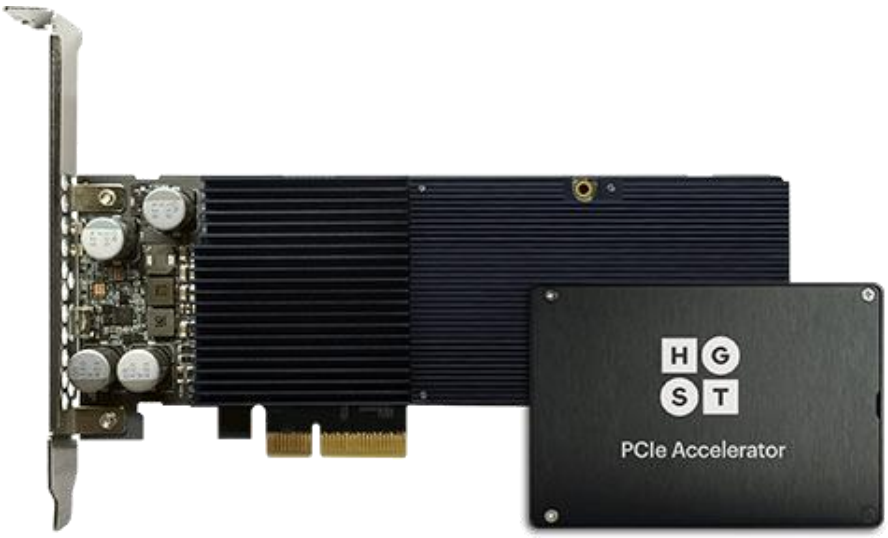
HGST Device Manager
Version 3.2
User Guide



Document Number: 61600-00180-309

Document Version: 3.09

Software Version: HDM 3.2

Revision Date: May 31, 2016

3RD Edition (Document Version 3.09)

May 31, 2016

The following paragraph does not apply to the United Kingdom or any country where such provisions are inconsistent with local law: HGST, INC., PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer or express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of this publication. HGST may make improvements or changes in any products or programs described in this publication at any time.

It is possible that this publication may contain reference to, or information about, HGST products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that HGST intends to announce such HGST products, programming, or services in your country.

Technical information about this product is available by contacting your local HGST representative or on the Internet at *http://www.hgst.com*.

HGST may have patents or pending patent applications covering the subject matter in this document. The furnishing of this document does not give you any license to these patents.

End-User License Agreement

IMPORTANT: PLEASE READ THIS END USER LICENSE AGREEMENT CAREFULLY. DOWNLOADING, INSTALLING OR USING THE ACCOMPANYING SOFTWARE CONSTITUTES ACCEPTANCE OF THIS END USER LICENSE AGREEMENT.

HGST, INC. AND ITS AFFILIATES (COLLECTIVELY, "HGST") IS WILLING TO LICENSE THE SOFTWARE ONLY IF YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS END USER LICENSE AGREEMENT (THE "EULA").

1. BY DOWNLOADING, INSTALLING OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED IN THIS END USER LICENSE AGREEMENT, YOU INDIVIDUALLY AND ON BEHALF OF THE ENTITY THAT YOU REPRESENT (COLLECTIVELY, "LICENSEE") CONSENT TO BE BOUND BY THIS EULA AND THE AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED IN THIS AGREEMENT, THEN (A) DO NOT DOWNLOAD, INSTALL OR USE THE SOFTWARE RETURN THE SOFTWARE TO HGST.

   1.1. No Sale. This EULA governs Licensee's use of and access to the Software (as defined below) for any Software first placed in use on or after the release date of this EULA (the "Release Date"). Licensee agrees and acknowledges that all Software is licensed and not sold, and that all use of "purchase" or "sell" in connection with any Software shall be deemed to mean "license." Any term in Licensee's purchase order or other document purporting to require the sale of Software instead of the license thereof is hereby rejected by HGST and is not part of the Agreement.

   1.2. No Implied Licenses; Licensed Material. Nothing in this EULA shall be deemed to provide to Customer any right, title or interest in (i) any product sold by HGST or an authorized reseller of HGST (a "Product"); (ii) any instance of a program, module, feature, function, service, application, operation or capability of the software HGST intentionally delivers under this EULA; or (iii) any documentation related to any Software (the "Documentation"), other than the express license rights set forth in Section 3 hereof.

2. Definitions. In this EULA, the following capitalized terms shall have the meaning set forth below:

   2.1. "Agreement" shall mean this EULA.

   2.2. "Affiliate" of a Party means such Party's parent corporation, an entity under the control of such Party's parent corporation at any tier or an entity controlled by such Party at any tier. For these purposes, "control" shall mean the power to direct or cause the direction of the management and policies of the entity, whether through the ownership of more than 50% of the outstanding voting interests in such entity or otherwise.

   2.3. "Party" means either Licensee or HGST and "Parties" means both Licensee and HGST.

   2.4. "Software" shall mean the software utility tool delivered under this EULA.

   2.5. "Third Party" means an individual or entity other than Licensee, HGST and HGST's Affiliates.

   2.6. "User" means an individual able to use or gain access to any Software functionality.

3. License Grant.

   3.1. Grant. Subject to the terms and conditions of this Agreement, HGST hereby grants to Licensee, and Licensee accepts, upon delivery of the Software, a nonexclusive, nontransferable license to install and use (subject to the limitations in Sections 3.2 and 3.3) each copy of the Software, in executable form only, supplied by HGST, and to use the accompanying Documentation, only for Licensee's internal use in connection with the testing and support of HGST hardware products. Licensee may make a reasonable number of copies of the Software for inactive backup and archival purposes.

   3.2. No Right to Sublicense or Assign. Except to the extent otherwise required by applicable law or expressly provided for assignment generally in the Agreement, including without limitation in this EULA, no license provided in this Section 3 is sublicensable, transferable or assignable by

Licensee, including by operation of law, change of control, merger, purchase or otherwise, without the prior written consent of HGST in each instance. Other than as expressly permitted by the foregoing, any attempted sublicense, transfer or assignment by Licensee shall be null and void.

3.3. License Prohibitions. Notwithstanding anything to the contrary in this EULA, Licensee shall not, alone, through a User, an Affiliate or a Third Party (or allow a User, an Affiliate or a Third Party to):

(a) sell, transfer, license, lease pledge, loan or otherwise make available to Third Parties any Software;

(b) modify any Software;

(c) reverse compile, reverse assemble, reverse engineer or otherwise translate all or any portion of any Software;

(d) pledge, rent, lease, share, distribute, sell or create derivative works of any Software;

(e) use any Software on a time sharing, service bureau, application service provider (ASP), rental or other similar basis;

(f) make copies of any Software, except as provided for in the license grant above;

(g) transfer any Software from a Product supplied by HGST to any hardware supplied by a Third Party;

(h) remove, alter or deface (or attempt any of the foregoing) proprietary notices, labels or marks in any Software;

(i) distribute any copy of any Software to any Third Party, including without limitation selling any Product with Software in a secondhand market;

(j) use any Embedded Software other than with Products supplied by HGST or an authorized reseller of HGST or for more than the number of Products purchased from HGST or an authorized reseller of HGST;

(k) disclose the results of testing or benchmarking of any Software to any Third Party without the prior written consent of HGST;

(l) use any Update or Upgrade beyond those to which Licensee is entitled or with any Software to which Licensee does not have a valid, current license;

(m) circumvent or disable HGST's copyright protection mechanisms or license management mechanisms;

(n) use the Software in violation of any applicable law or to support any illegal activity; or

(o) attempt any of the foregoing.

HGST expressly reserves the right to seek all available legal and equitable remedies to prevent any of the foregoing and to recover any lost profits, damages or costs resulting from any of the foregoing.

3.4. Legal Exception. Licensee agrees that, to the extent that any applicable laws (including without limitation national laws implementing EC Directive 91/250 on the Legal Protection of Computer Programs) give Licensee the right to reverse engineer any Software to make it interoperable without HGST's consent, before Licensee exercises any such rights, Licensee shall notify HGST of such desire and, no later than sixty (60) days following receipt of such request, HGST may decide either: (a) to perform the work to achieve such interoperability and charge its then-standard rates for such work to Licensee; or (b) to permit Licensee to reverse engineer parts of the Software only to the extent necessary to achieve such interoperability. Only if and after HGST, at its sole discretion, partly or completely denies Licensee's request, shall Licensee exercise its statutory rights.

3.5. Third Party Software. Portions of the Software may be owned by one or more Third Parties and licensed to HGST. HGST and Licensee intend and agree that Software owned by Third Parties and supplied hereunder is being sublicensed to Licensee, that such Third Parties retain ownership

of and title to such Software, network and that such Third Parties may directly enforce Licensee's obligations hereunder to protect their respective interests in such Software. The warranty and indemnification provisions set forth herein shall not apply to Software owned by Third Parties and supplied hereunder. The licenses/terms applicable to such Third Party licenses are set forth in Exhibit A.

3.6. Software Support. Unless otherwise agreed to as part of the Agreement, HGST does not offer or provide any technical support or other services under this EULA.

4. Confidentiality. Subject to any other confidentiality restrictions set forth in the Agreement, Licensee agrees the Software and Documentation are the confidential information of HGST. Licensee shall maintain the Software and Documentation in confidence, using the same degree of care it uses for its own confidential information of a similar nature, which shall be no less than reasonable and customary efforts in the software industry.

5. Software Warranty. TO THE MAXIMUM EXTENT PERMITTED BY LAW, EXCEPT AS PROVIDED FOR IN THE AGREEMENT, THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS". HGST, INCLUDING ITS AFFILIATES AND SUPPLIERS, MAKES NO OTHER WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT WITH RESPECT TO THE SOFTWARE AND THE DOCUMENTATION. IN ADDITION, HGST DOES NOT WARRANT THAT THE SOFTWARE'S OPERATION WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT ALL DEFECTS WILL BE CORRECTED.

6. Exclusive Remedies and Limitation of Liability.

6.1. Definitions. For purposes of the exclusive remedies and limitations of liability set forth in this Section 7, HGST shall be deemed to include its directors, officers, employees, agents, representatives, shareholders, subcontractors, licensors and suppliers; and "damages" shall be deemed to refer collectively to all injury, damage, loss or expense incurred.

6.2. Maximum Liability. Except as may otherwise be provided in the Agreement, HGST's maximum aggregate liability arising from or relating to this EULA or the use of the Software and Products shall not to exceed the total amount paid by Licensee to HGST during the twelve (12) month period that immediately preceded the event that gave rise to the applicable claim.

6.3. Limitation of Liability. NOTWITHSTANDING ANY OTHER PROVISION OF THIS EULA BUT SUBJECT IN ALL RESPECTS TO THE AGREEMENT, TO THE MAXIMUM EXTENT PERMITTED BY LAW, HGST, INCLUDING ITS AFFILIATES AND SUPPLIERS, SHALL NOT BE LIABLE FOR INCIDENTAL, SPECIAL, EXEMPLARY, CONSEQUENTIAL OR OTHER INDIRECT DAMAGES, INCLUDING BUT NOT LIMITED TO LOSS OF DATA, LOSS OR INTERRUPTION OF USE, LOST PROFITS, SAVINGS OR REVENUES OF ANY KIND, WHETHER OR NOT HGST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS PROVISION SHALL APPLY EVEN IN THE EVENT OF THE FAILURE OF AN EXCLUSIVE REMEDY.

7. Term And Termination.

7.1. Term. The term of this EULA is while any Software is under Licensee's control or possession, unless terminated earlier (the "License Term").

7.2. Termination. HGST shall have the right to terminate this EULA (a) upon thirty (30) days' prior written notice to Licensee if Licensee has not cured any material breach of this EULA by the end of such thirty (30) day notice period; (b) if Licensee has not paid an invoice to HGST regarding such Software sixty (60) days after such invoice is due, at any time thereafter upon written notice to Licensee; (c) if the Software is found to infringe or misappropriate any Third Party's intellectual property or proprietary rights, or becomes subject to an injunction prohibiting its use, or HGST reasonably believes either of the foregoing conditions is likely to occur; or (d) as otherwise provided for in the Agreement. Upon termination of this EULA for any reason, (i) Licensee shall be responsible for payment for all purchase orders delivered to Licensee by HGST before the effective date of termination; and (ii) Licensee shall destroy all copies of the Software under Licensee's control or possession. Further, the licenses granted under this EULA for Embedded Software

shall automatically terminate when Licensee's lawful possession of the applicable Product provided hereunder ceases, unless earlier terminated as provided herein.

7.3. Survival. The provisions of Sections 1, 2, 3.5, 3.6, 3.8, 3.9, 4, 5.3, 6, 7, 8.3, and 9 of this EULA shall survive any termination in accordance with their terms, in addition to any accrued obligations.

8. Miscellaneous.

8.1. Choice of Law; Jurisdiction. The Agreement shall be governed by the laws of the State of California, excluding (a) its conflicts of law rules that would result in the application of the law of any other jurisdiction; and (b) the United Nations Convention for the International Sale of Goods. The federal and state courts located in San Jose, California shall have exclusive jurisdiction with respect to any dispute arising under this Agreement.

8.2. Assignment. Neither Party may assign its rights and obligations hereunder without the prior written consent of the other Party. Notwithstanding the foregoing, either Party may assign this EULA to any Affiliate of the other or to an acquirer (by purchase, merger or otherwise) of all or substantially all of such Party's business or assets relating to this EULA, provided that (i) the assignee agrees in writing to be bound by the terms and conditions of this EULA, (ii) neither the assignor nor assignee are in default hereunder. Any attempted assignment other than as permitted shall be null and void.

8.3. Non-Waiver. No course of dealing or failure of either party to strictly enforce any term, right or condition of this Agreement shall be construed as a waiver of such term, right or condition. In the event of any breach or threatened breach of Section 3, Licensee agrees that HGST will suffer irreparable damage for which HGST will have no adequate remedy at law. Accordingly, in addition to any other remedy, HGST shall be entitled to injunctive and other equitable remedies to prevent or restrain such breach or threatened breach, without the necessity of proving actual damages or posting any bond.

8.4. Export Regulations. Licensee understands that HGST is subject to regulation by the U.S. government and its agencies, which prohibit export or diversion of certain technical products and information to certain countries and individuals. Licensee warrants to HGST that Licensee will comply in all respects with all export and re-export restrictions applicable to the technology and documentation provided hereunder.

