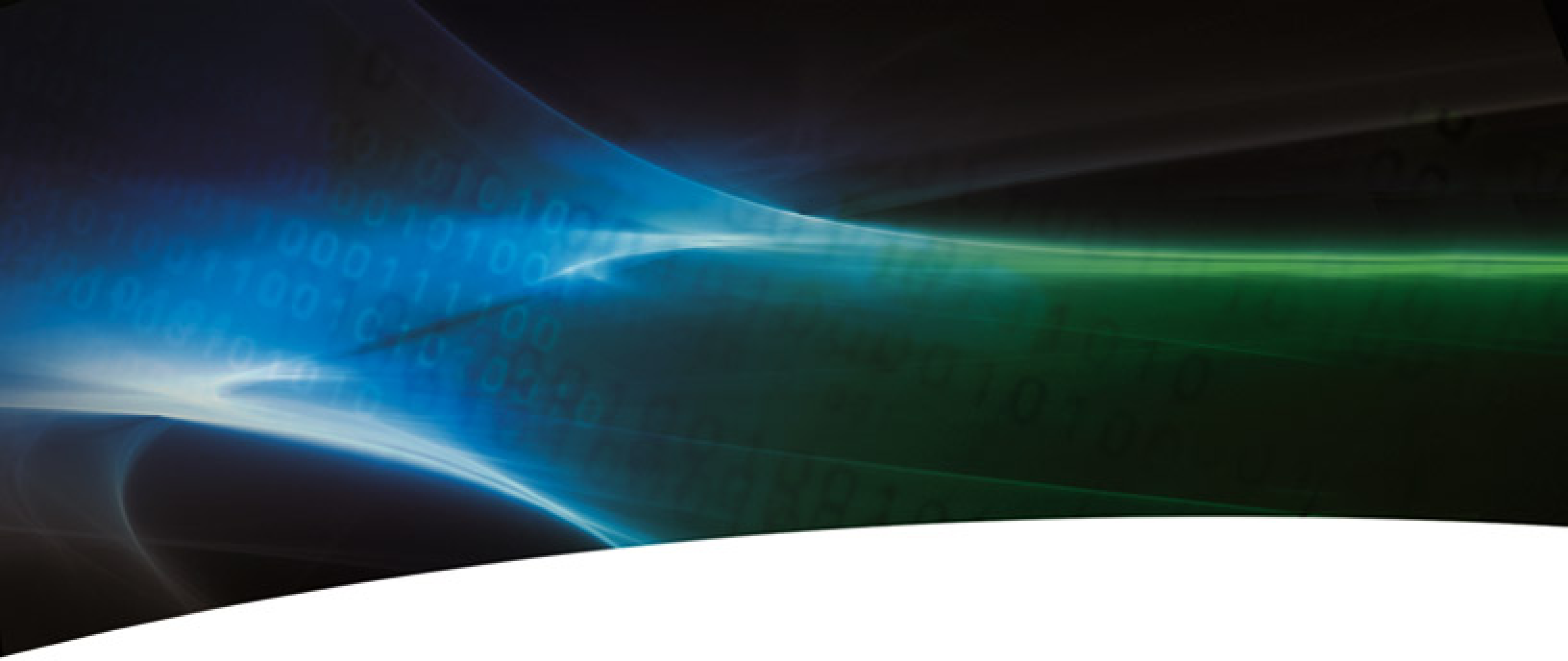


IBM ioMemory VSL 2.3.10

User Guide for Linux

Copyright Notice

Part Number: D0001565-003_1
Published: August 20, 2012

Introduction

Overview

Congratulations on your purchase of a IBM solid-state storage device. This guide explains how to install, troubleshoot, and maintain the software for your IBM High IOPS Adapters.

NOTE Throughout this manual, when you see a reference to an **IBM High IOPS Adapter**, you may substitute your particular device(s), such as an IBM High IOPS Adapter or each of the two IBM High IOPS Adapters of an IBM High IOPS Duo Adapter.

Attention **Products with Multiple Devices**: Some products, such as IBM High IOPS Duo Adapters, are actually comprised of **multiple IBM High IOPS Adapters**. If your product consists of multiple IBM High IOPS Adapters, you will manage each IBM High IOPS Adapter as an independent device.

For example, if you have an IBM High IOPS Duo Adapter, you can independently attach, detach, and/or format each of the two IBM High IOPS Adapters. Each of the two devices will be presented as an individual device to your system.

About ioMemory Devices

Designed around a revolutionary silicon-based storage architecture, IBM High IOPS Adapters are the world's most advanced NAND flash storage devices, with performance comparable to DRAM and storage capacity on par with today's hard disks – giving you the power to improve storage performance by orders of magnitude. IBM High IOPS Adapters allow every computer to exceed the I/O performance of an enterprise SAN.

IBM High IOPS Adapters are data accelerators designed specifically to improve the bandwidth for I/O-bound applications. They are no-compromise solutions for the toughest computational challenges faced by data centers today, putting them in a league of their own.

About the ioMemory Virtual Storage Layer (VSL)

More than just a hardware driver, the ioMemory® Virtual Storage Layer™ (VSL) is the "secret sauce" that gives IBM High IOPS Adapters their amazing performance. The VSL™ is a hybrid of the RAM virtualization subsystem and the disk I/O subsystem, combining the best of both worlds. It appears like a disk to interface well with block-based applications and software. At the same time, it runs like RAM underneath to maximize performance. This provides the following game-changing benefits:

- Performance: The ioMemory VSL offers direct and parallel access to multiple CPU cores, enabling near-linear performance scaling, consistent performance across different read/write workloads, and low latency with minimal interrupts and context switching
- Extensibility: The ioMemory VSL enables flash-optimized software development, making each ioMemory module a flexible building block for building a flash-optimized data center.

About Flashback Protection Technology

High IOPS memory devices have a finite life span of writes operations. Over the life of the product, NAND flash will eventually wear out as part of it's normal life cycle. To extend the life of the adapter, Flashback™ redundancy is designed to mitigate the loss of failed memory location without interrupting normal operation.

This real-time NAND flash redundancy works at the chip-level so these losses are handled without sacrificing user capacity, performance, and provides additional protection above and beyond ECC (Error Correction Code) for soft failures.

IBM's Flashback Protection™ technology, with self-healing properties, ensures higher performance, minimal failure, and longer endurance than all other flash solutions.

System Requirements

Please read the *IBM ioMemory VSL Release Notes* for more information on this release.

Hardware Requirements

- **Hardware Requirements**: These depend on your device (including device capacity, generation, and configuration). Please see the *IBM High IOPS Hardware Installation Guide* for requirements on the following:
  - PCIe Slot
  - Cooling
  - Power
- **Supported Devices**: Also see the *IBM High IOPS Hardware Installation Guide* for a list of supported IBM High IOPS Adapters.

  Attention Second generation adapters are not compatible with this version of the ioMemory VSL. If you have a newer IBM High IOPS Adapter that is not on this list, you may need to upgrade your system and all of your devices to ioMemory VSL 3.x software and firmware versions.
- **RAM Requirements**: The *IBM ioMemory VSL Release Notes* contains memory (RAM) requirements for this version of the software.

Supported Linux Distributions

Attention The following distributions are supported. Some distribution versions may have binary packages available for download. If your version does not have a binary package available, you can build the installation package from the available source package. Check the download folders for available packages.

- Red Hat Enterprise Linux (RHEL) 5 (up to 5.8), 6 (up to 6.2)
- SUSE Linux Enterprise Server (SLES) 10 SP4, 11, 11 SP1, 11 SP2

Software Installation

NOTE All commands require administrator privileges. Use sudo or log in as "root" to run the install.

Installation Overview

1. If you are installing this version of ioMemory VSL on a system with previous versions of the ioMemory VSL installed, you will need to uninstall the ioMemory VSL package and the utilities. See the Common Maintenance Tasks section for instructions. Once you have uninstalled the packages, return to this page.
2. Install the latest version of the ioMemory VSL. You can install the ioMemory VSL as:
   - A pre-compiled binary package
   - A source-to-build package

     NOTE Follow the instructions under Installing RPM Packages to determine whether pre-compiled binary packages are available for your kernel version or if you need to build the package from source.
3. Install utilities and management software (included in driver installation instructions).
4. Load the ioMemory VSL Driver and Set the Options.
5. Upgrade the Firmware to the latest version, if needed (recommended).

Installing RPM Packages

To install the Linux ioMemory VSL and utilities on SUSE and RHEL:

1. You will need to install a version of the ioMemory VSL that is built for your kernel. To determine what kernel version is running on your system, use the following command at a shell prompt:

```
$ uname -r
```

2. Compare your kernel version with the binary versions of the software available at http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174.
    - If there is a binary version of the software that corresponds to your kernel version, download that. **For example**:

    ```
    iomemory-vsl-<kernel-version>-<VSL-version>.x86_64.rpm
    ```

    - If there is no binary version of the software corresponding to your kernel, download the source package. **For example**:

    ```
    iomemory-vsl-<VSL-version>.src.rpm
    ```

    NOTE Exact package names may vary, depending on software and kernel version chosen.

3. Download all of the support RPM packages from http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174. These packages provide utilities, firmware, and other files.
    - **Examples:**

Package	What is installed
`fio-util-<VSL-version>.x86_64.rpm`	ioMemory VSL utilities – **Recommended**
`fio-firmware-<firmware-version>.noarch.rpm`	Firmware archive – **Recommended**
`libfio-<version>.x86_64.rpm`	Libraries needed for management tools – **Recommended**
`fio-common-<VSL-version>.x86_64.rpm`	Files required for the init script – **Recommended**
`fio-sysvinit-<VSL-version>.x86_64.rpm`	Init script – **Recommended**
`fio-smis-<version>.x86_64.rpm`	SMI-S CIM Provider – Optional see Appendix E- SMI-S Interface
`fio-snmp-agentx-<version>.x86_64.rpm`	Agentx SNMP subagent – Optional, **you must install `net-snmp` before you install this package**, see Appendix D- Setting Up SNMP for details.
`fio-snmp-mib-<version>.x86_64.rpm`	SNMP MIBs – Optional see Appendix D- Setting Up SNMP
`libfio-dev-<version>.x86_64.rpm`	Management SDK – Optional
`libfio-doc-<version>.x86_64.rpm`	Management SDK – Optional

4. Build the ioMemory VSL software from source: if you downloaded the software source package, you must now follow the instructions in Building the Package from Source. If you downloaded a binary version of the software, continue.

5. Change to the directory to where you downloaded the installation packages.

6. Enter the following command to install the custom-built software package. Use the package name that you just copied/downloaded into that directory.

```
rpm -Uvh iomemory-vsl-<kernel-version>-<VSL-version>.x86_64.rpm
```

7. Enter the following commands to install the support files:

```
rpm -Uvh lib*.rpm
rpm -Uvh fio*.rpm
```

The ioMemory VSL and utilities are installed to the following locations:

Package Type	Installation Location
ioMemory VSL	`/lib/modules/<kernel-version>/extra/fio/iomemory-vsl.ko`
Utilities	`/usr/bin`
Firmware	`/usr/share/fio/firmware`
SNMP MIB	`/usr/share/fio/mib`

NOTE Under SLES 10, the `fio-sysvinit` package will install with an error:

```
Service udev has to be enabled for service iomemory-vsl
```

After the package is installed on a SLES 10 system, follow the instructions in Loading the ioMemory VSL Facility (Driver) to modify the `/etc/init.d/iomemory-vsl` file so the `sysvinit` script will function properly.

NOTE IBM High IOPS Management Application is a free browser-based solution for managing IBM High IOPS Adapter. The IBM High IOPS Management Application installation software and user documentation are also available from http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174.

Once the packages are installed, continue to Loading the Software Driver later in the section.

Building the Package from Source

The ioMemory VSL is distributed as a source package. If a binary version of the software is not available, you will need to build the ioMemory VSL from source. Use the source package that is made for your distribution. Source packages from other distributions are not guaranteed to work.

1. Download current ioMemory VSL source and support packages from http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174

   NOTE The exact source package to download depends on your operating system, but it is an `RPM` package.

2. Change directory to wherever you downloaded the source package.

3. Follow the instructions below to create an installation package tailored to your needs.

Building an RPM Installation Package

To build an RPM installation package:

1. Install the prerequisite files for your kernel version.

   NOTE Some of the prerequisite packages may already be in the default OS installation. If your system is not configured to get packages over the network, then you may need to mount your install CD/DVD.

   - On RHEL 5/6, you need `kernel-devel`, `kernel-headers`, `rpm-build`, GCC4, and `rsync`.

     ```
     $ yum install kernel-devel kernel-headers rpm-build gcc rsync
     ```

     Attention `yum` may not install the correct (matching) kernel versions of `kernel-devel` or `kernel-headers` packages. `yum` will download the latest version. Use the following command to force `yum` to download the exact versions:

     ```
     yum install kernel-headers-`uname -r` kernel-devel-`uname
     -r` gcc rsync rpm-build
     ```

     If the exact versions are no longer available in the repository, then you will have to manually search for them on the Internet.

   - On SLES 10/11 you need `kernel-syms`, `make`, `rpm-build`, GCC4, and `rsync`.

