



**Intel® Optane™ Persistent
Memory Software Specification
for Windows* and Linux*
*Revision 2.03***

Reference Number 611922

Table of Contents

Preface	1
Legal	1
Revision History	2
2.03	2
2.02	2
2.01	4
2.00	4
Supporting Documents	5
1. Introduction	6
1.1. Document Formatting	6
1.2. Launching the CLI	6
1.3. Command Syntax	6
1.4. PMem Module Identification	7
1.5. Capacities	8
1.6. PMem Module Long Operations (DEVICE BUSY)	9
1.7. PMem Modules in non-POR configuration	9
1.8. PMem Modules Must Be Present	9
1.9. Output Format	10
1.10. Microsoft* Windows* Notes and Limitations	10
1.11. Debug / Verbose	10
1.12. CPU Maximum Mapped Memory Limiting	11
2. Commands	12
2.1. PMem Module Discovery	12
2.1.1. Show Topology	12
2.1.2. Show Socket	15
2.1.3. Show Device	17
2.1.4. Show Memory Resources	33
2.1.5. Show System Capabilities	37
2.2. Provisioning	42
2.2.1. Create Memory Allocation Goal	43
2.2.2. Show Memory Allocation Goal	48
2.2.3. Dump Memory Allocation Settings	52
2.2.4. Load Memory Allocation Goal	54
2.2.5. Delete Memory Allocation Goal	58
2.3. Instrumentation	60
2.3.1. Show Sensor	60
2.3.2. Change Sensor Settings	65

2.3.3. Show Device Performance	67
2.4. Persistent Memory Provisioning	70
2.4.1. Show Persistent Memory	71
2.5. Support and Maintenance	74
2.5.1. Help	74
2.5.2. Version	75
2.5.3. Show Device Firmware	76
2.5.4. Update Firmware	79
2.5.5. Modify Device	82
2.5.6. Show Preferences	84
2.5.7. Change Preferences	86
2.5.8. Dump Support Data	88
2.6. Debug	90
2.6.1. Start Diagnostic	90
2.6.2. Show Error Log	94
2.6.3. Dump Debug Log	96
2.6.4. Show ACPI Tables	98
2.6.5. Show Device Platform Configuration Data	100
2.6.6. Delete Device Platform Configuration Data	101
2.6.7. Inject Error	104
2.6.8. Show Command Access Policy	108
2.6.9. Show Command Effect Log	110
2.7. Playback and Record	113
2.7.1. Start Session	118
2.7.2. Dump Session	120
2.7.3. Load Session	121
2.7.4. Show Session	122
2.7.5. Stop Session	123
3. Addendum	125
3.1. Diagnostic Events	125
3.1.1. Quick Health Check Events	125
3.1.2. Platform Configuration Check Events	129
3.1.3. Security Check Events	132
3.1.4. Firmware Consistency and Settings Check Events	133



Preface

Legal

Notice: This document contains information on products in the design phase of development. The information here is subject to change without notice. Do not finalize a design with this information.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software, or service activation. Learn more at intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice.

Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel, the Intel logo, Intel Optane, and Xeon are trademarks of Intel Corporation in the U. S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Revision History

2.03

- Updated definitions of terms non-functional and unmanageable in show-device.
- Poison use note added to inject-error.
- Goal creation request limitations list updated to describe conditional exclusion of population violation PMem modules.
- Added -tag target documentation to start-session.
- Added StagedFWActivatable, QuiesceRequired, and ActivationTime to show-firmware.
- Removed type 20 SMBIOS option.
- Added AveragePower, Average12vPower, Average1_2vPower to show-device.
- Updated product name references throughout documentation and select user facing text in CLI.
- Added quick diagnostic event 544.
- Changed the PopulationViolationState attribute to PopulationViolation.
- Removed change device passphrase command.
- Removed change device security command.
- Removed enable device command.
- Removed erase device data command.

2.02

- Removed Show System Host command.
- Removed the following preferences in Show Preferences and Change Preferences:
 - APPDIRECT_GRANULARITY - now always uses 1 GiB granularity
- Added new output attributes to Show Device command.
 - General: SKUViolation, MixedSKU, PopulationViolationState
 - FIS 1.3 (missing attributes): MaxMediaTemperature, MaxControllerTemperature
 - FIS 2.1: MemoryBandwidthBoostFeature, MemoryBandwidthBoostMaxPowerLimit, MemoryBandwidthBoostAveragePowerTimeConstant, AveragePowerReportingTimeConstant, ThermalThrottleLossPercent
 - FIS 2.2: S3ResumeOptIn
 - FIS 2.3: SVNDowngradeOptIn, SecureErasePolicyOptIn, FwActivateOptIn and

FWActiveAPIVersion.

- Removed: PeakPowerBudget
- Added option nfit to the Show Region command.
- Added new section: CPU Maximum Mapped Memory Limiting.
- Added new section: PMem modules in non-POR configuration.
- Added new section: Playback and Record.
- Added new addendum section: Diagnostic Events.
- Added note to Create/Load Memory Allocation Goal command explaining that security lockstate disabled no longer prevents creating a new goal request.
- Updated Show Device command attribute LatchedLastShutdownStatus values to align with FIS description.
- Updated Show Memory Resources command output format to utilize a table format and include DDR and PMem module allocations.
- Updated Show Sensor command:
 - Updated output format, including definition of attributes AlarmThreshold and AlarmEnabled.
 - Removed SettableThresholds, SupportedThresholds, MaxMediaTemperature, MaxControllerTemperature, EnabledState.
 - Added MaxTemperature, AlarmEnabled.
- Updated Start Diagnostic command events:
 - Added platform diagnostic events 631, 632, 633.
 - Updated platform diagnostic events 624, 625, 626, 627.
 - Removed firmware diagnostic event 907.
- Updated description of PMem module target for Create Memory Allocation Goal command to describe limitations.
- Reorganized help output to be grouped logically and align closer to the documentation order.
- Fixed some errors, typos in text.
- Updated show firmware return data attributes:
 - Added Fw activation time, staged fw activatable and fw quiesce required.
- Updated documentation to explain HealthState in greater detail for Show Device, Show Namespace, and Show Persistent Memory commands.
- Updated documentation for Show Sensor command to clarify which sensor attributes apply to a given sensor.

2.01

- Removed Show Events and Acknowledge Event commands.
- Removed ActionRequired output value in Show Device command.
- Removed the following preferences in Show Preferences and Change Preferences:
 - PERFORMANCE_MONITOR_ENABLED
 - PERFORMANCE_MONITOR_INTERVAL_MINUTES
 - EVENT_MONITOR_ENABLED
 - EVENT_MONITOR_INTERVAL_MINUTES
 - EVENT_LOG_MAX
- Removed the Show Topology command [-d | -display] option.
- Added notes to delete-pcd, load-goal and create-goal commands.
- Fixed some errors in text.
- Added details about Maximum Mapped Memory Limiting.
- Removed requirement for "-recover" flag for updating the firmware of non-functional PMem modules. All PMem modules are now updated over the fastest interface possible, automatically. Note that PMem module Flash SPI recovery is still a separate flow and still requires the "-recover FlashSPI" parameter and a full SPI image.