8.5. Restricted Party. Licensee represents to HGST that Licensee is not a Restricted Party, which shall be deemed to include any person or entity: (1) located in or a national of Cuba, Iran, Libya, North Korea, Sudan, Syria, or any other countries that may, from time to time, become subject to U.S. export controls for anti-terrorism reasons or with which U.S. persons are generally prohibited from engaging in financial transactions; (2) on the U.S. Department of Commerce Denied Person's List, Entity List, or Unverified List; U.S. Department of the Treasury list of Specially Designated Nationals and Blocked Persons; or U.S. Department of State List of Debarred Parties; (3) engaged in activities involving nuclear materials or weapons, missile or rocket technologies, or proliferation of chemical or biological weapons; or (4) affiliated with or a part of any non-U.S. military organization. Licensee shall not sell, resell, loan, disclose, or otherwise transfer any Software, Product or Documentation to any Restricted Person without prior, express written authorization from HGST and the appropriate U.S. Government agency. If Licensee becomes a Restricted Person during the Term, this EULA shall automatically terminate and Licensee shall immediately cease all use of the Software.

8.6. United States Government Restricted Rights. The Software and Documentation is provided with restricted rights and qualifies as "commercial items" consisting of "commercial computer software" and "computer software documentation" as such terms are defined and used at FAR (48 C.F.R.) 2.101 and FAR 12.212. Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(f)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c) (1) and (2) of the Commercial Computer Software-Restricted Rights at 48 C.F.R. S:52.227-19, as applicable. For clarity, this Section 9.6 shall also be construed to apply to the state or local government counterparts, or the foreign counterpart specified as the Territory, as applicable.

8.7. Order of Precedence; Entire Agreement. To the extent of any conflict among this EULA, any additional terms in an agreement signed by Licensee and HGST, any HGST "click-accept" agreement, any terms on a purchase order and HGST's terms and conditions of sale, the order of precedence shall be (a) an agreement signed by Licensee and HGST; (b) this EULA; (c) a "click-accept" agreement, including any agreement regarding the Features to be included as part of the Software; (d) HGST's terms and conditions of sale; and (e) HGST's purchase order acceptance (collectively, (a)-(e), after applying the order of precedence, the "Agreement"). The terms and conditions contained in the Agreement supersede all prior oral or written understandings between the Parties and shall constitute the entire agreement between the Parties with respect to the subject matter of the Agreement. This Agreement shall not be modified or amended except by a writing signed by Licensee and HGST.

8.8. Construction; Severability. The headings used in this Agreement are for reference purposes only and will not be deemed to limit, expand or in any way affect the interpretation of any term or provision hereof. If any provision or part hereof shall be held to be invalid or unenforceable for any reason, then the meaning of such provision or part hereof shall be construed so as to render it enforceable to the extent feasible. If no feasible interpretation would save such provision or part hereof, it shall be severed herefrom, but without in any way affecting the remainder of such provision or any other provision contained herein, all of which shall continue in full force and effect unless such severance effects such a material change as to render the Agreement unreasonable.

Exhibit A

Third Party Licenses

The Software contains open source components as set forth in this Exhibit A. Please refer to the table below for the open source components and the associated license(s) for the terms and conditions for their use. The source code for the components listed in **Table 1** below may be obtained from HGST. HGST may provide the open source code to you on a physical storage medium for a charge covering the cost of performing such distribution, which may include, without limitation the cost of media, shipping and handling.

Table 1

License	Component	License Available at
MIT	libuv	*https://github.com/joyent/libuv/blob/master/LICENSE*
Public Domain	MurmurHash3 Function	*https://code.google.com/p/smhasher*
Apache Version 2.0	Android Open Source Project (hashtable)	*https://android.googlesource.com/platform/system/core/+/master/libcutils/hashmap.c*
MIT	uint128 class	*http://codef00.com/code/uint128.h*
MIT	Jansson JSON String Dump Utility	*http://www.digip.org/jansson*

Notwithstanding anything to the contrary in this EULA, this EULA does not prohibit Licensee from modifying, or reverse engineering to debug any such modifications to, any component of the Software that is subject to an LGPL or GPL license. Any such component is provided under the terms of the applicable LGPL or GPL license, as specified in Table 1.

Please also be informed that the Software may contain other Third Party Software listed in Table 2 below.

Table 2

License	Component	Copyright Notices	License Available at
N/A	-		

Conventions

The following icon and text conventions are used throughout this document to identify additional information of which the reader should be aware.

Conventions		Description
CAUTION		This icon denotes the use of extreme caution and the user must exercise good judgment according to previous experience before advancing to the next procedure. The icon also indicates the existence of a hazard that could result in equipment or property damage, or equipment failure if the instructions are not observed.
NOTE		This icon denotes additional or related information that the user may find useful. It also identifies any information that relates to the safe operation of the equipment, software, or related items.
Bold.	**Text**	Used to indicate **important technical notes**.
Bold Italic	***Text***	Used to indicate ***critical instructions***.
Light Blue Italic	*Text*	Used to indicate a hyperlink or "jump" to a related topic or subtopic. In addition, the text may be **bold**.

Revision History

Revision	Date	Page(s)	Description
-300	12/20/2014	All	Initial release.
-301	03/02/2015	All	Preliminary release.
-302	03/20/2015	19	Topic; Shared Library (Linux Installation) struck from user guide. Obsolete specification.
		20	Topic; Shared Library (Windows Installation) struck from user guide. Obsolete specification. Also added note explaining the necessity of using the HGST In-house Driver.
		21	Topic; Device References. Note 3 struck due to inaccuracy.
		23	Topic; Output Types. Minor edit of Sentence #2 to include `mini` (MINI) output type.
		29	Topic; `alias` command updated.
		15	Topic; Beta Limitations. Statement updated to reflect latest beta release.
		31	Topic; `configure-smart` options `temperature` and `availablespare` documented.
		35	Topic; `generate-report` Examples updated to include both redirection and console output.
		37	Topic; `get-feature` NVMe Feature `--feature` options documented.
		40	Topic; `get-log-page` NVMe `--page` options documented.
		45	Topic; `get-statistics` Example syntax corrected.
		46, 47, 48	Topic; `get-statistics` category examples updated.
		53	Topic; `resize` command. Added table listing valid resize ranges for `resize` command.
-303	04/15/2015	All	Struck references to “Beta” in preparation for GA Release.
		All	Global Search/Replace; The default is “mini”.
		viii	Exhibit A; struck Public Domain component “murmur3 hash function” from Table 1. Obsolete specification.
		15	Topic; “Beta Limitations” changed to “GA Limitations” and note re-worded to reflect GA software release.
		19	Topic; Linux Installation. Installation Notes updated.
		20	Topic; Windows Installation. Installation Notes updated.
		22	Figure 1 updated to show `mini` output format.
		28-52	The example output, where applicable, is now the `mini`, rather than the `text`, output format.
		26	Table 5, Command Exit Status Codes matrix updated.
-303	(continued)	47	Topic; `monitor` Command. Cautionary note added in regards to the `--interval` and `--count` values.
		48	Topic; `reset-to-defaults` Command. Cautionary note added in regards to power cycle of system.
		49	Topic; `resize` Command. Cautionary notes added in regards to `resize` command functionality and support.

Revision	Date	Page(s)	Description
-304	07/17/2015	Title	Title page information updated.
		viii	Exhibit A; updated to reflect legal requirements.
		22	Topic; Linux Directory Trees vs. Windows Device Instance IDs. Informational only.
			Topic; Short Form Syntax. Table of "Long Form" and "Short Form" equivalent options implemented in HDM 3.1.
			Topic; Configuration Files. The short form -c option documented for the `--config PATH` option.
		24	Topic; Output Types. The short form `-o` option documented for the `--output-format FORMAT` option.
		29	Topic; Command Confirmation developed for destructive commands.
		34	Topic; configure-trace. The short form `-f` option documented for the `--file FILE` parameter.
		54-56	Topic: `manage-namespaces` developed for HDM 3.1.
		57	Topic; `manage-power` developed for HDM 3.1.
		60	Topic; `prepare-for-removal` developed for HDM 3.1.
		65	Topic; `secure-purge` developed for HDM 3.1.
-305	07/30/2015	32	Topic; `configure-smart` command. Note added to clarify available spare threshold limitations vs. currently available spare threshold value and read-only mode.
		57	Topic; `manage-power` command. Note added in regards to proper usage of command and the possible adverse consequences to changing the power state of the device.
		62	Topic; `resize` command. Updated table listing the minimum and maximum ranges according to drive capacity.
-306	11/09/2015	Title	Title updated to reflect HDM 3.2 Alpha release.
		20	Topic; Linux DEB Installation documented.
		37-39	Topic; `format` command. Complete re-write of topic, with explanation of valid option combinations for SAS and NVMe drives.
		41-42	Topic; `get-capabilities` command documented.
-307	05/12/2016	17	Table 1; list of Supported Products updated to included Ultrastar SAS series of SSDs.
		18	Topic; Operating Systems. Listed Ubuntu Linux 16.04 POWER8, Little-Endian (LE).
		35	Topic; `configure-smart` command. Updated functional description of `--clear-all`, `--clear`, `--restore` and `--restore-all` options.
		69	Topic; clarified the `overwrite` option for the `sanitize` command.
-308	05/12/2016	Many	Mass changes to all sections for accuracy, brevity, and to broaden document to apply to SAS SSDs as well.

Revision	Date	Page(s)	Description
-309	05/26/2016	Many	Widespread changes to document, EULA, and footers to reflect that this document is for a GA release.

Table of Contents

List of Tables

List of Figures

Scope

Overview

The HGST Device Manager (HDM) is designed to efficiently assist in the administration of HGST Solid-State Drive (SSD) devices. While many IT organizations have embraced the benefits of solid-state technology, many of the current hard disk drive utilities have proven woefully inadequate in the management of enterprise solid-state devices. HDM provides the ability to easily update, manage, and configure SSDs.

Audience

This user guide is intended for system administrators, network administrators, and other IT professionals. It is therefore written specifically for a technically advanced audience; it is not intended for end-users that will eventually purchase the commercially available product. The *user*, as referenced throughout this guide, is primary concerned with industrial, commercial and enterprise networking applications.

Features

Feature	Description
Cross-Platform Solution	Simplifies the management of SSDs in Linux 64-Bit and Windows 64-bit environments.
Command Line Interface (CLI)	A consistent command line interface that supports the use of Linux scripts or Windows batch files to automate tasks. The user can choose from four (4) modes of output: `mini`, `text`, `json` and `csv`. The `mini` option (default) presents the output in an abridged format, the `text` output is a human-readable format, the `json` output is for a scripting environment, and the `csv` format can be used for data mining purposes and analysis.

Specifications

Specification	Description
Overview	Software solution that operates independently of host hardware for managing multiple drives.
Solid-State Device Compatibility	*See Supported Products*.
Operating Systems	See *Operating Systems* under *Prerequisites*.
Capabilities	Integrated solid-state device solution; firmware management, namespace management, power management, report generation, secure erase and secure purge, factory reset and format, resize, statistics, diagnostic capture, and S.M.A.R.T. data retrieval.
Management	CLI utilities to manage solid-state devices.

Supported Products

HDM can discover and manage the following SSDs as outlined in Table 1: Supported Products.

√ *HDM 3.2 requires that Firmware Version KMGNP110 or later be installed on the SN100 or SN150 Series SSDs; otherwise, the execution of the* `resize`, `reset-to-defaults` *and* `manage-namespaces` *commands will return an error.*

Table 1: Supported Products

Product Family	Description
Ultrastar® SN100 Series	SN100 NVMe PCIe 3.0 x4 Lane 2.5-Inch SFF Solid-State Drives
Ultrastar® SN150 Series	SN150 NVMe PCIe 3.0 x4 Lane HH-HL Edge Card Solid-State Drives
Ultrastar® SSD800MH	Ultrastar 2.5-Inch 12Gbps SAS-3 SFF MLC HE Solid-State Drives
Ultrastar® SSD800MM	Ultrastar 2.5-Inch 12Gbps SAS-3 SFF MLC ME Solid-State Drives
Ultrastar® SSD800MH.B	Ultrastar 2.5-Inch 12Gbps SAS-3 SFF MLC HE Solid-State Drives
Ultrastar® SSD1000MR	Ultrastar 2.5-Inch 12Gbps SAS-3 SFF MLC RI Solid-State Drives
Ultrastar® SSD1600MM	Ultrastar 2.5-Inch 12Gbps SAS-3 SFF MLC ME Solid-State Drives
Ultrastar® SSD1600MR	Ultrastar 2.5-Inch 12Gbps SAS-3 SFF MLC RI Solid-State Drives

Supported Controllers

HDM can discover and manage Serial Attached SCSI (SAS) SSDs managed by the following controllers outlined in Table 2: Supported Controllers.

Table 2: Supported Controllers

Controller Vendor	Controller Family
Avago	SAS/SATA 92xx Host Bus Adapters
Avago	SAS/SATA 93xx Host Bus Adapters
Avago	MegaRAID SATA + SAS 92xx RAID Controllers
Avago	MegaRAID SATA + SAS 93xx RAID Controllers
ATTO	ExpressSAS SAS/SATA H6xx Host Bus Adapters
ATTO	ExpressSAS SAS/SATA H6xx Host Bus Adapters

Installation

Overview

This section addresses issues regarding the compatibility, system requirements, installation and configuration of HDM.

Prerequisites

The prerequisite platform or software described below should be installed and configured before you install HDM.

Operating Systems

√

HDM 3.2 only supports the kernels that are supplied with the operating systems distributed by the OS vendors. If the user should compile any other kernel versions into the operating systems, then the configuration shall not be considered officially supported by HDM.

Table 3: Supported Operating Systems

Operating System	Description
Red Hat Enterprise Linux (RHEL)	RHEL 6.6 and 6.7, x86_64 with supplied kernel RHEL 7, 7.1, and 7.2, x86_64 with supplied kernel RHEL 7.2, POWER Little-Endian with supplied kernel RHEL 7.2, POWER Big-Endian with supplied kernel
Community Enterprise Linux (CentOS)	CentOS 6.7, x86_64 with supplied kernel CentOS 7.2, x86_64 with supplied kernel
SUSE Linux Enterprise Server (SLES)	SLES 11 SP2, 11 SP3, and 11 SP4, x86_64 with supplied kernel SLES 11 SP4, POWER Big-Endian with supplied kernel SLES 12 and 12 SP1, x86_64 with supplied kernel
Ubuntu Server Linux	Ubuntu 14.04, 15.04, and 16.04, x86_64, with supplied kernel Ubuntu 15.04 and 16.04, POWER Little-Endian with supplied kernel
Microsoft Windows	Windows Server 2008 R2, 64-bit Windows Server 2012, 64-bit Windows Server 2012 R2, 64-bit Windows 7, 64-bit Windows 8 and 8.1, 64-bit

User Privileges

Table 4: User Privileges

Prerequisite	Description
Linux Distributions	root
Microsoft Windows	Administrator

Supported Drivers

The minimum version of the HGST NVMe driver required by HDM 3.2 is 1.0.1.50625. The corresponding driver file names will be prefixed by "HGST-NVMe-10150625".