     ```
     $ zypper install kernel-syms make rpm gcc rsync
     ```

2. To build an RPM installation package for the current kernel, run this command:

```
$ rpmbuild --rebuild iomemory-vsl-<VSL-version>.src.rpm
```

   When using a .rpm source package for a non-running kernel, run this command:

```
$ rpmbuild --rebuild --define 'rpm_kernel_version <kernel-version>'
iomemory-vsl-<VSL-version>.src.rpm
```

   The new `RPM` package is located in a directory that is indicated in the output from the `rpmbuild` command. To find it, look for the "wrote" line. In the following example, the RPM packages are located in the `/usr/src/redhat/RPMS/x86_64/` directory.

```
...
Processing files: iomemory-vsl-source-2.2.3.66-1.0.x86_64.rpm
Requires(rpmlib): rpmlib(PayloadFilesHavePrefix) <= 4.0-1
rpmlib(CompressedFileNames) <= 3.0.4-1
Obsoletes: iodrive-driver-source
Checking for unpackaged file(s): /usr/lib/rpm/check-files
/var/tmp/iomemory-vsl-2.2.3.66-root
Wrote:
/usr/src/redhat/RPMS/x86_64/iomemory-vsl-2.6.18-128.el5-2.2.3.66-1.0.x86_
/usr/src/redhat/RPMS/x86_64/iomemory-vsl-source-2.2.3.66-1.0.x86_64.rpm
```

3. Make a note of the RPM location; you will need this information later in the installation.

4. You have now built installation packages for your distribution and kernel.

5. Copy your custom-built software installation RPM package into the directory where you downloaded the installation packages.

6. Return to the Installing RPM Packages section of this guide.

Loading the ioMemory VSL Driver

To load the ioMemory VSL driver:

1. Run this command:

   ```
   $ modprobe iomemory-vsl
   ```

   NOTE The ioMemory VSL automatically loads at system boot. The IBM High IOPS Adapter is now available to the OS as `/dev/fiox`, where *x* is a letter (i.e., a, b, c, etc.).

   - For SLES 10 systems, the above modprobe command works to manually load the driver, however to automatically start the driver at boot time with UDEV, edit the `/etc/init.d/iomemory-vsl` file to modify the init parameter and change `udev` to `boot.udev` as specified below.

     NOTE You must install the `fio-sysvinit` package before you can edit the `/etc/init.d/iomemory-vsl` file.

     After the change, the file should look like this:

     ```
     ### BEGIN INIT INFO
     # Provides:       iomemory-vsl
     # Required-Start: boot.udev
     ```

   - On SLES systems, you must also allow unsupported modules for this command to work.
     - **SLES 11**: Modify the `/etc/modprobe.d/iomemory-vsl.conf` file and uncomment the appropriate line:

       ```
       # To allow the ioMemory VSL driver to load on SLES11,
       uncomment below
       allow_unsupported_modules 1
       ```

     - **SLES 10**: Modify the `/etc/sysconfig/hardware/config` file so the `LOAD_UNSUPPORTED_MODULES_AUTOMATICALLY` sysconfig variable is set to `yes`, for example:

       ```
       LOAD_UNSUPPORTED_MODULES_AUTOMATICALLY=yes
       ```

2. To confirm the IBM High IOPS Adapter is attached, run the `fio-status` utility from the command line. The output lists each drive and its status (attached or not attached).

   Attention If the IBM High IOPS Adapter is not automatically attaching, check the /etc/modprobe.d files to see if the `auto_attach` option is turned off (set to 0).

Controlling ioMemory VSL Loading

You can control driver loading either through the init script or through udev.

In newer Linux distributions, users can rely on the udev device manager to automatically find and load drivers for their installed hardware at boot time, though udev can be disabled and the init script used in nearly all cases. For older Linux distributions without this functionality, users must rely on a boot-time init script to load needed drivers. IBM provides an init script in `/etc/init.d/iomemory-vsl` to load the VSL driver in SLES10 distributions.

Using the init Script

On systems where udev loading of the driver doesn't work, or is disabled, the init script may be enabled to load the driver at boot. On some distros it may be enabled by default.

NOTE The init Script is part of the `fio-sysvinit` package, which must be installed before you can enable it.

You can disable this loading of the ioMemory VSL with the following command:

```
$ chkconfig --del iomemory-vsl
```

To re-enable the ioMemory VSL loading in the init script, use the following command:

```
$ chkconfig --add iomemory-vsl
```

For further details on the init script, see Using the Init Script later in this section.

Using udev

On systems that rely on udev to load drivers, users need to modify an ioMemory VSL's options file if they want to prevent udev from auto-loading the ioMemory VSL at boot time. To do this, locate and edit the `/etc/modprobe.d/iomemory-vsl.conf` file that already has the following line:

```
# blacklist iomemory-vsl
```

To disable loading, remove the "#" from the line and save the file.

With the blacklist command in place, restart Linux. The ioMemory VSL will not be loaded by udev.

To restore the udev-loading of the ioMemory VSL, replace the "#" to comment out the line.

On either udev or init script systems

Users can disable the loading of the ioMemory VSL at boot time, and thus prevent the auto-attach process for diagnostic or troubleshooting purposes on either udev or init script systems. Follow the steps in the Disabling Auto-Attach section to disable or re-enable the auto-attach functionality.

Alternatively, you can prevent the ioMemory VSL from loading by appending the following parameter at the kernel command line of your boot loader:

```
iodrive=0
```

However, this method is not preferred as it prevents the ioMemory VSL from functioning at all, thus limiting the amount of troubleshooting you can perform.

Using the Init Script

The ioMemory install process places an init script in `/etc/init.d/iomemory-vsl`. In turn, this script uses the setting options found in the options file in `/etc/sysconfig/iomemory-vsl`. The options file must have ENABLED set (non-zero) for the init script to be used:

```
ENABLED=1
```

The options file contains documentation for the various settings---two of which, `MOUNTS` and `KILL_PROCS_ON_UMOUNT`, are discussed further in the Handling ioMemory VSL Unloads section later in this document.

Mounting Filesystems

Because the ioMemory VSL does not load by the standard means (in the `initrd`, or built into the kernel), using the standard method for mounting filesystems (`/etc/fstab`) for filesystems hosted on the IBM High IOPS Adapter does not work. To set up auto-mounting of a filesystem hosted on an IBM High IOPS Adapter:

1. Add the filesystem mounting command to `/etc/fstab` as normal.
2. Add the 'noauto' option to `etc/fstab` as in the two following sample entries.

```
/dev/fioa /mnt/fioa ext3 defaults,noauto 0 0
/dev/fiob1 /mnt/ioDrive ext3 defaults,noauto 0 0
```

(where the `a` in `fioa` can be `a`, `b`, `c`, etc., depending on how many IBM High IOPS Adapters you have installed in the system).

To have the init script mount these drives after the ioMemory VSL is loaded and unmounted and before the ioMemory VSL is unloaded, add a list of mount points to the options file using the procedure documented there.

For the filesystem mounts shown in the earlier example, the line in the options file would look like this:

```
MOUNTS="/mnt/fioa /mnt/iodrive"
```

Handling ioMemory VSL Unloads

Special consideration must be taken during ioMemory VSL unload time. By default, the init script searches for any processes holding open a mounted filesystem and kills them, thus allowing the filesystem to be unmounted. This behavior is controlled by the option `KILL_PROCS_ON_UMOUNT` in the options file. If these processes are not killed, the filesystem cannot be unmounted. This may keep the ioMemory VSL from unloading cleanly, causing a significant delay on the subsequent boot.

Setting the Software Options

This section explains how to set ioMemory VSL options. For more information about setting specific options, see Appendix C- Using Module Parameters.

One-Time Configuration

ioMemory VSL options can be set at install time, on the command line of either `insmod` or `modprobe`. For example, to set the `auto_attach` ioMemory VSL option to 0, run the command:

```
$ modprobe iomemory-vsl auto_attach=0
```

This option takes effect only for this load of this ioMemory VSL; subsequent calls to `modprobe` or `insmod` will not have this option set.

Persistent Configuration

To maintain a persistent setting for an option, add the desired option to `/etc/modprobe.d/iomemory-vsl.conf` or a similar file. To prevent the IBM High IOPS Adapters from auto-attaching, add the following line to the `iomemory-vsl.conf` file:

```
options iomemory-vsl auto_attach=0
```

This ioMemory VSL option then takes effect for every subsequent ioMemory VSL load, as well as on autoload of the ioMemory VSL during boot time.

Upgrading the Firmware

With the ioMemory VSL loaded, you need to check to ensure that the IBM High IOPS Adapter's firmware is up-to-date. To do this, run the `fio-status` command-line utility.

If the output shows that the device is running in minimal mode, download the latest firmware from http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174, then use the IBM High IOPS Management Application software or the `fio-update-iodrive` utility to upgrade the firmware.

Attention **Upgrade Path**: There is a specific upgrade path that you must take when upgrading IBM High IOPS Adapter. Consult the *Release Notes* for this ioMemory VSL release before upgrading IBM High IOPS Adapters.

Your IBM High IOPS Adapter may have a minimum firmware label affixed (for example, "MIN FW: XXXXXX"). This label indicates the minimum version of the firmware that is compatible with your device.

Attention Do not attempt to downgrade the firmware on any IBM High IOPS Adapter, doing so may void your warranty.

When installing a new IBM High IOPS Adapter along with existing devices, you must upgrade all of the currently installed devices to the latest available versions of the firmware and ioMemory VSL before installing the new devices.

Consult the *Release Notes* for this ioMemory VSL release for any upgrade considerations.

Upgrading VMware Guest OS

If you are using your IBM High IOPS Adapter with a VMware guest OS (using VMDirectPathIO), you must cycle the power on the host after you upgrade the device(s). Just restarting the virtual machine won't apply the change.

Using the Device as Swap

To safely use the IBM High IOPS Adapter as swap space requires passing the preallocate_memory kernel module parameter. The recommended method for providing this parameter is to add the following line to the `/etc/modprobe.d/iomemory-vsl.conf` file:

```
options iomemory-vsl preallocate_memory=1072,4997,6710,10345
```

- Where 1072,4997,6710,10345 are serial numbers obtained from `fio-status`.

A 4K sector size format is required for swap—this reduces the ioMemory VSL memory footprint to reasonable levels. Use `fio-format` to format the IBM High IOPS Adapter with 4k sector sizes.

NOTE Be sure to provide the serial numbers for the IBM High IOPS Adapter, not an adapter, when applicable.

NOTE The `preallocate_memory` module parameter is necessary to have the device usable as swap space. See Appendix C- Using Module Parameters for more information on setting this parameter.

Attention You must have enough RAM available to enable the IBM High IOPS Adapter with pre-allocation enabled for use as swap. Attaching an IBM High IOPS Adapter, with pre-allocation enabled, without sufficient RAM may result in the loss of user processes and system instability.

Consult the *IBM ioMemory VSL Release Notes* for RAM requirements with this version of the ioMemory VSL.

NOTE The `preallocate_memory` parameter is recognized by the ioMemory VSL at load time, but the requested memory is not actually allocated until the specified device is attached.

Using the Logical Volume Manager

The Logical Volume Manager (LVM) volume group management application handles mass storage devices like IBM High IOPS Adapters if you add the IBM High IOPS Adapter as a supported type:

1. Locate and edit the /etc/lvm/lvm.conf configuration file.
2. Add an entry similar to the following to that file:

```
types = [ "fio", 16 ]
```

The parameter "16" represents the maximum number of partitions supported by the device.

NOTE If using LVM or MD, do not use udev to load the ioMemory VSL driver. The init script will ensure that the LVM volumes and MD devices are detached before attempting to detach the IBM High IOPS Adapter.

Configuring RAID

You can configure two or more IBM High IOPS Adapters into a RAID array using software-based RAID solutions.

NOTE If you are using RAID1/Mirrored and one device fails, be sure to run `fio-format` on the replacement device (not the existing, good device) before rebuilding the RAID. Following are some examples of some common RAID configurations using the `mdadm` utility.

Attention The Linux kernel RAID 5 implementation performs poorly at high data rates. This is an issue in the Linux kernel. Alternatives include using RAID 10, or possibly a third-party RAID stack.

RAID 0

To create a striped set, where `fioa` and `fiob` are the two IBM High IOPS Adapters you want to stripe, run this command:

```
$ mdadm --create /dev/md0 --chunk=256 --level=0 --raid-devices=2 /dev/fioa
/dev/fiob
```

Making the Array Persistent (Existing after Restart)

NOTE On some versions of Linux, the configuration file is in `/etc/mdadm/mdadm.conf`, not `/etc/mdadm.conf`.

Inspect `/etc/mdadm.conf`. If there are one or more lines declaring the devices to inspect, make sure one of those lines specifies "partitions" as an option. If it does not, add a new DEVICE line to the file specifying "partitions" like this:

```
DEVICE partitions
```

Also add a device specifier for the fio IBM High IOPS Adapters:

```
DEVICE /dev/fio*
```

To see if any updates are needed to `/etc/mdadm.conf`, issue the following command:

```
$ mdadm --detail --scan
```

Compare the output of this command to what currently exists in `mdadm.conf` and add any needed sections to `/etc/mdadm.conf`.

NOTE For example, if the array consists of two devices, there will be three lines in the output of the command that are not present in the `mdadm.conf` file: one line for the array, and two device lines (one line for each device). Be sure to add those lines to the `mdadm.conf` so it matches the output of the command.

For further details please see the `mdadm` and `mdadm.conf` man pages for your distribution.