2.00

- Show Device return data attribute updates:
 - Added AveragePower, Average12vPower, Average1_2vPower.
 - Changed Speed to be correctly reported in MT/s.
- Help output has been updated for clarity.
- Options smbus and ddrt have been added to all commands (except Show Version and Help) for explicit transport selection. Option ddrt is the default if not specified.
- Set Sensor property NonCriticalThreshold is renamed to AlarmThreshold.
- Show Sensor result data has been updated for clarity.
 - CurrentState has been removed.
 - Property NonCriticalThreshold is renamed to AlarmThreshold.
 - Properties LowerThresholdNonCritical, UpperThresholdNonCritical, LowerThresholdCritical, UpperThresholdCritical, UpperThresholdFatal have been replaced with ThrottlingStopThreshold, ThrottlingStartThreshold, ShutdownThreshold.
 - Only valid properties are displayed for given sensor type.

- Start Diagnostic result data has been updated for clarity
 - For each test group, subtest execution and result is now presented.
 - Diagnostic event code is presented with the test/subtest result.
- Diagnostic events are included in this document.
- Removed property FirstFastRefresh from command Modify Device.
- Added option 'master' to specify master passphrase on Change Device Passphrase and Erase Device Data.
- Added commands Show Command Access Policy and Show SMBIOS.
- The following commands result data output has been modified for clarity.
 - Show Device Platform Configuration Data
 - Show SMBIOS
 - Show ACPI Tables
 - Show Error Log
 - Show Register
 - Delete Platform Configuration Data
 - Load Goal
- Show Topology devices are sorted by topology order.

Supporting Documents

- *Software-Firmware Interface Specification (611932)*
- *Intel® Optane™ Persistent Memory Firmware Interface Specification (576640)*

1. Introduction

Command line access to Intel® Optane™ persistent memory module (PMem module) management functionality is available through the Command Line Interface (CLI) component.

1.1. Document Formatting

Throughout this specification, the syntax of each command is documented in a shaded text box. Items in [brackets] are optional and in (parenthesis) are required. For options, targets and properties, each possible value is separated by a '|' meaning "or" and the default value is italicized. For example, the following command syntax specifies that the verb "set" is required followed by an optional option "help". The target `-example` is required and a value for ExampleID can optionally be provided. It also specifies an optional property "Test" in which valid values are "Test1" or "Test2" with Test1 being the default.

```
ipmctl set [-h|-help] -example [ExampleID] [Test=(Test1|Test2)]
```

1.2. Launching the CLI

The CLI application is launched from the OS/UEFI shell. To run a single command, supply the command and parameters immediately following the binary name. The command will run, display the resulting data and return control to the shell. If no command is provided, the CLI will display the help information as documented in the Help command.

```
ipmctl (command)
```

1.3. Command Syntax

The command line syntax is case insensitive and is interpreted in English only. It follows the DMTF SM CLP standard (DSP0214) with the exception of the target portion of the command. Generally the form of a user request is:

```
ipmctl <verb>[<options>][<targets>][<properties>]
```

A command has a single verb which represents the action to be taken. Following the verb are zero or more options which modify the action of the verb; overriding the default behavior with explicitly requested behavior. Options generally have a short and long form (e.g., `-a` | `-all`). Generally one or more targets are required to indicate the object of the

action. However, there are a few cases where a target is not required. Finally, zero or more properties defined as a key/value pair can be used to modify the target.

The ordering of the command parts shown above is enforced (e.g., a target cannot precede an option). However, ordering within the command parts is not enforced (e.g., options can be provided in any order, targets can be provided in any order and properties can be provided in any order). For example, the following command will generate a syntax error because the option `-all` comes after the target `-system` and therefore the command parts are mis-ordered (verb then target then option).

```
ipmctl show -system -all
```

Whereas, these commands are both valid even though the options are specified in different order because the command parts are provided in the correct order (verb then options then target).

```
ipmctl show -d Capacity -u B -dimm  
ipmctl show -u B -d Capacity -dimm
```

If an option or target allows input of more than one value, it is expected as a comma separated list without whitespace between the values. For example, `"-display value1,value2"` instead of `"-display value1, value2"`.

Properties follow the syntax `[PROPERTIES]=Value`.

1.4. PMem Module Identification

Throughout this document, PMem modules are referenced by one of two unique IDs: DimmHandle & DimmUID. Either ID may be used for commands that utilize a `-dimm` target.

For example, each of the following are equivalent:

```
ipmctl show -d DimmHandle,DimmUID -dimm 8089-a2-1748-00000001  
ipmctl show -d DimmHandle,DimmUID -dimm 0x0001  
ipmctl show -d DimmHandle,DimmUID -dimm 1  
---DimmID=0x0001---  
DimmHandle=0x0001  
DimmUID=8089-a2-1748-00000001
```

The `-dimm` target also accepts a comma separated list.

```
ipmctl show -d DimmHandle,DimmUID -dimm 0x0001,0x1001
ipmctl show -d DimmHandle,DimmUID -dimm 0x0001,8089-a2-1748-00000002
---DimmID=0x0001---
    DimmHandle=0x0001
    DimmUID=8089-a2-1748-00000001
---DimmID=0x1001---
    DimmHandle=0x1001
    DimmUID=8089-a2-1748-00000002
```

The default DimmID output format can be modified by changing the CLI_DEFAULT_DIMM_ID property of using the [Change Preferences](#) command. For example, with `CLI_DEFAULT_DIMM_ID=UID`:

```
ipmctl show -d DimmHandle,DimmUID -dimm 0x0001
---DimmID=8089-a2-1748-00000001---
    DimmHandle=0x0001
    DimmUID=8089-a2-1748-00000001
```

DimmHandle

The PMem module handle is formatted as 0xABCD.

- A = Socket
- B = Memory Controller
- C = Channel
- D = Slot

DimmUID

The unique identifier of the PMem module formatted as VVVV-ML-MMYYSNSNSNSN or VVVV-SNSNSNSN (if the manufacturing information is not available) where:

- VVVV = VendorID
- ML = ManufacturingLocation
- MMYT = ManufacturingDate
- SNSNSNSN = SerialNumber

1.5. Capacities

Capacities can be displayed in bytes (B), binary multiples of bytes (MiB, GiB, TiB), or decimal multiples of bytes (MB, GB, TB) where:

- 1 mebibyte (MiB) = 1024^2 bytes
- 1 gibibyte (GiB) = 1024^3 bytes

- 1 tebibyte (TiB) = 1024^4 bytes
- 1 megabyte (MB) = 1000^2 bytes
- 1 gigabyte (GB) = 1000^3 bytes
- 1 terabyte (TB) = 1000^4 bytes

Capacities are displayed to 1/1000 precision (except bytes). Therefore, a capacity may be displayed as 0.000 if the value is less than 0.001. The default display of capacities is configurable with the command [Change Preferences](#). The input and display of capacities in an individual command can also be overridden by supplying the -units option. Capacities can be input in either decimal or hexadecimal format with a '0x' prefix.

