Device Mapper Multipath Configuration Guide for HP StorageWorks Arrays

Second Edition

Part number: AA-RW8RB-TE
Second edition: December 2006

Legal and notice information

Hewlett-Packard Company makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The information contained in this document is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Hewlett-Packard Company shall not be liable for technical or editorial errors or omissions contained herein. The information is provided “as is” without warranty of any kind and is subject to change without notice. The warranties for Hewlett-Packard Company products are set forth in the express limited warranty statements for such products. Nothing herein should be construed as constituting an additional warranty.

Device Mapper Multipath Configuration Guide for HP StorageWorks Arrays

Contents

Tables

About this guide

This guide describes the Device Mapper Multipath software and provides information to help you:

- Determine hardware and software prerequisites
- Install Device Mapper Multipath software
- Use and manage Device Mapper Multipath software

NOTE:

This second edition of Device Mapper Multipathing for HP StorageWorks Arrays inherits all the features available in the first edition and offers a range of new features and OS support. For more information, see whats new.

Intended audience

This document is intended for users who are responsible for installing, configuring, and maintaining Device Mapper Multipath in their Linux server environment, assuming that you are familiar with Linux system administration, including hardware and software installation.

Document conventions and symbols

Table 1 Document conventions

Convention	Element
Medium blue, underlined text (http://www.hp.com)	Web site addresses
Bold font	• Key names • Text typed into a GUI element, such as into a box • GUI elements that are clicked or selected, such as menu and list items, buttons, and check boxes
Italic font	Text emphasis
`Monospace font`	• File and directory names • System output • Code • Text typed at the command line
`Monospace, italic font`	• Code variables • Command-line variables
`Monospace, bold font`	Emphasis of file and directory names, system output, code, and text typed at the command line

IMPORTANT:

Provides clarifying information or specific instructions.

NOTE:

Provides additional information.

HP technical support

Telephone numbers for worldwide technical support are listed on the HP support web site: http://www.hp.com/support/.

Collect the following information before calling:

- Technical support registration number (if applicable)
- Product serial numbers
- Product model names and numbers
- Applicable error messages
- Operating system type and revision level
- Detailed, specific questions

For continuous quality improvement, calls may be recorded or monitored.

HP recommends that customers sign up online using the Subscriber's choice web site: http://www.hp.com/go/e-updates.

- Subscribing to this service provides you with e-mail updates on the latest product enhancements, newer versions of drivers, and firmware documentation updates as well as instant access to numerous other product resources.
- After signing up, you can locate your products by selecting **Business support** and then **Storage** under Product Category.

Helpful web sites

For additional product information, see the following web sites:

- http://www.hp.com
- http://www.hp.com/country/us/eng/prodserv/storage.html
- http://www.hp.com/support/
- http://www.docs.hp.com
- http://christophe.varoqui.free.fr/
- http://christophe.varoqui.free.fr/multipath.html
- http://sourceware.org/dm/

1 Abbreviations

Table 2 lists the abbreviations and their definitions used in this document

Table 2 Abbreviations and their definitions

Abbreviations/Acronyms	Definition
DM	Device Mapper
GUI	Graphical User Interface
HBA	Host Bus Adapter
I/O	Input/Output
LUN	Logical Unit Number
LVM	Logical Volume Manager
CLI	Command Line Interpreter
OS	Operating System
RAID	Redundant Array of Independent (or Inexpensive) Disks
RHEL	Red Hat Enterprise Linux
SAN	Storage Area Network
SLES	SuSE LINUX Enterprise Server
UID	Unique Identifier
WWID	World-Wide Identifier
WWN	World-Wide Name
FC	Fibre Channel
SCSI	Small Computer System Interface

2 Introduction to Device Mapper Multipath

This chapter discusses the following:

- Overview
- Device Mapper Multipath Operation
- Software components
- Features

NOTE:

This second edition of Device Mapper Multipath can be used on limited environments. Do not use Device Mapper Multipath for mission critical applications. Caution should be exercised when implementing any new software into the production environment. At this time, Device Mapper Multipath is only recommended for test environments and non-mission critical applications.

