on Windows 2003 SP2

Windows Services for UNIX 3.5 can be downloaded from Microsoft’s web site. IBRIX Fusion does not require installation of the entire SFU 3.5 package. What you install depends on what your site supports, but you must install the Server for NIS component. When you run the SFU Setup Wizard, you must select at least that component.

Services for UNIX on Windows 2003 R2

You do not need to install SFU on servers running the R2 edition. It is built in.

1. On the Active Directory server, open the Control Panel, open Add Remove Programs, and click on **Add Remove Windows Components**.
2. Check the box for **Active Directory Services** and click **Details**.
3. Check the box for **Identity Management for UNIX** and click **Details**.
4. Check the box for **Server for NIS**.
5. Click **OK** to close the Identity Management for UNIX window, click **OK** again to close the Active Directory Services window, and click **Next** to install.

7.5 Configure Groups and Users on the Active Directory Server

7.5.1 Create Administrative User/Group

It is necessary to have an administrative user in Active Directory mapped to Linux root (UID 0) in order to extend root’s permissions on the file system to the Windows side. You can create
a new user or modify an existing user, but this user must be assigned the UID of 0 on its Properties > UNIX Attributes tab.

Alternatively, you can create or modify an administrative group in Active Directory, with all of its members having root privileges on Fusion files and folders. This group must be assigned the GID of 0 on the group’s Properties > UNIX Attributes tab, and be mapped to the root group on Linux having GID 0. Keep in mind, however, that the Linux root group might have a lower level of permissions than root itself (for example, it might not have write permission). If you go this route, make sure the permissions on the Linux root group are rwx before mapping.

Mapping a single user to UID 0 may be more secure than granting the same level of control over all Fusion files to multiple users.

### 7.5.2 Create Proxy User and Delegate Folder Control

The proxy user queries the Active Directory server on behalf of the client to find mappings from Linux UIDs/GIDs to Windows SSIDs. This user is required and must be defined in the Fusion Manager with the `-ibrix_activedirectory` command, as well as being created in Active Directory.

1. Log in to the Active Directory’s Main Catalog server and open the Active Directory Users and Computer window.
2. Under the domain where the user will be created, right-click **Users**, then select **New**, then select **User**.
3. On the Create New Object - User screen, add the user. Two fields are required, Full name and User logon name. You can use a name like `fusion_proxy` for both, but it can be any name of your choice. The domain is automatically assigned. Click **Next**. Assign a password and password policy. Click **Next**. Click **Finish**.
4. Right-click the **Users** folder, select **Delegate Control** to open the delegation wizard, and click **Next** to open the Users or Groups window.
5. Click **Add** to open the Select Users, Computers, or Groups window. Add your new user (`fusion_proxy`) in the Enter Object Names field. Click **Next** to open the Tasks to Delegate window.
6. Select **Create a Custom Task to Delegate**.
7. Click **Next** to open the Active Directory Object Type window. Check **Only the Following Objects**. Scroll to and check **User Objects**. Click **Next** to open the Permissions window.
8. Check **Property-Specific**. The property names vary by server version:
   - On Windows Server 2003 SP2: Scroll to and check **Read msSFU30GidNumber** and **Read msSFU30UidNumber**.
   - On Windows Server 2003 R2: Scroll to and check **Read gidNumber** and **Read uidNumber**.
9. Click **Next**. Click **Finish**.

If you create any other OUs (Organizational Units) in Active Directory and users in them will access the IBRIX file system, delegate control for these OUs to the proxy user also.

### 7.5.3 “Unknown” IBRIX Windows User

The “unknown” IBRIX Windows user is displayed as the owner of a file when the FusionClient cannot resolve a user mapping. This user is required and must be defined in the Fusion Manager with the `-ibrix_activedirectory` command. You can assign any name to this user.
7.5.4 Other Windows Fusion Client Users

All Windows users that will access the IBRIX file system must be assigned a UID and GID on their UNIX Attributes tab. If you want to map these to specific Linux users, use the IDs from the Linux side, for example the users in /etc/passwd. If specific mappings are not important, you can accept the next available UID and GID as generated by Active Directory when the users are added.

Unmapped users are granted the Others permissions as defined by the mode mask settings. Unmapped users cannot create new files or directories in the file system even if they have such permissions on Windows.

7.6 Configure Active Directory Settings on Fusion Manager

For the Fusion Manager, first configure Active Directory settings, and then register each individual client on the Fusion Manager.

To enter Active Directory settings using the Fusion Manager CLI, run ibrix_activedirectory on the Fusion Manager, entering the proxy user name and unknown Windows user name, with passwords, as created in Active Directory.

On Windows Server 2003 R2, the -E and -F arguments are required. Use the field names gidNumber and uidNumber as values.

Note! Specify the proxy user name in the format "domainname\username" where domainname is the name of the NIS domain in Active Directory. The double quotes are required.

```
```

The second of these examples applies to Windows Server 2003 R2:

```
<ibrixhome>/bin/ibrix_activedirectory -c -d fm1.ibrix.com -i 192.168.1.1 -u "cs\fusion_proxy" -p proxy12345 -W fusion_winuser
```

7.7 Register Windows Fusion Clients and Start Service

The Active Directory setup must be complete before registering Windows Fusion Clients with the Fusion Manager.

All Fusion Clients must be registered with the Fusion Manager before they can mount an IBRIX file system. Windows Fusion Clients are registered on the client itself. Repeat this procedure on each Windows Fusion Client.

Note! You may encounter problems with client access caused by firewall settings. We recommend you turn the firewall off while testing. When you turn the firewall back on, open ports 1234 and 9000 through 9010 for IBRIX Fusion use.

1. Launch the Windows Fusion Client GUI and navigate to the Registration tab (Figure 7-1).
2. Select the client’s IP address from the drop-down list.
3. Enter the Fusion Manager name into the FM Host Name field.
4. Check **Recover Registration** to avoid having to re-register this client if you reinstall it. This option automatically retrieves the Fusion Client’s id from the Fusion Manager.

5. To start the Windows Fusion Client service, check **Start Service After Registration**.

6. Click **Register**.

7. On the Active Directory Settings tab, click **Verify** to validate that the proxy user has access to Active Directory for query mapping.

8. On the client’s Mount tab, select the Fusion Manager in the drop-down list if necessary and enter the file system name. To mount the IBRIX file system to a Windows drive letter, click **Mount**.

9. If you are using Remote Desktop to access the client and the drive letter does not display, log out and log back in. This is a known limitation of Windows Terminal Services when exposing new drives.