It is strongly recommended to upgrade Serial Attached SCSI (SAS) Host Bus Adapters (HBAs) or RAID controllers to the latest driver.

Table 5: Supported Drivers

Prerequisite	Description
HGST NVMe Windows Driver	Windows NVMe Driver provided by HGST, Inc. **Note**: HDM is not compatible with Windows In-box NVMe drivers due to the limitations of these drivers.
HGST NVMe Linux Driver	Linux NVMe Driver provided by HGST, Inc.
Linux In-box NVMe Drivers	NVMe drivers included with Linux Distributions.
HBA and RAID controller Drivers	Drivers provided by vendors of Host Bus Adapters and RAID controllers.

System Hardware Requirements

Table 6: System Hardware Requirements

Prerequisite	Description
CPU	Intel Single-Core 64-bit Processor AMD x86_64 Processor
Display	1024 x 768 or higher resolution monitor.
Hard Disk Drive	2.0 MB available disk space minimum.
Memory	512MB RAM
Network	Internet connection required for downloads and upgrades.
Optical Drive	CD-ROM/DVD-ROM optional.
Peripherals	Keyboard and Mouse (Printer optional).

Installation Packages

HDM installers are available for a variety of platforms. It is the responsibility of the user to be knowledgeable of the specific platform on which HDM will be installed. An authorized representative will e-mail the desired version of HDM to the user, or the user may download the package(s) from the HGST Support website. The packages must be unpacked in a default directory or a temporary directory. It is recommended that the user create the default or temporary directories off the root of the drive hierarchy. Table 7: Installation Packages lists the available installer packages and descriptions.

Table 7: Installation Packages

Part Number	Installation Package	Description
66000-00053-003	HGST Device Manager-3.2.0-win64.msi	For Windows 64-bit platforms
66000-00054-003	hdm-core-3.2.0-x.yy-x86_64.rpm	For RHEL-based Linux 64-bit x86 platforms
66000-00080-001	hdm-core-3.2.0-x.yy-ppc64le.rpm	For RHEL-based Linux 64-bit POWER Little Endian platforms
66000-00081-001	hdm-core-3.2.0-x.yy-amd64.deb	For Debian-based Linux 64-bit x86 platforms
66000-00082-001	hdm-core-3.2.0-x.yy-ppc64el.deb	For Debian-based Linux 64-bit POWER Little Endian platforms
66000-00TBD-001	hdm-core-3.2.0-x.yy-ppc64.deb	For Debian-based Linux 64-bit POWER Big Endian platforms
Note: x = build number, yy = build type		

Event Logging

HDM supports the use of event logging as a method of tracking drive health and operation. Event messages are usually logged to and operating system event log such as the Event Viewer (Windows) or `/var/log/syslog` (Linux).

It is strongly recommended that the user completely review each installation procedure respective to the individual platform, including any Installation Notes.

It is strongly recommended that the user uninstall any previous versions of HDM before installing the latest version.

The installation packages are only compatible with 64-bit systems.

The installer will check whether the necessary prerequisite packages are present before continuing the installation.

Ultrastar PCIe NVMe drives require an NVMe driver to function. Most operating systems supported by HDM provide an inbox NVMe driver. HGST also provides an NVMe driver for Linux and Windows. HDM is compatible with all HGST NVMe drivers as well as NVMe drivers for all supported Linux operating systems. HDM is not compatible with inbox NVMe drivers for Windows operating systems due to the limitations of these drivers.

Linux RPM Installation

To install the RPM package(s):

1. Logon to the system with `root` privileges.
2. Create a temporary installation folder or directory.
3. Download the HDM 3.2 installable (RPM) file to the temporary installation folder or directory. *See Installation Packages to determine which `rpm` file to download*.
4. Open a terminal window and change directories to the temporary installation folder or directory.
5. Use the `rpm` command to install the RPM file, e.g., `rpm -ivh <HDM RPM file name>`.

Uninstall Procedure

To uninstall HDM:

1. Logon to the system with `root` privileges.
2. Open a terminal window or console.
3. Run `rpm -e <hdm-core>`.

Linux DEB Installation

To install the DEB package(s):

1. Logon to the system with `root` privileges.
2. Create a temporary installation folder or directory.
3. Download the HDM 3.2 installable (DEB) file to the temporary installation folder or directory. *See Installation Packages to determine which `deb` file to download*.
4. Open a terminal window and change directories to the temporary installation folder or directory.
5. Use the `dpkg` command to install the DEB file, e.g., `dpkg -i <HDM RPM file name>`.

Uninstall Procedure

To uninstall HDM:

1. Logon to the system with `root` privileges.
2. Open a terminal window or console.
3. Run `dpkg -r hdm`.

Windows Installation

The user must install the HGST In-house Driver before using HDM to manage HGST devices under Windows.

To install HDM under Windows:

1. Logon to the system with Administrator privileges.
2. Create a temporary installation folder or directory.
3. Download the HDM 3.2 `msi` file: `HGST Device Manager-3.2.0-win64.msi` to the temporary installation folder or directory.
4. Double-click the **Installer Package (msi)**.

   Notes:

   1. If a Microsoft SmartScreen filter message appears informing you that the package being installed cannot be verified, click **More Information** and then allow the package to be installed.
   2. Enter any administrative credentials if so prompted.
5. The HGST Device Manager Setup Window appears.
6. Click **Next** to continue. The End-User License Agreement will appear.
7. Review the terms of the EULA and then place a check mark in the **I accept the terms in the License Agreement** check box.
8. Click **Next** to continue. You are prompted to choose the location of the working directory. The default path will appear in the text box, but you can use the **Change…** option to navigate to an alternate installation directory.
9. Click **Next** to continue. You are now prompted to install HGST Device Manager.
10. Click **Install**. The HDM CLI will be installed.

Uninstall Procedure

To uninstall HDM:

1. Open the Control Panel window.

   1.1. Navigate to the Add/Remove Programs panel, or;

   1.2. Navigate to the Programs and Features panel.
2. Click the application “HGST Device Manager” or “HDM” in the list to select it.
3. Click the Uninstall button.
4. Confirm the removal of HDM.

Installation Notes

The user will notice that after HDM is installed:

1. All the HDM-related files are contained in the installation path that was specified in the Setup Wizard.
2. The **Add/Remove Programs** panel or **Programs and Features** panel under the **Control Panel** will contain the application of “HGST Device Manager”.
3. All command execution for the CLI requires the use of a terminal / command window and administrator privileges.

Overview

This section explains the usage and capabilities of the Command Line Interface (CLI) and provides basic and advanced device management and diagnostic functions. The CLI can also be used in situations where automated tasks can be compiled as scripts or batch files.

Command Execution

The syntax for command execution is consistent across the various platforms. In this section, the commands are presented in the platform neutral form of **hdm**. The user should have a practical knowledge of navigating the command line interface for the specific system platform.

Command Syntax

The commands and options use the same syntax across the platforms. The spaces or delimiters are taken literally, while the brackets are ignored:

hdm <command> <command arguments> <device references>

Where **hdm** is a "meta" command that invokes the various management or administrative commands, while <command> is the actual operation.

Command Arguments

The <command arguments> (command arguments) may consist of zero or multiple combinations of options or flags. An option is considered an input parameter that requires a user supplied value, while flags modify the command behavior.

Device References

A *device-reference* describes how a device is accessed when executing a command and may be expressed as shown in Table 8: Device References. *See Linux Directory Trees vs. Windows Device Instance IDs*.

Table 8: Device References

Argument	Description	Example
--uid (-u)	The unique identifier (UID) of the device.	--uid 1C58STM0001A6758HUSPR3216AHP301
--path (-p)	The OS native path or device instance ID.	--path /dev/*nvme0*
--alias (-a)	The HDM internal alias.	--alias myssd01

Notes:

1. The unique identifier is device specific and HDM uses the following unique identifiers:
   1.1. NVMe Controllers: A concatenation of the PCI Vendor ID, Serial Number, Model Number (Identify Controller Structure).
   1.2. NVMe Namespaces: IEEE Extended Unique Identifier (EUI64) (Identify Namespace Structure).
   1.3. SAS Drives: WWNN.
2. While the most intuitive method may be to reference a device using the OS native path, the enumeration of the devices, the controller paths, and namespaces may change after a reboot.

Device Reference Examples

For example, the user may want to obtain the properties of a device using the `get-info` command. The user must therefore specify the *device-reference*; the OS path, an alias or a device unique identifier.

OS Path Example

The ***<device-reference>*** argument is the **OS path** and follows the `--path` input option:

```
hdm get-info --path /dev/nvme0
```

Alias Example

The ***<device-reference>*** argument is the **alias** that was assigned to the device and follows the `--alias` input option:

```
hdm get-info --alias myssd01
```

Device Unique Identifier Example

The ***<device-reference>*** argument is the **device unique identifier** assigned to the device and follows the `--uid` input option:

```
hdm get-info --uid 1C58STM001A6758HUSPR3216AHP301
```

Linux Directory Trees vs. Windows Device Instance IDs

It is important to understand that Linux and Windows manage devices using different methods. Linux implements the `/dev/` directory tree scheme where the `/dev/` directory tree contains all the "device" files. Examples of device paths on Linux systems are: `/dev/nvme0` and `/dev/sda`

However, under Windows, HDM uses the device instance paths or device instance IDs assigned by the system. A device instance ID is a unique identification string assigned by the Plug and Play (PnP) Manager to each device node in the device tree. The format of the string consists of an instance ID concatenated to a device ID, i.e., `<device-ID>\<instance-specific-ID>`. The device instance ID is persistent across system restarts. An example of a Windows device instance ID is:
`\\?\pci#ven_1c58&dev_0003&subsys_00031c58&rev_05 #4&11cc8299&0&0009#{2accfe60-c130-11d2-b082-00a0c91efb8b}`

These entities are shown in the output of most HDM commands using the Device Path property.

Short Form Syntax

Table 9: Short Form Command Arguments lists the "short form" syntax that is available for the following <command args> (command arguments).

Table 9: Short Form Command Arguments

Long Form	Short Form	Example
--output-format FORMAT	-o	hdm get-state **-o** mini –path /dev/nvme0
--config PATH	-c	hdm get-state **-c** $HOME/hdm.ini --path /dev/nvme0
--path PATH	-p	hdm get-state **-p** /dev/nvme0
--uid UID	-u	hdm get-state **-u** 1C58STM0001A6758HUSPR3216AHP301
--alias ALIAS	-a	hdm get-state **-a** myssd01
--file FILE	-f	hdm configure-trace **-f** nvme0trace --level 3
--interval INTERVAL	-i	hdm get-statistics **-i** 14 --category performance --path /dev/nvme0

Configuration Files

The --config PATH or –c option allows the user to specify a configuration file other than the default. The option expects the user to specify the path to the configuration file, hdm.ini. The configuration file is usually located in the $HOME/.hdm directory by default. If the user does not specify $HOME, then the current directory is used.

The** --config PATH **option is for advanced usage and is not recommended for normal situations.

Listing the Installed Devices

The first command that the user should run is **hdm scan**. The **hdm scan** command will list information about all HDM supported devices that are installed in the host system and currently recognized by the operating system. These names can then be used in the execution of subsequent commands. For example, **get-info** can use either the {--path PATH |--uid UID |--alias ALIAS} options to extract drive information.

```
[user@host]$hdm scan
       [1C58STM00019C205HUSPR3232AHP301]
         Device Type        = Ultrastar SN150
         Device Path        = /dev/nvme0
         UID                = 1C58STM00019C205HUSPR3232AHP301
         Alias              = @nvme0
         Vendor Name        = HGST
         Model Name         = HUSPR3232AHP301
         Vendor Id          = 1C58
         Device Id          = 0003

       [000CCA0060077E80]
         Device Type        = NVMe Namespace
         Device Path        = /dev/nvme0n1
         UID                = 000CCA0060077E80
         Alias              = @nvmens0
         Parent Type        = Ultrastar SN150
         Parent Device Path = /dev/nvme0
         Parent UID         = 1C58STM00019C205HUSPR3232AHP301
         Namespace ID       = 1

       [5000CCA02B022D0C]
         Device Type        = Ultrastar SSD1600MR
         Device Path        = /dev/sda
         UID                = 5000CCA02B022D0C
         Alias              = @scsi0
         Vendor Name        = HGST
         Model Name         = HUSMR1616ASS200

       Results for scan: Operation succeeded on 3 of 3 devices.
```

Figure 1: Listing the Installed Devices

Output Types

The `--output-format FORMAT` or `-o` option can be specified with any HDM command for flexibility of output after successful command execution. The user substitutes `mini` (for MINI output), `text` (for ASCII text output), `json` (for JSON output), or `csv` (for Comma-Separated Values output) for the `FORMAT` value. If no output format is specified, HDM will default to MINI output.

