NAME

vdostats - get configuration and statistics from a running VDO volume

SYNOPSIS

vdostats [--**verbose**] [--**human-readable**] [--**si**] [--**all**] [--**version**] [device...]

DESCRIPTION

The **vdostats** utility display statistics for each configured (or specified) VDO device.

The default output format is a table with the following columns, similar to that of the Linux **df** utility:

Device The path to the VDO volume

1K-blocks
The total number of 1K blocks allocated for a VDO volume (= physical volume size * block size / 1024)

Used The total number of 1K blocks used on a VDO volume (= physical blocks used * block size / 1024)

Available
The total number of 1K blocks available on a VDO volume (= physical blocks free * block size / 1024)

Use% The percentage of physical blocks used on a VDO volume (= used blocks / allocated blocks * 100)

Space Saving%
The percentage of physical blocks saved on a VDO volume (= [logical blocks used - physical blocks used] / logical blocks used)

VERBOSE OUTPUT

The --**verbose** option displays VDO device statistics in YAML format for the specified VDO devices. The following fields will continue to be reported in future releases. Management tools should not rely upon the order in which any of the statistics are reported.

version
The version of these statistics.

release version
The release version of the VDO.

data blocks used
The number of physical blocks currently in use by a VDO volume to store data.

overhead blocks used
The number of physical blocks currently in use by a VDO volume to store VDO metadata.

logical blocks used
The number of logical blocks currently mapped.

physical blocks
The total number of physical blocks allocated for a VDO volume.

logical blocks
The maximum number of logical blocks that can be mapped by a VDO volume.

1K-blocks
The total number of 1K blocks allocated for a VDO volume (= physical volume size * block size / 1024)

1K-blocks used
The total number of 1K blocks used on a VDO volume (= physical blocks used * block size / 1024)

1K-blocks available
The total number of 1K blocks available on a VDO volume (= physical blocks free * block size / 1024)

used percent
The percentage of physical blocks used on a VDO volume (= used blocks / allocated blocks * 100)

saving percent
The percentage of physical blocks saved on a VDO volume (= [logical blocks used - physical blocks used] / logical blocks used)

block map cache size
The size of the block map cache, in bytes.

write policy
The write policy (sync or async). This is configured via **vdo modify --writePolicy=**policy.

block size
The block size of a VDO volume, in bytes.

completed recovery count
The number of times a VDO volume has recovered from an unclean shutdown.

read-only recovery count
The number of times a VDO volume has been recovered from read-only mode (via **vdo start --forceRebuild**).

operating mode
Indicates whether a VDO volume is operating normally, is in recovery mode, or is in read-only mode.

recovery progress (%)
Indicates online recovery progress, or **N/A** if the volume is not in recovery mode.

compressed fragments written
The number of compressed fragments that have been written since the VDO volume was last restarted.

compressed blocks written
The number of physical blocks of compressed data that have been written since the VDO volume was last restarted.

The remaining fields are primarily intended for software support and are subject to change in future releases; management tools should not rely upon them.

compressed fragments in packer
The number of compressed fragments being processed that have not yet been written.

slab count
The total number of slabs.

slabs opened
The total number of slabs from which blocks have ever been allocated.

slabs reopened
The number of times slabs have been re-opened since the VDO was started.

journal disk full count
The number of times a request could not make a recovery journal entry because the recovery journal was full.

journal commits requested count
The number of times the recovery journal requested slab journal commits.

journal entries batching
The number of journal entry writes started minus the number of journal entries written.

journal entries started
The number of journal entries which have been made in memory.

journal entries writing
The number of journal entries in submitted writes minus the number of journal entries committed to storage.

journal entries written
The total number of journal entries for which a write has been issued.

journal entries committed
The number of journal entries written to storage.

journal blocks batching
The number of journal block writes started minus the number of journal blocks written.

journal blocks started
The number of journal blocks which have been touched in memory.

journal blocks writing
The number of journal blocks written (with metadatata in active memory) minus the number of journal blocks committed.

journal blocks written
The total number of journal blocks for which a write has been issued.

journal blocks committed
The number of journal blocks written to storage.

slab journal disk full count
The number of times an on-disk slab journal was full.

slab journal flush count
The number of times an entry was added to a slab journal that was over the flush threshold.

slab journal blocked count
The number of times an entry was added to a slab journal that was over the blocking threshold.

slab journal blocks written
The number of slab journal block writes issued.

slab journal tail busy count
The number of times write requests blocked waiting for a slab journal write.

slab summary blocks written
The number of slab summary block writes issued.

reference blocks written
The number of reference block writes issued.

block map dirty pages
The number of dirty pages in the block map cache.

block map clean pages
The number of clean pages in the block map cache.

block map free pages
The number of free pages in the block map cache.

block map failed pages
The number of block map cache pages that have write errors.

block map incoming pages
The number of block map cache pages that are being read into the cache.

block map outgoing pages
The number of block map cache pages that are being written.

block map cache pressure
The number of times a free page was not available when needed.

block map read count
The total number of block map page reads.

block map write count
The total number of block map page writes.

block map failed reads
The total number of block map read errors.

block map failed writes
The total number of block map write errors.

block map reclaimed
The total number of block map pages that were reclaimed.

block map read outgoing
The total number of block map reads for pages that were being written.

block map found in cache
The total number of block map cache hits.

block map discard required
The total number of block map requests that required a page to be discarded.

block map wait for page
The total number of requests that had to wait for a page.

block map fetch required

The total number of requests that required a page fetch.

block map pages loaded
The total number of page fetches.

block map pages saved
The total number of page saves.

block map flush count
The total number of flushes issued by the block map.

invalid advice PBN count
The number of times the index returned invalid advice

invalid rollover PBN count
The number of times the index returned invalid advice for a retried query after a block has reached the maximum reference count.

dedupe deadlock avoidance count
The number of times deduplication was aborted in order to avoid a potential deadlock with other requests.

no space error count
The number of write requests which failed due to the VDO volume being out of space.

read only error count
The number of write requests which failed due to the VDO volume being in read-only mode.

instance
The VDO instance.