With these changes, on most systems the RAID 0 array will be created automatically upon restart. However, if you have problems accessing `/dev/md0` after restart, run the following command:

```
$ mdadm --assemble --scan
```

You may also want to disable udev loading of the ioMemory VSL driver, if needed, and use the init script provided for driver loading. Please see the Using the Init Script section of this guide for further details on how to use the init script.

NOTE In SLES 11, you may need to run the following commands to make sure these services are run on boot:

```
chkconfig boot.md on
chkconfig mdadmd on
```

RAID 1

To create a mirrored set using the two IBM High IOPS Adapters `fioa` and `fiob`, run this command:

```
$ mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/fioa /dev/fiob
```

RAID 10

To create a striped, mirrored array using four IBM High IOPS Adapters (fioa, fiob, fioc, and fiod), run this command:

```
$ mdadm --create /dev/md0 -v --chunk=256 --level=raid10 --raid-devices=4
/dev/fioa /dev/fiob /dev/fioc /dev/fiod
```

Building a RAID10 Across Multiple Devices

In a RAID10 configuration, sets of two disks are mirrored, and then those mirrors are striped. When setting up a RAID10 across multiple IBM High IOPS Adapters, it is best to make sure that no mirror resides solely on the two IBM High IOPS Adapters that comprise an a single product (such as an IBM High IOPS Duo Adapter).

In order to get the data to lay out properly,

- Use the `--layout=n2` option when creating the RAID10 (though it should be the default)
- Ensure that no two IBM High IOPS Adapters from the same device are listed side by side.

The following sample code shows some recommended configurations.

NOTE The following commands assume that all IBM High IOPS Adapters have been freshly formatted with the `fio-format` utility.

Attention The ordering of the fiox devices is critical.

```
# 2 Duos RAID10
$ mdadm --create --assume-clean --level=raid10 --layout=n2 -n 4 /dev/md0 \
 /dev/fioa /dev/fioc \
 /dev/fiob /dev/fiod
# Mirror groups are: fioa,fioc and fiob,fiod

# 3 Duos RAID10
$ mdadm --create --assume-clean --level=raid10 --layout=n2 -n 6 /dev/md0 \
 /dev/fioa /dev/fiod \
 /dev/fioc /dev/fiof \
 /dev/fioe /dev/fiob

# 4 Duos RAID10
$ mdadm --create --assume-clean --level=raid10 --layout=n2 -n 8 /dev/md0 \
 /dev/fioa /dev/fiod \
 /dev/fioc /dev/fiof \
 /dev/fioe /dev/fioh \
 /dev/fiog /dev/fiob

# 8 Duos RAID10
$ mdadm --create --assume-clean --level=raid10 --layout=n2 -n 16 /dev/md0 \
 /dev/fioa /dev/fiod \
 /dev/fioc /dev/fiof \
 /dev/fioe /dev/fioh \
 /dev/fiog /dev/fioj \
 /dev/fioi /dev/fiol \
 /dev/fiok /dev/fion \
 /dev/fiom /dev/fiop \
 /dev/fioo /dev/fiob
```

Understanding Discard (TRIM) Support

With this version of the ioMemory VSL, Discard (also known as TRIM) is enabled by default.

Discard addresses an issue unique to solid-state storage. When a user deletes a file, the device does not recognize that it can reclaim the space. Instead the device assumes the data is valid.

Discard is a feature on newer filesystem releases. It informs the device of logical sectors that no longer contain valid user data. This allows the wear-leveling software to reclaim that space (as reserve) to handle future write operations.

Discard (TRIM) on Linux

Discard is enabled by default in the release. However, for discard to be implemented, the Linux distribution must support this feature, and discard must be turned on.

In other words, if your Linux distribution supports discard, and discard is enabled on the system, then discard will be implemented on your IBM High IOPS Adapter.

Under Linux, discards are not limited to being created by the filesystem, discard requests can also be generated directly from userspace applications using the kernel's discard ioctl.

Attention There is a known issue that ext4 in Kernel.org 2.6.33 or earlier may silently corrupt data when discard is enabled. This has been fixed in many kernels provided by distribution vendors. Please check with your kernel provider to be sure your kernel properly supports discard. For more information, see the Errata in the *Release Notes* for this version of the software

NOTE On Linux, MD and LVM do not currently pass discards to underlying devices. Thus any IBM High IOPS Adapter that is part of an MD or LVM array will not receive discards sent by the filesystem.

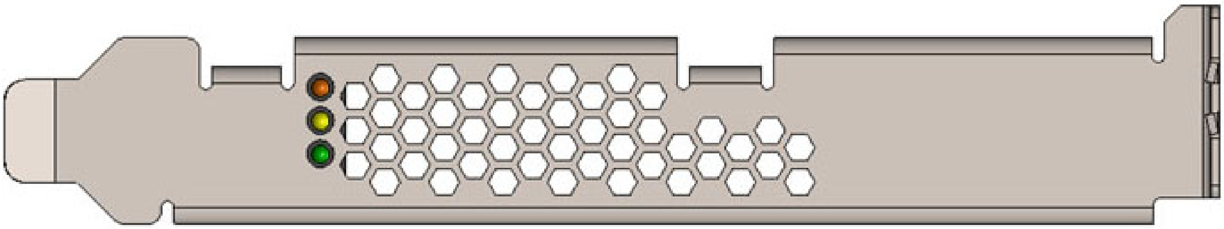
The LVM release included in Red Hat 6.1 supports passing discards for several targets, but not all (RHEL 6.1 documentation). Please see your distribution's documents for exact details.

Maintenance

The ioMemory VSL includes software utilities for maintaining the device. You can also install SNMP as a monitoring option.

Device LED Indicators

The IBM High IOPS Adapter includes three LEDs showing drive activity or error conditions. The LEDs on your device should be similar to this configuration:



This table explains the information that these LEDs convey:

Green	Yellow	Amber	Indicates	Notes
○	○	○	Power off	
○	○	🟠	Power on (Driver not loaded and device not attached)	Load Driver and attach device
🟢	○	○	Power on, Driver loaded (device may not be attached)	You may need to attach the device
🟢	(Flashing) 🟡	○	Writing (Rate indicates volume of writes)	Can appear in combination with the Read LED indication
(Flashing) 🟢	○	○	Reading (Rate indicates volume of reads)	Can appear in combination with the Write LED indication
🟢	🟡	🟠	Location Beacon	

GUI Management

IBM High IOPS Management Application software is a free browser-based solution for managing IBM High IOPS Adapters. It is also available from http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174.

The IBM High IOPS Management Application software can perform many management functions, including:

- Firmware upgrades
- Low-level formatting
- Attach and detach actions
- Device status and performance information
- Configure Swap and Paging
- Generate bug reports

Command-line Utilities

Several command-line utilities are included in the installation packages for managing your IBM High IOPS Adapter:

- `fio-attach`
- `fio-beacon`
- `fio-bugreport`
- `fio-detach`
- `fio-format`
- `fio-pci-check`
- `fio-snmp-agentx`
- `fio-status`
- `fio-sure-erase`
- `fio-update-iodrive`

For more information on command-line utilities, see Appendix A- Utilities Reference

Common Maintenance Tasks

The following are the most common tasks for maintaining your IBM High IOPS Adapter using command-line utilities.

NOTE All commands require administrator privileges. Log in as "root" or use sudo to run the commands.

NOTE If you came to this section from the Software Installation section, **return** to that section after you uninstall previous versions of the ioMemory VSL and utilities.

Unloading the ioMemory VSL Driver

To unload the ioMemory VSL, run this command:

```
$ modprobe -r iomemory-vsl
```

Uninstalling the ioMemory VSL RPM Package

Versions 1.2.x

Remove prior versions of the ioMemory VSL software using this command:

```
$ rpm -e iodrive-driver
```

Versions 2.x.x

With versions 2.x.x of the ioMemory VSL, you must specify the kernel version of the package you are uninstalling. Run this command to find the installed driver packages:

```
$  rpm -qa | grep -i iomemory
```

Sample output:

```
iomemory-vsl-2.6.18-194.el5-2.2.2.82-1.0
```

Uninstall the ioMemory VSL by running a command similar to this example (specify the kernel version of the driver you wish to uninstall):

```
$ rpm -e iomemory-vsl-2.6.18-194.el5-2.2.0.82-1.0
```

Uninstalling the ioMemory VSL Utilities, IBM High IOPS Management Application 2.x, and Other Support Packages

To uninstall the support RPM packages, run this command (adding or removing package names as needed):

```
$ rpm -e fio-util fio-snmp-agentx fio-common fio-firmware iomanager-gui
iomanager-jre libfio libfio-doc libfusionjni fio-sysvinit fio-smis
fio-snmp-mib libfio-dev
```

Disabling Auto-Attach

When the ioMemory VSL is installed, it is configured to automatically attach any devices when the ioMemory VSL is loaded. Sometimes you may want to disable the auto-attach feature. To do so:

1. Edit the following file:

   ```
   /etc/modprobe.d/iomemory-vsl.conf
   ```

2. Add the following line to that file:

   ```
   options iomemory-vsl auto_attach=0
   ```

3. Save the file. To re-enable auto-attach, simply edit the file and either remove that line or change it to the following:

   ```
   options iomemory-vsl auto_attach=1
   ```

Unmanaged Shutdown Issues

Unmanaged shutdowns due to power loss or other circumstances can force the IBM High IOPS Adapter to perform a consistency check during the restart. This may take several minutes or more to complete.

Although data written to the IBM High IOPS Adapter is not lost due to unmanaged shutdowns, important data structures may not have been properly committed to the device. This consistency check repairs these data structures.

Disabling the ioMemory VSL

The ioMemory VSL automatically loads by default when the operating system starts. You can disable ioMemory VSL auto-load for diagnostic or troubleshooting purposes.

To disable ioMemory VSL auto-load:

1. Append the following parameter to the kernel command line of your boot loader:

   ```
   iodrive=0
   ```

   The ioMemory VSL won't load, so the device won't be available to users.

   NOTE You can also uninstall the ioMemory VSL to keep it from loading, or move it out of the `/lib/modules/<kernel_version>` directory.

2. Proceed with troubleshooting to correct the problem. If the problem is outdated firmware, use `iodrive=1` to place the ioMemory VSL in minimal mode. You can then use `fio-update-iodrive` or the IBM High IOPS Management Application application to update the firmware.

3. Use either the `fio-attach` utility or the IBM High IOPS Management Application to attach the IBM High IOPS Adapter to the operating system.

Performance and Tuning

IBM High IOPS Adapters provide high bandwidth, and high Input/Output Operations per Second (IOPS), and are specifically designed to achieve low latency.

As IBM High IOPS Adapters improve in IOPS and low latency, the device performance may be limited by operating system settings and BIOS configuration. These settings may need to be tuned to take advantage of the revolutionary performance of IBM High IOPS Adapters.

While IBM devices generally perform well out of the box, this section describes some of the common areas where tuning may help achieve optimal performance.

Disabling DVFS

Dynamic Voltage and Frequency Scaling, or DVFS, are power management techniques that adjust the CPU voltage and/or frequency to reduce power consumption by the CPU. These techniques help conserve power and reduce the heat generated by the CPU, but they adversely affect performance while the CPU transitions between low-power and high-performance states.

These power-savings techniques are known to have a negative impact on I/O latency and maximum IOPS. When tuning for maximum performance, you may benefit from reducing or disabling DVSF completely, even though this may increase power consumption.

DVFS, if available, should be configurable as part of your operating systems power management features as well as within your system's BIOS interface. Within the operating system and BIOS, DVFS features are often found under the Advanced Configuration and Power Interface (ACPI) sections; consult your computer documentation for details.

Limiting ACPI C-States

Newer processors have the ability to go into lower power modes when they are not fully utilized. These idle states are known as ACPI C-states. The `C0` state is the normal, full power, operating state. Higher C-states (`C1`, `C2`, `C3`, etc.) are lower power states.

While ACPI C-states save on power, they are known to have a negative impact on I/O latency and maximum IOPS. With each higher C-state, typically more processor functions are limited to save power, and it takes time to restore the processor to the `C0` state. When tuning for maximum performance you may benefit from limiting the C-states or turning them off completely, even though this may increase power consumption.

Setting ACPI C-State Options

If your processor has ACPI C-states available, you can typically limit/disable them in the BIOS interface (sometimes referred to as a Setup Utility). APCI C-states may be part of of the Advanced Configuration and Power Interface (ACPI) menu; consult your computer documentation for details.

C-States Under Linux

Newer Linux kernels have drivers that may attempt to enable APCI C-states even if they are disabled in the BIOS. You can limit the C-state in Linux (with or without the BIOS setting) by adding the following to the kernel boot options:

```
processor.max_cstate=1
```

In this example, the maximum C-state allowed will be `C1`. To completely disable C-states, use a value of `0`.

Appendix A- Utilities Reference

The ioMemory VSL installation packages include various command-line utilities, installed by default to `/usr/bin`. These provide a number of useful ways to access, test, and manipulate your device.

Attention There are some additional utilities installed in the `/usr/bin` directory that are not listed below. Those additional utilities are dependencies (used by the main VSL utilities), and you should not use them directly unless Customer Support advises you to do so.