1.6. PMem Module Long Operations (DEVICE BUSY)

PMem modules may be busy while processing some operations. These are defined as 'Long Operations' in the Firmware Interface Specification (FIS). While a long operation is in progress, some requests may be rejected as indicated by 'DEVICE BUSY' status. Commands may be retried and are expected to succeed when the long operation is complete. Show Device may be used to determine if ARS (ARSStatus) or Overwrite PMem module (OverwriteStatus) long operations are in progress.

1.7. PMem Modules in non-POR configuration

Memory populations are evaluated based on the "Enforce Population POR" setup option in UEFI Firmware to "Enforce Supported Populations" or "Enforce Validated Populations". When either option is selected, some memory may fall outside of the enforced population. The PopulationViolation attribute shows this status of PMem modules. Some commands may not be supported by management software for PMem modules in population violation.

Commands not supported for PMem modules in non-POR configuration are: * Create goal * Show goal * Dump goal * Load goal * Delete goal * Show performance * Change sensor * Show namespace * Create namespace * Delete namespace * Support * Modify device

1.8. PMem Modules Must Be Present

If PMem modules are not present then the functionality via the CLI will be unavailable.

1.9. Output Format

By default, the CLI formats the output of each command as text to facilitate readability. However, the CLI supports changing the output format from the default text to Extensible Markup Language (XML) with the output option. For example, to retrieve the output of the help command in XML format rather than text, add the output option.

```
ipmctl help -output nvmlxml
```

Changes the output format to XML. Refer to the XML Document Type Definition (DTD) Specification for detailed information about the XML output format of each command.

1.10. Microsoft* Windows* Notes and Limitations

The Windows driver that enables ipmctl communication to Intel's PMem modules prevents executing commands that change configuration of any PMem module when there is a related logical disk (namespace) associated with that PMem module. This is done to protect user data. If a logical disk (namespace) is associated with the target PMem module, the command will return an error. The logical disk (namespace) must first be deleted before attempting to execute commands that change configuration.

Generally, all commands that retrieve status will succeed regardless of logical disk presence.

Injecting a Fatal Media error is unsupported on Microsoft Windows. Contact Microsoft for assistance in performing this action.

1.11. Debug / Verbose

The CLI provides the capability to view debug messages when executing a command by specifying the verbose mode output option. This is intended to aid diagnosing a problem with the CLI itself and may be useful when submitting a defect.

Changes the output format to XML and view debug message for the command.

```
ipmctl show -verbose -dimm
```

1.12. CPU Maximum Mapped Memory Limiting

The CPU maximum mapped memory limit (MML) will always be enforced. This MML is found in the Die SKU Information Table (6) of the ACPI PCAT table and can be retrieved using the Show Socket command. When creating a new memory allocation goal request, the total capacity is calculated and compared with the MML. If the total requested mapped memory capacity exceeds the MML then it will be reduced and a warning will be displayed to inform the user.

Details for reducing memory allocation goal due to memory mapped limit:

- If the calculated TotalRequestedMappedMemory is more than the maximum mapped memory limit, then the requested mapped memory will be reduced in the following order:
 1. Asymmetrical AD
 2. Symmetrical AD
 3. Memory mode
- For Platform in 1LM mode:

$$\text{TotalRequestedMappedMemory} = \text{Requested AD Capacity} + \text{DDR size}$$

- For Platform in 2LM mode (with MemoryMode in request):

$$\text{TotalRequestedMappedMemory} = \text{Requested AD Capacity} + \text{Requested MM Capacity}$$

- For Platform in 2LM mode (without MemoryMode in request):

$$\text{TotalRequestedMappedMemory} = \text{Requested AD Capacity} + \text{DDR size (DDR is no longer a cache)}$$

2. Commands

2.1. PMem Module Discovery

PMem module Identifiers The default PMem module identifier "DimmID" is equivalent to DimmHandle as described in the command [Show Device](#). This may be changed to DimmUID using the command [Change Preferences](#). In all commands accepting the -dimm target as input, either the DimmHandle or the DimmUID may be used to identify a PMem module.

PMem Module Manageability Beyond discovery, most PMem module operations require the ability of host software to manage the PMem module. Manageability is determined by the interface format code, the vendor identifier, device identifier and the firmware API version and can be retrieved with the command [Section Show Device](#).

PMem module Compatibility If the host software detects that the system is populated with PMem modules with incompatible SKUs or the PMem modules are configured in violation of their license, it will indicate such and operate in read-only mode. In this case, the host software will not allow changes to the PMem modules and their associated capacity.

2.1.1. Show Topology

NAME

ipmctl-show-topology - Shows the topology of the memory installed

SYNOPSIS

```
ipmctl show [OPTIONS] -topology [TARGETS]
```

DESCRIPTION

Shows the topology of the memory installed in the host server. Use the command ipmctl-show-device to view more detailed information about a PMem module.

OPTIONS

-a

-all

Shows all attributes.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

-u (B|MB|MiB|GB|GiB|TB|TiB)

-units (B|MB|MiB|GB|GiB|TB|TiB)

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS

-dimm [DimmIDs]

Restricts output to specific PMem modules by optionally supplying the PMem module target and one or more comma-separated PMem module identifiers. The default is to display all PMem modules.

-socket [SocketIDs]

Restricts output to the PMem modules installed on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to display all sockets.

NOTE

If ACPI PMTT table is not present, then DDR4 memory will not be displayed in the filtered socket list.

EXAMPLES

Displays the system memory topology.


```
ipmctl show -topology
```

RETURN DATA

Displays a table with the attributes listed below for each memory PMem module installed in the host server.

MemoryType

(Default) The DIMM type. One of:

- Unknown
- DDR4
- Logical Non-volatile Device

Capacity

(Default) The raw capacity of the PMem module as reported in the SMBIOS Type 17 table.

DimmID

(Default) The DIMM identifier. For DRAM DIMMs, the DimmID is "N/A".

PhysicalID

(Default) The PMem module physical identifier (i.e., SMBIOS Type 17 handle).

DeviceLocator

(Default) The string that identifies the physically-labeled socket or board position where the PMem module is located.

SocketID

The processor socket identifier (i.e., NUMA node) where the PMem module is installed.

DieID

The processor die identifier where the PMem module is installed.

MemControllerID

The associated memory controller identifier.

ChannelID

The associated channel. For DRAM DIMMs, the channel identifier is "N/A".