Only the features described in this document are supported with this edition of Device Mapper Multipath. Future editions will provide additional functionality and usability.

Overview

Device Mapper is a new infrastructure in Linux 2.6 kernel that provides a generic way to create virtual layers of block devices. It supports stripping, mirroring, snapshots, concatenation and multipathing. The multipathing feature is provided in 2.6 kernel by the combination of Device Mapper Multipath kernel modules and multipath-tools user-space package.

Device Mapper Multipath operation

Device Mapper Multipath allows hosts to route I/O over the multiple paths available to an end storage unit (LUN). A path refers to the connection from an HBA port to a storage controller port. When an active path through which I/O happens fails, Device Mapper Multipath reroutes the I/O over other available paths. In a Linux host, when there are multiple paths to a storage controller, each path appears as a separate block device and hence results in multiple block devices for single LUN. Device Mapper Multipath creates a new Multipath block device for those devices having the same LUN WWN. For example, a host with two HBAs attached to a storage controller with two ports via a single unzoned FC switch, sees four block devices: `/dev/sda, /dev/sdb, /dev/sdc`, and `/dev/sdd`. Device Mapper Multipath creates a single block device, `/dev/mpath/mpath1` that reroutes I/O through those four underlying block devices.

Software components

This section discusses the software components for Device Mapper Multipath support:

- **dm-multipath kernel module** – Routes I/O and does failover to paths and path groups
- **multipath configuration tool** – Provides commands to configure, list, and flush Multipath devices. The command is run in `rc.sysinit` during startup, and by `udev`, whenever a block device is added

- **multipathd daemon** -- Monitors paths to check if faulty paths have been fixed. When paths revert, `multipathd` may also initiate path group switches to ensure that the optimal path group is being used. Also, it is possible to interactively modify a multipath device
- **kpartx utility** – Creates Device Mapper Multipath devices for partitions on a device. You must use this command for MS DOS based partitions with Device Mapper Multipath

Features

This second edition of Device Mapper Multipath supports features and OS that were supported in the first edition and also adds new features and OS support.

Device Mapper Multipath provides the following features:

- Allows the multivendor Storage RAID systems and host servers equipped with multivendor Host Bus Adaptors (HBAs) redundant physical connectivity along the independent Fibre Channel fabric paths available
- Monitors each path and automatically reroutes (failover) I/O to an available functioning alternate path, if an existing connection fails
- Provides an option to perform failback of the LUN to the repaired paths
- Implements failover or failback actions transparently without disrupting applications
- Monitors each path and notifies if there is a change in the path status
- Facilitates I/O load balancing among the multiple paths
- Provides CLI with display options to configure and manage Multipath features
- Provides all Device Mapper Multipath features support for any LUN newly added to the host
- Provides an option to have customized names for the Device Mapper Multipath devices
- Provides persistency to the Device Mapper Multipath devices across reboots if there are any change in the Storage Area Network
- Provides policy based path grouping for the user to customize the I/O flow through specific set of paths

What's new?

Device Mapper Multipath provides the following additional features as part of this second edition:

- Provides support for the latest OS distributions from RedHat and Novell
- Provides support for the HP StorageWorks Arrays (EVA/XP Arrays) with latest firmware revisions
- Provides larger LUN configuration on the hosts (except RHEL4 U3)
- Provides multiple array/host support in a SAN
- Provides online LUN addition support
- Provides online LUN deletion support
- Provides regular expression support for the vendor/product information to configure multipathing in `multipath.conf` file

NOTE:

Caution to be exercised while editing the `multipath.conf` file with the latest parameters for this edition. For the correct parameters, see Configuring Device Mapper Multipath for HP StorageWorks Arrays.

3 Managing Device Mapper Multipath for HP StorageWorks Arrays

This chapter discusses the following:

- Preparing for installation
- Device Mapper Multipath support matrix
- Setting up Device Mapper Multipath for HP StorageWorks Arrays
- Configuring Device Mapper Multipath for HP StorageWorks Arrays
- Starting Device Mapper Multipath for HP StorageWorks arrays
- Using Device Mapper Multipath for HP StorageWorks arrays

NOTE:

This second edition of Device Mapper Multipath can be used on limited environments. Do not use Device Mapper Multipath for mission critical applications. Caution should be exercised when implementing any new software into a production environment. At this time, Device Mapper Multipath is only recommended for test environments and non-mission critical applications.