![Figure 7-1 Windows Fusion Client GUI – Registration Tab](image)

If Active Directory settings are changed on the Fusion Manager later, use the FusionClient’s Update function to copy the new settings to each client, and test the settings on any single client to verify the proxy account is correct and the client can connect to the Active Directory server.

1. On every client’s Active Directory Settings tab, click **Update** to copy settings to the client.

2. On a single client, click **Verify** to test the settings. A success message indicates all clients can communicate, and a failure message indicates a settings error on one or more clients.

The IBRIX service needs to be restarted after the update.

**Automatically Starting FusionClient Service**

When the client is functioning to your satisfaction, change the client service to automatically start when the machine is booted. The FusionClient service is installed to start manually by default. To do this:

2. Scroll to and right-click on **Fusion Client** in the services list. Select **Properties**.
3. Set the Startup Type to Automatic. Click **OK**.
7.8 Importing UIDs/GIDs into Active Directory

If you have many FusionClient users, use this procedure to import their Linux UIDs/GIDs into the Active Directory server with SFU.

You only need to do this once. Thereafter, add new UIDs/GIDs directly to the Active Directory as new users. You must explicitly activate imported users after adding them, because the Active Directory server immediately disables newly created users.

1. From any Segment Server, get the contents of the passwd file:
   # getent passwd > /tmp/passwd

2. Edit tempPasswdFile to remove users that don’t need to be imported to Windows.

3. Enter x into the password field (the second field) for all entries.

4. Get the contents of the group file:
   # getent group > /tmp/group

5. Edit tempGroupFile to remove groups that don’t need to be imported to Windows.


7. On Windows Server 2003 SP2 only, install Windows Services for Unix (SFU) 3.5 to the default location on the Active Directory server/domain controller. In the User Name Mapping install window, select Password and Group file and enter c:\temp\passwd for the password file and c:\temp\group for the group file.

8. On Windows Server 2003 SP2 only, create the file <SFU program directory>\nis\ldif.log. (Windows does not create this file when SFU is installed.)

9. Open a DOS command window and change directory to c:\temp.

10. Import user UIDs and GIDs into the Active Directory server:
    nis2ad -y ibrix -a <Active Directory domain name> -r yes -d C:\temp -m group
    nis2ad -y ibrix -a <Active Directory domain name> -r yes -d C:\temp -m passwd

    \for Windows 2000
    %systemroot%\system32\ldifde -m -i c:\passwd.1df
    \for Windows 2003
    %systemroot%\system32\ldifde -m -f c:\passwd.1df

11. Open the Active Directory Users and Computer window to verify that the user data was imported.

7.9 Using the Windows Fusion Client GUI

The Windows Fusion Client GUI is the client interface to the Fusion Manager. To open the GUI, double-click the desktop icon or select the IBRIX Client program from the Start menu. The client program is organized into tabs by function:

- **Status**: Shows the client’s Fusion Manager registration status and mounted IBRIX file systems, and provides access to the IAD log for troubleshooting.

- **Registration**: Registers the client with a Fusion Manager (refer to Section 7.7, *Register Windows Fusion Clients and Start Service*).

- **Mount**: Mounts an IBRIX file system. Select the Cluster Name from the drop-down list (the cluster name is the Fusion Manager name). Enter the IBRIX file system name to mount. Select a drive, then click Mount. As noted in Section 7.7, *Register Windows Fusion Clients and Start Service*, if you are using Remote Desktop to access the client and the drive letter does not display, log out and log back in.

- **Umount**: Unmounts an IBRIX file system.
• **Tune Host:** Tunable parameters include the NIC to prefer (The client uses the Cluster interface by default unless a different network interface is preferred for it.), the communications protocol (UDP or TCP), and the number of server threads to use.

• **Active Directory Settings:** Displays current Active Directory settings.

Refer to the client’s online help if necessary.

### 7.10 Managing Access Control Lists

Because the Windows FusionClient may be operating in a “mixed-mode” environment in which the same file objects are accessed from both Linux and Windows, system administrators must consider the differences between the Linux and Windows models of file access permissions.

#### 7.10.1 Windows Access Controls

The Linux standard file mask assigns read, write, and execute permissions on files and folders to three classes of user (owner, group, others).

Windows defines permissions on files and folders in the Access Control List (ACL), a data structure that consists of Access Control Entries (ACEs) which **allow** or **deny** a given permission on the file to a given user or group.

ACEs can be explicit or inherited. An **explicit ACE** is assigned directly to the object by the owner or an administrator, while an **inherited ACE** is inherited from the parent directory.

ACEs are governed by precedence rules:

- An explicit deny ACE overrides an explicit allow ACE, and an inherited deny ACE overrides an inherited allow ACE. For example, if an explicit allow ACE grants a user read-write permission, but an explicit deny ACE denies the same user the write permission, the effective permission for this user is read only.

- An explicit ACE overrides an inherited ACE. For example, if an explicit allow ACE grants the user read-write permission and an inherited deny ACE denies this same user the write permission, the resulting permission for this user is still read-write.

An ACL that is assigned to a file created by IBRIX Fusion defines up to three special explicit allow ACEs derived from the file mask, in addition to any other explicit and inherited ACEs the file might have.

#### 7.10.2 Linux Mode Mask to Special ACEs Mapping

The FusionClient maps a file’s mode mask to a set of up to three special explicit Allow ACEs, as shown in Table 7-1: one for the Windows user that corresponds to the file UID, another for the Windows group that corresponds to the file GID, and the third for the built-in Windows group Everyone that corresponds to the file others class of user.

<table>
<thead>
<tr>
<th>Linux Class</th>
<th>Windows Account</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner (owning user)</td>
<td>Owner special ACE</td>
</tr>
<tr>
<td>Group (owning group)</td>
<td>Group special ACE</td>
</tr>
<tr>
<td>Other</td>
<td>Everyone special ACE</td>
</tr>
</tbody>
</table>

Table 7-1  Linux Mode Mask/Windows Special ACE Mappings
The permissions for each special ACE are set according to the bits in each category. If all bits in some categories are cleared, no corresponding special ACE is added to the file ACL and no explicit Deny ACE is generated.

7.10.3 User Mapping

**Owner Mapping.** Each file/directory in Linux has a UID that defines its owner and should be mapped to a corresponding Windows user. Refer to Section 7.5, *Configure Groups and Users on the Active Directory Server.*

If the user mapping can be resolved, this user is used as the owner in the Owner special ACE, and displayed as the owner of the file.

If the user mapping cannot be resolved, an “unknown” Windows user is used instead. The unknown user must be defined on the Fusion Manager.

**Group Mapping.** Each file/directory in Linux has a GID that defines its owner group, with access rights as specified by the mode mask. A Windows group can be mapped to a corresponding Linux GID.