MINI Output

The MINI output is an abridged, simplified format of the command output. The use of `mini` is optional; the format is output by default if the user does not specify the `text`, `json`, or `csv` formats. For example:

hdm get-state --output-format mini --path /dev/nvme0

hdm get-state -o mini --path /dev/nvme0

OR

hdm get-state --path /dev/nvme0

The output will be in the `mini` output format:

```
[/dev/nvme0]
  Device Type           = Ultrastar SN150
  Device Path           = /dev/nvme0
  UID                   = 1C58STM00019C20AHUSPR3232AHP301
  Alias                 = @nvme0
  Device Status         = Ready
  Life Gauge            = 100
  Power Consumption     = 7.795313
  Dual Port Mode        = Single Port
  Encryption Mode       = No Encryption

Results for get-state: Operation succeeded.
```

Figure 2: MINI Output Example

Text Output

The user can specify the `text` option to specify that the output appear on the computer screen as shown in the following figure. The output is presented in a tabular, human-readable format.

hdm get-state --output-format text --path /dev/nvme0

```
================================================================================
                            Results for get-state
================================================================================
+-----------------------------------------------------------------------------+
| Device Type                       | Ultrastar SN150                         |
+-----------------------------------------------------------------------------+
| Device Path                       | /dev/nvme0                              |
+-----------------------------------------------------------------------------+
| UID                               | 1C58STM00019C20AHUSPR3232AHP301         |
+-----------------------------------------------------------------------------+
| Alias                             | @nvme0                                  |
+-----------------------------------------------------------------------------+
| Device Status                     | Ready                                   |
+-----------------------------------------------------------------------------+
| Life Gauge                        | 100                                     |
+-----------------------------------------------------------------------------+
| Power Consumption                 | 7.788625                                |
+-----------------------------------------------------------------------------+
| Dual Port Mode                    | Single Port                             |
+-----------------------------------------------------------------------------+
| Encryption Mode                   | No Encryption                           |
+-----------------------------------------------------------------------------+
                        Result: Operation succeeded.
================================================================================
```

Figure 3: Text Output Example

JSON Output

JSON is an acronym for JavaScript Object Notation and is an open standard format that uses human-readable text to transmit data objects consisting of attribute-value pairs. The format is often employed for a scripting environment. To specify JSON output, use the `--output-format json` option.

```
hdm get-state --output-format json --path /dev/nvme0
```


```
{
  "status":0,
  "results":[
    {
      "ref":{"path":"/dev/nvme0"},
      "device_type":2004,
      "device_path":"/dev/nvme0",
      "uid":"1C58STM00019C20AHUSPR3232AHP301",
      "alias":"@nvme0",
      "device_status":3000,
      "life_gauge":100,
      "power_consumption":7.550063,
      "dual port mode":1003,
      "encryption_mode":1006,
      "status":0
    }
  ]
}
```


Figure 4: JSON Output Example

CSV Output

CSV is an acronym for Comma-Separated Values (a.k.a. Character-Separated Values). The CSV file type is a compilation of tabular data (numbers/text) in a plain-text form; the file is a sequence of characters, with no interpretation of binary numbers, divided into records that are subdivided into fields. These fields are separated by a literal comma (,). The user should specify the filename extension of `*.csv` to easily identify the file.

To specify CSV output, use the `--output-format csv` option. Figure 5 shows an example of CSV output within a text editor. It is then possible to import the CSV file into a spreadsheet program, or use the CSV data contained in the file for data mining purposes or analysis.

```
hdm get-smart --output-format csv --path /dev/nvme0
```


```
DEVICE_REFERENCE,STATUS,DEVICE_TYPE,DEVICE_PATH,UID,ALIAS,DATA_UNITS_READ
,DATA_UNITS_WRITTEN,HOST_READ_COMMANDS,HOST_WRITE_COMMANDS,AVAILABLE_SPAR
E,AVAILABLE_SPARE_THRESHOLD,AVAILABLE_SPARE_CRITICAL_WARNING,PERCENTAGE_U
SED,TEMPERATURE,TEMPERATURE_THRESHOLD,TEMPERATURE_CRITICAL_WARNING,DEVICE
_RELIABILITY_CRITICAL_WARNING,MEDIA_READ_ONLY_MODE_CRITICAL_WARNING,VOLAT
ILE_MEMORY_BACKUP_DEVICE_FAILURE_CRITICAL_WARNING,CONTROLLER_BUSY_TIME,PO
WER_CYCLES,POWER_ON_HOURS,UNSAFE_SHUTDOWNS,MEDIA_AND_DATA_INTEGRITY_ERROR
S,ERROR_INFORMATION_LOG_ENTRIES,RAW_READ_ERROR_COUNT,SOFT_READ_ERROR_COUN
T,TOTAL_ERASE_COUNT,MAX_ERASE_COUNT,TOTAL_PROGRAM_FAILURES,TOTAL_BLOCK_ER
ASE_FAILURES,POWER_BACKUP_FAULTS,NAND_UNITS_READ,NAND_UNITS_WRITTEN,SMART
_STATUS
/dev/nvme0,Success,Ultrastar
SN150,/dev/nvme0,1C58STM00019C20AHUSPR3232AHP301,@nvme0,260567422,3149475
58,18553929271,17450482756,100,10,false,0,40,90,false,false,false,false,7
513,45,3748,19,0,1,,,,,,,,,Normal
```


Figure 5: CSV Output Example

Commands

Table 10: HDM Commands lists the valid commands for HDM 3.2 that must be specified prior to an input or value parameter.

While HDM commands and options are usually not case-sensitive, the user should be aware that paths, UIDs, aliases, etc. are case sensitive.

Table 10: HDM Commands

Command	Description
alias	Sets or clears a user supplied name for a device.
capture-diagnostics	Captures diagnostic data from a physical device.
configure-smart	Configures SMART thresholds or clears warnings on physical devices.
configure-trace	Configures the trace level and destination file.
format	Performs a low-level format on devices.
generate-report	Retrieves information and statistics for devices.
get-capabilities	Information about the commands supported about the physical devices.
get-feature	Retrieves feature data from devices.
get-info	Retrieves properties from devices.
get-log-page	Retrieves log pages from devices.
get-smart	Retrieves SMART statistics and thresholds from devices.
get-state	Retrieves state information from devices.
get-statistics	Retrieves statistics from devices.
help	Prints command help information.
locate	Enables or disables the LED beacon for physical devices.
manage-firmware	Displays and updates firmware on physical devices.
manage-namespaces	Displays and manages namespaces on a capable physical device.
manage-power	Displays and controls power states of a physical device.
monitor	Continuously retrieves device information at a specified interval.
prepare-for-removal	Prepares physical devices for safe removal from the system.
reset-to-defaults	Resets physical devices to factory default configuration.
resize	Changes the user capacity of physical devices.
sanitize	Performs a sanitize operation to erase all user data on physical devices.
scan	Lists the storage devices attached to this host.
secure-erase	Securely erases all user data from physical devices.
secure-purge	Performs a secure purge on physical devices.
version	Display HDM build and version information.

Command Support

Table 11: Supported Commands cross-references the HDM commands supported in the current release. The user should review the release notes for details.

Table 11: Supported Commands

	Device Type	
Command	**Ultrastar NVMe SSDs**	**Ultrastar SAS SSDs**
alias	Yes	Yes
capture-diagnostics	Yes	Yes
configure-smart	Yes	No
configure-trace	Yes	Yes
format	Yes	Yes
generate-report	Yes	Yes
get-capabilities	Yes	Yes
get-feature	Yes	No
get-info	Yes	Yes
get-log-page	Yes	No
get-smart	Yes	No
get-state	Yes	Yes
get-statistics	Yes	No
help	Yes	Yes
locate	Yes	No
manage-firmware	Yes	Yes
manage-namespaces	Yes	No
manage-power	Yes	No
monitor	Yes	Yes
prepare-for-removal	Yes	Yes
reset-to-defaults	Yes	No
resize	Yes	No
sanitize	No	Yes
scan	Yes	Yes
secure-erase	Yes	No
secure-purge	Yes	No
version	Yes	Yes

Help System

Complete Command Listing

The `help` command provides assistance for all supported HDM commands. If `help` is used without any arguments, it will list the available `commands` as shown below.

```
usage: hdm <command>              [-o, --output-format FORMAT] [-c, --config PATH]
                                  {-p, --path PATH | -u, --uid UID | -a, --alias ALIAS}
                                  <command args>

OPTIONS:
  -o, --output-format FORMAT      Output format for the command. Valid output formats are "text",
                                  “mini”, "csv", and "json". The default is "mini".
  -c, --config PATH               Path to the HDM configuration file.
  -u, --uid UID                   Device unique identifier.
  -p, --path PATH                 Platform path of the device as shown in the output of hdm scan.
  -a, --alias ALIAS               User supplied name for the device.

COMMANDS:
  alias                           Sets or clears a user supplied name for a device.
  capture-diagnostics             Captures diagnostic data from a physical device.
  configure-smart                 Configures SMART thresholds and clears warnings on physical devices.
  configure-trace                 Configures the trace level and destination file.
  format                          Performs a low-level format on devices.
  generate-report                 Retrieves information and statistics for devices.
  get-capabilities                Information about the commands supported by the physical devices.
  get-feature                     Retrieves feature data from devices.
  get-info                        Retrieves properties from devices.
  get-log-page                    Retrieves log pages from devices.
  get-smart                       Retrieves SMART statistics and thresholds from devices.
  get-state                       Retrieves state information from devices.
  get-statistics                  Retrieves statistics from devices.
  help                            Prints command help information.
  locate                          Enables or disables the LED beacon for physical devices.
  manage-firmware                 Displays and updates firmware on physical devices.
  manage-namespaces               Displays and manages namespaces on a capable physical device.
  manage-power                    Displays and controls power states of a physical device.
  monitor                         Continuously retrieves device information at a specified interval.
  prepare-for-removal             Prepares physical devices for safe removal from the system.
  reset-to-defaults               Resets physical devices to factory default configuration.
  resize                          Changes the user capacity of physical devices.
  sanitize                        Performs a sanitize operation to erase all user data on physical devices.
  scan                            Lists the storage devices attached to this host.
  secure-erase                    Securely erases all user data from physical devices.
  secure-purge                    Performs a secure purge on physical devices.
  version                         Display HDM build and version information.

Run 'hdm help <command>' for more information about a particular hdm command.
For more information about HDM, refer to the HDM User Guide and Release Notes documents.
```

Specific Command Help

If the <command> follows the help command, e.g., hdm help *scan*, it will list only the options that are relevant to the scan command:

```
hdm help scan
usage: hdm scan                [-o, --output-format FORMAT] [-c, --config PATH]

Lists the storage devices attached to this host.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini" "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
```

Command Confirmation

HDM commands that can perform destructive operations that result in the deletion of user and system data require user confirmation. The format, manage-namespaces, reset-to-defaults, resize, sanitize, secure-erase, and secure-purge commands prompt for confirmation. The following confirmation prompt will appear for destructive commands:

```
Press 'y' to continue
```

Press 'y' to confirm. The user can press any other key to cancel the operation. The user may override the prompt using the --force option.

Command Exit Status Codes

HDM may return one or more exit status codes to indicate the outcome of an executed command. Table 12: Command Exit Status Codes lists the possible status results that may occur.

Table 12: Command Exit Status Codes

Exit Status	Description
0	Success
1	Invalid user supplied parameter or value.
2	Device error.
3	I/O error.
4	Permissions error.
5	System error.
6	Qualified success.
7	Internal error.

alias

The `alias` command allows a user to assign or remove a user-readable name, or a name supplied by the user to a device, as a substitute for the system assigned name. The *device-reference* must refer to a single physical device. The user can then specify the use of the alias in all subsequent HDM commands that require a *device-reference*.

Device aliases are assigned by default to all devices the very first time the HDM scan command is run.

Synopsis

```
usage: hdm alias                [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [--name NAME] [--clear-alias]

Sets or clears a user supplied name for a device.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                “mini”.
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --name NAME                   The name to be used for this operation.
  --clear-alias                 Clears the alias for the specified device.
```

Examples

To assign an alias to a device:

`hdm alias --name alias-name --path /dev/nvme0`

OR

`hdm alias --name nvme32Tn1 --path /dev/nvme0`

To remove an alias from a device:

`hdm alias-name --clear-alias --path/dev/nvme0`

OR

`hdm nvme32Tn1 --clear-alias --path/dev/nvme0`

capture-diagnostics

The `capture-diagnostics` command will capture all diagnostic data for a device and compile it to a specific container file that the user can then send to technical support for analysis. The contents of the captured data vary by drive type. The *device-reference* must refer to a single physical device (logical devices like NVMe namespaces are not supported).

Synopsis

```
usage: hdm capture-diagnostics   [-o, --output-format FORMAT] [-c, --config PATH]
                                 {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                 [-f, --file FILE] [--name NAME]

Captures diagnostic data from a physical device.

OPTIONS:
  -o, --output-format FORMAT     Output format for the command. Valid output formats
                                 are "text", "mini", "csv", and "json". The default
                                 is "mini".
  -c, --config PATH              Path to the HDM configuration file.
  -u, --uid UID                  Device unique identifier.
  -p, --path PATH                Platform path of the device as shown in the output
                                 of hdm scan.
  -a, --alias ALIAS              User supplied name for the device.
  -f, --file FILE                File name or path.
  --name NAME                    The name to be used for this operation.
```

Options

Option	Description
`-f, --file FILE`	This option will specify the file name of the compressed archive that contains all the captured data files. The `*.zip` filename extension is appended to the file name. The option is only valid if the user does not specify the `--name` option.
`--name NAME`	This option will specify the base name of the compressed archive containing all the diagnostic data files. The name can begin with a directory name to output the archive into an alternate directory. This option is only valid if the user does not use the `-f, --file` option. The drive serial number, drive firmware version, current date, current time, and `*.zip` extension are appended to the base name with underscore separators to create the compressed archive name: `basename_<serial_number>_<current_fw_version>_<date>_<time>.zip`

Examples

```
hdm capture-diagnostics --file nvme0diags --path /dev/nvme0
```

OR

```
hdm capture-diagnostics --name nvme0diags --path /dev/nvme0
```

configure-smart

The `configure-smart` command will allow SMART thresholds to be set, cleared, or restored to default values. The `--set`, `--clear` and `--restore` options may be specified only once on a single command line and combinations of `--set`, `--clear`, `--restore`, `–clear-all` and `--restore-all` options are not allowed in the same command invocation.