Only the features described in this document are supported with this edition of Device Mapper Multipath. Future editions will provide additional functionality and usability.

Preparing for Device Mapper Multipath configuration

Before configuring the Device Mapper Multipath, verify that your system components support Device Mapper Multipath and if you need to upgrade the hardware or software.

To prepare your system for the installation of Device Mapper Multipath:

- Ensure that the Device Mapper and Device Mapper Multipath packages available with OS distributions are installed in the host system and verify that their versions match as per the following information:
  - **For RHEL4 U3:**

    ```
    device-mapper-1.02.02-3.0.rpm
    device-mapper-multipath-0.4.5-12.0.RHEL4.rpm
    ```
  - **For RHEL4 U4:**

    ```
    device-mapper-1.02.07-4.0.RHEL4.rpm
    device-mapper-multipath-0.4.5-16.1.RHEL4.rpm
    ```
  - **For SLES9 Service Pack 3 - kernel errata 2.6.5-7.267:**

    ```
    device-mapper-1.01.01-1.6.i586.rpm
    multipath-tools-0.4.5-0.16.i586.rpm or later
    ```

NOTE:

Upgrade the system to `errata 267` and edit the parameter in the file `/etc/sysconfig/hotplug` to "`HOTPLUG_USE_SUBFS=no`" and rebuild the `initrd` using the `mkinitrd` command.

NOTE:

Download for the `multipath-tools-0.4.5-0.16.i586.rpm` package is available at: http://download.novell.com. A valid user account at Novell web site is required to download the above package.

- **For SLES10:**

  `device-mapper-1.02.03-8.2.rpm`

  `multipath-tools-0.4.6-25.8.rpm`

- Ensure that the latest supported HBA drivers are installed in the system

For information on supported servers, HBAs, drivers, and storage arrays, see Table 3.

Device Mapper Multipath support matrix

Table 3 lists the hardware and software supported by this second edition of Device Mapper Multipath for HP StorageWorks Arrays.

Table 3 Device Mapper Multipath supported hardware and software

System feature	Supported hardware and software
Operating system versions	Red Hat Enterprise Linux 4 Update 3 Red Hat Enterprise Linux 4 Update 4 SuSE LINUX Enterprise Server 9 - Service Pack 3 errata-267 SuSE LINUX Enterprise Server 10
Host Bus Adapters (HBA)	**Qlogic:** • FCA2214 PCI-X 1-port 2Gbps Fibre Channel • FCA2214DC PCI-X 2-port 2Gbps Fibre Channel • A7538A PCI-X 1-port 2Gbps Fibre Channel • A6826A PCI-X 2-port 2Gbps Fibre Channel • FC1142SR PCI-E 1-port 4Gbps Fibre Channel • FC1242SR PCI-E 2-port 4Gbps Fibre Channel • FC1143 PCI-X 1-port 4Gbps Fibre Channel • FC1243 PCI-X 2-port 4Gbps Fibre Channel • AB379A PCI-X 2-port 4Gbps Fibre Channel **Emulex:** • A8002A/LPe1150 PCI-E 1-port 4Gbps Fibre Channel • A8003A/LPe11002 PCI-E 2-port 4Gbps Fibre Channel • AD167A PCI-X 2.0 1-port 4Gbps Fibre Channel • AD168A PCI-X 2.0 2-port 4Gbps Fibre Channel • EZ & E2 2 Gbps Mezzanine for p-class Blade Servers
Servers	ProLiant Blade Servers, ProLiant x86, ProLiant AMD64, ProLiant EM64T Servers Integrity Servers
Supported arrays	EVA 3000 (HSV101) VCS 4.004 EVA 5000 (HSV111) VCS 4.004 EVA 4000 (HSV200) XCS 5.110/6.000 EVA 6000 (HSV210) XCS 5.110/6.000 EVA 8000 (HSV210) XCS 5.110/6.000 XP128* fw rev 21-14-18-00/00 XP1024* fw rev 21-14-18-00/00 XP10000 fw rev 50-07-30-00/00 XP12000 fw rev 50-07-30-00/00 *If you use XP128/1024, the system modes 140 and 293 need to be switched on, otherwise the XP will not respond to a SCSI inquiry to code page 0x83 with a unique serial number (scsi_id tool). This would prevent the Device Mapper multipathing from detecting the redundant paths.
HBA drivers	Emulex: version 8.1.6.6 or later (for SLES10), 8.0.16.27 or later (for RHEL4 and SLES9) available at: http://h18006.www1.hp.com/products/storageworks/4gbpciehba/index.html Qlogic: version 8.01.06-5 or later, available at: http://h18006.www1.hp.com/products/storageworks/fca2214/index.html