If the mapping can be resolved, this group is used as the owning group in the Group special ACE. If the mapping cannot be resolved, the Group special ACE is not added to the file ACL.

7.10.4 ACL to Mode Mask Mapping

If a special ACE is modified by the Windows client, the corresponding bits in the file mode mask are updated. Likewise, if the mode mask is modified by the Linux client, the corresponding permission in the special ACEs are updated.

Note that inherited ACEs do not affect the file mode mask, only special ACEs do. For example, if we have a special ACE for Everyone with read permission, and an inherited ACE for Everyone with read-write-execute permissions, the corresponding permission in the file mode mask for others is set to read only. The write-execute permissions of the inherited ACE are ignored in the mapping.

When an explicit deny ACE is added to a file’s ACL, the corresponding allow permissions are removed for group and others in the file mode mask, and the corresponding special explicit ACEs will be updated accordingly.

An inherited deny ACE has no effect on the mode mask. Any explicitly set rights override inherited restrictions.

To view a Windows ACL, right click any file on a Windows 2003 server and select Properties. Click the Security tab. The Security Properties tab for your file will look like the one shown in Figure 7-2 (if you clicked on a directory, the additional permission List Directory Contents is also included). This is the ACL for a file named postinfo.html. The progression of windows shows the permissions granted to the group Everyone by this file’s ACL.

Each Windows permission that is listed as allowed or denied on the Security tab is actually a grouping of related special permissions, as shown on the Permission Entry window.
Figure 7-2   Windows ACL and permissions window

There are three permissions important to IBRIX in the Permissions Entry window: Read Data, Write Data, and Execute File. These map directly to Read, Write, and Execute in the Linux mode mask, as shown in Table 7-2.

<table>
<thead>
<tr>
<th>Linux Permission</th>
<th>Windows Permission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>Read Data</td>
</tr>
<tr>
<td>Write</td>
<td>Write Data</td>
</tr>
<tr>
<td>Execute</td>
<td>Execute</td>
</tr>
</tbody>
</table>

7.10.5 File Creation, Ownership and Administrative Rights

Only the Windows root user (mapped to UID 0), a member of the root group (mapped to GID 0), or the owner of a file can change the permissions assigned to a file or directory.

If a file or folder already has ACLs assigned on its native system, for example Linux POSIX ACLs, they cannot be changed on the other system. Only limited access is allowed, and no ACL translation is performed.
Only mapped users can create files, so all Windows users who will create IBRIX files must be mapped. Any Windows user not mapped to a Linux UID has read access to the file system but cannot create files or directories.

The mode mask of a new file or directory is initialized as follows.

- Owner is granted read-write-execute permissions.
- Group and Other are granted read permission only if the file’s parent had read permission for these classes of user. In addition, the execute permission is set on directories for Group and Other.

A Linux-like permission schema can be implemented by setting a umask. In the absence of a umask, the mode mask is initialized according to the schema described above.

The mode mask of a file or directory is initialized to 766 or 777 respectively, and then adjusted to the umask.

The umask defines what bits to clear in the target file mode mask. For example a umask of 022 allows for an initial file mode mask permission of 755—rwx for owner, and rx for groups and others. An administrator can restrict it by setting the umask to 077, which clears all bits in the file mode mask for group and others. Setting the umask to 000 allows the maximum permission of 766 for files and 777 for directories.
Chapter 8  Managing File Allocation

8.1 How File Allocation Works ................................................................. 87
8.2 How IBRIX Fusion Evaluates File Allocation Settings ................................ 88
8.3 Using Hostgroups for File Allocation Settings .................................................. 89
8.4 When File Allocation Settings Take Effect on Fusion Clients .......................... 89
8.5 Guidelines for Using File Allocation CLI Commands ................................ 89
8.6 Setting File and Directory Allocation Policies ........................................ 89
8.7 Setting Segment Preferences .................................................................. 90
  8.7.1 Creating a Pool of Preferred Segments ..................................................... 90
  8.7.2 Preferring One Segment for a User or Group .......................................... 90
  8.7.3 Restoring the Default Segment Preference .............................................. 91
8.8 Unpreferring Segments ................................................................................. 91
8.9 Tuning Allocation Policy Settings .............................................................. 92
8.10 Restoring the Default Allocation Policy .................................................... 92
8.11 Listing Allocation Policies .......................................................................... 92

8.1 How File Allocation Works

IBRIX Fusion allocates new files and directories to segments according to the allocation policy and segment preferences that are in effect for a client. An allocation policy is an algorithm that determines the segments that are selected when clients write to an IBRIX file system. Different allocation policies may be set for files and directories. Segment preferences override the allocation policy by designating the specific segments to which new data is written. Table 8-1 lists the available allocation policies.

By default, IBRIX file systems use the RANDOM allocation policy, and all segments are preferred. These defaults are adequate for most cluster environments. In some cases IBRIX Customer Support may recommend that you change the defaults as follows to optimize file storage for your system:

- Change the default file and directory allocation policy.
- Prefer a pool of segments for storing all new files and directories.
- Prefer one segment for storing files and directories created by a specific user or group.
- Declare the first segment where a file or directory will be stored.

Caution! Changing the default allocation policy and segment preferences will alter IBRIX file system storage behavior. Contact IBRIX Customer Support before changing either of these defaults.

The allocation policy and segment preferences are set locally on Fusion Clients, but on the NFS or CIFS server for NFS or CIFS clients. Fusion Clients directly write to Segment Servers, according to the file allocation settings in effect for them. In contrast, NFS and CIFS clients write to an IBRIX file system only via their NFS or CIFS server, therefore the allocation policy and segment preferences that are in effect on the NFS or CIFS server determine the segments where the data is stored.
To choose a segment, a Fusion Client or NFS/CIFS server evaluates the allocation policy and segment preferences that are defined for it. After all segments in the IBRIX file system have reached their *spillover watermark* for file storage, IBRIX Fusion applies the AUTOMATIC policy algorithm to choose a segment that is beneath its *create watermark* setting. This segment is then used continuously until its create watermark is reached, after which IBRIX Fusion again applies the AUTOMATIC policy to choose a different segment that is beneath its create watermark.