***Ultrastar NVMe PCIe Devices Only**: Users should be aware that if the* `availablespare` *threshold is exceeded, the device will enter Read-Only Mode. Users should be careful not to specify too high a value for the available spare threshold, or that the value be greater than the currently available spare value.*

***Ultrastar NVMe PCIe Devices Only**: Calling the HDM get-smart or get-log-page commands will also cause all warning conditions to be cleared. This behavior is defined by the NVMe specification.*

Synopsis

```
usage: hdm configure-smart    [-o, --output-format FORMAT] [-c, --config PATH]
                              {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                              [--set NAME=VALUE] [--clear-all] [--clear NAME]
                              [--restore NAME] [--restore-all]

Configures SMART thresholds and clears warnings on physical devices.

OPTIONS:
  -o, --output-format FORMAT   Output format for the command. Valid output formats
                               are “text”, “mini”, “csv”, and “json”. The default is
                               “mini”.
  -c, --config PATH            Path to the HDM configuration file.
  -u, --uid UID                Device unique identifier.
  -p, --path PATH              Platform path of the device as shown in the output of
                               hdm scan.
  -a, --alias ALIAS            User supplied name for the device.
  --set NAME=VALUE             Sets the value of a specific SMART threshold.
  --clear-all                  Clears all threshold warnings.
  --clear NAME                 Clears the specified threshold warning.
  --restore NAME               Restores the specified threshold to its default value.
  --restore-all                Restores all thresholds to default values.
```

Options

--set NAME=VALUE	**Usage**
temperature=(Celsius)	The `temperature` value is used to set the SMART temperature threshold. The threshold is expressed in degrees `Celsius`. For example, to set the temperature threshold at `60°C`, the expression would be `--set temperature=60`.
availablespare=(percentage)	The `availablespare` value is used to either set or clear the available spare threshold. The threshold is expressed as a `percentage`, e.g., to set the threshold at 10%, `--set availablespare=10`.
--clear-all	The `--clear-all` option will clear both the `temperature` and `availablespare` SMART threshold warnings.
--clear NAME	The `--clear` option will clear either the `temperature` or `availablespare` SMART threshold warnings. **Note**: This option is not valid for NVMe devices; the `--clear-all` option must be used to clear SMART thresholds for NVMe devices.
--restore NAME	The `--restore` option will restore either the `temperature` or `availablespare` thresholds to the default values.
--restore-all	The `--restore-all` option will restore both the `temperature` and `availablespare` thresholds to the default values.

Examples

To set the SMART `temperature` threshold:

```
hdm configure-smart --set temperature=60 --path /dev/nvme0
```

To set the SMART `availablespare` threshold:

```
hdm configure-smart --set availablespare=20 --path /dev/nvme0
```

To clear all the threshold `temperature` and `availablespare` warnings:

```
hdm configure-smart --clear-all --path /dev/nvme0
```

To restore the SMART `temperature` threshold to its default value:

```
hdm configure-smart --restore temperature --path /dev/nvme0
```

To restore the SMART `availablespare` threshold to its default value:

```
hdm configure-smart --restore availablespare --path /dev/nvme0
```

To restore all the SMART thresholds to default values:

```
hdm configure-smart --restore-all --path /dev/nvme0
```

Note: *The following is an example of an invalid* `configure-smart` *command, with the* `--set` *and* `--clear` *options being invoked within the same command line:*

```
hdm configure-smart --set temperature=60 --clear availablespare
--path /dev/sda
```

configure-trace

The `configure-trace` command allows the user configure the HDM trace level and specify the path and filename for the HDM trace output. The `--level` parameter determines the verbosity of output to be recorded in the trace file. The trace file specified by the `-f, --file` parameter will create the filename in the specified path if it does not exist. If the file name currently exists then the trace output will be appended to the file. The messages in the trace file are generated by HDM and record a variety of error conditions, including the error return codes from the OS, status codes from the device, and internal HDM errors.

Synopsis

```
usage: hdm configure-trace     [-o, --output-format FORMAT] [-c, --config PATH]
                               [-f, --file FILE] [--level LEVEL]

Configures the trace level and destination file.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -f, --file FILE               File name or path
  --level LEVEL                 Trace level to be used for tracing.
                                0 - do not trace any messages
                                1 - includes only error-class messages
                                2 - includes error-class and warning-class messages
                                3 - includes all trace messages
```

Options

Option	Value	Description
`-f, --file FILE`	`path/trace filename`	Specify the path and filename for the trace data. The file will be compiled in the `$HOME/.hdm` directory if the path is not specified.
`--level LEVEL`	`0, 1, 2, 3`	The `LEVEL` value specifies the trace level to be used for message tracing: `0` (no tracing), `1` (error-class), `2` (error class/warning class) or `3` (trace all messages).

Example

```
hdm configure-trace --file nvme0trace --level 2
```

format

The `format` command will perform a low-level format on the targeted device.

The command is destructive. The user should backup all data currently stored on the device. All user data will be destroyed.

The combination of `format` *options are dependent upon the device type.* *See Valid Option Combinations*.

Notes:

1. The `--sector size`, `--dif-level`, and `--metadata-size` parameters are optional for NVMe devices; however, the user must specify the above parameters when targeting NVMe devices configured with multiple namespaces.
2. If any of the optional parameters are not specified, the current device settings will be used to perform the format operation.
3. If the format command is run against an NVMe controller, the format parameters will be applied to all NVMe namespaces on the device.
4. The user will be prompted to confirm unless the `--force` option is specified.

Synopsis

```
usage: hdm format                [-o, --output-format FORMAT] [-c, --config PATH]
                                 {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                 [--sector-size SIZE] [--dif-level LEVEL]
                                 [--metadata-size SIZE] [--protection-interval NUM]
                                 [--force]

Performs a low-level format on devices.

OPTIONS:
  -o, --output-format FORMAT     Output format for the command. Valid output formats
                                 are "text", "mini", "csv", and "json". The default
                                 is "mini".
  -c, --config PATH              Path to the HDM configuration file.
  -u, --uid UID                  Device unique identifier.
  -p, --path PATH                Platform path of the device as shown in the output
                                 of hdm scan.
  -a, --alias ALIAS              User supplied name for the device.
  --sector-size SIZE             Size of each sector in bytes.
  --dif-level LEVEL              Level of DIF protection.
  --metadata-size SIZE           Size of metadata in bytes.
  --protection-interval NUM      Number of protection information intervals for this
                                 operation.
  --force                        Bypasses all confirmation messages.
```

Options

--sector-size SIZE

The SIZE value specifies the new size of the sector on the device or namespace.

Value	Result
512	512-byte sectors.
520	520-byte sectors.
528	528-byte sectors.
4096	4096-byte sectors.
4160	4160-byte sectors.
4224	4224-byte sectors.

--metadata-size SIZE

The SIZE value specifies the new size for the metadata that includes information about the sector and the protection information. A SIZE of 0 is only valid if no protection level is specified, while a SIZE of 8 is necessary when formatting the device with a valid --dif-level LEVEL (protection level).

Note: This option is not valid for Ultrastar SAS SSD devices.

Value	Result
0	0 bytes metadata (only valid if --dif-level is 0).
8	8 bytes metadata (only valid if --dif-level is > 0).

--dif-level LEVEL

The LEVEL value specifies the DIF (protection) level.

Value	Result
0	Type 0 protection (no protection).
1	Type 1 protection (not valid if --metadata-size is 0).
2	Type 2 protection (not valid if --metadata-size is 0).
3	Type 3 protection (not valid if --metadata-size is 0).

--protection-interval NUM

The NUM value specifies the number of protection intervals per sector.

Note: This option is not valid for Ultrastar NVMe PCIe devices.

Value	Result
1	One (1) whole protection interval per sector.
8	Eight (8) protection intervals interleaved within each sector.

--force

The --force option will override the confirmation prompt and execute the command.

Valid Option Combinations

The `format` command only supports specific combinations of options, depending upon the device type. The following table lists the valid option combinations.

Device Type	Valid Option Combinations	
Ultrastar SAS SSDs	`--sector-size`	`512, 520, 4096, 4160`
	`--dif-level`	`0, 1, 2`
	`--protection-interval`	`1`
	`--sector-size`	`528, 4224`
	`--dif-level`	`0`
	`--protection-interval`	`1`
	`--sector-size`	`4096, 4160`
	`--dif-level`	`2`
	`--protection-interval`	`8`
Ultrastar NVMe SSDs	`--sector-size`	`512, 4096`
	`--dif-level`	`0, 1, 2, 3`
	`--metadata-size`	`0` when `--dif-level` is `0` `8` when `--dif-level` is `>0`
	`--protection-interval`	Not valid.

Examples

Device Format

To perform a default format of a device:

```
hdm format --path /dev/nvme0
```

To format a device with DIF protection:

```
hdm format --sector-size 512 --metadata-size 8 --dif-level 3 --path /dev/nvme0
```

Namespace Format

To perform a default format of a namespace:

```
hdm format --path /dev/nvme0n1
```

To format a namespace with DIF protection:

```
hdm format --sector-size 512 --metadata-size 8 --dif-level 3 --path /dev/nvme0n1
```

generate-report

The `generate-report` command will compile device, state, statistics, and S.M.A.R.T. information for all HDM supported devices in the system or for a specific device. The user can redirect the output to a file using the OS-specific `stdout` or `stderr` operators. The user must specify the *device-reference* if compiling a report for a single device; otherwise, a report will be compiled for all HDM supported devices installed in the system.

The set of properties shown in the `hdm generate-report` *command may vary by device type.*

Synopsis

```
usage: hdm generate-report     [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}

Retrieves information and statistics for devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
```

Examples

To generate a report for a specific device in the system and redirect the output to a text file:

```
hdm generate-report --path /dev/nvme0 > nvme0rpt.txt
```

To generate a report for all HDM supported devices in the system and redirect the output to a text file:

```
hdm generate-report > nvmeallrpt.txt
```

OR

To generate a report for a specific device in the system and display the output on the console:

```
hdm generate-report --path /dev/nvme0
```

To generate a report for all HDM supported devices in the system and display the output on the console:

```
hdm generate-report
```

get-capabilities

The `get-capabilities` command retrieves information about the HDM commands supported for the specified device. The user must specify the *device-reference* if retrieving the capabilities for a single device. The `--command` clause allows the user to determine if a specific HDM command is supported by the device(s). The user can also use multiple instances of the `--command` clause to determine whether the device supports those specific commands.

An error message will appear if the command is a valid HDM command but is not supported by device specified by the device-reference clause.

Synopsis

```
usage: hdm get-capabilities    [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                               [--command CMD]

Information about the commands supported by the physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --command CMD                 The HDM CLI command.
```

Example

To retrieve all the supported HDM commands for a specific device:

```
hdm get-capabilities --path /dev/nvme0
```

To determine whether a specific HDM command is supported for a device:

```
hdm get-capabilities --command command --path /dev/nvme0
```

To determine whether multiple HDM commands are supported for a device:

```
hdm get-capabilities --command command --command command
--command command --path /dev/nvme0
```

Sample Output

In the following example, `get-capabilities` command returned ***all*** the HDM commands supported by the device `/dev/nvme0`:

```
hdm get-capabilities --path /dev/nvme0

[/dev/nvme0]
  Device Type = Ultrastar SN150
  Device Path = /dev/nvme0
  UID         = 1C58STM00019C205HUSPR3232AHP301
  Alias       = @nvme0
  Command     = alias
  Command     = capture-diagnostics
  Command     = configure-smart
  Command     = format
  Command     = generate-report
  Command     = get-feature
  Command     = get-info
  Command     = get-log-page
  Command     = get-smart
  Command     = get-state
  Command     = get-statistics
  Command     = locate
  Command     = manage-firmware
  Command     = manage-namespaces
  Command     = manage-power
  Command     = monitor
  Command     = prepare-for-removal
  Command     = reset-to-defaults
  Command     = resize
  Command     = secure-erase
  Command     = secure-purge

Results for get-capabilities: Operation succeeded.
```

In the following example, multiple instances of the `--command` clause were used to determine if the device `/dev/nvme0` supported the HDM `manage-namespaces` ***and*** `format` commands:

```
hdm get-capabilities --path /dev/nvme0 --command manage-namespaces --command format

[/dev/nvme0]
  Device Type = Ultrastar SN150
  Device Path = /dev/nvme0
  UID         = 1C58STM00019C205HUSPR3232AHP301
  Alias       = @nvme0
  Command     = format
  Command     = manage-namespaces

Results for get-capabilities: Operation succeeded.
```

get-feature

The `get-feature` command will retrieve and display a specific feature that is supported by the target device. The `ID` of the `FEATURE` is specified after the `--feature` option and can be expressed as a whole number or in hexadecimal (1 or 0x01).

Synopsis

```
usage: hdm get-feature          [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [--feature FEATURE]

Retrieves feature data from devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --feature FEATURE             Identifier of the feature. This value must be an
                                integer (e.g. 1) or a prefixed hexadecimal number
                                (e.g. 0x01).
```

Options

`--feature` Value	Description
1 (0x01)	NVMe Feature 01h, Arbitration.
2 (0x02)	NVMe Feature 02h, Power Management.
3 (0x03)	NVMe Feature 03h, LBA Range Type; only supported for Ultrastar NVMe namespaces.
4 (0x04)	NVMe Feature 04h, Temperature Threshold.
5 (0x05)	NVMe Feature 05h, Error Recovery.
6 (0x06)	NVMe Feature 06h, Volatile Write Cache.
7 (0x07)	NVMe Feature 07h, Number of Queues.
8 (0x08)	NVMe Feature 08h, Interrupt Coalescing.
9 (0x09)	NVMe Feature 09h, Interrupt Vector Configuration; not supported by Ultrastar NVMe firmware (controller or namespace).
10 (0x0a)	NVMe Feature 0Ah, Write Atomicity.
11 (0x0b)	NVMe Feature 0Bh, Asynchronous Event Configuration.
12 (0x0c)	NVMe Feature 0Ch, Autonomous Power State Transition.
128 (0x80)	NVMe Feature 80h, Software Progress Marker.
129 (0x81)	NVMe Feature 81h, Host Identifier.
130 (0x82)	NVMe Feature 82h, Reservation Notification Mask; not supported by Ultrastar NVMe firmware (controller or namespace).
131 (0x83)	NVMe Feature 83h, Reservation Persistence; not supported by Ultrastar NVMe firmware (controller or namespace).

Example

```
hdm get-feature --feature 7 --path /dev/nvme0
```

Sample Output

```
[/dev/nvme0]
  Device Type  = Ultrastar SN150
  Device Path  = /dev/nvme0
  UID          = 1C58STM0001A6758HUSPR3216AHP301
  Alias        = @nvme0
  Feature Id   = 7
  Feature Info =
    00000000  7f00 7f00                                      [....            ]

Results for get-feature: Operation succeeded.
```

get-info

The `get-info` command will retrieve the static properties of a device, such as the device type, device path, UID, model name, serial number, etc.