NOTE:

- On any single host, the HBAs must be from the same HBA vendor.
- Does not support coexistence with other multipathing products.
- Does not support Active-Passive Storage Arrays.

Setting up Device Mapper Multipath for HP StorageWorks Arrays

This section discusses the following:

- Configuring HBA parameters
- Setting up Device Mapper Multipath daemons

Configuring HBA parameters

You must configure HBA parameters for the Device Mapper Multipath. The HBA time outs are typically setup for non Device Mapper Multipath environments, where longer time outs are necessary. This is because the only alternative is to send an error to the application. However, with Multipath, errors like cable failures must be intimated at the earliest so that the Multipath layer can quickly take action and redirect the I/O to another path.

This section discusses the following:

- Configuring QLogic HBA parameters
- Configuring Emulex HBA parameters

Configuring QLogic HBA parameters

To configure QLogic HBA parameter, complete the following steps:

1. For QLogic `2xxx` family of HBAs, edit the `/etc/modprobe.conf` file in RHEL4 hosts and `/etc/modprobe.conf.local` file in SLES9/SLES10 hosts with the following values:

   ```
   options qla2xxx qlport_down_retry=1 ql2xfailover=0 ql2xretrycount=5
   ```

2. Rebuild the `initrd` by executing the following script:

   ```
   /opt/hp/src/hp_qla2x00src/make_initrd
   ```

3. Reboot the host.

Configuring Emulex HBA parameters

To configure Emulex HBA parameter, complete the following steps:

1. For Emulex `lpfc` family of HBAs:
   - In RHEL4 hosts, edit the `/etc/modprobe.conf` file with the following values:

     ```
     options lpfc lpfc_nodev_tmo=14 lpfc_lun_queue_depth=16 lpfc_discovery_threads=1
     ```

   - In SLES9 hosts, edit the `/etc/modprobe.conf.local` file with the following values:

     ```
     options lpfc lpfc_nodev_tmo=14 lpfc_lun_queue_depth=16 lpfc_discovery_threads=1
     ```

   - In SLES10 hosts, edit the `/etc/modprobe.conf` file with the following values:

     ```
     options lpfc lpfc_lun_queue_depth=16 lpfc_discovery_threads=1
     ```

2. Rebuild the `initrd` by executing the following script:

   ```
   /opt/hp/hp-lpfc/make_initrd
   ```

3. Reboot the host.

NOTE:

For additional information, refer the Installation and Reference Guide of the respective HBA Drivers.

Setting up Device Mapper Multipath daemons

You must set the Device Mapper Multipath daemons to start at boot time. The `multipathd` daemon does not automatically start while booting the system.