### Table 8-1 File Allocation Policies

<table>
<thead>
<tr>
<th>Policy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOMATIC</td>
<td>Sets a no-preference policy, which instructs IBRIX Fusion to pick the strategy to use. In this version of IBRIX, it picks RANDOM, the default policy.</td>
</tr>
<tr>
<td>DIRECTORY</td>
<td>Allocates files to the segment where its parent directory sits.</td>
</tr>
<tr>
<td>LOCAL</td>
<td>Allocates files to a Segment Server’s local segments.</td>
</tr>
<tr>
<td>RANDOM</td>
<td>Allocates files to a randomly chosen segment among preferred segments. This is the default policy.</td>
</tr>
<tr>
<td>ROUNDROBIN</td>
<td>Allocates files to preferred segments in segment order, returning to the first segment (or the designated starting segment) when a file has been allocated to the last segment.</td>
</tr>
<tr>
<td>STICKY</td>
<td>Allocates files to one segment until the segment’s storage limit (set by the segment’s create watermark) is reached, then moves to the next segment as determined by the AUTOMATIC algorithm.</td>
</tr>
<tr>
<td>NONE</td>
<td>Sets directory allocation policy only. Causes the directory allocation policy to revert to its default, which is the policy set for file allocation. Use NONE only to set file and directory allocation to the same policy.</td>
</tr>
</tbody>
</table>

### 8.2 How IBRIX Fusion Evaluates File Allocation Settings

Fusion Clients and NFS/CIFS servers use the following precedence rules to evaluate the file allocation settings that are in effect for them:

- **If the host uses the default file and directory allocation policies and segment preferences:** The RANDOM primary policy is applied, and a segment is chosen from among all segments.

- **If the host uses a non-default file and/or directory allocation policy and the default segment preferences:** Only the file or the directory allocation policy is applied, and a segment is chosen from among all available segments.

- **If the host uses a non-default file and/or directory allocation policy and non-default preferences:** A segment is chosen according to these rules:
  a. If a segment has been preferred for the user or group that created the file/directory, store the file/directory in that segment if there is room, otherwise proceed to the next rule.
  b. From the pool of preferred segments, select a segment according to the allocation policy set for the client and store the file in that segment if there is room. If all segments in the pool are full, proceed to the next rule.
  c. Use the AUTOMATIC algorithm to choose a segment with enough storage room from among all segments and store the file there.
8.3 Using Hostgroups for File Allocation Settings

The Fusion Manager file allocation commands include an argument (-g GROUPLIST) that can take one or more hostgroups. Specifying a hostgroup is a convenient way to configure file allocation settings for a set of Fusion Clients. To configure file allocation settings on all Fusion Clients, specify the clients hostgroup.

8.4 When File Allocation Settings Take Effect on Fusion Clients

Whereas file allocation settings are immediately executed on Segment Servers, for Fusion Clients a file allocation intention is stored in the Fusion Manager. When IBRIX services start on a Fusion Client, the client queries the Fusion Manager for the file allocation settings that it should use, and then it implements them. If IBRIX services are already running on a Fusion Client when you configure new file allocation settings, you can force the client to query the Fusion Manager for the intention by running ibrix_client or ibrix_lwhost --a on the Fusion Client, or by rebooting the Fusion Client.

8.5 Guidelines for Using File Allocation CLI Commands

Follow these guidelines when using the CLI commands to perform any file allocation configuration tasks:

- To perform a task for NFS/CIFS clients, specify NFS/CIFS servers for the -h HOSTLIST argument.
- To perform a task for Fusion Clients, either specify individual Fusion Clients for the -h HOSTLIST argument or specify a hostgroup for the -g GROUPLIST argument.

8.6 Setting File and Directory Allocation Policies

You can set a non-default file or directory allocation policy for NFS/CIFS servers and Fusion Clients. Before using the CLI commands to do this, see the guidelines on page 89.

You can specify the first segment where the policy should be applied, but in practice this is useful only for the ROUNDROBIN policy.

Allocation policy names are case-sensitive, and must be entered as uppercase letters (for example, RANDOM).

**Note!** If your file and directory allocation policies are different and you want to make them the same, first set a new file allocation policy and then set the directory allocation policy to NONE. The directory allocation policy then assumes the value that is set for the file allocation policy. This is the default directory allocation policy in effect until it is explicitly changed.

- **Set a file allocation policy using the Fusion Manager CLI**

  `<ibrixhome>/bin/ibrix_fs_tune -f FSNAME { -h HOSTLIST | -g GROUPLIST } -p POLICY [-S STARTSEGNUM]`

  For example, to set the ROUNDROBIN policy on NFS-exported IBRIX file system ifs1 on Segment Server s1.ibrix.com for files only, starting at segment ilv1:

  `<ibrixhome>/bin/ibrix_fs_tune -f ifs1 -h s1.ibrix.com -p ROUNDROBIN -S ilv1`
8.7 Setting Segment Preferences

There are two ways to prefer segments for Fusion Clients, NFS/CIFS servers, or hostgroups:

- Prefer a pool of segments for the clients to use.
- Prefer a single segment for files created by a specific user or group on the clients.

Both methods may be in effect at the same time. For example, you may prefer a segment for a user, and then prefer a pool of segments for the clients on which the user will be working.

8.7.1 Creating a Pool of Preferred Segments

A segment pool can consist of individually selected segments, all segments local to an NFS/CIFS server, or all segments. Clients will apply the allocation policy that is in effect for them to choose a segment from the segment pool.

**Note!** All segments are always created in the preferred condition. You might want to have some segments preferred and others unpreferred, and if so, first pick a single segment and prefer it. This action unprefers all other segments, and once they are unpreferred, you can work with the segments one at a time, preferring and unpreferring as required. Also note that by design, the system cannot have zero preferred segments. If you have only one segment preferred and unprefer it, all segments become preferred.

When preferring multiple pools of segments (for example, one for Fusion Clients and one for NFS/CIFS clients), make sure that no segment appears in both pools.

**Create a pool of preferred segments**

Use the first command to specify the pool by logical volume name (*LVNAMELIST*) and the second to specify by segment number (*SEGNUMLIST*). Use the second command and the *LOCAL* keyword to create a pool of all segments on NFS/CIFS servers. Use it with the *ALL* keyword to restore the default segment preferences (for more information see page 91).

```
<ibrixhome>/bin/ibrix_fs_tune -f FSNAME {-h HOSTLIST|-g GROUPLIST} -s LVNAMELIST
<ibrixhome>/bin/ibrix_fs_tune -f FSNAME {-h HOSTLIST|-g GROUPLIST} -S {SEGNUMLIST|ALL|LOCAL}
```

8.7.2 Preferring One Segment for a User or Group

When preferring segments, do not assign more than one segment to the same user or group.

**Prefer a segment for a user**

```
<ibrixhome>/bin/ibrix_fs_tune -A -U UID -f FSNAME -s LVNAME {-h HOSTLIST|-g GROUPLIST}
```
Chapter 8 Managing File Allocation

8.7.3 Restoring the Default Segment Preference

The default is for all IBRIX file system segments to be preferred.