The set of properties shown in the `hdm get-info` *command may vary by device type.*

Synopsis

```
usage: hdm get-info             [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}

Retrieves properties from devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", “mini”, "csv", and "json". The default
                                is “mini”.
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output
                                of hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
```

Example

```
hdm get-info --path /dev/nvme0
```

NVMe Controller Sample Output

```
hdm get-info --path /dev/nvme0 --output-format mini

[/dev/nvme0]
  Device Type               = Ultrastar SN150
  Device Path               = /dev/nvme0
  UID                       = 1C58STM000198437HUSPR3232AHP301
  Alias                     = @nvme0
  Vendor Name               = HGST
  Model Name                = HUSPR3232AHP301
  Serial Number             = STM0001994D7
  Vendor ID                 = 1C58
  Device ID                 = 0003
  Firmware Version          = KMGNP698
  Default Capacity          = 3200631791616
  Capacity                  = 3200631791616
  Namespace Count           = 4
  Dual Port Support         = No
  MPIO Support              = No
  Namespace Sharing Support = No
  Encryption Support        = No
  PCI Domain Number         = 0000
  PCI Bus Number            = 01
  PCI Device Number         = 00
  PCI Function Number       = 0
  PCI Max Link Speed        = PCIe 3.0 (8GT/S)
  PCI Max Link Width        = x4
  PCI Current Link Speed    = PCIe 3.0 (8GT/S)
  PCI Current Link Width    = x4

Results for get-info: Operation succeeded.
```

NVMe Namespace Sample Output

```
hdm get-info --path /dev/nvme0n1 --output-format mini

[/dev/nvme0n1]
  Device Type               = NVMe Namespace
  Device Path               = /dev/nvme0n1
  UID                       = 000CCA0060077E80
  Alias                     = @nvmens0
  Parent Type               = Ultrastar SN150
  Parent Device Path        = /dev/nvme0
  Parent UID                = 1C58STM00019C205HUSPR3232AHP301
  Namespace ID              = 1
  Size                      = 10000003072
  Sector Count              = 19531256
  Sector Size               = 512
  Metadata Size             = 0
  DIF Level                 = None
  Private                   = Yes

Results for get-info: Operation succeeded.
```

SCSI Device Sample Output (managed by non-RAID HBA)

```
hdm get-info --path /dev/sda --output-format mini

[/dev/sda]
  Device Type               = Ultrastar SSD1600MR
  Device Path               = /dev/sda
  UID                       = 5000CCA02B022D0C
  Alias                     = @scsi0
  Vendor Name               = HGST
  Model Name                = HUSMR1616ASS200
  Serial Number             = 09V1631A
  Firmware Version          = SPGNA43D
  Default Capacity          = 1600321314816
  Capacity                  = 1600321314816
  Sector Count              = 3125627569
  Sector Size               = 512
  Metadata Size             = 0
  DIF Level                 = None
  Protection Interval       = 1
  MPIO Support              = Yes
  Encryption Support        = Yes
  Hardware Version          = PIKE BAY B.0

Results for get-info: Operation succeeded.
```

SCSI Device Sample Output (managed by MegaRAID RAID controller)

```
hdm get-info --alias @scsi1 --output-format mini

[@scsi2]
  Device Type               = Ultrastar SSD800MH-400
  UID                       = 5000CCA02B005870
  Alias                     = @scsi1
  Vendor Name               = HGST
  Model Name                = HUSMH8040ASS200
  Serial Number             = 2HV05WLA
  Firmware Version          = SMGNA274
  Default Capacity          = 400088457216
  Capacity                  = 400088457216
  Parent Type               = MegaRAID Controller
  RAID Controller ID        = 0
  RAID Device ID            = 20
  Sector Count              = 781422768
  Sector Size               = 512
  Metadata Size             = 0
  DIF Level                 = None
  Protection Interval       = 1
  MPIO Support              = Yes
  Encryption Support        = Yes
  Hardware Version          = PIKE BAY B.0

Results for get-info: Operation succeeded.
```

get-log-page

The `get-log-page` command will retrieve a log page as specified by the `--page` option. The `PAGE` value must either be an integer or the hexadecimal equivalent of the log page.

Synopsis

```
usage: hdm get-log-page        [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |--alias ALIAS}
                               --page PAGE

Retrieves log pages from devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --page PAGE                   Identifier of the page. This value must be either an
                                integer (e.g. 1) or a prefixed hexadecimal number
                                (e.g. 0xC1).
```

Options

`--page` Value	Description
`1 (0x01)`	NVMe Log Page 01h, Error Information; supported for Ultrastar NVMe controllers only.
`2 (0x02)`	NVMe Log Page 02h, SMART / Health Information. This is a raw dump of Log Page 2 (see the *Sample Output*); however the same information will appear in a detailed, parsed format when using the `get-smart` command.
`3 (0x03)`	NVMe Log Page 03h, Firmware Information; supported for Ultrastar NVMe controllers only. This is a raw dump of Log Page 3 (see the *Sample Output*); however the same information will appear in a detailed, parsed format when using the `get-info` command.
`193 (0xC1)`	NVMe Log Page C1h, Vendor-Unique Log Page C1h; supported for Ultrastar NVMe controllers only. This is a raw dump of Log Page C1h (0xC1) (see the *Sample Output*).

Example

```
hdm get-log-page --page 0x02 --path /dev/nvme0
```

OR

```
hdm get-log-page --page 2 --path /dev/nvme0
```

Sample Output

```
[/dev/nvme0]
  Device Type = Ultrastar SN150
  Device Path = /dev/nvme0
  UID         = 1C58STM00019C205HUSPR3232AHP301
  Alias       = @nvme0
  Page ID     = 2
  Page Data   =
    00000000  0032 0164 0a00 0000  0000 0000 0000 0000  [.2.d............]
    00000010  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000020  9d89 8000 0000 0000  0000 0000 0000 0000  [................]
    00000030  c3fa 2501 0000 0000  0000 0000 0000 0000  [..%.............]
    00000040  7732 b809 0000 0000  0000 0000 0000 0000  [w2..............]
    00000050  3591 e810 0000 0000  0000 0000 0000 0000  [5...............]
    00000060  9100 0000 0000 0000  0000 0000 0000 0000  [................]
    00000070  1d00 0000 0000 0000  0000 0000 0000 0000  [................]
    00000080  b500 0000 0000 0000  0000 0000 0000 0000  [................]
    00000090  1800 0000 0000 0000  0000 0000 0000 0000  [................]
    000000A0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000000B0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000000C0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000000D0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000000E0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000000F0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000100  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000110  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000120  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000130  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000140  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000150  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000160  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000170  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000180  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    00000190  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000001A0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000001B0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000001C0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000001D0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000001E0  0000 0000 0000 0000  0000 0000 0000 0000  [................]
    000001F0  0000 0000 0000 0000  0000 0000 0000 0000  [................]

Results for get-log-page: Operation succeeded.
```

get-smart

The `get-smart` command will retrieve the S.M.A.R.T. statistics and threshold values from a target device. As shown in the *Sample Output*, the command will return error rates, read/write rates, critical temperature data, and the current properties of the NAND media.

The set of properties shown in the `hdm get-smart` *command may vary by device type.*

Synopsis

```
usage: hdm get-smart           [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}

Retrieves SMART statistics and thresholds from devices.

OPTIONS:
  -o, --output-format FORMAT   Output format for the command. Valid output formats
                               are "text", "mini", "csv", and "json". The default is
                               "mini".
  -c, --config PATH            Path to the HDM configuration file.
  -u, --uid UID                Device unique identifier.
  -p, --path PATH              Platform path of the device as shown in the output of
                               hdm scan.
  -a, --alias ALIAS            User supplied name for the device.
```

Example

```
hdm get-smart --path /dev/nvme0
```

NVMe Controller Sample Output

```
hdm get-smart --path /dev/nvme0 --output-format mini

[/dev/nvme0]
  Device Type                                          = Ultrastar SN150
  Device Path                                          = /dev/nvme0
  UID                                                  = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                                = @nvme0
  Data Units Read                                      = 260567436
  Data Units Written                                   = 314947558
  Host Read Commands                                   = 18553931035
  Host Write Commands                                  = 17450482756
  Available Spare                                      = 100
  Available Spare Threshold                            = 10
  Available Spare Critical Warning                     = false
  Percentage Used                                      = 0
  Temperature                                          = 40
  Temperature Threshold                                = 90
  Temperature Critical Warning                         = false
  Device Reliability Critical Warning                  = false
  Media Read Only Mode Critical Warning                = false
  Volatile Memory Backup Device Failure Critical Warning = false
  Controller Busy Time                                 = 7513
  Power Cycles                                         = 45
  Power On Hours                                       = 3749
  Unsafe Shutdowns                                     = 19
  Media And Data Integrity Errors                      = 0
  Error Information Log Entries                        = 1
  SMART Status                                         = Normal

Results for get-smart: Operation succeeded.
```

NVMe Namespace Sample Output

```
hdm get-smart --path /dev/nvme0n1 --output-format mini

[/dev/nvme0n1]
  Device Type                                          = NVMe Namespace
  Device Path                                          = /dev/nvme0n1
  UID                                                  = 000CCA00600FFFFF
  Alias                                                = @nvmens0
  Data Units Read                                      = 260567436
  Data Units Written                                   = 314947558
  Host Read Commands                                   = 18553931036
  Host Write Commands                                  = 17450482757
  Available Spare                                      = 100
  Available Spare Threshold                            = 10
  Available Spare Critical Warning                     = false
  Percentage Used                                      = 0
  Temperature                                          = 40
  Temperature Critical Warning                         = false
  Device Reliability Critical Warning                  = false
  Media Read Only Mode Critical Warning                = false
  Volatile Memory Backup Device Failure Critical Warning = false
  Controller Busy Time                                 = 7513
  Power Cycles                                         = 45
  Power On Hours                                       = 3749
  Unsafe Shutdowns                                     = 19
  Media And Data Integrity Errors                      = 0
  Error Information Log Entries                        = 1
  SMART Status                                         = Normal

Results for get-smart: Operation succeeded.
```

get-state

The `get-state` command will retrieve the current state information of the target device, including the current encryption mode and remaining life.

The set of properties shown in the `hdm get-state` *command may vary by device type.*

Synopsis

```
usage: hdm get-state          [-o, --output-format FORMAT] [-c, --config PATH]
                              {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}

Retrieves state information from devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
```

Examples

```
hdm get-state --path /dev/nvme0
```

NVMe Controller Sample Output

```
hdm get-state --path /dev/nvme0 --output-format mini

[/dev/nvme0]
  Device Type           = Ultrastar SN150
  Device Path           = /dev/nvme0
  UID                   = 1C58STM00019C20AHUSPR3232AHP301
  Alias                 = @nvme0
  Device Status         = Ready
  Life Gauge            = 100
  Power Consumption     = 7.782563
  Dual Port Mode        = Single Port
  Encryption Mode       = No Encryption

Results for get-state: Operation succeeded.
```

NVMe Namespace Sample Output

```
hdm get-state --path /dev/nvme0n1 --output-format mini

[/dev/nvme0n1]
  Device Type           = NVMe Namespace
  Device Path           = /dev/nvme0n1
  UID                   = 000CCA00600FFFFF
  Alias                 = @nvmens0
  Life Gauge            = 100

Results for get-state: Operation succeeded.
```

SCSI Device Sample Output

```
hdm get-state --path /dev/sda --output-format mini

[/dev/sda]
  Device Type           = Ultrastar SSD1600MR
  Device Path           = /dev/sda
  UID                   = 5000CCA02B022D0C
  Alias                 = @scsi0
  Device Status         = Ready
  Life Gauge            = 100
  Power Consumption     = 9.000000
  Dual Port Mode        = Single Port
  Encryption Mode       = Full disk

Results for get-state: Operation succeeded.
```

Device Status Values

The values that are shown in the Device Status property and the corresponding enumerations, which are shown if using the JSON output format, are explained in the table below:

Device Status	Enumeration
Ready	3000
Secure Purge in Progress	3001
Secure Purge Failed: Power Cycle Required	3002
Secure Purge Interrupted: Retry Required	3003
Format in Progress	3004
Format Failed: Retry Required	3005
Sanitize in Progress	3006
Sanitize Failed: Retry Required	3007

get-statistics

The `get-statistics` command retrieves statistical data according to the specified `-category`. The set of valid intervals as specified by the `--interval` value may vary by device type and only apply to the `performance` category; the device stores various statistical fields that are used to review and analyze the device performance. There are fifteen sets of performance statistics (1-15) that can be collected at various intervals. If an `--interval` is not specified for the `performance` category, then the value that corresponds to the lifetime statistics of the device is used as the default. The command is only supported for physical devices.

Synopsis

```
usage: hdm get-statistics       [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [-i, --interval INTERVAL] [--category CATEGORY]

Retrieves statistics from devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  -i, --interval INTERVAL       Interval to be used for this operation.
  --category CATEGORY           Category to be used for this operation.
```

Options

Option	Value	Description
-i, --interval INTERVAL	1	Most recent five (5) minute accumulated set.
	2 – 12	Previous five (5) minute accumulated sets.
	13	The accumulated total of sets 1 through 12 that contain the previous hour of accumulated statistics.
	14	The statistical set accumulated since power-up.
	15	The statistical set accumulated during the entire lifetime of the device.
--category CATEGORY	performance	Performance statistics; the properties and values collected by this category will contain, if applicable, the properties and values defined in the Performance Log Page 37h (SCSI).
	power	Power statistics; the properties and values collected by this category will contain power consumption statistics. The power consumption value is in mW.
	queues	Queue statistics; the properties and values collected by this category are applicable to queues that are used by the device driver or firmware.
	commands	Command statistics; the properties and values collected by this category will contain read, write, verify and erase command execution statistics.
	temperature	Temperature statistics; the properties and values collected by this category will contain the recorded maximum, minimum, reference and sensor (main, inlet, daughter board) temperature data.
	miscellaneous	Miscellaneous statistics; the properties and values collected by this category are not specific to any of the previous categories.