For RHEL4 hosts:

Complete the following steps on your RHEL4 host to start the `multipathd` daemon at boot time:

1. Run the following command to check if the daemon is configured to start in any run level:

   ```
   # chkconfig --list multipathd
   ```

2. Run the following commands to start the `multipathd` daemon during boot time in the desired run level:

   ```
   # chkconfig [--level levels] multipathd on
   # chkconfig multipathd
   ```

3. Reboot the host.

For SLES9/SLES10 hosts:

Complete the following steps on your SLES9/SLES10 host to start the `multipathd` daemon at boot time:

1. Run the following commands to check if the multipath modules and multipath daemon is configured to load and start in any run level:

   ```
   # chkconfig --list boot.device-mapper

   # chkconfig --list boot.multipath

   # chkconfig --list multipathd
   ```

2. Run the following commands to load the multipath modules and start the `multipathd` daemon during boot time in the desired run level:

   ```
   # chkconfig boot.device-mapper [levels]

   # chkconfig boot.multipath [levels]

   # chkconfig multipathd [levels]
   ```

3. Reboot the host.

To verify whether the modules required by Device Mapper Multipath are loaded and the `multipathd` daemon has started automatically at boot time, run the following command and verify if `dm_multipath` and `dm_mod` modules are listed:

```
# lsmod | grep dm_multipath
# lsmod | grep dm_mod
```

Run the `# /etc/init.d/multipathd status` command to verify that the daemon is running.

Configuring Device Mapper Multipath for HP StorageWorks Arrays

Configuring Device Mapper Multipath for HP StorageWorks Arrays involves editing the multipathing configuration file that is distributed as part of the Device Mapper Multipath tools user space package. It also allows you to customize Device Mapper Multipath for a variety of storage subsystems present in the Storage Area Network (SAN).

For RHEL4 systems:

Following is the sample configuration file:

```
/usr/share/doc/device-mapper-multipath-0.4.5/multipath.conf.annotated
/usr/share/doc/device-mapper-multipath-0.4.5/multipath.conf.synthetic
```

For SLES9/SLES10 systems:

Following is the sample configuration file:

```
/usr/share/doc/packages/multipath-tools/multipath.conf.annotated
/usr/share/doc/packages/multipath-tools/multipath.conf.synthetic
```

The `multipath.conf.annotated` file contains a description of each of the parameters mentioned. The `multipath.conf.synthetic` file does not contain any description of the parameters.

Copy the `multipath.conf.annotated` file and rename it as `/etc/multipath.conf` and make the necessary changes to the device parameters for HP supported arrays.

In general, `multipath.conf` file consists of four sections that provides a variety of options to configure the attributes of a `multipath` device, such as I/O spreading policy, polling interval for subsequent path status verification, path selector algorithm and the program used by multipath to obtain a unique path identifier.

The `/etc/multipath.conf` file is divided into the following sections:

- System defaults (`defaults`)
- Black-listed devices (`devnode_blacklist`)
- Per storage array model settings (`devices`)
- Per multipath device settings (`multipaths`)

The per multipath device settings are used for the multipath device with a matching WWID value. The per storage array model settings are used for all multipath devices with matching vendor and product values. To determine the attributes of a multipath device, the per multipath settings are checked first, the per controller settings are checked second, and system defaults are checked at the last. The blacklisted device section provides various options to blacklist the devices and hence refrain Device Mapper Multipath from claiming those devices, based on the your requirements. The system defaults settings are applied for any multipath device, which does not fit into the other three sections.

For HP supported arrays, the multipath device attributes are configured using the section devices, which are per storage array model settings that are applied by matching vendor and product values. For each supported array model, a new sub section must be added in devices with the vendor and product values of the same array model. The attributes that are applied to the multipath devices belonging to the same array model are also added subsequently in the same sub section.

Table 4 lists the important attributes applied to HP supported arrays.

Table 4 HP supported array attributes

Attribute	Description	Supported values
`path_grouping_policy`	Used for path grouping policy to apply to multipath device hosted by this storage controller	- multibus: All valid paths in one priority group - group_by_prio: One priority group per path priority value
`path_checker`	Used for determining the state of the path	- tur
`path_selector`	Used to select the path selector algorithm to use for `mpath`. These algorithms are offered by the kernel `mpath` target	- round-robin 0
`failback`	Used to manage the time during path group failback	- immediate
`prio_callout`	Executable to obtain a path weight for a block device. Weights are summed for each path group to determine the next path group to use in case of path failure	- none - /sbin/mpath_prio_alua %n
`rr_weight`	Used to assign weights to the path	- uniform
`no_path_retry`	(n > 0) tells the number of retries until disable queuing (queues till n number of polling), or *fail* means immediate failure (no queuing), *queue* means never stop queuing (queue for ever till the path comes alive)	- **n (>0)** - **fail** - queue