- Prefer a segment for a group

```
<ibrixhome>/bin/ibrix_fs_tune -A -U UID -f FSNAME -S SEGMENT_NUMBER { -h HOSTLIST | -g GROUPLIST }
```

8.8 Unpreferring Segments

IBRIX Fusion disregards unpreferred segments when it applies the allocation policy and segment preferences to choose a segment for a new file. Therefore, if you add new segments because existing preferred segments have filled up, you can improve performance by unpreferring the filled segments and preferring the new ones.

- Unpreferring segments

```
<ibrixhome>/bin/ibrix_fs_tune -u -f FSNAME { -h HOSTLIST | -g GROUPLIST }
```

**Note!** All segments are always created in the preferred condition. You might want to have some segments preferred and others unpreferred, and if so, first pick a single segment and prefer it. This action unprefers all other segments, and once they are unpreferred, you can work with the segments one at a time, preferring and unpreferring as required. Also note that by design, the system cannot have zero preferred segments. If you have only one segment preferred and unprefer it, all segments become preferred.

- Unprefer a pool of segments

```
<ibrixhome>/bin/ibrix_fs_tune -u -f FSNAME { -h HOSTLIST | -g GROUPLIST } -s lvnamelist
```

- Unprefer a preferred segment for a user

```
<ibrixhome>/bin/ibrix_fs_tune -D -U UID -f FSNAME { -h HOSTLIST | -g GROUPLIST }
```

- Unprefer a preferred segment for a group

```
<ibrixhome>/bin/ibrix_fs_tune -D -U GID -f FSNAME { -h HOSTLIST | -g GROUPLIST }
```
8.9 Tuning Allocation Policy Settings

To optimize system performance you may want to globally change the following allocation policy settings for an IBRIX file system:

- File allocation policy (see Table 8-1).
- Starting segment number for applying changes to any of the preceding.
- Preallocation: Number of KB to preallocate for files. The default is a preallocated file size of 256 KB.
- Readahead: Number of KB in a file to pre-fetch. The default is 128 KB.
- NFS readahead: Number of KB in a file to pre-fetch on NFS systems. The default is 128 KB.

**Note!** Contact IBRIX Customer Support before tuning allocation policy settings, for guidance on selecting the best values for your installation.

**Tune allocation policy settings**

```
<ibrixhome>/bin/ibrix_fs_tune -f FSNAME -O [-p POLICY]
  [-S STARTSEGNUM] [-P prealloc (KB)] [-r readahead (KB)]
  [-N NFS readahead (KB)]
```

8.10 Restoring the Default Allocation Policy

If you have changed the file allocation policy for an IBRIX file system, you can restore it to the default value.

**Restore the default file allocation policy**

```
<ibrixhome>/bin/ibrix_fs_tune -f FSNAME{-h HOSTLIST|-g GROUPLIST} -p -U
```

8.11 Listing Allocation Policies

You can list the policies in effect for users, groups, or IBRIX file systems.

```
<ibrixhome>/bin/ibrix_fs_tune -l {u|g|f}
```

The arguments display policies for users (u), groups (g), or IBRIX file systems (f)
Chapter 9 Servicing IBRIX Fusion

9.1 Viewing IBRIX Fusion Software Version Numbers

The following output fields should show matching version numbers, unless you have installed special releases or patches:

- Fusion Manager version and IAD/IAS
- FILE SYSTEM and IAD/FS

If you discover mismatched version numbers, and you do not know of any reason for the mismatch (such as special releases or patches that you have installed), contact IBRIX Customer Support. A mismatch may or may not affect the operation of your cluster.

**Determine version numbers**

```
<ibrixhome>/bin/ibrix_version -l [-h HOSTLIST]
```

For each host, the output includes:

- Version number of the installed IBRIX file system.
- Version numbers of the IBRIX Administrative Daemon (IAD) and IBRIX File System module.
- Version numbers for the operating system and the OS kernel.
- Processor architecture.

To determine version numbers for all Segment Servers, include `-S`:

```
<ibrixhome>/bin/ibrix_version -l -S
```

To determine version numbers for all Fusion Clients, include `-C`. 
9.2 Viewing Your IBRIX Fusion License Terms

The IBRIX Fusion license file is stored in the `<ibrixhome>` directory on the Fusion Manager.

- **View license terms**

  `<ibrixhome>/bin/ibrix_license -i`

  **Sample Output**

  Support Number : IBRIX-10000-001
  Site Number : 36578
  Clients Limit : 1000
  Segment Server Limit : 12
  Support Start Date : ja-10-2007
  License Expiration Date : ja-10-2008
  Days Left : 200
  Features : quotas,migrate,failover
  Active : Yes
  Clients Count : 1000
  Segment Server Count : 12

9.3 Changing the IBRIX User Password

The IBRIX user password is created during installation. This is the password users enter to log on to the Fusion Manager GUI. This password can be changed on the Fusion Manager with the Linux `passwd` command. Enter the command below. You will be prompted to enter the new password.

- **Change the GUI user password using Linux passwd**

  # passwd ibrix

9.4 Viewing Disk Space Information for an IBRIX File System

Because an IBRIX file system is distributed among segments on many Segment Servers, disk space utilities such as `df` must be provided with collated disk space information about each Segment Server that holds segments in a given IBRIX file system. The Fusion Manager periodically collects this information and collates it for `df`.

IBRIX provides a disk space utility for clients, `ibrix_df`, which provides utilization data for an entire file system, or for file-only or directory-only segments in a data-segregated file system.

- **Check disk space utilization on a Linux Fusion Client**

  `<ibrixhome>/bin/ibrix_df [-F|-D]`

  Use the `-F` argument to see data for file-only segments and the `-D` argument to see data for directory-only segments. Output fields are defined in Table 9-1.
9.5 Viewing Logs

IBRIX Fusion provides logs for the Fusion Manager, Segment Servers, and Fusion Clients. Contact IBRIX Customer Support for assistance in interpreting log files. You may be asked to tar the logs and email them to IBRIX.

9.6 Viewing Operating Statistics for Segment Servers

At periodic intervals the Segment Servers report the following types of statistics to the Fusion Manager:

- Summary: General operational statistics, including CPU usage, disk throughput, network throughput, and operational state. To view information on the possible operational states, see Table 3-1 on page 24.
- IO: Aggregate statistics about reads and writes.
- Network: Aggregate statistics about network inputs and outputs.
- Memory: Statistics about available total, free, and swap memory.
- CPU: Processor and CPU activity statistics.
- NFS: Statistics about NFS client and server activity.