ChannelPos

The PMem module position in the channel.

NodeControllerID

The node controller identifier.

BankLabel

The string that identifies the physically-labeled bank where the PMem module is located.

2.1.2. Show Socket

NAME

ipmctl-show-socket - Shows basic information about the physical processors

SYNOPSIS

```
ipmctl show [OPTIONS] -socket [TARGETS]
```

DESCRIPTION

Shows basic information about the physical processors in the host server.

OPTIONS

-a

-all

Shows all attributes.

NOTE

The all and display options are exclusive and may not be used together.

-d (attributes)

-display (attributes)

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The `-ddrt` and `-smbus` options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

-u (B|MB|MiB|GB|GiB|TB|TiB)

-units (B|MB|MiB|GB|GiB|TB|TiB)

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS

-socket [SocketIDs]

Restricts output to the PMem modules installed on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to display all sockets.

EXAMPLES

Displays information about all the processors.

```
ipmctl show -socket
```

Lists all properties for socket 1.

```
ipmctl show -socket 1
```

Retrieves specific properties for each processor.

```
ipmctl show -d MappedMemoryLimit -socket
```

RETURN DATA

Displays a table with the attributes listed below for each physical processor installed in the host server.

SocketID

(Default) The processor socket identifier.

MappedMemoryLimit

(Default) The maximum amount of memory that is allowed to be mapped into the system physical address space for this processor based on its SKU.

TotalMappedMemory

(Default) The total amount of memory that is currently mapped into the system physical address space for this processor.

2.1.3. Show Device

NAME

ipmctl-show-device - Shows information about one or more PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -dimm [TARGETS]
```

DESCRIPTION

Shows information about one or more PMem modules.

OPTIONS

-a

-all

Shows all attributes.

NOTE	The all and display options are exclusive and may not be used together.
-------------	---

-d (attributes)

-display (attributes)

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

NOTE	The all and display options are exclusive and may not be used together.
-------------	---

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

-u (B|MB|MiB|GB|GiB|TB|TiB)**-units (B|MB|MiB|GB|GiB|TB|TiB)**

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS**-dimm [DimmIDs]**

Restricts output to specific PMem modules by supplying the PMem module target and one or more comma-separated PMem module identifiers. The default is to display all PMem modules.

-socket [SocketIDs]

Restricts output to the PMem modules installed on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to display all sockets.

NOTE

If ACPI PMTT table is not present, then DDR4 memory will not be displayed in the filtered socket list.

EXAMPLES

Lists a few key fields for each PMem module.

```
ipmctl show -dimm
```

Lists all properties for PMem module 0x0001.

```
ipmctl show -a -dimm 0x0001
```

Retrieves specific properties for each PMem module.