Example

```
hdm get-statistics --interval 14 --category performance --path /dev/nvme0
```

Sample Output --category performance

```
[/dev/nvme0]
  Device Type                                = Ultrastar SN150
  Device Path                                = /dev/nvme0
  UID                                        = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                      = @nvme0
  Category                                   = Performance
  Interval                                   = 14
  Host Read Commands                         = 93613753
  Host Read Blocks                           = 3027373912
  Average Read Size                          = 32.338987
  Host Read Cache Hit Commands               = 7161
  Host Read Cache Hit Percentage             = 0
  Host Read Cache Hit Blocks                 = 61328
  Average Read Cache Hit Size                = 8.564167
  Host Read Commands Stalled                 = 0
  Host Read Commands Stalled Percentage      = 0
  Host Write Commands                        = 6694738310
  Host Write Blocks                          = 164139680392
  Average Write Size                         = 24.517714
  Host Write Odd Start Commands              = 0
  Host Write Odd Start Commands Percentage   = 0
  Host Write Odd End Commands                = 0
  Host Write Odd End Commands Percentage     = 0
  Host Write Commands Stalled                = 11613529
  Host Write Commands Stalled Percentage     = 0
  NAND Read Commands                         = 12649124192
  NAND Read Blocks                           = 404771974144
  Average NAND Read Size                     = 32.000000
  NAND Write Commands                        = 9347350484
  NAND Write Blocks                          = 299115215488
  Average NAND Write Size                    = 32.000000
  NAND Read Before Write                     = 0
  Read Error Rate                            = 1.692628

Results for get-statistics: Operation succeeded.
```

Sample Output --category power

```
[/dev/nvme0]
  Device Type                                = Ultrastar SN150
  Device Path                                = /dev/nvme0
  UID                                        = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                      = @nvme0
  Category                                   = Power
  Processor Power Consumption                = 6613
  NAND Power Consumption                     = 1187

Results for get-statistics: Operation succeeded.
```

Sample Output --category queues

```
[/dev/nvme0]
  Device Type                                = Ultrastar SN150
  Device Path                                = /dev/nvme0
  UID                                        = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                      = @nvme0
  Category                                   = Queues
  Number of I/O Completion Queues Allocated  = 128
  Number of I/O Submission Queues Allocated  = 128

Results for get-statistics: Operation succeeded.
```

Sample Output `--category commands`

```
[/dev/nvme0]
  Device Type                                = Ultrastar SN150
  Device Path                                = /dev/nvme0
  UID                                        = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                      = @nvme0
  Category                                   = Commands
  Total Re-Writes                            = 0
  Total Write Bytes Processed                = 165052617228288
  Total Uncorrected Write Errors             = 0
  Flash Write Commands                       = 2336002848
  Read Errors Corrected Without Delays       = 11529379176
  Read Errors Corrected With Possible Delays = 0
  Total Re-Reads                             = 0
  Total Read Errors Corrected                = 11529379176
  Total Read Bytes Processed                 = 0
  Total Uncorrected Read Errors              = 0
  Flash Read Commands                        = 6811526551
  Total XOR Recovered                        = 0
  Total Corrected Bit Count                  = 11529379176
  Total Uncorrected Erase Errors             = 0
  Flash Erase Commands                       = 8922807
  Manufacturers Defect Count                 = 0
  Grown Defect Count                         = 0
  Maximum Erase Count of User Data           = 72
  Maximum Erase Count of System Data         = 0

Results for get-statistics: Operation succeeded.
```

Sample Output `--category temperature`

```
[/dev/nvme0]
  Device Type                                = Ultrastar SN150
  Device Path                                = /dev/nvme0
  UID                                        = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                      = @nvme0
  Category                                   = Temperature
  Maximum Temperature                        = 77
  Minimum Temperature                        = 24
  Total Time Over Reference Temperature      = 0
  Power On Time When At Max Temperature      = 429
  Power On Time When At Min Temperature      = 90044
  Main Sensor Temperature                    = 40
  Inlet Sensor Temperature                   = 31
  Daughterboard 1 Sensor Temperature         = 31
  Daughterboard 2 Sensor Temperature         = 32


Results for get-statistics: Operation succeeded.
```

Sample Output `--category miscellaneous`

```
[/dev/nvme0]
  Device Type                                = Ultrastar SN150
  Device Path                                = /dev/nvme0
  UID                                        = 1C58STM00019C20AHUSPR3232AHP301
  Alias                                      = @nvme0
  Category                                   = Miscellaneous
  Firmware Download Number                   = 5
  PFA Counter                                = 0

Results for get-statistics: Operation succeeded.
```

locate

The `locate` command will either activate or deactivate the beacon LED on the device(s), allowing the user to find the physical device(s) installed in the host system. The user must specify the *device-reference* for the target device. The `--enable` option is the default action that activates the beacon if it is inactive. The `--disable` option will deactivate the beacon if it is currently active. The command is only valid for physical devices.

Synopsis

```
usage: hdm locate              [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                               [--disable] [--enable]

Enables or disables the LED beacon for physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", “mini”, "csv", and "json". The default is
                                “mini”.
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --disable                     Executes the disable action.
  --enable                      Executes the enable action.
```

Examples

To enable the beacon LED for a device:

```
hdm locate --enable --path /dev/nvme0
```

To disable the beacon LED for a device:

```
hdm locate --disable --path /dev/nvme0
```

manage-firmware

The `manage-firmware` command will list the firmware version(s) present on the device; the user can also perform interrelated firmware management tasks. The command is only valid for physical devices.

Synopsis

```
usage: hdm manage-firmware      [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [--list] [--load] [-f, --file FILE] [--activate]
                                [--slot NUMBER] [--reset]

Displays and updates firmware on physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command.  Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --list                        Displays information from the operation.
  --load                        Loads a firmware image onto the specified devices.
  -f, --file FILE               File name or path.
  --activate                    Activates a firmware image.
  --slot NUMBER                 The slot number.
  --reset                       Resets the specified devices.
```

Options

Option	Description
`--list`	The `--list` switch will perform the default action and display information about the firmware image(s) if no other options are specified.
`--load`	The `--load` switch will install the firmware, as specified by the `--file` parameter, onto the target device(s); this option also requires the use of the `--slot` switch.
`-f, --file FILE`	The `FILE` value is the path and filename of the image. The `-f, --file FILE` option is required with the `--load` switch.
`--activate`	The `--activate` switch is used to enable the firmware on the device as per the required `--slot NUMBER` option. This switch may be used along with the `--load` switch to immediately activate the firmware upon the successful load. This option is required with the `--load` flag for Ultrastar SAS SSDs.
`--slot NUMBER`	The `--slot` switch specifies the slot number to target or devices that support multiple firmware images. This switch is required with the `--load` and `--activate` switches. The user should be aware that `Slot 1` is read-only. This option is required for Ultrastar NVMe devices. **Note**: This option is not valid for Ultrastar SAS SSDs.
`--reset`	The `--reset` switch is used to perform a bus reset on the slot (`--slot NUMBER`) and indicates that the `--activate` switch should automatically perform a controller/bus reset after completing the activation. The switch is only valid when used with the `--activate` switch. **Note**: This option is not valid for Ultrastar SAS SSDs.

Examples

To list information about the firmware image(s):

```
hdm manage-firmware --path /dev/nvme0 --output-format mini
```

To load a firmware image onto an NVMe controller:

```
hdm manage-firmware --load --file /firmware/filename.bin --slot 2 --path /dev/nvme0
```

To activate a firmware image on an NVMe controller:

```
hdm manage-firmware --activate --slot 2 --path /dev/nvme0
```

To activate a firmware image and perform a controller / bus reset on an NVMe controller:

```
hdm manage-firmware --activate --reset --slot 2 --path /dev/nvme0
```

To load a firmware image onto a SCSI device:

```
hdm manage-firmware --load --file /firmware/filename.bin --path /dev/sda
```

NVMe Controller Sample Output

```
hdm manage-firmware --path /dev/nvme0 --output-format mini
 [/dev/nvme0]
  Device Type              = Ultrastar SN150
  Device Path              = /dev/nvme0
  UID                      = 1C58STM00019EE74HUSPR1616AHP301
  Alias                    = ns0
  Running Firmware Version = KMGNP6D0 (Loaded from Slot 3)
  Slot 1 (Read-only)       = KMGNP6D0
  Slot 2                   = KMGNP6D7
  Slot 3                   = KMGNP690 (Next Running Firmware Slot)
  Slot 4                   = KMGNP69E

Results for manage-firmware: Operation succeeded.
```

Terminology

The HDM manage-firmware listing for NVMe controllers contains terminology that is explained in the following table:

Term	Description
`(Read-only)`	The firmware slot is read-only. The firmware image in this slot cannot be updated.
`(Loaded from Slot x)`	The firmware slot from which the currently running firmware image was loaded.
`(Next Running Firmware Slot)`	The firmware slot containing the firmware image that will become the running firmware following the next device reset or system power-cycle.

The first firmware slot on an NVMe controller is read-only as defined by the NVMe specification. The HDM manage-firmware list command indicates this by showing the `(Read-only)` *designator next to* `Slot 1`.

manage-namespaces

The `manage-namespaces` command allows the user to perform interrelated namespace management tasks on NVMe controllers. The user can target multiple namespaces by specifying multiple `--id` *`namespace-id`* options in the command syntax; the *device-reference* must refer to a physical device.

Synopsis

```
usage: hdm manage-namespaces   [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                               [--list] [--create] [--attach] [--detach] [--delete]
                               [--size SIZE] [--id ID] [--force]

Displays and manages namespaces on a capable physical device.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command.  Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --list                        Displays information from the operation.
  --create                      Executes the create action.
  --attach                      Executes the attach action.
  --detach                      Executes the detach action.
  --delete                      Executes the delete action.
  --size SIZE                   Size in gigabytes.
  --id ID                       The identifier to be used for this operation.
  --force                       Bypasses the confirmation messages.
```

Options

Option	Description
`--list`	The `--list` option will present information regarding the namespace(s) currently present on the device(s). This is the default action if none of the remaining options are specified.
`--create`	This option will create a new namespace or namespaces on the target device; the `--size` option is required. The ID integer can be specified using the `--id` option. If no ID integer is specified, then a single namespace will be created having the lowest available namespace identifier. The namespace will be formatted with the default LBA format and DIF level (for Ultrastar NVMe controllers this is 512-bytes sector size, 0 metadata size and a DIF level of 0 (zero).
`--attach`	This option will attach (if possible) the namespace specified by the `--id` option integer.
`--detach`	This option will detach the namespace specified by the `--id` option integer.
`--delete`	This option will delete the namespace specified by the `--id` option integer. A permanent namespace cannot be deleted. The namespace is automatically detached before deletion.
`--size SIZE`	This option will set, in gigabytes, the size of the namespace. This option is only valid when used with the `--create` option.
`--id ID`	The `--id` option allows the user to target an existing namespace or multiple namespaces using the actual ID integer value created by the user, e.g., `--id` *1*., `--id` *2*, etc. This option is required when using the `--create`, `--attach`, `--delete` or `--detach` options.

`--force`	The `--force` option will execute the operation without any user confirmation. The user will be prompted to confirm the `--create`, `--attach`, `--delete` or `--detach` operations if the `--force` option is not specified.

Examples

To list all the current namespaces on an NVMe controller:

```
hdm manage-namespaces --list --path /dev/nvme0 --output-format mini
```

OR

```
hdm manage-namespaces --path /dev/nvme0 --output-format mini

[/dev/nvme0]
  Device Type  = Ultrastar SN150
  Device Path  = /dev/nvme0
  UID          = 1C58STM00019EE74HUSPR1616AHP301
  Alias        = @nvme0
  Namespace 1  = 600 GB
  Namespace 3  = Detached
  Namespace 4  = 100 GB

Results for manage-namespaces: Operation succeeded.
```

To create a new NVMe namespace:

```
hdm manage-namespaces --create --id 3 --size 1600 --path /dev/nvme0
```

To delete an NVMe namespace:

```
hdm manage-namespaces --delete --id 3 --path /dev/nvme0
```

To attach an existing detached NVMe namespace:

```
hdm manage-namespaces --attach --id 3 --path /dev/nvme0
```

To detach an NVMe namespace:

```
hdm manage-namespaces –detach --id 3 --path /dev/nvme0
```

manage-power

The `manage-power` command will list the available power states for a device and allows the user to apply a specific power state to a device. The *device-reference* must be a physical device, as namespaces are not valid.

***Ultrastar NVMe PCIe Devices Only**: Issuing some HDM commands, or performing a host reboot/power cycle after using the `manage-power` command to change the power state will revert the power state to its previous value. The user should therefore perform and complete all other device tasks before attempting to alter the power state.*

Synopsis

```
usage: hdm manage-power        [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                               [--list] [--change] [--state STATE]

Displays and controls power states of physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command.  Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --list                        Displays information from the option.
  --change                      Executes the change action.
  --state STATE                 The state to be used for this operation.
```

Options

Option	Description
`--list`	The `--list` option will present information regarding the power states that are available for the device. This is the default action if none of the remaining options are specified.
`--change`	This option will change and apply the power state on the targeted device.
`--state STATE`	This option specifies the power state that should be applied to the targeted device; the option is only valid when used with the `--change` option. The valid values for the power-state are those listed with the prefix of "`Power State`".

Examples

To list the available power states for a device:

```
hdm manage-power --list --path /dev/nvme0 --output-format mini
```

OR

```
hdm manage-power --path /dev/nvme0 --output-format mini

[/dev/nvme0]
  Device Type   = Ultrastar SN150
  Device Path   = /dev/nvme0
  UID           = 1C58STM00019EE74HUSPR1616AHP301
  Alias         = @nvme0
  Power State 1 = 25 W (Active)
  Power State 2 = 20 W
  Power State 3 = 15 W
  Power State 4 = 10 W
  Power State 5 = 10 W [no I/O]

Results for manage-power: Operation succeeded.
```

To apply a valid power state to a device:

```
hdm manage-power --path /dev/nvme0 --change --state 3
```

monitor

The `monitor` command will retrieve device information, state, SMART and statistics for all supported devices in the system, or for a specific device according to the *device-reference*. The data is retrieved and refreshed by the value specified by the `--interval` option. The command will continue to run until it is killed by the user or until the number of `--count` iterations has been completed.

The user should be aware that while there are no minimum or maximum `--interval` or `--count` values, no extreme edge cases have been tested. The user is therefore advised to use discretion when specifying the values for these options.

Synopsis

```
usage: hdm monitor              [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [--interval INTERVAL] [--count NUM]

Continuously retrieves device information at a specified interval.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command.  Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  -i, --interval INTERVAL       Interval to be used for this operation.
  --count NUM                   Number of times to execute the operation.
```

Options

Option	Description
`-i, --interval INTERVAL`	The `INTERVAL` value is expressed in seconds, e.g., `-i 30` or `--interval 30` would specify an interval of thirty (30) seconds between the logging of each event.
`--count NUM`	The `NUM` value expresses the number of iterations to complete before ending the process.

Examples

To monitor a specific device at intervals of 30 seconds:

```
hdm monitor --interval 30 --path /dev/nvme0
```

To monitor all HDM supported devices on the system at an interval of 30 seconds:

```
hdm monitor --interval 30
```

To monitor a specific device at intervals of 30 seconds but limit the polling to 10 iterations:

```
hdm monitor --interval 30 --count 10 --path /dev/nvme0
```

To monitor all HDM supported devices on the system at an interval of 30 seconds but limit the polling to 10 iterations:

```
hdm monitor --interval 30 --count 10
```

prepare-for-removal

The `prepare-for-removal` command will instruct the operating system to stop issuing I/O to the target device; the device may then be safely removed from the system. The user must reboot the host or power-cycle the system before the OS will detect the device. This command is only valid for physical devices.