Utility	Purpose
`fio-attach`	Makes an IBM High IOPS Adapter available to the OS
`fio-beacon`	Lights the IBM High IOPS Adapter's external LEDs
`fio-bugreport`	Prepares a detailed report for use in troubleshooting problems
`fio-detach`	Temporarily removes an IBM High IOPS Adapter from OS access
`fio-format`	Used to perform a low-level format of an IBM High IOPS Adapter
`fio-pci-check`	Checks for errors on the PCI bus tree, specifically for IBM High IOPS Adapters
`fio-snmp-agentx`	SNMP sub-agent that implements the SNMP FUSION-IODRV-MIB for the ioMemory VSL
`fio-status`	Displays information about the device
`fio-sure-erase`	Clears or purges data from the device
`fio-update-iodrive`	Updates the IBM High IOPS Adapter's firmware

NOTE There are `-h` (Help) and `-v` (Version) options for all of the utilities. Also, `-h` and `-v` cause the utility to exit after displaying the information.

fio-attach

Description

Attaches the IBM High IOPS Adapter and makes it available to the operating system. This creates a block device in `/dev` named `fiox` (where `x` is `a`, `b`, `c`, etc.). You can then partition or format the IBM High IOPS Adapter, or set it up as part of a RAID array. The command displays a progress bar and percentage as it operates.

NOTE In most cases, the ioMemory VSL automatically attaches the device on load and does a scan. You only need to run `fio-attach` if you ran `fio-detach` or if you set the ioMemory VSL's auto_attach parameter to 0.

Syntax

```
fio-attach <device> [options]
```

where `<device>` is the name of the device node (`/dev/fctx`), where *x* indicates the card number: 0, 1, 2, etc. For example, `/dev/fct0` indicates the first IBM High IOPS Adapter installed on the system.

You can specify multiple IBM High IOPS Adapters. For example, `/dev/fct1 /dev/fct2` indicates the second and third IBM High IOPS Adapters installed on the system. You can also use a wildcard to indicate all IBM High IOPS Adapters on the system. For example, `/dev/fct*`

Option	Description
`-q`	Quiet: disables the display of the progress bar and percentage.

fio-beacon

Description

Lights the IBM High IOPS Adapter's LEDs to locate the device. You should first detach the IBM High IOPS Adapter and then run `fio-beacon`.

NOTE This utility always turns the LEDs on, unless you specifically use the `-0` option.

Syntax

```
fio-beacon <device> [options]
```

where `<device>` is the name of the device node (`/dev/fctx`), where *x* indicates the card number: 0, 1, 2, etc. For example, `/dev/fct0` indicates the first IBM High IOPS Adapter installed on the system.

Options	Description
`-0`	Off: (Zero) Turns off the three LEDs
`-1`	On: Lights the three LEDs
`-p`	Prints the PCI bus ID of the device at `<device>` to standard output. Usage and error information may be written to standard output rather than to standard error.

fio-bugreport

Description

Prepares a detailed report of the device for use in troubleshooting problems. The results are saved in the `/tmp` directory in the file that indicates the date and time the utility was run.

Example:

```
/tmp/fio-bugreport-20100121.173256-sdv9ko.tar.bz2
```

Syntax

```
fio-bugreport
```

Notes

This utility captures the current state of the device. When a performance or stability problem occurs with the device, run the `fio-bugreport` utility and send the output to http://www.ibm.com/systems/support for assistance in troubleshooting.

Sample Output

```
-bash-3.2# fio-bugreport
Collecting fio-status -a
Collecting fio-status
Collecting fio-pci-check
Collecting fio-pci-check -v
Collecting fio-read-lebmap /dev/fct0
Collecting fio-read-lebmap -x /dev/stdout/dev/fct0
Collecting fio-read-lebmap -t /dev/fct0
Collecting fio-get-erase-count/dev/fct0
Collecting fio-get-erase-count -b /dev/fct0
Collecting lspci
Collecting lspci -vvvvv
Collecting lspci -tv
Collecting messages file(s)
Collecting procfusion file(s)
Collecting lsmod
Collecting uname -a
Collecting hostname
Collecting sar -r
Collecting sar
Collecting sar -A
Collecting syslog file(s)
Collecting proc file(s)
Collecting procirq file(s)
Collecting dmidecode
Collecting rpm -qa iodrive*
Collecting find /lib/modules
Please send the file /tmp/fio-bugreport-20090921.173256-sdv9ko.tar.bz2
along with your bug report to support@fusionio.com The file is in the /tmp
directory.
```

For example, the filename for a bug report file named `/tmp/fiobugreport-20090921.173256-sdvk0.tar.bz2` indicates the following:

- Date (20090921)
- Time (173256, or 17:32:56)
- Misc. information (`sdv9ko.tar.bz2`)

fio-detach

Description

Detaches the IBM High IOPS Adapter and removes the corresponding `fctx` IBM High IOPS Adapter block device from the OS. The `fio-detach` utility waits until the device completes all read/write activity before executing the detach operation. By default, the command also displays a progress bar and percentage as it completes the detach.

Attention Before using this utility, ensure that the device you want to detach is **NOT** currently mounted and in use.

Syntax

```
fio-detach <device> [options]
```

where `<device>` is the name of the device node (`/dev/fctx`), where *x* indicates the card number: 0, 1, 2, etc. For example, `/dev/fct0` indicates the first IBM High IOPS Adapter installed on the system.

You can specify multiple IBM High IOPS Adapters. For example, `/dev/fct1 /dev/fct2` indicates the second and third IBM High IOPS Adapters installed on the system. You can also use a wildcard to indicate all IBM High IOPS Adapters on the system. For example, `/dev/fct*`

Options	Description
`-i`	Immediate: Causes a forced immediate detach (does not save metadata). This will fail if the device is in use by the OS.
`-q`	Quiet: Disables the display of the progress bar and percentage.

Notes

With this version of ioMemory VSL, attempting to detach an IBM High IOPS Adapter may fail with an error indicating that the device is busy. This typically may occur if the IBM High IOPS Adapter is part of a software RAID (0,1,5) volume, is mounted, or some process has the device open.

The tools fuser, mount, and lsof can be helpful in determining what is holding the device open.

fio-format

Description

NOTE IBM High IOPS Adapters ship pre-formatted, so `fio-format` is generally not required except to change the logical size or block size of a device, or to erase user data on a device. To ensure the user data is truly erased, use `fio-sure-erase`.

Performs a low-level format of the IBM High IOPS Adapter. By default, `fio-format` displays a progress-percentage indicator as it runs.

Attention Use this utility with care, as it deletes all user information on the device.

NOTE Using a larger block (sector) size, such as 4096 bytes, can significantly reduce worst-case ioMemory VSL host memory consumption. However, some applications are not compatible with non-512-byte sector sizes.

NOTE If you do not include the `-s` option, the device size defaults to the advertised capacity. If used, the `-s` option must include the size or percentage indicators.

Syntax

```
fio-format [options] <device>
```

where `<device>` is the name of the device node (`/dev/fctx`), where *x* indicates the device number: 0, 1, 2, etc. For example, `/dev/fct0` indicates the first IBM High IOPS Adapter installed on the system.

Options	Description
`-b <size B|K>`	Set the block (sector) size, in bytes or KiBytes (base 2). The default is 512 bytes. For example: `-b 512B` or `-b 4K` (`B` in `512B` is optional).
`-f`	Force the format size, bypassing normal checks and warnings. This option may be needed in rare situations when `fio-format` does not proceed properly. (The "Are you sure?" prompt still appears unless you use the `-y` option.)
`-q`	Quiet mode: Disable the display of the progress-percentage indicator.
`-s <size M|G|T|%>`	Set the device capacity as a specific size (in TB, GB, or MB) or as a percentage of the advertised capacity: • `T` Number of terabytes (TB) to format • `G` Number of gigabytes (GB) to format • `M` Number of megabytes (MB) to format • `%` Percentage, such as 70% (the percent sign must be included).
`-o <size B|K|M|G|T|%>`	Over-format the device size (to greater than the advertised capacity), where the maximum size equals the maximum physical capacity. If a percentage is used, it corresponds to the maximum physical capacity of the device. (Size is required for the `-o` option; see the `-s` option above for size indicator descriptions.) Attention Before you use this option, please discuss your use case with Customer Support by contacting http://www.ibm.com/systems/support.
`-y`	Auto-answer "yes" to all queries from the application (bypass prompts).

fio-pci-check

Description

Checks for errors on the PCI bus tree, specifically for IBM High IOPS Adapters. This utility displays the current status of each IBM High IOPS Adapter. It also prints the standard PCI Express error information and resets the state.

NOTE It is perfectly normal to see a few errors (perhaps as many as five) when `fio-pci-check` is initially run. Subsequent runs should reveal only one or two errors during several hours of operation.

Syntax

```
fio-pci-check [options]
```

Options	Description
`-d <value>`	1 = Disable the link; 0 = bring the link up (Not recommended)
`-e`	Enable PCI-e error reporting.
`-f`	Scan every device in the system.
`-i`	Print the device serial number. This option is invalid when the ioMemory VSL is loaded.
`-r`	Force the link to retrain.
`-v`	Verbose: Print extra data about the hardware.

fio-snmp-agentx

Description

This utility is an SNMP sub-agent that implements the SNMP FUSION-IODRV-MIB for the ioMemory VSL. `fio-snmp-agentx` communicates with the SNMP master agent via the agentx protocol.

Syntax

```
fio-snmp-agentx [options]
```

Option	Description
`-f`	Force the sub-agent to run in the foreground instead of as a daemon.
`-l <log file>`	Log file to use.
`-s`	Send errors to stderr instead of to syslog.

fio-status

Description

Provides detailed information about the installed devices. This utility operates on either `fctx` or `fiox` devices. The utility depends on running as root and having the ioMemory VSL loaded. If no ioMemory VSL is loaded, a smaller set of status information is returned.

`fio-status` provides alerts for certain error modes, such as a minimal-mode, read-only mode, and write-reduced mode, describing what is causing the condition.

Syntax

```
fio-status [<device>] [<options>]
```

where `<device>` is the name of the device node (`/dev/fctx`), where *x* indicates the card number: 0, 1, 2, etc. For example, `/dev/fct0` indicates the first IBM High IOPS Adapter installed on the system.

If `<dev>` is not specified, `fio-status` displays information for all cards in the system. If the ioMemory VSL is not loaded, this parameter is ignored.

Options	Description
`-c`	Count: Report only the number of IBM High IOPS Adapters installed.
`-d`	Show basic information set plus the total amount of data read and written (lifetime data volumes). This option is not necessary when the -a option is used.
`-fk`	Format key: Force alternate <"key=value"> format output, which may be easier for scripts and parsing programs to read.
`-a`	Print all available information for each device.

Basic Information: If no options are used, `fio-status` reports the following basic information:

- Number and type of devices installed in the system
- ioMemory VSL version

Adapter information:

- Adapter type
- Product number
- External power status
- PCIe power limit threshold (if available)
- Connected IBM High IOPS Adapters

Block device information:

- Attach status
- Product name
- Product number
- Serial number
- PCIe address and slot

- Firmware version
- Size of the device, out of total capacity
- Internal temperature (average and maximum, since ioMemory VSL load) in degrees Centigrade
- Health status: healthy, nearing wearout, write-reduced or read-only
- Reserve capacity (percentage)
- Warning capacity threshold (percentage)

Data Volume Information: If the -d option is used, the following data volume information is reported *in addition* to the basic information:

- Physical bytes written
- Physical bytes read

All Information: If the `-a` option is used, all information is printed, which includes the following information *in addition* to basic and data volume information:

Adapter information:

- Manufacturer number
- Part number
- Date of manufacture
- Power loss protection status
- PCIe bus voltage (avg, min, max)
- PCIe bus current (avg, max)
- PCIe bus power (avg, max)
- PCIe power limit threshold (watts)
- PCIe slot available power (watts)
- PCIe negotiated link information (lanes and throughput)

Block device information:

- Manufacturer's code
- Manufacturing date
- Vendor and sub-vendor information
- Format status and sector information (if device is attached)
- FPGA ID and Low-level format GUID

- PCIe slot available power
- PCIe negotiated link information
- Card temperature, in degrees Centigrade
- Internal voltage (avg and max)
- Auxiliary voltage (avg and max)
- Percentage of good blocks, data and metadata
- Lifetime data volume statistics
- RAM usage

Error Mode Information: If the ioMemory VSL is in minimal mode, read-only mode, or write-reduced mode when `fio-status` is run, the following differences occur in the output:

- Attach status is "Status unknown: Driver is in MINIMAL MODE:"
- The reason for the minimal mode state is displayed (such as "Firmware is out of date. Update firmware.")
- "Geometry and capacity information not available." is displayed.
- No media health information is displayed.

fio-sure-erase

Attention As a best practice, do not use this utility if there are any IBM High IOPS Adapters installed in the system that you do not want to clear or purge. First remove any devices that you do not want to accidentally erase. Once the data is removed with this utility it is gone forever. **It is not recoverable.**

Attention Before you use this utility, be sure to back up any data that you wish to preserve.

NOTE After using `fio-sure-erase`, format the device using `fio-format` before using the device again.

Attention If the device is in Read-only mode, perform a format using `fio-format` before running `fio-sure-erase`. If the device is in Minimal mode, then `fio-sure-erase` cannot erase the device. Updating the firmware may take the device out of Minimal Mode. If the device remains in Minimal mode, contact Customer Support at http://www.ibm.com/systems/support for further assistance.

In order to run `fio-sure-erase`, the block device **must be detached**. See the `fio-detach` section for more information.