The following example provides the recommended attributes and values for HP supported arrays:

NOTE:

The `product` string of storage controller contains 16 characters (including spaces) and the `vendor` string contains 8 characters (including spaces) before editing the `/etc/multipath.conf` file. To ease out the manual editing of `devices` section in `multipath.conf` file, the wild card, regular expression support for the `product` and `vendor` string is also available. For Example, in `/etc/multipath.conf` file, the `product` string for the HSV101 arrays can be given the value as "HSV101*" instead of keeping it as "HSV101 \(C\)COMPAQ".

For EVA3000-(HSV101)

```
device {
       vendor   "HP"
       product   "HSV101 \(C\)COMPAQ"
       path_grouping_policy  group_by_prio
       getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
       path_checker  tur
       path_selector  "round-robin 0"
       prio_callout  "/sbin/mpath_prio_alua %n"
       rr_weight  uniform
       rr_min_io  100
       failback  immediate
       no_path_retry  60
  }
```

For EVA5000-(HSV111)

```
device {
       vendor   "COMPAQ"
       product   "HSV111 \(C\)COMPAQ"
       path_grouping_policy  group_by_prio
       getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
       path_checker  tur
       path_selector  "round-robin 0"
       prio_callout  "/sbin/mpath_prio_alua %n"
       rr_weight  uniform
       rr_min_io  100
       failback  immediate
       no_path_retry  60
  }
```

For EVA4000/6000

```
device {
       vendor  "HP"
       product  "HSV200"
       path_grouping_policy  group_by_prio
       getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
       path_checker  tur
       path_selector  "round-robin 0"
       prio_callout  "/sbin/mpath_prio_alua %n"
       rr_weight  uniform
       rr_min_io  100
       failback  immediate
       no_path_retry  60
  }
```

For EVA8000

```
device {
       vendor  "HP”
       product  "HSV210"
       path_grouping_policy  group_by_prio
       getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
       path_checker  tur
       path_selector  "round-robin 0"
       prio_callout  "/sbin/mpath_prio_alua %n"
       rr_weight  uniform
       rr_min_io  100
       failback  immediate
       no_path_retry  60
  }
```

For XP arrays

```
device {
       vendor  "HP"
       product  "OPEN-*"
       path_grouping_policy  multibus
       getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
       rr_weight  uniform
       rr_min_io  100
       path_checker  tur
       failback  immediate
       no_path_retry  10
  }
```

NOTE:

In XP arrays, there are different LUNs such as OPEN-<x>, 3390-3A, 3390-3B, OP-C:3390-3C, 3380KA, 3380-KB, OP-C:3380-KC where x = {3,8,9,K,T,E,V}. The product strings for XP LUNs are based on the above emulation types. Since each product string requires a new device sub section, a new device section must be added for each emulation type.

Since regular expression is supported in /etc/multipath.conf. OPEN-* is sufficient for the product string for all the XP LUNs with different OPEN emulations.

After editing the /etc/multipath.conf file for HP StorageWorks Arrays, you must restart Device Mapper Multipath. See Starting Device Mapper Multipath for HP StorageWorks arrays for details.

Starting Device Mapper Multipath for HP StorageWorks Arrays

Whenever an attribute for the LUNs of any of the HP supported storage array changes by editing the multipath.conf file, the configuration has to be reloaded and the multipathd daemon has to be restarted by running the following commands:

```
# /sbin/multipath -v0
# /etc/init.d/multipathd restart
```

The above steps are required only if the attributes are changed. You do not manually execute the steps on every reboot because these are also part of boot time scripts.

To view the status of the multipath devices, multipath utility is run with the following options:

```
# /sbin/multipath -ll
```

For details on status maps in multipath -ll for Device Mapper multipath devices, see Using Device Mapper Multipath for HP StorageWorks arrays.