```
ipmctl show -d HealthState,LockState -dimm
```

RETURN DATA

The default behavior is to display a table with the default attributes listed below; applying options changes the output to a more detailed format. Limited information (noted in the table below) is applicable if the PMem module is not manageable by the software as indicated by the "ManageabilityState" property.

NOTE | Some data is endian-swapped for human readability.

DimmID

(Default) The PMem module identifier.

Capacity

(Default) The usable capacity of the PMem module as reported by the firmware.

HealthState

(Default) Overall PMem module health. One of:

- Healthy
- Noncritical: Maintenance is required.
- Critical: Features or performance are degraded due to failure.
- Fatal: Critical internal state failure (DPA Failure, Internal Buffer Failure, AIT Failure, etc.) is non-recoverable and data loss has occurred or is imminent. In this case, the firmware will disable the media and access to user data and operations that require use of the media will fail.
- Non-functional: The PMem module is detected and manageable, though some commands and capabilities may be limited. The PMem module has limited communication or another error preventing complete functionality. Common causes include:
 - DDRT memory interface training failure
 - Expected region mapping to SPA range unable to be found
- Unmanageable: The PMem module has an incompatible firmware API version or hardware revision or is unresponsive (possibly due to a communication interface failure or a firmware/hardware error).
- Unknown: Unable to determine the PMem module health state. Refer to *HealthStateReason* for *HealthState* details.

HealthStateReason

Indicates why the PMem module is in the current Health State. One or more of:

- None
- Percentage Remaining less than 1%
- Package Sparing occurred
- CAP Self-Test warning
- Percentage Remaining is 0
- Die Failure
- AIT DRAM disabled
- CAP Self-Test failure
- Critical internal state failure
- Performance degraded
- CAP Self-Test communication failure

Refer to the *Intel® Optane™ Persistent Memory Firmware Interface Specification*, document number 556488, Section 4.8 SMART and Health.

InterfaceFormatCode

A comma-delimited list of the JEDEC standard format interface codes for the PMem module where each code is formatted as: code (JEDEC Description or "Unknown").

ManageabilityState

Ability of the PMem module host software to manage the PMem module. Manageability is determined by the interface format code, the vendor identifier, device identifier and the firmware API version. One of:

- Manageable: The PMem module is manageable by the software.
- Unmanageable: The PMem module is not supported by this version of the software.

PopulationViolation

Memory populations are evaluated based on the "Enforce Population POR" setup option in UEFI Firmware. If enforcement of POR populations is selected, then some PMem module memory may be in population violation. See [\[PMem modules in non-POR configuration\]](#) for details. One of:

- Yes: The PMem module is in population violation.
- No: The PMem module is not in population violation.

PhysicalID

The PMem module physical identifier (i.e., SMBIOS Type 17 handle).

DimmHandle

The PMem module handle formatted as 0xABCD.

- A = Socket

- B = Memory Controller
- C = Channel
- D = Slot

DimmUID

The unique identifier of the PMem module formatted as VVVV-ML-MMYYSNSNSNSN or VVVV-SNSNSNSN (if the manufacturing information is not available) where:

- VVVV = VendorID
- ML = ManufacturingLocation
- MMYT = ManufacturingDate
- SNSNSNSN = SerialNumber

SocketID

The processor socket identifier (i.e., NUMA node) where the PMem module is installed.

MemControllerID

The associated memory controller identifier.

ChannelID

The associated channel.

ChannelPos

The PMem module position in the channel.

MemoryType

The memory type. One of:

- Unknown
- DDR4
- Logical Non-Volatile Device

VendorID

The vendor identifier of the PMem module. This value is presented in big endian format.

DeviceID

The device identifier of the PMem module. This value is presented in big endian format.

RevisionID

The revision identifier of the PMem module.

**SubsystemVendorID**

The vendor identifier of the non-volatile memory subsystem controller. This value is presented in big endian format.

SubsystemDeviceID

The device identifier of the non-volatile memory subsystem controller.

SubsystemRevisionID

The revision identifier of the non-volatile memory subsystem controller retrieved from NFIT. This field uses a different encoding than ControllerRevisionID.

ManufacturingInfoValid

If the manufacturing location and date are valid. One of:

- 0: Not valid
- 1: Valid

ManufacturingLocation

The manufacturing location assigned by the vendor or "N/A" if ManufacturingInfoValid is 0.

ManufacturingDate

The manufacturing date assigned by the vendor or "N/A" if ManufacturingInfoValid is 0.

SerialNumber

The serial number assigned by the vendor. This value is presented in big endian format.

PartNumber

The part number assigned by the vendor

DeviceLocator

A string describing the physically-labeled socket or board position where the memory device is located from the SMBIOS Type 17 Memory Device table.

BankLabel

A string that identifies the physically labeled bank where the memory device is located from the SMBIOS Type 17 Memory Device table.

DataWidth

The width in bits used to store user data from the SMBIOS Type 17 Memory Device table.

TotalWidth

The width in bits for data and error correction and/or data redundancy from the

SMBIOS Type 17 Memory Device table.

Speed

The maximum capable speed of the device in megatransfers per second (MT/s) from the SMBIOS Type 17 Memory Device table.

FormFactor

The PMem module form factor (i.e., SMBIOS Type 17 Memory Device Form Factor). One of:

- Unknown
- DIMM
- SODIMM

LockState

The current security state of the persistent memory on the PMem module. One or more of:

- Unknown - The security state cannot be determined (e.g., when the PMem module is not manageable by the software).
- Disabled - Security is not enabled.
- Unlocked - Security is enabled and unlocked.
- Locked - Security is enabled and locked.
- Frozen - A reboot is required to change the security state.
- Exceeded - The passphrase limit has been reached. A power cycle is required to change the security state.
- MP Exceeded - The master passphrase limit has been reached. A power cycle is required to change the security state.
- Not Supported - Security is not supported on the PMem module.

SVNDowngrade

The Opt-in value of SVN Downgrade security opt-in feature for PMem module. One of :

- Unknown
- Disabled
- Enabled

SecureErasePolicy

The Opt-in value of Secure erase policy opt-in feature for PMem module. One of :

- Unknown
- No Master Passphrase

- Master Passphrase Enabled

S3ResumeOptIn

The Opt-in value of S3 Resume security opt-in feature for PMem module. One of :

- Unknown
- UnsecureS3
- SecureS3

FwActivateOptIn

The Opt-in value of Fw Activate security opt-in feature for PMem module. One of :

- Unknown
- Disabled
- Enabled

FWVersion

(Default) The BCD-formatted revision of the active firmware in the format PN.RN.SV.bbbb where:

- PN = 2-digit product number
- RN = 2-digit revision number
- SN = 2-digit security revision number
- bbbb = 4-digit build version

Value may be N/A if the PMem module is not manageable by the software.

FWAPIVersion

The firmware supported interface revision in the format aa.bb where:

- aa = 2-digit major version
- bb = 2-digit minor version

The firmware interface is intended to be backwards compatible. Therefore, the host software allows management of PMem modules where this version is less than or equal to the version stored in the host software. Value may be N/A if the PMem module is not manageable by the software.

The following information is only applicable when the PMem module is manageable by the software as indicated by the "ManageabilityState".

FWActiveAPIVersion

The firmware interface revision locked in the BIOS API handshake in the format aa.bb where:

- aa = 2-digit major version
 - bb = 2-digit minor version
- Value may be N/A if the PMem module is not manageable by the software.

ManufacturerID

The manufacturer identifier of the PMem module. This value is presented in big endian format.

ControllerRevisionID

The controller stepping and revision ID retrieved from the controller FW. This field uses a different encoding than SubsystemRevisionID.

IsNew

Whether or not the PMem module is incorporated with the rest of the PMem module in the system. One of:

- 0: Configured
- 1: The PMem module requires configuration.

MemoryCapacity

Usable PMem module Memory Mode capacity.

AppDirectCapacity

Usable PMem module App Direct capacity.

UnconfiguredCapacity

PMem module capacity that is inaccessible because it is not mapped into the system physical address space.

InaccessibleCapacity

PMem module capacity that is inaccessible due to:

- Licensing issue
- Platform configuration prevents accessing this capacity. For example, MemoryCapacity is configured and available on a PMem module but MemoryMode is not enabled by BIOS.

ReservedCapacity

PMem module capacity reserved for proper alignment.

AvgPowerLimit

If the PMem module firmware power management policy is enabled, the power limit in mW used for average power. Refer to FIS for allowable range and default value.

MemoryBandwidthBoostFeature

Returns if the Memory Bandwidth Boost Feature is currently enabled or not. One of:

- 0x0: Disabled
- 0x1: Enabled

MemoryBandwidthBoostMaxPowerLimit

The power limit used for limiting the Memory Bandwidth Boost Feature's power consumption [mW].

MemoryBandwidthBoostAveragePowerTimeConstant

The value used as a base time window for average power throttle [ms]. This range can be checked in the Max Memory Bandwidth Boost Average Time Constant and Average Power Time Constant Step from the Get Device Characteristics command.

- Default: 15000 ms

MaxAveragePowerLimit

Maximum average power limit [mW] supported by the PMem module.

MaxMemoryBandwidthBoostMaxPowerLimit

Maximum Memory Bandwidth Boost Power value [mW] that can be set for the PMem module. Will return 0 if unsupported by current FIS.

MaxAveragePowerTimeConstant

This field returns the maximum supported value of the Average Power Time Constant and Average Power Reporting Time constant [ms] that can be set via the Set Power Management Policy and the Optional Configuration Data Policy Commands.