Description

The `fio-sure-erase` is a command-line utility that securely removes data from IBM High IOPS Adapters. It complies with the "Clear" and "Purge" level of destruction from the following standards:

1. DOD 5220.22-M - Comply with instructions for Flash EPROM
2. NIST SP800-88- Comply with instructions for Flash EPROM

See below for more information on Clear and Purge support.

Syntax

```
fio-sure-erase [options] <device>
```

Where `<device>` is the name of the device node (`/dev/fctx`), where *x* indicates the card number: 0, 1, 2, etc. For example, `/dev/fct0` indicates the first IBM High IOPS Adapter installed on the system. Use `fio-status` to view this device node.

> NOTE **Products with Multiple Devices**: `fio-sure-erase` works on individual IBM High IOPS Adapters. For example, if you are planning to purge an IBM High IOPS Duo Adapter, you will need to perform this operation on each of the product's two IBM High IOPS Adapters.

Options	Description
`-p`	Purge instead of Clear: performs a write followed by an erase. For more information on Purge, see below. Attention Purging the device may take hours to accomplish, depending on the size of the device that needs to be purged.
`-y`	No confirmation: do not require a yes/no response to execute the utility.
`-q`	Quiet: do not display the status bar.

> NOTE If you run `fio-sure-erase` with no options, a Clear is performed. For more information, see below.

Each block of memory consists of uniform 1 bits or 0 bits.

Clear Support

A "Clear" is the default state of running `fio-sure-erase` (with no options), and refers to the act of performing a full low-level erase (every cell pushed to "1") of the entire NAND media, including retired erase blocks.

Metadata that is required for operation will not be destroyed (media event log, erase counts, physical bytes read/written, performance and thermal history), but any user-specific metadata will be destroyed.

The following describes the steps taken in the Clear operation:

1. Creates a unity map of every addressable block (this allows `fio-sure-erase` to address every block, including previously unmapped bad blocks).
2. For each block, performs an erase cycle (every cell is pushed to "1").
3. Restores the bad block map.
4. Formats the device (the purpose of this is to make the device usable again, the utility erases all of the headers during the clear).

Purge Support

A "Purge" is implemented by using the `-p` option with `fio-sure-erase`. Purge refers to the act of first overwriting the entire NAND media (including retired erase blocks) with a single character (every cell written to logical "0"), and then performing a full chip erase (every cell pushed to "1") across all media (including retired erase blocks).

Metadata that is required for operation will **not** be destroyed (media event log, erase counts, physical bytes read/written, performance and thermal history), but any user-specific metadata will be destroyed.

The following describes the steps taken in the Purge operation:

1. Creates a unity map of every addressable block (this allows `fio-sure-erase` to address every block, including previously unmapped bad blocks).
2. For each block, performs a write cycle (every cell written to "0").
3. For each block, performs an erase cycle (every cell pushed to "1").
4. Restores the bad block map.
5. Formats the drive (the purpose of this is to make the drive usable again, the utility erases all of the headers during the clear).

fio-update-iodrive

Attention You should back up the data on the IBM High IOPS Adapter prior to any upgrade as a precaution.

Description

Updates the IBM High IOPS Adapter's firmware. This utility scans the PCIe bus for all IBM High IOPS Adapters and updates them. A progress bar and percentage are shown for each device as the update completes.

Attention It is extremely important that the power not be turned off during a firmware upgrade, as this could cause device failure. If a UPS is not already in place, consider adding one to the system prior to performing a firmware upgrade.

Attention Note that when running multiple firmware upgrades in sequence, it is critical to load the driver after each firmware upgrade step. Otherwise the on-drive format will not be changed, and there will be data loss.

Attention Do not use this utility to downgrade the IBM High IOPS Adapter to an earlier version of the firmware. Doing so may result in data loss and void your warranty.

Attention The default action (without using the `-d` or `-s` option) is to upgrade all IBM High IOPS Adapters with the firmware contained in the `<iodrive_version.fff>` file. Confirm that all devices need the upgrade prior to running the update. If in doubt, use the `-p` (Pretend) option to view the possible results of the update.

Attention You must detach all IBM High IOPS Adapters before updating the firmware.

Attention **Upgrade Path**: There is a specific upgrade path that you must take when upgrading IBM High IOPS Adapter. Consult the *Release Notes* for this ioMemory VSL release before upgrading IBM High IOPS Adapters.

NOTE If you receive an error message when updating the firmware that instructs you to update the midprom information, contact Customer Support.

To update one or more specific devices:

- If the ioMemory VSL is loaded, use the `-d` option with the device number.

Syntax

```
fio-update-iodrive [options] <iodrive_version.fff>
```

where `<iodrive_version.fff>` is the path and firmware archive file provided by IBM. The default path is `/usr/share/fio/firmware`. This parameter is required.

Options	Description
`-d`	Updates the specified devices (by `fctx`, where *x* is the number of the device shown in `fio-status`). If this option is not specified, all devices are updated. Attention Use the `-d` or `-s` options with care, as updating the wrong IBM High IOPS Adapter could damage your device.
`-f`	Force upgrade (used primarily to downgrade to an earlier firmware version). If the ioMemory VSL is not loaded, this option also requires the `-s` option. Attention Use the -f option with care, as it could damage your card.
`-l`	List the firmware available in the archive.
`-p`	Pretend: Shows what updates would be done. However, the actual firmware is not modified.
`-q`	Runs the update process without displaying the progress bar or percentage.
`-y`	Confirm all warning messages.
`-s`	Updates the devices in the specified slots using '*' as a wildcard for devices. The slots are identified in the following PCIe format (as shown in lspci): `[[[[<domain>]:]<bus>]:][<slot>][.[<func>]]`

All three external LED indicators light up during the update process.

Appendix B- Monitoring the Health of ioMemory Devices

This section describes how the health of IBM High IOPS Adapters can be measured and monitored in order to safeguard data and prolong device lifetime.

NAND Flash and Component Failure

An IBM High IOPS Adapter is a highly fault-tolerant storage subsystem that provides many levels of protection against component failure and the loss nature of solid-state storage. As in all storage subsystems, component failures may occur.

By pro-actively monitoring device age and health, you can ensure reliable performance over the intended product life.

Health Metrics

The ioMemory VSL manages block retirement using pre-determined retirement thresholds. The IBM High IOPS Management Application and the `fio-status` utilities show a health indicator that starts at 100 and counts down to 0. As certain thresholds are crossed, various actions are taken.

At the 10% healthy threshold, a one-time warning is issued. See the Health Monitoring Techniques section below for methods for capturing this alarm event.

At 0%, the device is considered unhealthy. It enters *write-reduced* mode, which somewhat prolongs its lifespan so data can be safely migrated off. In this state the IBM High IOPS Adapter behaves normally, except for the reduced write performance.

After the 0% threshold, the device will soon enter *read-only* mode – any attempt to write to the IBM High IOPS Adapter causes an error. Some filesystems may require special mount options in order to mount a read-only block device in addition to specifying that the mount should be read-only.

For example, under Linux, `ext3` requires that `"-o ro,noload"` is used. The `"noload"` option tells the filesystem to not try and replay the journal.

Read-only mode should be considered a final opportunity to migrate data off the device, as device failure is more likely with continued use.

The IBM High IOPS Adapter may enter failure mode. In this case, the device is offline and inaccessible. This can be caused by an internal catastrophic failure, improper firmware upgrade procedures, or device wearout.

NOTE For service or warranty-related questions, contact the company form which you purchased the device.

NOTE For products with multiple IBM High IOPS Adapters, these modes are maintained independently for each device.

Health Monitoring Techniques

`fio-status`: Output from the `fio-status` utility shows the health percentage and device state. These items are referenced as `"Media status"` in the sample output below.

```
Found 3 ioMemory devices in this system
Fusion-io driver version: 3.0.6 build 364

Adapter: Single Adapter
        Fusion-io ioDrive 1.30TB, Product Number:F00-001-1T30-CS-0001,
SN:1133D0248, FIO SN:1134D9565
...
Media status: Healthy; Reserves: 100.00%, warn at 10.00%; Data: 99.12%
Lifetime data volumes:
   Physical bytes written: 6,423,563,326,064
   Physical bytes read   : 5,509,006,756,312
```

IBM High IOPS Management Application: In the Device Report tab, look for the Reserve Space percentage in the right column. The higher the percentage, the healthier the drive is likely to be.

SNMP/SMI-S: On Windows or Linux, see the corresponding appendix for details on how to configure SNMP or SMI-S health indicators.

The following Health Status messages are produced by the `fio-status` utility:

- Healthy
- Read-only
- Reduced-write
- Unknown

Software RAID and Health Monitoring

Software RAID stacks are typically designed to detect and mitigate the failure modes of traditional storage media. The IBM High IOPS Adapter attempts to fail as gracefully as possible, and these new failure mechanisms are compatible with existing software RAID stacks. An IBM High IOPS Adapter in a RAID group will fail to receive data at a sufficient rate if a) the device is in a write-reduced state, and b) it is participating in a write-heavy workload. In this case, the device will be evicted from the RAID group. A device in read-only mode will be evicted when write I/Os are returned from the device as failed. Catastrophic failures are detected and handled just as though they are on traditional storage devices.

Appendix C- Using Module Parameters

The following table describes the module parameters you can set by editing the `/etc/modprobe.d/iomemory-vsl.conf` file and changing their values.

Each module parameter in the configuration file must be preceded by `options iomemory-vsl`. The `/etc/modprobe.d/iomemory-vsl.conf` file has some example parameters that are commented out. You may use these examples as templates and/or uncomment them in order to use them.

NOTE These changes must be completed before the ioMemory VSL is loaded in order to take effect.

Module Parameter	Default (min/max)	Description
`auto_attach`	1	`1` = Always attach the device on on driver load. `0` = Don't attach the device on driver load.
`fio_dev_wait_timeout_secs`	30	Number of seconds to wait for `/dev/fio*` files to show up during driver load. For systems not using `udev`, this should be set to `0` to disable the timeout and avoid an unneeded pause during driver load.
`force_minimal_mode`	0	`1` = Force minimal mode on the device. `0` = Do not force minimal mode on the device.
`parallel_attach`	1	`1` = Enable parallel attach of multiple devices. `0` = Disable parallel attach of multiple devices.
`preallocate_memory`	No devices selected	For the selected devices, pre-allocate all memory necessary to have the drive usable as swap space. Where the `<value>` for this parameter is a comma-separated list of device serial numbers.
`tintr_hw_wait`	0 (0, 255)	Interval (microseconds) to wait between hardware interrupts. Also known as interrupt coalescing. `0` is off.
`use_workqueue`	3 (1 or 3)	Linux only: `3` = use standard OS I/O elevators; `0` = bypass.

NOTE Other than `preallocate_memory`, module parameters are global — they apply to all IBM devices in the computer.

Appendix D- Setting Up SNMP

The `fio-snmp-agentx` SNMP agent is an RFC 2741-compliant AgentX sub-agent. It can work with any RFC-compliant SNMP agent, such as Net-SNMP. The master SNMP agent defers queries to `fio-snmp-agentx` for supported MIBs.

SNMP Master Agent

The `fio-snmp-agentx`, provided in the `fio-util` package, requires an already-installed SNMP master agent. The SNMP master agent must support and be configured for AgentX connections (see http://www.ietf.org/rfc/rfc2741.txt). The `fio-snmp-agentx` is tested and verified with Net-SNMP, which is the typical SNMP agent provided with most Linux distributions.

There are several agents available that support this functionality. If you choose to use Net-SNMP, then use the instructions in the following sections to configure and launch it.

Launching the SNMP Master Agent

Install the Net-SNMP package using the package manager for your version of Linux.

Red Hat

Use the following command to install Net-SNMP on Red Hat:

```
yum install net-snmp rsync
```

Other Linux Versions

Use the standard system package manager to install the `Net-SNMP` package on your Linux distribution. The `fio-snmp-mib` package places MIB files in `/usr/share/fio/mib`.

Configuring the Master Agent

You can configure the Net-SNMP master agent daemon to set the network communications parameters, security, and other options by using the `snmpd.conf` text file. The location of this file is system-dependent; often it is in `/etc/snmp` or `/usr/share/snmp`.

A simple `snmpd` configuration file might include the following:

```
# set standard SNMP variables
syslocation "Data room, third rack"
syscontact itguy@example.com
# required to enable the AgentX protocol
master agentx
agentxsocket tcp:localhost:705
#set the port that the agent listens on (defaults to 161)
agentaddress 161
# simple access control (some form of access control is required)
rocommunity public
```

Running the Master Agent

Once you install and configure the master agent, you must start or restart the `snmpd` daemon for the new parameters to take effect. You can simply run `snmpd` from its installed location (often `/usr/sbin` – see the `snmpd` man page for options). It typically needs root privileges to run properly. You can also use the `snmpd` startup script in `/etc/init.d or /etc/rc.d/init.d`. If you are concerned about security, use the more advanced SNMPv3 access control instead of the `rocommunity` and `rwcommunity` access control directives as outlined in the relevant man page.

Fusion-io SNMP AgentX Subagent

Attention The SNMP agent requires the libvsl RPM package. This should have been installed as part of the ioMemory VSL installation. It is available for download at http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174

Installing the IBM SNMP Subagent

1. Download the IBM SNMP packages from http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174.
2. Install the package using your operating systems package manager. For instance, on Red Hat, run the following:

   ```
   rpm -Uvh fio-snmp-*.rpm
   ```

The SNMP package places its MIB files in `/usr/share/fio/mib`.