View statistics

<ibrixhome>/bin/ibrix_stats -1 [-s] [-c] [-m] [-i] [-n] [-f] [-h HOSTLIST]

Specify any of the following arguments to view that type of statistics, or omit all arguments to view all statistics:

- \(-s\): Summary statistics
- \(-c\): CPU statistics
- \(-m\): Memory statistics
- \(-i\): I/O statistics
- \(-n\): Network statistics
- \(-f\): NFS statistics

---

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>File system name.</td>
</tr>
<tr>
<td>BLOCKS</td>
<td>Number of blocks in the file system or segment.</td>
</tr>
<tr>
<td>BFREE</td>
<td>Number of unused blocks of storage or segment.</td>
</tr>
<tr>
<td>BAVAIL</td>
<td>Number of blocks available for user files or segment.</td>
</tr>
<tr>
<td>USED PERCENT</td>
<td>Percentage of total storage occupied by user files.</td>
</tr>
<tr>
<td>FILES</td>
<td>Number of files that can be created in this file system or segment.</td>
</tr>
<tr>
<td>FFREE</td>
<td>Number of unused file inodes in system or segment.</td>
</tr>
</tbody>
</table>
9.7 Starting and Stopping Processes

You can start, stop, and restart and display status for the processes that perform internal IBRIX Fusion functions on a machine. The following commands also control the operation of PostgreSQL on the machine. The PostgreSQL service is available at /usr/local/ibrix/init/.

- **Start and stop processes and view process status on the Fusion Manager**
  
  Execute the following command on the Fusion Manager:
  
  `/etc/init.d/ibrix_fusionmanager [start | stop | restart | status]`

- **Start and stop processes and view process status on a Segment Server**
  
  Execute the following command on the Segment Server:
  
  `/etc/init.d/ibrix_server [start | stop | restart | status]`

There are situations where a follow-up action is required after stopping and starting, or restarting, a Segment Server.

- **Start and stop processes and view process status on a Fusion Client**
  
  Execute the following command on the Fusion Client:
  
  `/etc/init.d/ibrix_client [start | stop | restart | status]`

9.8 Managing Long-Running Jobs

Use the ibrix_jobs command to manage long-running jobs on IBRIX file systems, for example rebalancing and replication, and to list information about them.
Stopping a job poses no risks for an IBRIX file system. The Fusion Manager completes all operations in process when the command starts. Because of this, depending on when a job is stopped, the number of files in a segment may be different from the count in that segment before the run began. You can view information about long-running jobs, such as rebalancing or replication, and stop them if you need to prioritize other jobs on the Segment Servers.

**List jobs**

```bash
<ibrixhome>/bin/ibrix_jobs -l [-f FSLIST]
```

For each job, the output lists the file system where the job is running, the job type (for example, rebalance) and the job name. Running the command with `-i` instead of `-l` breaks the job information into jobs and subjobs, where the subjobs indicate machines in the job. For example:

```bash
<ibrixhome>/bin/ibrix_jobs -i ifs0
FileSystem: ifs0

<table>
<thead>
<tr>
<th>JOB</th>
<th>RUNNING?</th>
</tr>
</thead>
<tbody>
<tr>
<td>ifs0_rebalance_20070508_171335</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SUBJOB</th>
<th>RUNNING?</th>
</tr>
</thead>
<tbody>
<tr>
<td>lab13-11_rebalance_20070508_171335</td>
<td>Yes</td>
</tr>
<tr>
<td>lab13-22_rebalance_20070508_171335</td>
<td>Yes</td>
</tr>
</tbody>
</table>
```

**Stop jobs**

```bash
<ibrixhome>/bin/ibrix_jobs -k -j JOBLIST [-F]
```

To force jobs to stop, include the `-F` argument.

**Pause or stop a CFR job**

A continuous file replication (CFR) job can be paused or killed with variations of the same command. To pause:

```bash
<ibrixhome>/bin/ibrix_cfrjob -p -f FSNAME [-j JOBID]
```

Pauses a continuous replication or run-once job on the named source file system. The `JOBID` parameter identifies a specific job to pause. To obtain job ids, run `ibrix_cfrjob -l`; the ids are the first column in the output table. You can omit the `JOBID` parameter to pause a continuous replication job on the named file system as only one such job can be running at a time, but you must include it to pause a run-once job.

To stop a CFR job, run the same command but with `-k` in place of `-p`.

### 9.9 Keeping Segment Server and Configuration Database Information Consistent

A Segment Server’s information about an IBRIX file system must remain current in order to maintain access to the file system. IBRIX recommends that you execute `ibrix_health` on a regular basis to monitor the health of this information. If the information becomes outdated on a Segment Server, execute `ibrix_dbck -o` to resynchronize the server’s information with the configuration database. For information on `ibrix_health`, see page 26.

**Run Segment Server health check**

```bash
<ibrixhome>/bin/ibrix_health -i -h HOSTLIST
```
If the last line of the output reports Passed, IBRIX file system information on the Segment Server and Fusion Manager is congruent.

## Repair Segment Server information

```
<ibrixhome>/bin/ibrix_dbck -o -f FSNAME [-h HOSTLIST]
```

To repair information on all Segment Servers, omit the `-h HOSTLIST` argument.

### 9.10 Checking and Repairing IBRIX File Systems

**Caution!** Do not run `ibrix_fsck` in corrective mode without the direct guidance of an IBRIX Customer Support representative. If run improperly, the command can cause data loss and IBRIX file system damage.

**Caution!** Do NOT run `e2fsck` (or any other off-the-shelf fsck program) on an any part of an IBRIX file system. If you do, you will damage the file system!

The `ibrix_fsck` command can detect and repair IBRIX file system inconsistencies, which are a symptom of file system corruption. Corruption can occur for many reasons, including abnormal shutdown due to hardware failure, a power failure, switching off the system without proper shutdown, and a failed migration.

The command runs in four phases and has two running modes, analytical and corrective. You must run the phases in order, without skipping any, and you must run all of them:

- **Phase 0** checks host connectivity and the consistency of the segment byte blocks and repairs them in corrective mode.
- **Phase 1** checks segments and repairs them in corrective mode. Results are stored in `<home>/web/fsck`.
- **Phase 2** checks the file system and repairs it in corrective mode. Results are also stored in `<home>/web/fsck`.
- **Phase 3** moves files from `lost+found` on each segment to the global `lost+found` directory on the root segment for the file system.

If an IBRIX file system is showing evidence of corruption, contact IBRIX Customer Support to discuss the situation. A representative will ask you to run `ibrix_fsck` in analytical mode and, based on the output, will recommend a course of action, including assistance in running the command in corrective mode.