AveragePowerTimeConstantStep

This field returns the increments in milliseconds allowed by the firmware when setting the Average Power Time Constant and Average Power Reporting Time Constant via the Set Power Management Policy and the Optional Configuration Data Policy Commands.

AveragePower

This field returns the average power in milliwatts that each PMem module consumes over the Average Power Reporting Time constant.

Average12vPower

This field returns the 12V average power in milliwatts that each PMem module consumes over the Average Power Reporting Time constant.

Average1_2vPower

This field returns the 1.2V average power in milliwatts that each PMem module consumes over the Average Power Reporting Time constant.

PackageSparingCapable

Whether or not the PMem module supports package sparing. One of:

- 0: False

- 1: True

PackageSparingEnabled

Whether or not the PMem module package sparing policy is enabled. One of:

- 0: Disabled
- 1: Enabled

PackageSparesAvailable

The number of spare devices available for package sparing.

LatchedLastShutdownStatus

The status of the last shutdown of the PMem module. One or more of:

- Unknown: The last shutdown status cannot be determined.
- PM ADR Command Received: Power management ADR command received.
- PM S3 Received: Power management S3 command received.
- PM S5 Received: Power management S5 command received.
- DDRT Power Fail Command Received: DDR power fail command received.
- PMIC 12V/DDRT 1.2V Power Loss (PLI)
- PM Warm Reset Received: Power management warm reset received.
- Thermal Shutdown Received: Thermal shutdown triggered.
- Controller's FW State Flush Complete: Flush Completed.
- Viral Interrupt Received: Viral interrupt received.
- Surprise Clock Stop Received: Surprise clock stop received.
- Write Data Flush Complete: Write data flush completed.
- PM S4 Received: Power management S4 command received.
- PM Idle Received: Power management idle received.
- DDRT Surprise Reset Received: Surprise reset received.
- Extended Flush Complete.

UnlatchedLastShutdownStatus

The status of the last shutdown status of the PMem module. Same fields as the Latched Last Shutdown Status with the only difference that LLS details on a dirty shutdown is logged, even if the Latch System Shutdown Status was not enabled. One or more of:

- Unknown: The last shutdown status cannot be determined.
- PM ADR Command: Power management ADR command received.
- PM S3: Power management S3 command received.
- PM S5: Power management S5 command received.

- DDRT Power Fail Command: DDR power fail command received.
- PMIC 12V/DDRT 1.2V Power Loss (PLI)
- PM Warm Reset: Power management warm reset received.
- Thermal Shutdown: Thermal shutdown triggered.
- Controller's FW State Flush Complete: Flush Completed.
- Viral Interrupt: Viral interrupt received.
- Surprise Clock Stop: Surprise clock stop received.
- Write Data Flush Complete: Write data flush completed.
- PM S4: Power management S4 command received.
- DDRT Surprise Reset: Surprise reset received
- PM Idle Received: Power management idle received.
- Extended Flush Complete.

ThermalThrottleLossPercent

The average performance loss percentage due to thermal throttling in current boot of the PMem module.

LastShutdownTime

The time the system was last shutdown.

ModesSupported

A list of the modes supported by the PMem module. Refer to the command Show System Capabilities to determine the modes supported by the platform. One or more of:

- Memory Mode: PMem modules act as system memory under the control of the operating system. In Memory Mode, any DDR in the platform will act as a cache working in conjunction with the PMem module.
- App Direct: PMem modules and DDR act as independent memory resources under direct load/store control of the application.

SecurityCapabilities

The security features supported by the PMem module. Zero or more of:

- Encryption: The PMem module supports persistent memory encryption by setting a passphrase.
- Erase: The PMem module is erasable.

MasterPassphraseEnabled

This property indicates if master passphrase is enabled. If it is disabled, then it cannot be enabled. One of:

- 0: Disabled - Cannot be enabled.

- 1: Enabled - Master passphrase can be changed. Cannot be disabled.

ConfigurationStatus

The status of the PMem module memory configuration. One of:

- Valid: The configuration is valid.
- Not Configured: The PMem module has not been configured.
- Failed - Bad configuration: The configuration is corrupt.
- Failed - Broken interleave: This PMem module is part of an interleave set that is not complete.
- Failed - Reverted: The configuration failed and was reverted to the last known good configuration.
- Failed - Unsupported: The configuration is not compatible with the installed BIOS.
- Unknown: The configuration cannot be determined.

SKUViolation

The configuration of the PMem module is unsupported due to a license issue. One of:

- 0: False
- 1: True

ARSStatus

The address range scrub (ARS) operation status for the PMem module. The status is a reflection of the last requested ARS, but not necessarily within the current platform power cycle. One of:

- Unknown - The ARS operation status cannot be determined.
- Not started - An ARS operation has not started.
- In progress - An ARS operation is currently in progress.
- Completed - The last ARS operation has completed.
- Aborted - The last ARS operation was aborted.

OverwriteStatus

The overwrite PMem module operation status for the PMem module. One of:

- Unknown - The overwrite PMem module operation status cannot be determined. This may occur if the status gets overwritten due to a different long operation running on this PMem module.
- Not started - An overwrite PMem module operation was not started on the last boot.
- In progress - An overwrite PMem module operation is currently in progress.
- Completed - An overwrite PMem module operation completed and a reboot is

required to use the PMem module.

AveragePowerReportingTimeConstantMultiplier

The value that is used to multiply the base time value used for average power consumption measurements. Can be set to a value between 0 and 32. The default value is 0.

AveragePowerReportingTimeConstant

The value, in milliseconds, used to determine the time constant for reporting the average power consumption measurements. Can be set to a value between 100 and 12000, by increments of 100. The default value is 1000.

ViralPolicy

Whether viral policies are enabled on the PMem module. One of:

- 0: Disabled - This is the default.
- 1: Enabled - The persistent memory on the PMem module will be put into read-only mode if the host operating system software detects an uncorrectable error situation and indicates a viral state in order to prevent the spread of damage.

ViralState

Whether the PMem module is currently viral. One of:

- 0: Not Viral
- 1: Viral - The viral policies of the PMem module have switched the persistent memory to read-only mode due to the host operating system software detecting an uncorrectable error situation and indicating a viral state.

AitDramEnabled

If the PMem module AIT DRAM is enabled. One of:

- 0: Disabled - The device will suffer performance degradation if the AIT DRAM becomes disabled.
- 1: Enabled

BootStatus

The initialization status of the PMem module as reported by the firmware in the boot status register. One or more of:

- Unknown - The boot status register cannot be read.
- Success - No errors were reported during initialization.
The following statuses indicate that the media is not functional and therefore access to user data and operations that require use of the media will fail.
- Media Not Ready - The firmware did not complete media training.
- Media Error - The firmware detected an error during media training.

- Media Disabled - The firmware disabled the media due to a critical issue.
The following statuses indicate that communication with the firmware is not functional.
- FW Assert - The firmware reported an assert during initialization.

BootStatusRegister

The raw hex value of the PMem module Boot Status Register of the PMem module

LatchSystemShutdownState

Status of the latch. Specifies whether the PMem module will latch the SMART Last Shutdown Status and SMART Dirty Shutdown Count.

- 0: Disabled - This is the default.
- 1: Enabled

PreviousPowerCycleLatchSystemShutdownState

The status of the latch during the previous power cycle.

- 0: Disabled - This is the default.
- 1: Enabled

ExtendedAdrEnabled

Specifies whether extended ADR flow is enabled in the FW.

- 0: Disabled
- 1: Enabled

PpcExtendedAdrEnabled

Specifies whether extended ADR flow was enabled in the FW during the last power cycle.

- 0: Disabled
- 1: Enabled

ErrorInjectionEnabled

Error injection status.

- 0: Disabled - This is the default.
- 1: Enabled

MediaTemperatureInjectionEnabled

Media temperature injection status.

- 0: Disabled - This is the default.
- 1: Enabled

SoftwareTriggersEnabled

Software trigger status.

- 0: Disabled - This is the default.
- 1: At least one software trigger enabled.

SoftwareTriggersEnabledDetail

Comma separated list of software triggers currently enabled. One or more of:

- None
- Package Sparing
- Fatal Error
- Percentage Remaining
- Dirty Shutdown

PoisonErrorInjectionsCounter

This counter is incremented each time the set poison error is successfully executed.

PoisonErrorClearCounter

This counter is incremented each time the clear poison error is successfully executed.

MediaTemperatureInjectionsCounter

This counter is incremented each time the media temperature is injected.

SoftwareTriggersCounter

This counter is incremented each time a software trigger is enabled.

MaxMediaTemperature

The highest die temperature reported in degrees Celsius. This value is persistent through Power Loss as well as not effected by Overwrite PMem module or Media Temperature Error Injection.

MaxControllerTemperature

The highest controller temperature reported in degrees Celsius. This value is persistent through Power Loss as well as not effected by Overwrite PMem module.

MixedSKU

One or more PMem modules in the system have different SKUs. One of:

- 0: False
- 1: True - In this case, the host software operates in a read-only mode and does not allow changes to the PMem modules and their associated capacity.

2.1.4. Show Memory Resources

NAME

ipmctl-show-memory-resources - Shows PMem module and DDR memory allocation

SYNOPSIS