Running and Configuring the IBM SNMP Subagent

1. Configure the subagent by creating a fio-snmp-agentx.conf file.
2. Store this .conf file in the `/opt/fio/etc/snmp` directory.
3. At a minimum, set the agent network parameters in this file similar to the following:

```
# required to enable the AgentX protocol

agentxsocket tcp:localhost:705
```

This must match the AgentX network parameters in the `snmpd.conf` file for the master agent. For further AgentX configuration information, consult the man pages or visit http://www.net-snmp.org.

The `fio-snmp-agentx startup` script will launch automatically at boot time once the installation and configuration is complete.

Manually Running the IBM SNMP Subagent

If you need to run the IBM SNMP Subagent manually, follow these steps:

1. After the SNMP master agent is started, start the subagent by running this command:

```
/usr/bin/fio-snmp-agentx
```

   This command launches the IBM subagent using the Net-SNMP configuration file named `fio-snmp-agentx.conf`. This file must reside in one of the `/opt/fio/etc/snmp` directory.

2. You can now view the IBM High IOPS Adapter management information using an SNMP MIB browser or by using a network management system accessing `FIOioDrv.mib` (in `/usr/share/fio/mib`).

Subagent Log File

The IBM SNMP subagent can maintain a log file regarding its own activities. This file is separate from the MIB, as it includes entries on the subagent's communications with the master agent, including any errors or intermittent issues.

To have the subagent maintain this log file, include the `-l` parameter and a path to the log file as part of the command in running the subagent. For example, this command:

```
fio-snmp-agentx -l /usr/snmp/subagent.log
```

keeps the subagent log file as `subagent.log`, in the `/usr/snmp` directory.

The IBM SNMP subagent is now ready to monitor your device.

Using the SNMP Sample Config Files

When you install SNMP, the following sample config files are available:

- `/usr/share/doc/fio-snmp-agentx/conf/snmpd.conf/` (master agent)
- `/usr/share/doc/fio-snmp-agentx/conf/fio-snmp-agentx.conf/` (sub-agent)

To customize and use the sample config files,

1. Rename your `snmpd.conf` file (such as to `snmpd-orig.conf`) and your `fio-snmp-agentx.conf` file (such as to `fio-snmp-agentx-orig.conf`). The `snmpd.conf` file usually reside in `/etc/snmp` or `/usr/share/snmp`. The `fio-snmp-agentx.conf` file resides in the `/opt/fio/etc/snmp` directory.
2. From the `/usr/share/doc/fio-snmp-agentx/conf/` directory, copy the sample `snmpd.conf` file and the sample `fio-snmp-agentx.conf` file to the appropriate directories.
3. Edit the sample files you copied and save your changes as `snmpd.conf` and `fio-snmp-agentx.conf`.

Enabling SNMP Test Mode

When the SNMP Agentx runs, it reads the `fio-snmp-agentx config` file:

```
####################################################################
# Example config file for fio-snmp-agentx SNMP AgentX subagent.
#
# Fusion-io, Inc. #

agentxsocket tcp:localhost:705

# test_mode_enabled
# set to 1, true or yes to enable 0, false or no to disable (default: false)
test_mode_enabled true
# traps_enabled
traps_enabled true
# testmode_file
# name of test mode file (default: testmode.ini)
testmode_file testmode.ini

# update_delay
# delay between agent polling requests in milliseconds (default: 250)
update_delay 100

# mib_select
# set to fio for FUSIONIO-IODRV-MIB or cpq for CPQIODRV-MIB (default: fio)
mib_select fio
####################################################################
```

Conditions for test mode are described below.

1. If the Admin has set the `test_mode_enabled` parameter from TRUE to FALSE, the IBM SNMP does not try to run test mode. Instead, it continues processing data as usual from the ioMemory VSL, storing the data in the MIB.
2. If the CONF file says that `test_mode_enabled` is TRUE, the SNMP subagent reads the `testmode.ini` is read periodically by the subagent to check for any changes. A sample testmode.ini file is installed in `/usr/share/doc/fio-snmp-agentx/conf`.
3. If the `testmode.ini` file shows the test mode is set to ON, then it engages the test mode.

4. If test mode is ON, the SNMP Agentx reads the next line, `TestModeIndex`, to identify which ioMemory module (DIMM) to test. The number in this parameter is the PCIe device number shown using `fio-status` such as:

   ```
   PCI:01:00.0
   ```

   The first two numerals identify the PCIe bus number (in this case, 01). This bus number is reported in *hexadecimal*, whereas the `TestModeIndex` in the `testmode.ini` file must be specified in decimal. The converted number should be entered into `testmode.ini`. The `TestModeIndex` must be a valid bus number of an IBM High IOPS Adapter installed in the system.

The IBM SNMP subagent now replaces any existing ioMemory VSL data it may have (for the IBM High IOPS Adapter specified by TestModeIndex) with any populated fields in the list of parameters. If a field is not populated, Agentx retains the existing data and reports it to the MIB. If there is a value in a field, then the Agentx replaces that data and reports it to the MIB.

The subagent continues in test mode until the `.INI` file parameter is set to OFF. The test mode information is described in the `testmode.ini` file :

```
# SNMP Test Mode sample file.
# These values may be used to test the SNMP subsystem when it is in test
mode.

[SNMP Agent Test Mode]
TestMode                = off
TestModeIndex           = 0

# InfoState: Note that the following states may change, but current
definitions are:
#    0 = unknown
#    1 = detached
#    2 = attached
#    3 = minimal mode
#    4 = error
#    5 = detaching
#    6 = attaching
#    7 = scanning
#    8 = formatting
#    9 = updating firmware
#   10 = attach
#   11 = detach
#   12 = format
#   13 = update
```

```
InfoState                = 2

InfoInternalTemp         = 45
InfoAmbientTemp          = 35
InfoWearoutIndicator     = 2  ; 2=normal, 1=device is wearing out.
InfoWritableIndicator    = 2  ; 2=normal, 1=non-writable, 0=write-reduced,
3=unknown
InfoFlashbackIndicator   = 2  ; 2=normal, 1=flashback protection degraded.

ExtnTotalPhysCapacityU   = 23
ExtnTotalPhysCapacityL   = 215752192
ExtnUsablePhysCapacityU  = 21
ExtnUsablePhysCapacityL  = 7852192
ExtnUsedPhysCapacityU    = 4
ExtnUsedPhysCapacityL    = 782330816
ExtnTotalLogCapacityU    = 18
ExtnTotalLogCapacityL    = 2690588672
ExtnAvailLogCapacityU    = 14
ExtnAvailLogCapacityL    = 3870457856

ExtnBytesReadU       = 18
ExtnBytesReadL       = 3690588672
ExtnBytesWrittenU  = 4
ExtnBytesWrittenL   = 2578550816

InfoHealthPercentage     = 95

InfoMinimalModeReason    = 7            ; 0=unknown, 1=fw out of date,
2=low power,
                                        ; 3=dual plane failure, 5=internal,
6=card limit,
                                        ; 7=not in minimal mode,
8=unsupported OS,
                                        ; 9=low memory

InfoReducedWriteReason   = 0            ; 0=none, 1=user requested, 2=no md
blocks,
                                        ; 3=no memory, 4=failed die,
5=wearout,
                                        ; 6=adapter power, 7=internal,
8=power limit
```

```
InfoMilliVolts           = 12000
InfoMilliVoltsPeak       = 12100
InfoMilliVoltsMin        = 11900
InfoMilliWatts           = 6000
InfoMilliWattsPeak       = 15000
InfoMilliAmps            = 500
InfoMilliAmpsPeak        = 1000

InfoAdapterExtPowerPresent   = 1      ; 1=present, 2=absent

InfoPowerlossProtectDisabled = 2      ; 1=powerloss protection available but
disabled
                                      ; 2=any other powerloss protection
condition
```

SNMP MIB Support

The following SNMP MIB fields are supported in Linux:

fusionIoDimmMibRevMajor	fusionIoDimmInfoAdapterType
fusionIoDimmMibRevMinor	fusionIoDimmInfoAdapterPort
fusionIoDimmMIBCondition	fusionIoDimmInfoAdapterSerialNumber
fusionIoDimmInfoIndex	fusionIoDimmInfoAdapterExtPowerPresent
fusionIoDimmInfoStatus	fusionIoDimmInfoPowerlossProtectDisabled
fusionIoDimmInfoName	fusionIoDimmInfoInternalTempHigh
fusionIoDimmInfoSerialNumber	fusionIoDimmInfoAmbientTemp
fusionIoDimmInfoPartNumber	fusionIoDimmInfoPCIBandwidthCompatibility
fusionIoDimmInfoSubVendorPartNumber	fusionIoDimmInfoPCIPowerCompatibility
fusionIoDimmInfoSparePartNumber	fusionIoDimmInfoActualGoverningLevel
fusionIoDimmInfoAssemblyNumber	fusionIoDimmInfoLifespanGoverningLevel
fusionIoDimmInfoFirmwareVersion	fusionIoDimmInfoPowerGoverningLevel
fusionIoDimmInfoDriverVersion	fusionIoDimmInfoThermalGoverningLevel
fusionIoDimmInfoUID	fusionIoDimmInfoLifespanGoverningEnabled
fusionIoDimmInfoState	fusionIoDimmInfoLifespanGoverningTgtDate
fusionIoDimmInfoClientDeviceName	fusionIoDimmExtnIndex

fusionIoDimmInfoBeacon	fusionIoDimmExtnTotalPhysCapacityU
fusionIoDimmInfoPCIAddress	fusionIoDimmExtnTotalPhysCapacityL
fusionIoDimmInfoPCIDeviceID	fusionIoDimmExtnTotalLogCapacityU
fusionIoDimmInfoPCISubdeviceID	fusionIoDimmExtnTotalLogCapacityL
fusionIoDimmInfoPCIVendorID	fusionIoDimmExtnBytesReadU
fusionIoDimmInfoPCISubvendorID	fusionIoDimmExtnBytesReadL
fusionIoDimmInfoPCISlot	fusionIoDimmExtnBytesWrittenU
fusionIoDimmInfoWearoutIndicator	fusionIoDimmExtnBytesWrittenL
fusionIoDimmInfoFlashbackIndicator	fusionIoDimmExtnFormattedBlockSize
fusionIoDimmInfoWritableIndicator	fusionIoDimmExtnCurrentRAMUsageU
fusionIoDimmInfoInternalTemp	fusionIoDimmExtnCurrentRAMUsageL
fusionIoDimmInfoHealthPercentage	fusionIoDimmExtnPeakRAMUsageU
fusionIoDimmInfoMinimalModeReason	fusionIoDimmExtnPeakRAMUsageL
fusionIoDimmInfoReducedWriteReason	fusionIoDimmWearoutTrap
fusionIoDimmInfoMilliVolts	fusionIoDimmNonWritableTrap
fusionIoDimmInfoMilliVoltsPeak	fusionIoDimmFlashbackTrap
fusionIoDimmInfoMilliVoltsMin	fusionIoDimmTempHighTrap
fusionIoDimmInfoMilliWatts	fusionIoDimmTempOkTrap
fusionIoDimmInfoMilliWattsPeak	fusionIoDimmErrorTrap
fusionIoDimmInfoMilliAmps	fusionIoDimmPowerlossProtectTrap
fusionIoDimmInfoMilliAmpsPeak	

Appendix E- SMI-S Interface

Attention The SMI-S provider has a new CIM namespace: `root/fio`

Introduction to the SMI-S Interface

The SMI-S interface is based on Web-Based Enterprise Management (WBEM) and provides a Common Information Model (CIM) model that represents the IBM High IOPS Adapter and associated software, in accordance with existing Distributed Management Task Force (DMTF), Storage Networking Industry Association (SNIA), and Storage Management Initiative Specification (SMI-S) standards. This model permits backward-compatible extension, accommodating new hardware and software features developed by IBM.

It is assumed that the reader is versed in WBEM, SMI-S and DMTF standards. This document and associated model may change at any time as feedback is received.

References

CIM Schema v2.22
http://www.dmtf.org/standards/cim/cim_schema_v2220

DMTF DSP1011, Physical Asset Profile
http://www.dmtf.org/standards/published_documents/DSP1011_1.0.2.pdf

DMTF DSP1023, Software Inventory Profile
http://www.dmtf.org/standards/published_documents/DSP1023_1.0.1.pdf

DMTF DSP1033, Profile Registration Profile
http://www.dmtf.org/standards/published_documents/DSP1033_1.0.0.pdf

DMTF DSP1075 PCI Device Profile
http://www.dmtf.org/standards/published_documents/DSP1075_1.0.0.pdf

DMTF DSP1002, Diagnostics Profile
http://www.dmtf.org/standards/published_documents/DSP1002_2.0.0.pdf

SMI-S v1.4 Architecture
http://www.snia.org/sites/default/files/SMI-Sv1.4r6_Architecture.book_.pdf

SMI-S v1.4 Common Profiles
http://www.snia.org/sites/default/files/SMI-Sv1.4r6_CommonProfiles.book_.pdf

SMI-S v1.4 Host Profiles
http://www.snia.org/sites/default/files/SMI-Sv1.4r6_Host.book_.pdf

SMI-S v1.4 Common Diagnostic Model
http://www.dmtf.org/standards/mgmt/cdm/

Installing the SMI-S WMI Provider on Linux

The IBM SMI-S provider implements a standard WBEM interface based on DMTF and SNIA standards for remote management of IBM High IOPS Adapters. The provider is a CMPI-based provider and should work with popular CIMOMs including SFCB, OpenPegasus, and OpenWBEM. We also have a version that supports WMI on Windows.