IBRIX strongly recommends that you use corrective mode only with the direct guidance of an IBRIX Customer Support representative. Corrective mode is complex and difficult to run safely. Using it improperly can damage both data and the file system. Analytical mode is completely safe, by contrast.

**Note!** During an `ibrix_fsck` run, an INFSCK flag is set on the IBRIX file system to protect it. If an error occurs during the run, you must explicitly clear the INFSCK flag as described below, or you will be unable to mount the file system.

### Analyze the integrity of an IBRIX file system on all segments

1. Turn off automated failover (see Section 4.5.3), unmount all NFS clients and stop NFS, and unmount the IBRIX file system.
2. Run the following commands one time each:
   
   ```
   <ibrixhome>/bin/ibrix_fsck -f FSNAME -p 0 [-s LVNAME]
   <ibrixhome>/bin/ibrix_fsck -f FSNAME -p 1 [-s LVNAME]
   [-B BLOCKSIZE] [-b ALTSUPERBLOCK]
   <ibrixhome>/bin/ibrix_fsck -f FSNAME -p 2 -m MOUNTPOINT
   ```

   ![](image)
9.11 Backing Up the IBRIX Fusion Configuration

Backups of the IBRIX Fusion Manager configuration file and database are needed to recover from Fusion Manager or Segment Server failures, or to undo an unwanted configuration changes. An autobackup utility on the Fusion Manager, controlled by the `ibrix_monitor` command, maintains the backups of these files.

Whenever a change is made to the configuration file or database, the Fusion Manager creates the necessary backup file. The location of the backups, and the number of backup files to keep, are set with `ibrix_monitor`.

Current backup settings can be viewed by running `ibrix_monitor -l`.

The autobackup defaults are:

- Backup host: localhost
- Backup path: `/var/lib/ibrix/backup`
- Number of backups to keep: 10

When the autobackup utility is turned on, it creates backups according to its configuration settings. You can force an immediate back up at any time.

Configure the autobackup utility

1. Specify the backup host:
   `<ibrixhome>/bin/ibrix_monitor -h HOSTNAME`
2. Specify the backup directory:
   `<ibrixhome>/bin/ibrix_monitor -p BACKUP_PATH`
3. Specify the maximum number of backup files to keep:
   `<ibrixhome>/bin/ibrix_monitor -k KEEP`

Turn the autobackup utility on or off

`<ibrixhome>/bin/ibrix_monitor -c [start|stop|on|off]`

Force a backup

`<ibrixhome>/bin/ibrix_monitor -f`

9.12 Recovering the Fusion Manager

It is necessary to recover the Fusion Manager in two situations:

- The Fusion Manager fails and must be replaced. The database and a configuration must be restored. In this situation, bring your backup Fusion Manager on line or install the Fusion Manager software on a replacement Fusion Manager and then perform the recovery procedure described below to restore backup files to the replacement machine.
- If you install the Fusion Manager software on a new machine, be sure to assign the name and IP address of the failed Fusion Manager to the new machine.
• You have made a configuration change that you want to roll back. Only the configuration must be restored. In this situation, perform the recovery procedure on the existing Fusion Manager to restore the backup created immediately prior to the configuration change.

You cannot start the recovery process without backups being made ahead of time. We recommend you enable automatic backups on the Fusion Manager; refer to Section 9.11, Backing Up the IBRIX Fusion Configuration for information on enabling automatic backups and the ibrix_monitor command.

Note! The replacement Fusion Manager must be the same IBRIX version as the failed Fusion Manager. The recovery process does not take the place of an upgrade.

IBRIX Fusion uses two files for backups, one for the configuration and one for the database. The ibrix_recover command must be run twice, each time restoring a different file. Backup files and their corresponding checksum files are stored in a user-selected directory location with a similar autogenerated name, and when copying the files to the replacement machine, be sure to copy four files:

• A configuration file (fm_config_filename.tgz) and its corresponding checksum file (fm_config_filename.tgz.md5sum).
• A database file (fm_database_filename.sql) and its corresponding checksum file (fm_database_filename.sql.md5sum).

Caution! Recovering the Fusion Manager can result in data loss if improperly performed. Contact IBRIX Customer Support for assistance in performing the recovery procedure.

Recover the Fusion Manager

1. Install the Fusion Manager software on a replacement machine.
2. Make sure that you have root access to the Fusion Manager.
3. Stop all IBRIX services on the replacement Fusion Manager:
   
   ```
   # /etc/init.d/ibrix_fusionmanager stop
   ```
4. Copy the four backup files to a local directory on the replacement Fusion Manager, or make them available over the network.
5. Run ibrix_recover from the <ibrixhome>/sbin directory on the replacement Fusion Manager. When prompted, enter the path name of the backup file, either fm_config_filename.tgz or fm_database_filename.sql.
6. Run ibrix_recover a second time to recover the second database file. When prompted, enter the path name of the second backup file.
7. Start IBRIX services:
   
   ```
   # /etc/init.d/ibrix_fusionmanager start
   ```
8. If quotas were in effect before running the recovery process, restore them with:
   
   ```
   # <ibrixhome>/bin/ibrix_edquota -r
   ```

   Recommended: Set up a new standby Fusion Manager, preferably a hot-spare.

9.13 Recovering a Segment Server

If a Segment Server fails, you will need to contact IBRIX Customer Support to obtain a replacement machine and then perform the following recovery procedure on the replacement machine. The procedure involves installing the Segment Server software on a replacement machine and then copying the local configuration data from the Fusion Manager to the replacement.

Note! The Segment Server software installation will fail. This is expected behavior. Continue with the procedure to recover the Segment Server.
Caution! Recovering a Segment Server can result in damage to IBRIX file systems. Please contact IBRIX Customer Support before executing this procedure.

Recover a Segment Server

1. Install the Segment Server software on the replacement machine (for installation instructions, see the IBRIX Fusion Installation and Upgrade Guide). The installation will fail and display the error message, `seg server is already registered with FusionManager`. Ignore this error message and continue with the procedure.