```
ipmctl show [OPTIONS] -memoryresources
```

DESCRIPTION

Shows PMem module and DDR memory allocation information for this platform.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

-u (B|MB|MiB|GB|GiB|TB|TiB)

-units (B|MB|MiB|GB|GiB|TB|TiB)

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

EXAMPLES

Shows the PMem module and DDR memory allocation.

```
ipmctl show -memoryresources
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

Returns a table containing the attributes listed below.

NOTE

Capacities from unmanageable PMem modules are not included in the following aggregated totals.

Volatile DDR Capacity

Total DDR capacity that is used as volatile memory.

Volatile PMem module Capacity

Total PMem module capacity that is used as volatile memory.

Total Volatile Capacity

Total DDR and PMem module capacity that is used as volatile memory.

AppDirect PMem module Capacity

Total PMem module capacity used as persistent memory.

Total AppDirect Capacity

Total DDR and PMem module capacity used as persistent memory.

Cache DDR Capacity

Total DDR capacity used as a cache for PMem modules.

Total Cache Capacity

Total DDR capacity used as a cache for PMem modules.

Inaccessible DDR Capacity

Total DDR capacity that is inaccessible.

Inaccessible PMem module Capacity

Total PMem module capacity that is inaccessible due to any of:

- Platform configuration prevents accessing this capacity. For example, MemoryCapacity is configured but MemoryMode is not enabled by platform FW (current Memory Mode is 1LM).

- Capacity is inaccessible because it is not mapped into the System Physical Address space (SPA). This is usually due to platform firmware memory alignment requirements.
- Persistent capacity that is reserved. This capacity is the persistent memory partition capacity (rounded down for alignment) less any App Direct capacity. Reserved capacity typically results from a Memory Allocation Goal request that specified the Reserved property. This capacity is not mapped to System Physical Address space (SPA).
- Capacity that is unusable because it has not been configured.
- PMem module configured capacity but SKU prevents usage. For example, AppDirectCapacity but PMem module SKU is MemoryMode only.

Total Inaccessible Capacity

Total capacity of DDR and PMem module that is inaccessible.

Physical DDR Capacity

Total physical DDR capacity populated on the platform.

Physical PMem module Capacity

Total physical PMem module capacity populated on the platform.

Total Physical Capacity

Total physical capacity populated on the platform.

DETAILS

PMem modules are partitioned into Memory and Persistent partitions. Memory partitions are aligned on a 1 GiB boundary by ipmctl with the Persistent partition consuming the remaining capacity. Any capacity that falls outside the Memory and Persistent partitions is InaccessibleCapacity and is not useable. If the PMem module is configured for 100% Memory Mode, then the Memory partition consumes all of the capacity and the Persistent partition has none.

The mode determines how the DDR capacities are allocated. In 1LM and AppDirect, all DDR capacity is used as additional volatile memory. In MemoryMode, all DDR capacity that the CPU will support is used as a cache for the PMem module.

Platform firmware alignment restrictions may result in some capacity from the Memory and Persistent partitions not mapped to System Physical Address space (SPA). This memory is considered InaccessibleCapacity and is not usable.

The definitions and calculations that follow intend to describe how each of the values are determined.

Definitions:

Intel PMem module Current Config

See the *Intel® Optane™ Persistent Memory Software-Firmware Interface Specification* for details.

PMem module Partition Info

PMem module partition information provided by PMem module firmware. See the *Intel® Optane™ Persistent Memory Module Firmware Interface Specification* for details.

PMem module Physical Capacity (PC)

Raw capacity (total usable) reported by PMem module Partition Info

PMem module Memory Partition Capacity (MPC)

Volatile capacity reported by PMem module Partition Info

PMem module Persistent Partition Capacity (PPC)

Persistent capacity reported by PMem module Partition Info

PMem module Volatile Memory Capacity (VMC)

Usable volatile memory capacity as reported by platform FW via the *Intel PMem module Current Config* → *Volatile Memory Size Mapped into SPA* field

PMem module Persistent Memory Capacity (PMC)

Usable persistent memory capacity as reported by platform FW via the *Intel PMem module Current Config* → *Persistent Memory Size Mapped into SPA* field

DDR Total Capacity (DDRTC)

Total amount of memory available for use on the populated DDRs

Calculations:

```
if (CurrentMode == 1LM) then
  DDRCacheCapacity = 0
  DDRVolatileCapacity = DDRTC
else if (CurrentMode == 2LM) then
  DDRCacheCapacity = DDRTC
  DDRVolatileCapacity = 0
```

```
TotalVolatileCapacity = VMC + DDRVolatileCapacity
```

```
InaccessibleCapacity = PC - PMC
if (CurrentMode != 1LM) then
  InaccessibleCapacity -= VMC (rounded down for alignment)
```

2.1.5. Show System Capabilities

NAME

ipmctl-show-system-capabilities - Shows the platform supported PMem module capabilities.

SYNOPSIS

```
ipmctl show [OPTIONS] -system -capabilities
```

DESCRIPTION

Shows the capabilities of the current platform as determined by platform firmware (BIOS) and this application.

OPTIONS

-a

-all

Shows all attributes.

NOTE	The all and display options are exclusive and may not be used together.
-------------	---

-d (attributes)

-display (attributes)

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

NOTE	The all and display options are exclusive and may not be used together.
-------------	---

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The `-ddrt` and `-smbus` options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

-u (B|MB|MiB|GB|GiB|TB|TiB)

-units (B|MB|MiB|GB|GiB|TB|TiB)

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

EXAMPLES

Displays the supported PMem module capabilities

```
ipmctl show -system -capabilities
```

RETURN DATA

The default behavior is to return the default attributes listed below; the options can be used to expand or restrict the output.

PlatformConfigSupported

(Default) Whether the platform level configuration of PMem modules can be modified with the host software. One of:

- 0: Changes must be made in the BIOS.
- 1: The command **Create Memory Allocation Goal** is supported.