Using Device Mapper Multipath for HP StorageWorks Arrays

The multipath devices are created under /dev/mapper directory in RHEL4 and SLES10 hosts. These devices are the same as any other block devices present in the host and they can be used for any block or file level I/O operations, such as creating the file system. For the device during reboots, you can add a new name for any multipath device by adding the multipath sub-sections in the /etc/multipath.conf file. This is carried out with the attributes alias and the WWID of the multipath device present in that multipath subsection.

For example, when the following sub-section is added in the /etc/multipath.conf file for the LUN with WWID 3600508b30090f5d0d2a9d64590490022, a multipath device mydatadisk1 is created under /dev/mapper.

```
multipaths {
multipath {
wwid 3600508b30090f5d0d2a9d64590490022
alias mydatadisk1
path_grouping_policy multibus
path_checker tur
path_selector "round-robin 0"
}
```

You can use /dev/mapper/mydatadisk1 like any other block device.

The following section briefly explains the multipath map obtained by executing the multipath-ll command based on a sample multipath.conf:

The following is a sample `multipath.conf` (for RHEL4 U4) with parameters to support `EVA 4000` as a storage controller:

```
defaults {
multipath_tool  "/sbin/multipath -v0"
udev_dir  /dev
polling_interval  5
default_selector  "round-robin 0"
default_path_grouping_policy  failover
default_getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
default_prio_callout  "/bin/true"
default_features  "0"
rr_min_io  100
failback  immediate
}
multipaths {
multipath {
wwid  3600508b30090f5d0d2a9d64590490022
path_grouping_policy  multibus
path_checker  tur
path_selector  "round-robin 0"
}
multipath {
…….
…..  for other target
}
multipath {
…….
….. for other target
}
}
devices {

device {
vendor  "HP"
product  "HSV200"
path_grouping_policy  group_by_prio
getuid_callout  "/sbin/scsi_id -g -u -s /block/%n"
path_checker  tur
path_selector  "round-robin 0"
prio_callout  "/sbin/mpath_prio_alua %n"
rr_weight  uniform
rr_min_io  100
failback  immediate
no_path_retry  60
}

device {
…….
….. for targets from other storage controllers
}
}
```

For example, in a typical configuration, a Linux host with a dual port HBA will have connections to an EVA 4000 via two switches. In this case if all the paths are available, the host will have 4 I/O paths for any LUN presented from the EVA 4000. The `multipath.conf` file will enable the host to see the multipath map as follows.

```
3600508b30090f5d0cf46bd1a4c420023
[size=1 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [prio=100][active]
  \_ 3:0:1:3 sdad 65:208 [active][ready]
  \_ 2:0:2:3 sdo  8:224  [active][ready]
\_ round-robin 0 [prio=20][enabled]
  \_ 2:0:1:3 sdj  8:144  [active][ready]
  \_ 3:0:0:3 sdy  65:128 [active][ready]
```

```
3600508b30090f5d0d2a9d64590490022
[size=3 GB][features="1 queue_if_no_path"][hwhandler="0"]
\_ round-robin 0 [prio=120][active]
  \_ 3:0:2:4 sdaj 66:48  [active][ready]
  \_ 3:0:3:4 sdao 66:128 [active][ready]
  \_ 2:0:0:4 sdf  8:80   [active][ready]
  \_ 2:0:3:4 sdu  65:64  [active][ready]
```

The information in the map is presented by grouping the paths for a LUN with unique Identifiers such as, UID/WWN.

The size, features and the corresponding hwhandlers are shown followed by the unique LUN identifier.

The grouping of paths is done and presented in a map based on the I/O spreading policy. In the above example, for the LUN 3600508b30090f5d0d2a9d64590490022, the grouping is done with the policy as multibus and the devices sdaj, sdao, sdf, sdu belong to same path group. The state of the group will be either active or enabled depending on whether I/O is happening or not. And for the LUN 3600508b30090f5d0cf46bd1a4c420023, the grouping is done with the policy as group_by_prio and the devices sdad, sdo belong to one path group and sdj, sdy belong to different path group as they are from 2 controllers of different serial numbers from the EVA 4000 . The I/O always happens in the path group which is active. In the event of all paths failure in the active group, the failover occurs to the other path group which is enabled on changing that path group as active. When the paths are up again and if the failback parameter is set as immediate, the failback occurs to the earlier group and I/O will occur through the earlier group. For example, if 'group a' has sdad, sdo, and 'group b' has sdj, sdy, the failover occurs from 'group a' to 'group b' and the failback occurs from 'group b' to 'group a'.