In order to use the provider, a CIMOM must be installed and configured. The provider and associated MOF files must then be installed and registered with the CIMOM. The MOF files define the CIM objects available from the SMI-S provider.

Initially, the provider has been tested with SFCB on Linux and WMI on Windows.

1. Install the SFCB CIM broker (CIMOM).
    - On RHEL-like: Install `sblim-sfcc`, `sblim-sfcc-devel`, `cim-schema-2.21.0`, `sblim-sfcb`, `sblim-indication-helper`, `sblim-cmpi-base` and `sblim-testsuite`

      NOTE RPMs available for SLES, RHEL, and others on OpenSUSE Build Service.

2. Configure SFCB: Copy the file (`sfcb.cfg`) to `/etc/sfcb`

3. Install SMI-S provider: Install `fio-smis` package from IBM distribution and copy `/usr/lib/fio/libfiosmis.so` to `/usr/lib/sfcb` (or `/usr/lib64/sfcb` as appropriate).

4. Register the SMI-S provider with SFCB:

```
cd /usr/share/fio/cim

sh /usr/share/sblim-cmpi-base/provider-register.sh -r
fiosmis.registration cimv222-dmtf.mof
```

5. Restart SFCB:

```
$ /etc/init.d/sfcb restart
```

Linux Testing

The `wbemcli` utility can be used to test the SMI-S provider.

1. Query the provider for the ioMemory VSL version and the firmware version for each IBM High IOPS Adapter in the system:

   ```
   $ wbemcli -nl ei http://localhost:5988/root/cimv2:FIO_SoftwareIdentity
   ```

   The output should look something like this (values may change as development continues):

   ```
   localhost:5988/root/cimv2:FIO_SoftwareIdentity.InstanceID="Fusion-io
   drive driver"
   -InstanceID="Fusion-io drive driver"
   -TargetTypes=
   -OtherExtendedResourceTypeDescription=
   -MinExtendedResourceTypeRevisionNumber=
   . . .
   ```

2. Query the SMI-S provider for each IBM High IOPS Adapter's health:

   ```
   wbemcli -nl ei http://localhost:5988/root/cimv2:FIO_IoMemoryPort
   ```

   The output should look something like this (values may change as development continues):

   ```
   localhost:5988/root/cimv2:FIO_IoMemoryPort.DeviceID="fct1",CreationClassN
   . .
   ```

3. Query capacity and usage counters of a specific IBM High IOPS Adapter (in this case fct0):

   ```
   $ wbemcli -nl gi
   http://localhost:5988/root/cimv2:FIO_SSDStatistics.InstanceID="fct0"
   ```

   The output should look something like this (values may change as development continues):

   ```
   localhost:5988/root/cimv2:FIO_SSDStatistics.InstanceID="fct0"
   -InstanceID="fct0"
   -WriteOperations=0
   -ReadOperations=6887
   -PhysicalMBytesWritten=1523769
   . . .
   ```

The Linux SMI-S provider can be tested remotely with the `wbemcli` utility by replacing `localhost` in the examples above with the hostname or IP address of the remote host. This method cannot be used to test the Windows SMI-S provider remotely, however, since (of course) Windows doesn't follow the emerging standard.

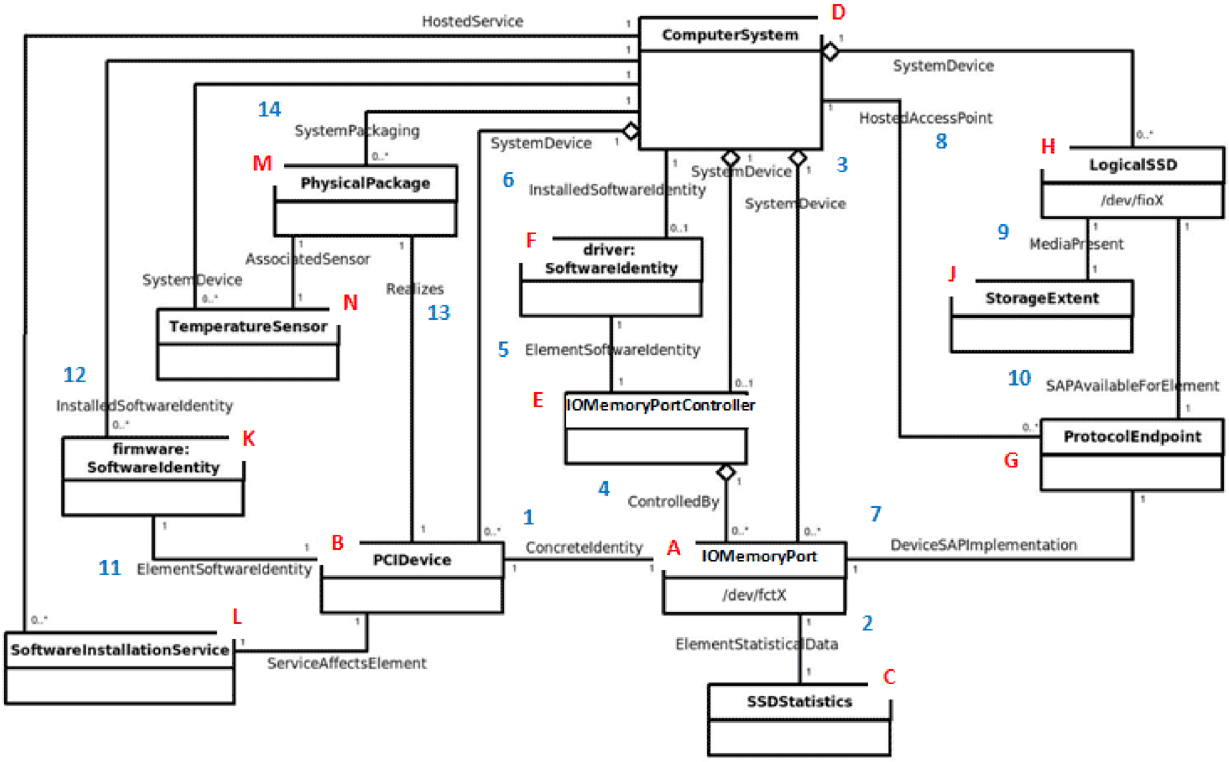
The SMI-S provider indications can be tested as well.

Description

SMI-S is a collection of specifications that traditionally focus on Storage Area Network (SAN) systems based on the SCSI command set, such as Fibre Channel, iSCSI, and SAS. However, the general pattern used to model these storage systems can be applied to solid-state, direct-attached storage systems such as those provided by IBM.

IBM High IOPS Adapters are modeled using the SMI-S patterns established in the Storage HBA, Direct Attached (DA) Ports, and Host Discovered Resources Profiles. The physical aspects of the IBM High IOPS Adapter and all firmware and ioMemory VSL software are modeled using published DMTF specifications, including the Physical Asset, Software Inventory, PCI Device Profiles, and Common Diagnostic Model Profile.

The following chart describes the IBM SMI-S CIM model, with IBM High IOPS Adapters and their associated firmware and software. For simplicity, the prefix FIO_ has been removed from the class names.



A: IOMemoryPort Class

The central instance of the model is of the `IOMemoryPort` class (A in the figure), a logical representation of the IBM High IOPS Adapter. It supports the extrinsic methods necessary to provision the drive. An instance of `PCIDevice` (B) and `IOMemoryPort` exist for each installed IBM High IOPS Adapter, and they are associated with instances of `ConcreteIdentity` (1). An instance of `SSDStatistics` (C), which contains important performance and capacity data for the device, is associated by an `ElementStatisticalData` association (2) to each `IOMemoryPort`. `IOMemoryPort` is scoped by an instance of the `ComputerSystem` class. The `SystemDevice` (3) aggregation aggregates `IOMemoryPort` within the containing `ComputerSystem`.

E: IOMemoryPortController Class

An instance of IOMemoryPortController (E) represents the ioMemory VSL used to control the installed IBM High IOPS Adapters. IOMemoryPortController specializes CIM_PortController, and it aggregates IoMemoryPort with the ControlledBy (4) aggregation. The software version and vendor information are represented by the SoftwareIdentity (F) instance that is associated to IOMemoryPortController (E) via ElementSoftwareIdentity (5). The SoftwareIdentity that represents the installed ioMemory VSL software is associated to the scoping ComputerSystem using the InstalledSoftwareIdentity association (6).

An instance of the ProtocolEndpoint class (G) represents both ends of the logical data path between the IOMemoryPort and the solid-state storage. This aspect of the model is derived from the pattern in the DA Ports Profile, where the port is both an initiator and target. ProtocolEndpoint is associated to the IOMemoryPort by DeviceSAPImplementation (7) and to the ComputerSystem by HostedAccessPoint (8).

H: LogicalSSD Class (Block Device)

The block device exposed to applications (file systems, database, and logical volume manager) is modeled using an instance of LogicalSSD (H), a subclass of CIM_DiskDrive. It is associated with a StorageExtent (J) using the MediaPresent association (9), but the StorageExtent will always be present. It is also associated to the ProtocolEndpoint (G) representing the IOMemoryPort using SAPAvailableForElement (10) and to the scoping ComputerSystem using SystemDevice (3).

IBM High IOPS Adapters, being PCIe devices, are also represented by an instance of the PCIDevice class (B). IOMemoryPort is an alternate representation of the PCIDevice and its associated control device. It is associated to it by the ConcreteIdentity association.

K: SoftwareIdentity

The ioMemory VSL software is also represented with SoftwareIdentity, which is associated to the PCIDevice by the ElementSoftwareIdentity association (11). The SoftwareIdentity (firmware) is associated to the scoping ComputerSystem by the InstalledSoftwareIdentity association (12). An instance of SoftwareInstallationService (L) is associated with each PCIDevice, which can be used to update device firmware.

M: Physical Aspects

The physical aspects of IBM High IOPS Adapters are represented by an instance of the PhysicalPackage class (M), which is associated to the PCIDevice by Realizes (13) and to the scoping ComputerSystem by SystemPackaging (14). The temperature sensors on IBM High IOPS Adapters are represented by an instance of TemperatureSensor (N) and is associated to the PhysicalPackage by AssociatedSensor.

Implementation

This section describes the arrangement of instances and associations for the IBM device CIM model. Not all class properties are described in detail. Consult the CIM schema for detailed description of all properties.

The device health is indicated by the value of the HealthLevel property. Values include: Healthy, Warning, Reduced

Write, and Read Only. These values are mapped to `standardHealthState` values – OK, Degraded/Warning, and Critical Failure – as appropriate.

Extrinsic methods for device provisioning include attach, detach, format, and update. The attach method creates a block device for the IBM High IOPS Adapter. Detach disables the block device. A format option enables users to specify the device size in either megabytes or a percentage. The update method allows users to upgrade the firmware on the device.

Device longevity is indicated by the value of the `HealthPercentage` property. `FlashbackAvailability` indicates whether or not this feature of the IBM High IOPS Adapter is online.

`IOMemoryPorts` are aggregated by `IOMemoryPortController` via the `ControlledBy` aggregation. Instances of `IOMemoryPort` are associated to their corresponding `PCIDevice` with the `ConcreteIdentity` association. The `IOMemoryPort` is a logical device of the scoping `ComputerSystem` and is indicated as such by the `SystemDevice` aggregation.

Products with two or more IBM High IOPS Adapters, such as the IBM High IOPS Duo Adapter do appear like two separate IBM High IOPS Adapters. For products with multiple devices, the `IOMemoryPort` class is extended to include information about the carrier card type, serial number, and external power connection for the product as a whole.

IOMemoryPort

One instance of `IOMemoryPort` exists for each IBM High IOPS Adapter installed in the `ComputerSystem`.

The `LocationIndicator` property reflects the state of the device indicator beacon (e.g., all LEDs on solid). Reading the value gives the current state of the indicator. Writing the value with "On" or "Off" turns the indicator on or off and can be used to determine the device's physical location.

SSDStatistics

One instance of `SSDStatistics` exists for each `IOMemoryPort` instance. Properties of this object provide performance and capacity information. Some of this information is only available when the drive is attached (i.e., the state of the associated `IOMemoryPort` is "Attached").

IOMemoryPortController

Only one instance of `IOMemoryPortContoller` exists, representing the ioMemory VSL software used to control `IOMemoryPorts`. The `IOMemoryPortController` specializes the `CIM_PortController`.

`IOMemoryPortController` is aggregated to the scoping `ComputerSystem` using the `SystemDevice` aggregation. `IOMemoryPortController` is associated with a `SoftwareInventory` instance representing the ioMemory VSL software properties via the `ElementSoftwareIdentity` association.

ProtocolEndpoint

One instance of ProtocolEndpoint exists for each instance of IOMemoryPort. It is associated to the IOMemoryPort using DeviceSAPImplementation and to LogicalSSD using SAPAvailableForElement. Because an IOMemoryPort represents both the initiator and target ports, only one ProtocolEndpoint per IOMemoryPort is needed to model the connection between IOMemoryPort and LogicalSSD.