The user must physically remove the device from the system and then reinstall the device for the operating system to detect it.

The prepare-for-removal command is not supported for SAS devices managed by RAID controllers.

Synopsis

```
usage: hdm prepare-for-removal   [-o, --output-format FORMAT] [-c, --config PATH]
                                 {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}

Prepares physical devices for safe removal from the system.

OPTIONS:
  -o, --output-format FORMAT     Output format for the command.  Valid output formats
                                 are "text", "mini", "csv", and "json". The default
                                 is "mini".
  -c, --config PATH              Path to the HDM configuration file.
  -u, --uid UID                  Device unique identifier.
  -p, --path PATH                Platform path of the device as shown in the output
                                 of hdm scan.
  -a, --alias ALIAS              User supplied name for the device.
```

Example

```
hdm prepare-for-removal --path /dev/nvme0
```

reset-to-defaults

The `reset-to-defaults` command will restore the targeted device(s) to factory default settings. Neither the firmware nor the statistical reporting is affected. The command is only valid for physical devices. The user will be prompted to confirm unless the `--force` option is specified.

The command is destructive. The user should backup all data currently stored on the device. All user data will be destroyed.

Synopsis

```
usage: hdm reset-to-defaults    [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [--force]

Resets physical devices to factory default configuration.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command.  Valid output formats
                                are "text", "mini", "csv", and "json". The default
                                is "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output
                                of hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --force                       Bypasses all confirmation messages.
```

Example

```
hdm reset-to-defaults --path /dev/nvme0
```

resize

The `resize` command will set the user capacity of a device according to the value of the `--size` parameter. The valid value for the `--size` parameter is dependent upon the device model and must be expressed in gigabytes (GB). For example, if the device is 2.0TB (2000GB), but the user wants to resize it to 1.6TB (1600GB), the expression would be `--size 1600`. The command is only valid for physical devices. The user will be prompted to confirm unless the `--force` option is specified.

The command is destructive. The user should backup all data currently stored on the device. All user data will be destroyed.

Synopsis

```
usage: hdm resize              [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                               [--size SIZE] [--force]

Changes the user capacity of physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --size SIZE                   Size in gigabytes.
  --force                       Bypasses the confirmation messages.
```

Note: The value for the `-size SIZE` must fall within the following ranges for Ultrastar NVMe controllers as listed in the following table:

Capacity	Minimum Resize Value (GB)	DW/D	Maximum Resize Value (GB)	DW/D
800GB	560	7.4	956	1.1
1.6TB	1120	7.5	1910	1.2
3.2TB	2240	7.5	3820	1.2
Note: DW/D = "Data Writes / Day" or "Data Writes per Day".				

Example

```
hdm resize --size 1600 --path /dev/nvme0
```

sanitize

The `sanitize` command allows the user to perform block, overwrite and cryptographic sanitization of physical devices.

The command is destructive. The user should backup all data currently stored on the device. All user data will be destroyed.

√ *The sanitize operation is prohibited by most SAS Host Bus Adapters and RAID controllers on Windows.*

√ *The* `sanitize` *command can run on Ultrastar SAS SSD devices for several hours before it completes. The* `get-state` *command (specifically, the* `Device Status` *output) can be used to determine whether the sanitize operation has been completed.*

√ *The user can use the* `--status` *option to display, as a running percentage, the completion status of the operation.*

Synopsis

```
usage: hdm sanitize             [-o, --output-format FORMAT] [-c, --config PATH]
                                {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                [--type TYPE] [--invert] [--count NUM]
                                [--pattern PATTERN] [-f, --file FILE]
                                [-w, --wait] [--status] [--force]

Performs a sanitize operation to erase all user data on physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command.  Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --type TYPE                   Type of this operation to perform.
  --invert                      Whether to invert the pattern as part of this
                                operation.
  --count NUM                   Number of times to execute the operation.
  --pattern PATTERN             Pattern that should be used for this operation.
  -f, --file FILE               File name or path.
  -w, --wait                    Blocks until this operation completes.
  --status                      Displays status of this operation.
  --force                       Bypasses all confirmation messages.
```

Options

--type TYPE

The TYPE specifies the type of sanitization operation. The valid values for TYPE are block, cryptographic and overwrite.

block	The block option will write a predetermined, device-specific pattern to all blocks containing user data.
cryptographic	The cryptographic option will delete the encryption key previously used to encrypt the user data.
overwrite	The overwrite option will write the pattern as specified by the --pattern option to all blocks containing user data. **Note**: The Ultrastar SAS SSDs do not currently support the overwrite sanitize type.

--pattern PATTERN

The PATTERN value specifies the pattern that will be used for the overwrite operation. This option is only valid for the overwrite operation; the option is invalid if the –f, --file option is specified. If neither the --pattern nor –f, -file options are specified, the default value used for the pattern is 0xFFFFFFFF.

-f, --file FILE

The FILE value specifies the path and filename of a file containing a data pattern that will written to all user data blocks during an overwrite operation. The option is only valid for the overwrite operation; the option is invalid if the --pattern option is specified.

--invert

The --invert flag indicates that the --pattern option or that the value within the filename as specified by the –f, --file option should be inverted each time it is written. This option is only valid for the overwrite operation; the -f, --file or --pattern options must also be specified. If the --invert flag is not used, then the data pattern will not be inverted (default behavior).

--count NUM

The NUM value indicates the number of times all user data blocks will be written during the overwrite operation, e.g., if --count 10, then the user data blocks will be overwritten a total of ten (10) times during the operation. The option is only valid for the overwrite operation; the -f, --file or --pattern options must also be specified. If the --count NUM option is not used, then the data blocks are written only once.

-w, --wait

The -wait | -w flag will cause the command to block any other HDM command line input from the console until the sanitize operation completes.

--status

The --status option will report, as a running percentage, the status of the sanitize operation.

--force

The --force option will override the confirmation prompt and execute the command.

Examples

Ultrastar SAS SSDs do not support the `overwrite` sanitize type.

Block Sanitization

To perform a `block` sanitize operation:

```
hdm sanitize --type block --path /dev/sda
```

Cryptographic Sanitization

To perform a `cryptographic` sanitize operation:

```
hdm sanitize --type cryptographic --path /dev/sda
```

Overwrite Sanitization

To perform a `overwrite` sanitize operation:

```
hdm sanitize --type overwrite --path /dev/sda
```

Using the `--status` Option

To display a running status of any sanitize operation, add the `--status` option:

```
hdm sanitize --type overwrite --status --path /dev/sda
```

Using the `--wait` Option

To prevent any HDM command input during any sanitize operation, add the `--wait` option:

```
hdm sanitize --type overwrite --wait --path /dev/sda
```

Using the `--pattern` and `--file` Options

```
hdm sanitize --type overwrite --pattern --file <path pattern file name>
--path /dev/sda
```

Using the `--invert` Option

```
hdm sanitize --type overwrite --pattern --file <path/patternfile>
--invert --path /dev/sda
```

Using the `--count` Option

```
hdm sanitize --type overwrite –pattern --file <path/patternfile>
--count 10 --path /dev/sda
```

scan

The `scan` command will enumerate all HDM supported storage devices installed in the host system.

The `hdm scan` *command will automatically assign aliases to NVMe controllers and namespaces. See* `alias` *for more information.*

The set of properties shown in the `hdm scan` *command may vary by device type.*

Synopsis

```
usage: hdm scan                [-o, --output-format FORMAT] [-c, --config PATH]

Lists the storage devices attached to this host.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
```

Example

```
hdm scan
```

Sample Output

```
[1C58STM00019C205HUSPR3232AHP301]
  Device Type         = Ultrastar SN150
  Device Path         = /dev/nvme0
  UID                 = 1C58STM00019C205HUSPR3232AHP301
  Alias               = @nvme0
  Vendor Name         = HGST
  Model Name          = HUSPR3232AHP301
  Vendor Id           = 1C58
  Device Id           = 0003

[000CCA0060077E80]
  Device Type         = NVMe Namespace
  Device Path         = /dev/nvme0n1
  UID                 = 000CCA0060077E80
  Alias               = @nvmens0
  Parent Type         = Ultrastar SN150
  Parent Device Path  = /dev/nvme0
  Parent UID          = 1C58STM00019C205HUSPR3232AHP301
  Namespace ID        = 1

[5000CCA02B022D0C]
  Device Type         = Ultrastar SSD1600MR
  Device Path         = /dev/sda
  UID                 = 5000CCA02B022D0C
  Alias               = @scsi0
  Vendor Name         = HGST
  Model Name          = HUSPR1616ASS200

[5000CCA02B005870]
  Device Type         = Ultrastar SSD800MH-400
  UID                 = 5000CCA02B005870
  Alias               = @scsi1
  Vendor Name         = HGST
  Model Name          = HUSMH8040ASS200
  Parent Type         = MegaRAID Controller
  RAID Controller ID  = 0
  RAID Device ID      = 20

Results for scan: Operation succeeded on 4 of 4 devices.
```

secure-erase

The `secure-erase` command will perform a secure erase of the target device using a specified erase level. The command applies the erase to the entire device and cannot target a specific namespace or partition. The operation will result in a reset of the LBA format to the default for the targeted device (for Ultrastar NVMe devices, the default LBA format is 512-byte sectors with no metadata or T10 DIF Protection). The user will be prompted to confirm unless the `--force` option is specified.

The command is destructive. The user should backup all data currently stored on the device. All user data will be destroyed.

Synopsis

```
usage: hdm secure-erase        [-o, --output-format FORMAT] [-c, --config PATH]
                               {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                               [--type TYPE] [--force]

Securely erases all user data from physical devices.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", "mini", "csv", and "json". The default is
                                "mini".
  -c, --config PATH             Path to the HDM configuration file.
  -u, --uid UID                 Device unique identifier.
  -p, --path PATH               Platform path of the device as shown in the output of
                                hdm scan.
  -a, --alias ALIAS             User supplied name for the device.
  --type TYPE                   Type of this operation to perform.
  --force                       Bypass all confirmation messages.
```

Options

Option	Value	Description
`--type TYPE`	`user`	If `TYPE = user`, then the command erases all user content present in the NVM subsystem.
	`cryptographic`	If `TYPE = cryptographic`, then the command will erase all content present in the NVM subsystem by deleting the encryption key with which the user data was previously encrypted. **Note**: The user should be aware that not all devices support the `cryptographic` option.

Examples

To perform a `user` data secure erase:

```
hdm secure-erase --type user --path /dev/nvme0
```

To perform a `cryptographic` secure-erase:

```
hdm secure-erase --type cryptographic --path /dev/nvme0
```

secure-purge

The `secure-purge` command will perform a purge on a physical device. Whereas *`secure-erase`* only erases the user data areas, `secure-purge` applies a sanitize/erase fill to the entire device and does not target a specific namespace or partition. The process will erase all the data (user and system areas) and then fills (overwrites) each addressable block of memory with a predetermined pattern. The user will be prompted to confirm unless the `--force` option is specified.

The command is destructive. The user should backup all data currently stored on the device. All user data will be destroyed.

√ *The user can use the `--status` option to display, as a running percentage, the completion status of the operation.*

√ *The `secure-purge` command can run on SN100/SN150 devices for several hours before it completes. The `get-state` command (specifically, the `Device Status` output) can be used to determine whether the purge operation has been completed.*

Synopsis

```
usage: hdm secure-purge          [-o, --output-format FORMAT] [-c, --config PATH]
                                 {-p, --path PATH |-u, --uid UID |-a, --alias ALIAS}
                                 [-w, --wait] [--status] [--force]

Performs a secure purge on physical devices.

OPTIONS:
  -o, --output-format FORMAT     Output format for the command.  Valid output formats
                                 are "text", "mini", "csv", and "json". The default
                                 is "mini".
  -c, --config PATH              Path to the HDM configuration file.
  -u, --uid UID                  Device unique identifier.
  -p, --path PATH                Platform path of the device as shown in the output
                                 of hdm scan.
  -a, --alias ALIAS              User supplied name for the device.
  -w, --wait                     Blocks until this operations completes.
  --status                       Displays status of this operation.
  --force                        Bypasses all confirmation messages.
```

Options

Option	Description
`-w, --wait`	The `--wait` or `-w` flag will cause the command to block any other HDM command line input from the console until operation completes.
`--status`	The `--status` option will report, as a running percentage, the completion status of the operation.
`--force`	The `--force` option will override the confirmation prompt and execute the command.

Examples

To perform a `secure purge` of the device with a confirmation prompt:

```
hdm secure-purge --path /dev/nvme0
```

Using the `--status` Option

To display a running status of any secure purge operation, add the `--status` option:

```
hdm secure-purge --status --path /dev/nvme0
```

Using the `--wait` Option

To prevent any HDM command input during any sanitize operation, add the `--wait` option:

```
hdm secure-purge --wait --path /dev/nvme0
```

version

The `version` command will obtain and display the current build and version of the software. The output consists of a string that displays the current software version.

Synopsis

```
usage: hdm version             [-o, --output-format FORMAT] [-c, --config PATH]

Display HDM build and version information.

OPTIONS:
  -o, --output-format FORMAT    Output format for the command. Valid output formats
                                are "text", “mini”, "csv", and "json". The default is
                                “mini”.
  -c, --config PATH             Path to the HDM configuration file.
```

Example

```
hdm version
```

Output

```
HDM Version = 3.2.0
  Build Date  = 2016-04-28T08:40:30
  Build Host  = HDMTEST-2
  Commit ID   = e24095ecaa75f64bc456fb6ebe762add87f5b9a8
  Ref Spec    = refs/heads/develop

Results for version: Operation succeeded.
```

Contact Information

General Information

Main Web Site:	*www.hgst.com*

Technical Support

Software Support:	*http://www.hgst.com/support/software-support*
Solid-State Drive Support:	*http://www.hgst.com/support/solid-state-drive-support*

Email Support and Telephone Support

Email Support:	*support@hgst.com*
Telephone Support:	1-855-778-2497. 24 x 7 Support. Please have the following information available when calling: Product Name, Model Number, Part Number and Operating System.

Index

HGST, a Western Digital company
3403 Yerba Buena Road
San Jose, CA 95135
Produced in the United States

United States: 800-801-4618 (Toll-Free)
International: 408-717-6000
FAX: 408-717-5000

May 31, 2016