The state of the path is given as [ready] [active] if the path is up and ready for I/O. If the path is down, this state will be shown as [faulty] [failed]. The path states will be updated periodically based on the polling interval as in `/etc/multipath.conf`, and this is taken care of by `multipathd` daemon.

Table 5 lists the basic operations supported by the `multipath` CLI utility provided with Device Mapper Multipath.

Table 5 Basic operations of Device Mapper Multipath

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td><code># multipath -F</code></td><td>Deletes all Device Mapper Multipath devices.</td></tr>
<tr><td><code># multipath -d</code></td><td>Displays potential paths, but do not create any device.</td></tr>
<tr><td><code># multipath</code></td><td>Creates Device Mapper Multipath devices.</td></tr>
<tr><td><code># multipath -l</code>
<code># multipath -ll</code>
<code># multipath -l|-ll <device&gt;</code></td><td>Displays the device status.</td></tr>
<tr><td><code># multipath -v2 <device&gt;</code></td><td>Configures and displays multipath map information.</td></tr>
<tr><td><code># multipath -v3</code></td><td>Configures and displays multipath map information.</td></tr>
<tr><td><code># multipath -p multibus [<device&gt;]</code>
<code># multipath -p group_by_prio[<device&gt;]</code></td><td>Sets group policy.</td></tr>
</table>

4 Known Issues

Following are the known issues of the Device Mapper Multipath:

- `ioctls` support is not available on a multipath device created by Device Mapper. As a result, SCSI inquiry like operations are not possible with Device Mapper Multipath devices.
- You have to manually enter the device properties for HP StorageWorks Arrays in the `/etc/multipath.conf` file.
- You have to manually do the device aliasing for the devices presented. See Starting Device Mapper Multipath for HP StorageWorks arrays for details.
- For RHEL4 U3 and SLES9 SP3 hosts with many LUNs, "`multipath -ll`" may take longer time to display the status of all the Device Mapper Multipath devices, if there is a change in the path status.
- In RHEL4 U3 and SLES9 hosts, device reset might occur, but will not impact device availability other than a possible I/O delay. Device reset occurs only during a path failure.
- Does not support boot/root partition on a SAN device
- During I/O on the multipath devices, there may be occasional path failures. However, the path failure will not have any impact on the application's stability, other than the possible I/O delay.

  Path failures occur frequently during heavy I/O load and results in slow I/O response. Device Mapper Multipath identifies this slow I/O response as a path failure and causes the path to fail.

  An example of the message displayed in the `/var/log/messages` file during occasional path failures follows:

  ```
  ...
  Oct  5 12:51:21 clipper1 kernel: SCSI error : <0 0 4 5> return code = 0x20000
  Oct  5 12:51:21 clipper1 kernel: end_request: I/O error, dev sdad, sector 4793376
  Oct  5 12:51:21 clipper1 kernel: end_request: I/O error, dev sdad, sector 4793378
  Oct  5 12:51:21 clipper1 kernel: device-mapper: dm-multipath: Failing path 65:208.
  Oct  5 12:51:22 clipper1 multipathd: 65:208: mark as failed
  Oct  5 12:51:22 clipper1 multipathd: xp128-5: remaining active paths: 3
  ...
  ```
- Provides only CLI support

5 Troubleshooting

Following are the troubleshooting procedures for Device Mapper Multipath issues:

- Execute the following steps whenever a new LUN is added or deleted at the Linux host to update the multipath maps in the kernel:
  1. `/etc/init.d/multipathd restart`
  2. `multipath -v <X>`
- Execute the following steps whenever multipath devices are deleted using `multipath -F` to create the multipath devices and to create the same for the partitions on the multipath devices:
  1. `multipath -v <X>`
  2. `/sbin/dmsetup ls --target multipath --exec "/sbin/kpartx -a -p -part"`