LogicalSSD

One instance of LogicalSSD, a subclass of CIM_DiskDrive, exists for each block device (/dev/fioX) exposed by an IBM High IOPS Adapter. Correlatable IDs are used, based on operating system device names. This enables client applications to associate block devices discovered through this model with resources discovered from other SMI-S models instrumented on the host system.

ComputerSystem aggregates LogicalSSDs via SystemDevice. The LogicalSSD instances are associated to their ProtocolEndpoints via SAPAvailableForElement. If the IOMemoryPort associated to the endpoint is not attached, then the Availability property is set to "Off Line," and the DeviceID property value is "Unknown."

StorageExtent

One instance of StorageExtent is associated with each LogicalSSD and represents the logical storage of the associated device.

SoftwareIdentity

One instance of SoftwareIdentity exists to represent the ioMemory VSL software. The firmware is also modeled using SoftwareIdentity but requires an instance for each IBM High IOPS Adapter installed. The IsEntity property has a value of True, indicating that the SoftwareIdentity instance corresponds to a discrete copy of the ioMemory VSL software or firmware. The MajorVersion, MinorVersion, RevisionNumber, and BuildNumber properties convey the driver/firmware version information. The Manufacturer property can be used to identify Fusion-io.

Another option for the firmware is to omit the InstalledSoftwareIdentity association with ComputerSystem, because the firmware is not really installed on ComputerSystem. This option would depend on how users want to model the firmware.

SoftwareInstallationService

An instance of SoftwareInstallationService exists for each PCIDevice and can be used to update the associated device's firmware.

PCIDevice

An instance of PCIDevice is instantiated for each IBM High IOPS Adapter (PCIe card) in the computer. Properties are set as follows:

- BusNumber – bus number where the PCIe device exists

- DeviceNumber – device number assigned to the PCI device for this bus.
- FunctionNumber – set to the function number for the PCI device.
- SubsystemID, SubsystemVendorID, PCIDeviceID, VendorID, and RevisionID are optional but can be populated if values can be extracted from the configuration registers of the PCI device.

PCIDevice is associated with IOMemoryPort, its alternate logical representation, using ConcreteIdentity. The PCIDevice is also associated with PhysicalPackage, representing the physical aspects of the IBM High IOPS Adapter, via Realizes.

PhysicalPackage

One instance of PhysicalPackage exists for each discrete, physical IBM High IOPS Adapter installed in the computer system. The Manufacturer, Model, SKU, SerialNumber, Version, and PartNumber properties can be used to describe these aspects of the physical card. PhysicalPackage is associated with PCIDevice via Realizes and the scoping ComputerSystem via SystemPackaging.

TemperatureSensor

One instance of TemperatureSensor exists for each PhysicalPackage. Temperature information for the drive is stored in the properties of this object.

Diagnostic Test

One instance of DiagnosticTest will exist. The RunDiagnostic() method will trigger a snapshot of device status for the specified ManagedElement which must be an instance of IoMemoryPort. The diagnostic run is synchronous and runs instantaneously. The resulting ConcreteJob object will associate to the originating DiagnosticTest instance and the respective IoMemoryPort instance that was specified (see Figure 2). At this time, RunDiagnostic() can only be used with the default DiagnosticSettingData provided.

Each run will add a single entry of DiagnosticSettingDataRecord and associated DiagnosticCompletionRecord in the DiagnosticLog. The RecordData property of the DiagnosticCompletionRecord will record critical device status at the time of the run. The format of the RecordData string can be found in the RecordFormat property.

The format is a series of status strings, each of which can hold one of the following values delimited by an asterisk (*) character: "Unknown", "OK", "Warning", or "Error". Currently, seven status values are recorded: WearoutStatus, WritabilityStatus, FlashbackStatus, TemperatureStatus, MinimalModeStatus, PciStatus and InternalErrorStatus. All of these should report "OK" under normal operating conditions.

WearoutStatus will be set to "Warning" when less than 10% reserve space is left on the device. It will be set to "Error" when there is no more reserved space.

- WritabilityStatus will be set to "Error" whenever the device is write throttling or in read-only mode. This can happen due to a variety of conditions including device wearout and insufficient power.
- FlashbackStatus will report "Warning" if a catastrophic error causes Flashback protection to be degraded.

- TemperatureStatus will report "Warning" when the device temperature is nearing the maximum safe temperature and "Error" when the maximum safe temperature is reached or surpassed.
- MinimalModeStatus will report either "Warning" or "Error" whenever the device is in minimal mode.
- PciStatus will report "Warning" or "Error" if there are compatibility problems with the host PCIe bus.
- InternalErrorStatus will report "Error" if there are any internal problems with the ioMemory VSL.

The CompletionState property will summarize the results and may be set to Unknown, OK, Warning or Failed. If any status is in error the state will report as Failed. Otherwise, if there is any warning status the state will report Warning. The Message property will be set to indicate the appropriate action if there are any warnings or errors.

DiagnosticSetting Data

There will be an instance of DiagnosticSettingData associated with the DiagnosticTest instance (see Figure 2). It records the default settings for each call to RunDiagnostic.

DiagnosticServiceCapabilities

There is an instance of DiagnosticServiceCapabilities associated with the DiagnosticTest instance which records the capabilities of the DiagnosticTest service.

DiagnosticLog

An instance of DiagnosticLog is associated with the DiagnosticTest instance and will store the results of each run.

DiagnosticSettingRecord

A copy of the default DiagnosticSettingData will be stored in a DiagnosticSettingDataRecord each time a diagnostic is run and will be associated with an instance of DiagnosticCompletionRecord.

DiagnosticCompletionRecord

An instance of DiagnosticCompletionRecord will store the results of each RunDiagnostic execution. The details are explained in DiagnosticTest.

RegisteredDiskDriveLiteProfile

Only one instance of this class is needed. It resides in the /root/interop namespace and indicates the implementation of the Disk Drive Lite Profile. The following properties are set as follows:

- InstanceID – set to "SNIA:DiskDriveLiteProfile-1.4.0"
- RegisteredOrganization – set to "11" (SNIA)
- RegisteredName – set to "DirectAccess Ports Profile"

- `RegisteredVersion` – set to `"1.4.0"`

RegisteredDAPortsProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the DA Ports Profile. The properties are set as follows:

- `InstanceID` – set to `"SNIA:DAPortsProfile-1.4.0"`
- `RegisteredOrganization` – set to `"11"` (SNIA)
- `RegisteredName` – set to `"DirectAccess Ports Profile"`
- `RegisteredVersion` – set to `"1.4.0"`

RegisteredStorageHBAProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the Storage HBA Profile. The properties are set as follows:

- `InstanceID` – set to `"SNIA:StorageHBAProfile-1.4.0"`
- `RegisteredOrganization` – set to `"11"` (SNIA)
- `RegisteredName` – set to `"Storage HBA Profile"`
- `RegisteredVersion` – set to `"1.4.0"`

RegisteredHostDiscoveredResourcesProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the Host Discovered Resources Profile. The properties are set as follows:

- `InstanceID` – set to `"SNIA:HostDiscoveredResourcesProfile-1.2.0"`
- `RegisteredOrganization` – set to `"11"` (SNIA)
- `RegisteredName` – set to `"Host Discovered Resources Profile"`
- `RegisteredVersion` – set to `"1.2.0"`

RegisteredPCIDeviceProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the PCI Device Profile. The properties are set as follows:

- `InstanceID` – set to `"DMTF:DSP1075-PCIDevice-1.0.0a"`
- `RegisteredOrganization` – set to `"2"` (DMTF)
- `RegisteredName` – set to `"PCIDevice Profile"`

- `RegisteredVersion` – set to "`1.0.0a`"

RegisteredSoftwareInventoryProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the Software Inventory Profile. The properties are set as follows:

- `InstanceID` – set to "`DMTF:DSP1023-SoftwareInventory-1.0.1`"
- `RegisteredOrganization` – set to "`2`" (DMTF)
- `RegisteredName` – set to "`Software Inventory Profile`"
- `RegisteredVersion` – set to "`1.0.1`"

RegisteredSoftwareUpdateProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the Software Update Profile. The properties are set as follows:

- `InstanceID` – set to "`DMTF:DSP1023-SoftwareUpdate-1.0.0`"
- `RegisteredOrganization` – set to "`2`" (DMTF)
- `RegisteredName` – set to "`Software Update Profile`"
- `RegisteredVersion` – set to "`1.0.0`"

RegisteredPhysicalAssetProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the Physical Asset Profile. The properties are set as follows:

- `InstanceID` – set to "`DMTF:PhysicalAssetProfile-1.0.2`"
- `RegisteredOrganization` – set to "`2`" (DMTF)
- `RegisteredName` – set to "`PhysicalAsset Profile`"
- `RegisteredVersion` – set to "`1.0.2`"

RegisteredSensorsProfile

Only one instance of this class is needed. It resides in the `/root/interop` namespace and indicates the implementation of the Sensors Profile. The properties are set as follows:

- `InstanceID` – set to "`SNIA:SensorsProfile-1.0.0`"
- `RegisteredOrganization` – set to "`11`" (SNIA)
- `RegisteredName` – set to "`Sensors Profile`"

- RegisteredVersion – set to "1.0.0"

RegisteredCommonDiagnosticProfile

Only one instance of this class is needed. It will reside in the /root/interop namespace and indicate the implementation of the Common Diagnostic Model Profile. The InstanceID property will be set to a value of " DMTF:DiagnosticsProfile-2.0.0a". The RegisteredOrganization property will be set to a value of "2" (DMTF). The RegisteredName property will be set to a value of "Diagnostics Profile". The RegisteredVersion property will be set to a value of "2.0.0a".

Indications

An indication will be generated periodically when a serious condition exists for a particular IBM High IOPS Adapter. The WBEM provider currently supports six types of indications. They alert users of the SMI-S provider to conditions such as imminent wearout, degradation of writability, degradation of the flashback feature, higher temperature, and internal error states.

The indications will be instances of the FIO_AlertIndication class which simply specializes the CIM_AlertIndication class.

The values for the properties of the FIO_AlertIndication instances are under development and may change as testing proceeds and feedback is received.

FIO_AlertIndication

Property	Value
`IndicationIdentifier`	See below for each type
`IndicationTime`	Timestamp when sent
`AlertingManagedElement`	IoMemoryPort.DeviceID=<device ID>
`AlertingElementFormat`	CIMObjectPath (2)
`AlertType`	Device Alert (5)
`PerceivedSeverity`	See below for each type
`ProbableCause`	See below for each type
`SystemCreationClassName`	"FIO_AlertIndication"
`SystemName`	<hostname>
`ProviderName`	"fiosmis"
`CorrelatedIndications`	Not used
`Description`	Class description
`OtherAlertType`	Not used
`OtherSeverity`	Not used
`ProbableCauseDescription`	Not used
`EventID`	Same as IndicationIdentifier
`OwningEntity`	<vendor>
`MessageID`	TBD
`Message`	TBD
`MessageArguments`	TBD

Reduced Writability Indication

The ioMemory VSL can dramatically reduce write throughput to manage device conditions such as excessive wear, high temperature, and insufficient power. The reduced writability indication is generated while the drive is in this mode. If the triggering condition is excessive wear, the `IoMemoryPort` health percentage will report 0% health.

Property	Value
`IndicationIdentifier`	`<mfr>":"<hostname>":write"`
`PerceivedSeverity`	Degraded/Warning (3)

ProbableCause	Threshold Crossed (52) Temperature Unacceptable (51) Power Problem (36)

Read-only Indication

When the drive has reached the end-of-life, it can no longer be written to and can only be read from. The read-only indication will be sent when this occurs. The `IoMemoryPort` health percentage will continue to report 0% health when this happens.

Property	Value
IndicationIdentifier	`<mfr>":"<hostname>":read_only"`
PerceivedSeverity	Degraded/Warning (3)
ProbableCause	Threshold Crossed (52)

Wearout Indication

As the drive wears out, this indication is generated as a warning when the drive health percentage drops below 10%, before write throughput is reduced.

Property	Value
IndicationIdentifier	`<mfr>":"<hostname>":wearout"`
PerceivedSeverity	Degraded/Warning (3)
ProbableCause	Threshold Crossed (52)

Flashback Indication

If a catastrophic part failure degrades the effectiveness of the flashback feature, this indication will be sent.

Property	Value
IndicationIdentifier	`<mfr>":"<hostname>":flashback"`
PerceivedSeverity	Degraded/Warning (3)
ProbableCause	Loss of Redundancy (88)

High Temperature Indication

This indication will be sent when the temperature of the card becomes excessive.

Property	Value
IndicationIdentifier	`<mfr>":"<hostname>":temperature"`
PerceivedSeverity	Critical (6)

`ProbableCause`	Temperature Unacceptable (51)

Error Indication

If the ioMemory VSL is in an error state the error indication will be sent.

Property	Value
`IndicationIdentifier`	`<mfr>":"<hostname>":error"`
`PerceivedSeverity`	Major (6)
`ProbableCause`	Other (1)

Appendix F- Documentation Permissions

The `AVR bootloader` and the `tree.h` file, which ship in binary form with the driver, contain content that have the following documentation copyright requirements:

AVR Bootloader

tree.h

following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

IBM Support

IBM High IOPS Adapter software and documentation are available on the web at the following address:

http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5083174.

IBM part number 81Y1047