512 byte emulation
Indicates whether 512 byte emulation is on or off for the volume.

current VDO IO requests in progress
The number of I/O requests the VDO is current processing.

maximum VDO IO requests in progress
The maximum number of simultaneous I/O requests the VDO has processed.

current dedupe queries
The number of deduplication queries currently in flight.

maximum dedupe queries
The maximum number of in-flight deduplication queries.

dedupe advice valid
The number of times deduplication advice was correct.

dedupe advice stale
The number of times deduplication advice was incorrect.

dedupe advice timeouts
The number of times deduplication queries timed out.

flush out
The number of flush requests submitted by VDO to the underlying storage.

write amplification ratio
The average number of block writes to the underlying storage per block written to the VDO device.

bios in...
bios in partial...
bios out...
bios meta...
bios journal...
bios page cache...
bios out completed...
bios meta completed...
bios journal completed...
bios page cache completed...
bios acknowledged...
bios acknowledged partial...
bios in progress...
These statistics count the number of bios in each category with a given flag. The categories are:

bios in
The number of block I/O requests received by VDO.

bios in partial
The number of partial block I/O requests received by VDO. Applies only to 512-byte emulation mode.

bios out
The number of non-metadata block I/O requests submitted by VDO to the storage device.

bios meta
The number of metadata block I/O requests submitted by VDO to the storage device.

bios journal
The number of recovery journal block I/O requests submitted by VDO to the storage device.

bios page cache
The number of block map I/O requests submitted by VDO to the storage device.

bios out completed
The number of non-metadata block I/O requests completed by the storage device.

bios meta completed
The number of metadata block I/O requests completed by the storage device.

bios journal completed
The number of recovery journal block I/O requests completed by the storage device.

bios page cache completed
The number of block map I/O requests completed by the storage device.

bios acknowledged
The number of block I/O requests acknowledged by VDO.

bios acknowledged partial
The number of partial block I/O requests acknowledged by VDO. Applies only to 512-byte emulation mode.

bios in progress
The number of bios submitted to the VDO which have not yet been acknowledged.

There are five types of flags:

read The number of non-write bios (bios without the REQ_WRITE flag set)

write The number of write bios (bios with the REQ_WRITE flag set)

discard
The number of bios with a REQ_DISCARD flag set

flush The number of flush bios (bios with the REQ_FLUSH flag set)

fua The number of "force unit access" bios (bios with the REQ_FUA flag set)

Note that all bios will be counted as either read or write bios, depending on the REQ_WRITE flag setting, regardless of whether any of the other flags are set.

KVDO module bios used
The current number of kernel "struct bio" structures allocated by the kernel VDO module.

KVDO module peak bio count
The peak number of kernel "struct bio" structures allocated by the kernel VDO module, since the module was loaded.

KVDO module bytes used
The current count of bytes allocated by the kernel VDO module.

KVDO module peak bytes used
The peak count of bytes allocated by the kernel VDO module, since the module was loaded.

read cache accesses
The number of times VDO searched the read cache.

read cache hits
The number of times VDO found an entry in the read cache for the desired data block, whether or not the block's contents had yet been read into memory.

read cache data hits
The number of times VDO found an entry in the read cache for the desired data block, where the block's contents had been read into memory.

OPTIONS

--**verbose**
Displays the utilization and block I/O (bios) statistics for the selected VDO devices.

--**human-readable**
Displays block values in readable form (Base 2: 1 KB = 2^10 bytes = 1024 bytes).

--**si** Modifies the output of the --**human-readable** option to use SI units (Base 10: 1 KB = 10^3 bytes = 1000 bytes). If the --**human-readable** option is not supplied, this option has no effect.

--**all** This option is only for backwards compatibility. It is now equivalent to --**verbose**.

--**version**
Displays the **vdostats** version.

device...
Specifies one or more specific volumes to report on. If this argument is omitted, **vdostats** will report on all devices.

EXAMPLES

The following example shows sample output if no options are provided:

```
Device              1K-blocks  Used       Available   Use% Space Saving%
/dev/mapper/my_vdo 1932562432 427698104  1504864328  22%  21%
```

With the --**human-readable** option, block counts are converted to conventional units (1 KB = 1024 bytes):

```
Device              Size   Used    Available   Use%   Space Saving%
/dev/mapper/my_vdo 1.8T   407.9G  1.4T        22%    21%
```

With the --**si** option as well, the block counts are reported using SI units (1 KB = 1000 bytes):

```
Device              Size   Used    Available   Use%    Space Saving%
/dev/mapper/my_vdo 2.0T   438G    1.5T        22%     21%
```

NOTES

The output may be incomplete when the command is run by an unprivileged user.

SEE ALSO

vdo(8).