2. Regenerate the `/etc/ibrix/iadconf.xml` file:
   ```
   <ibrixhome>/bin/register_server -p FusionManagerName -c ClusterIF -x
   ```

3. Start IBRIX services:
   ```
   /etc/init.d/ibrix_server start
   ```

4. On the Fusion Manager, run:
   ```
   <ibrixhome>/bin/ibrix_fm -r -h SegmentServerName
   ```

5. Restart `chkconfig`:
   ```
   chkconfig --add ibrix_server
   ```

6. Set up email notifications, if any were set up prior to the recovery, with `ibrix_event`. 
Chapter 10 Troubleshooting

10.1 Problems with IBRIX Services

10.1.1 Cannot Start Services on the Fusion Manager or a Segment Server

SELinux may be enabled. To determine the current state of SELinux, execute `getenforce`. If it returns `enforcing`, disable SELinux with either of these commands:

```
setenforce Permissive
setenforce 0
```

To permanently disable SELinux, edit its configuration file (`/etc/selinux/config`) to set `SELINUX=parameter` to either `permissive` or `disabled`. SELinux will be stopped at the next boot.

10.1.2 Cannot Start Services on a Linux Fusion Client

There are two possible causes:

1. SELinux is enabled. See the preceding topic for the fix.
2. The Fusion Client is not registered with the Fusion Manager. For information on registering Fusion Clients, see page 28.
10.2 Problems with the IBRIX File System

10.2.1 ibrix_pv -a Discovers Too Many or Too Few Devices

This happens when Segment Servers see devices multiple times. To prevent this, modify the LVM2 filter in /etc/lvm/lvm.conf to filter only on the devices IBRIX Fusion uses, which will change the output of lvmdiskscan.

By default, this filter finds all devices:

```
filter = [ "a/.*/" ]
```

The following filter specifies all sd devices:

```
filter = [ "a|^/dev/sd.*|", "r|^.*|" ]
```

Contact IBRIX Customer Support if you need assistance in modifying this line.

10.2.2 Cannot Mount on a Fusion Client

Check whether the IBRIX file system is mounted and functioning on the Segment Servers.

Check whether the mountpoint exists on the Fusion Client. If not, create the mountpoint locally on the client.

Verify that Fusion services have been started on the Fusion Client (see page 96).

10.2.3 NFS Clients Cannot Access an Exported IBRIX File System

An exported file system has become unmounted from one or more Segment Servers, causing IBRIX Fusion to automatically disable NFS on those servers. Fix the reason for the unmount, then remount the file system.

10.2.4 User Quota Usage Data Is Not Updating

Restart the quota monitor service to force a read of all quota usage data and updates usage counts to the Segment Servers in your cluster:

```
<ibrixhome>/init/ibrix_qm restart
```

10.3 Failover Problems

10.3.1 Cannot Failback from Failover Caused by Storage Subsystem Failure

When a storage subsystem fails and automated failover is turned on, the Fusion Manager will initiate its failover protocol: It will update the configuration database to record that segment ownership has transferred from primary servers to their standbys, and then it will attempt to migrate the segments to the standbys. However, no segments can migrate because neither the primary servers nor the standbys can access system storage devices, and so the failover is arrested.

Perform the following manual recovery procedure in this situation:

1. Restore the failed storage subsystem (for example, replace failed fibre-channel switches or replace a LUN that had been removed from the storage array).
2. Reboot the standby servers, which will allow the arrested failover to complete.
10.3.2 Cannot Failback Owing to a Storage Subsystem Failure

This problem is similar to the preceding one. If a storage subsystem fails after you have initiated a failback, the configuration database will record that the failback occurred even though segments never migrated back to the primary host. If you execute `ibrix_fs -i -f FSNAME`, the output will list “No” in the ONBACKUP field, indicating that the primary server now owns the segments, even though it does not. In this situation, you will be unable to complete the failback after you fix the storage subsystem problem.

Perform the following manual recovery procedure in this situation:

1. Restore the failed storage subsystem.
2. Reboot the primary server, which will allow the arrested failback to complete.

10.3.3 Fusion Client I/O Errors Following Segment Migration

Following successful segment migration to a different Segment Server, the Fusion Manager sends all Fusion Clients an updated map reflecting the changes, which enables the clients to continue I/O operations. If, however, the network connection between a Fusion Client and the Fusion Manager is not active, the client cannot receive the updated map, resulting in client I/O errors.

To remedy this problem, restore the network connection between the involved Fusion Clients and the Fusion Manager.

10.3.4 Recovering Fusion Manager High Availability When a Cable Is Disconnected

Note! IBRIX does not recommend a Fusion Manager High Availability configuration in which the IBRIX cluster network, Linux heartbeat program, and power source controller (for example, IPMI) all run over a single interface. However, if you have such a configuration and the cable servicing that interface is disconnected, follow these steps to recover.

Fusion Manager High Availability is designed to be used with IPMI on a separate network connection from the heartbeat connection. When using a single network connection, the behavior is unpredictable when the network cable is disconnected. When this occurs, neither system is reliably able to know the state of the other. Both Fusion Managers will attempt to shut down the other, but since they cannot communicate through IPMI, nothing will happen.

When the cable is plugged back in and the network connection is re-established, both systems want to be the primary and both systems attempt to shut down the other. One or the other should succeed, and in some cases, both with succeed and power each other down. To recover from this:

1. Power up the system that went down, or power up both systems if they both went down.
2. Put Fusion Manager High Availability into administrative mode on the machine or machines in Step 1:
   `<ibrixhome>/bin/ibrix_fmha -a`
3. Take the machine(s) out of administrative mode
   `<ibrixhome>/bin/ibrix_fmha -A`
4. If the systems do not recover through administrative mode, implement a recovery by reinstalling Fusion Manager High Availability with the current settings:
   `<ibrixhome>/fm-ha./install_fmha -c -y`
10.4 Windows Fusion Client Problems

10.4.1 Logged in But Getting a “Permission Denied” Message

The client cannot access the Active Directory server because you did not specify the domain name. Reconfigure the Active Directory settings, specifying the domain name (see page 80).

10.4.2 Verify Button in the Active Directory Settings Tab Does Not Work

This problem has the same cause as the preceding one.

10.4.3 Mounted Drive Does Not Appear in Windows Explorer

To make a drive appear in the Explorer, after mounting it log off, then log on, or reboot the machine. You can also open a DOS command window and access the drive manually.

10.4.4 Mounted Drive Not Visible When Using Terminal Server

Refresh the Explorer’s view of the system by logging off and then on.

10.4.5 Fusion Client Auto-Startup Interferes with Debugging

The Fusion Client is set to start automatically, which may interfere with debugging a Windows Fusion Client problem. To prevent this, reboot the machine in safe mode and change the Windows Fusion Client service mode to manual, so you can reboot without starting the Fusion Client:

1. Open the Services control manager (Control Panel > Administrative Tools > Services).
2. Right-click IBRIX Client Services and select Properties.
3. Change the startup type to Manual, then click OK.
4. Debug the client problem. When finished, switch the Windows Fusion Client service back to automatic startup at boot-time by repeating these steps and changing the startup type to Automatic.