Alignment

(Default) Capacity alignment requirement for all memory types as reported by the BIOS.

AllowedVolatileMode

(Default) The volatile mode allowed as determined by BIOS setup. One of:

- 1LM: One-level volatile mode. All memory resources in the platform are independently accessible, and not captive of the other resources.
- Memory Mode: PMem modules act as system memory under the control of the operating system. In Memory Mode, any DDR in the platform will act as a cache working in conjunction with the PMem modules.
- Unknown: The allowed volatile mode cannot be determined.

CurrentVolatileMode

(Default) The current volatile mode. One of:

- 1LM: One-level volatile mode. All memory resources in the platform are independently accessible, and not captive of the other resources.
- Memory Mode: PMem modules act as system memory under the control of the operating system. In Memory Mode, any DDR in the platform will act as a cache working in conjunction with the PMem modules.
- Unknown: The current volatile mode cannot be determined.

AllowedAppDirectMode

(Default) The App Direct mode allowed as determined by BIOS setup. One of:

- Disabled: App Direct support is currently disabled by the BIOS.
- App Direct: App Direct support is currently enabled by the BIOS.
- Unknown: The current App Direct support cannot be determined.

ModesSupported

A list of PMem module modes supported by the BIOS. Refer to the command **Show Device** to determine the modes supported by the individual PMem modules. One or more of:

- 1LM: One-level volatile mode. All memory resources in the platform are independently accessible, and not captive of the other resources.
- Memory Mode: PMem modules act as system memory under the control of the operating system. In Memory Mode, any DDR in the platform will act as a cache working in conjunction with the PMem modules.
- App Direct: PMem modules and DDR act as independent memory resources under direct load/store control of the application.

SupportedAppDirectSettings

The BIOS supported list of App Direct interleave settings in the format:

- x[Way] - [(IMCSize) iMC x (ChannelSize) Channel] followed by the input format for the command **Change Preferences**: (ByOne | (IMCSize)_(ChannelSize)).

RecommendedAppDirectSettings

The BIOS recommended list of App Direct interleave settings in the format:

- x[Way] - [(IMCSize) iMC x (ChannelSize) Channel] followed by the input format for the command **Change Preferences**: (ByOne | (IMCSize)_(ChannelSize)).

MinNamespaceSize

The minimum allowed namespace size as reported by the driver.

AppDirectMirrorSupported

If the BIOS supports App Direct mirroring. One of:

- 0: Not supported.
- 1: Supported

DimmSpareSupported

If the BIOS supports PMem module sparing. One of:

- 0: Not supported
- 1: Supported

AppDirectMigrationSupported

If the BIOS supports App Direct migration. One of:

- 0: Not supported
- 1: Supported

RenameNamespaceSupported

If the host software supports renaming a namespace. One of:

- 0: Not supported
- 1: Supported

GrowAppDirectNamespaceSupported

If the host software supports increasing the capacity of an App Direct namespace. One of:

- 0: Not supported
- 1: Supported

ShrinkAppDirectNamespaceSupported

If the host software supports decreasing the capacity of an App Direct namespace. One of:

- 0: Not supported
- 1: Supported

InitiateScrubSupported

If the platform and host software support initiating an address range scrub on the PMem modules in the system. One of:

- 0: Not supported
- 1: Supported

AdrSupported

Whether the platform supports asynchronous DRAM refresh (ADR). One of:

- 0: Not supported. If ADR is not supported, App Direct data integrity cannot be assured during system interruptions.
- 1: Supported

EraseDeviceDataSupported

Whether Erase Device Data is supported.

- 0: Not supported
- 1: Supported

EnableDeviceSecuritySupported

Whether Enable Device Security is supported.

- 0: Not supported
- 1: Supported

DisableDeviceSecuritySupported

Whether Change Device Security property Lockstate = Disabled is supported.

- 0: Not supported
- 1: Supported

UnlockDeviceSecuritySupported

Whether Change Device Security property Lockstate = Unlocked is supported.

- 0: Not supported
- 1: Supported

FreezeDeviceSecuritySupported

Whether Change Device Security property Lockstate = Frozen is supported.

- 0: Not supported
- 1: Supported

ChangeDevicePassphraseSupported

Whether Change Device Passphrase is supported.

- 0: Not supported
- 1: Supported

ChangeMasterPassphraseSupported

Whether Change Master Passphrase is supported.

- 0: Not supported
- 1: Supported

MasterEraseDeviceDataSupported

Whether Master Erase Device Data is supported.

- 0: Not supported
- 1: Supported

2.2. Provisioning

Next generation memory controller and media technologies provide a number of opportunities to optimize a server's memory subsystem for a given workload. PMem module memory tiering and mode configuration are supported by the CLI and described in this section.

A system with both DRAM and PMem modules can be configured to use memory tiering. The CLI supports configuring a "near" performance tier and a "far" capacity tier, called Memory Mode. In this two level memory (2LM) system the near tier acts as a cache for the far tier. When memory tiering is enabled, DRAM DIMMs are utilized as near memory. The near memory tier utilizes all available DRAM capacity. The far memory tier is created from PMem modules. Because the capacity of a PMem module is so large (relative to DRAM) it may not be necessary to use all available PMem module capacity as far memory (a typical near to far ratio is 1:8). So in addition to enabling memory tiering, the CLI provides for designating the amount of PMem module capacity to be used as far memory. The remaining capacity can be used as App Direct persistent memory. Changing the tiering configuration dynamically on a running system is not supported. The CLI supports the creation of a tiering "goal configuration" request which will take effect after a system reboot. A valid goal request results in a combination of Memory Mode capacity and/or one or more persistent memory regions after it is successfully parsed and activated by BIOS on the next boot. See the command [Create Memory Allocation Goal](#) for more details on configuring the system for memory tiering.

Since modifying the memory tiering and persistent memory configuration requires a reboot to take affect it is possible to have both a current configuration and a pending "goal" configuration. Once the system is rebooted the goal configuration is applied and becomes the current configuration. Prior to rebooting it is possible to see the goal configuration using the command [Show Memory Allocation Goal](#). It is also possible to delete a goal prior to the reboot. See the command [Delete Memory Allocation Goal](#) for more details.

NOTE

Changing a memory allocation goal modifies how the platform firmware maps persistent memory in the System Physical Address space (SPA) which may result in data loss or inaccessible data, but does not explicitly delete or modify user data found in persistent memory.

2.2.1. Create Memory Allocation Goal

NAME

ipmctl-create-goal - Creates a memory allocation goal on one or more PMem modules

SYNOPSIS

```
ipmctl create [OPTIONS] -goal [TARGETS] [PROPERTIES]
```

DESCRIPTION

Creates a memory allocation goal on one or more for the BIOS to read on the next reboot in order to map the PMem module capacity into the system address space. Persistent memory can then be utilized by creating a namespace.

NOTE	The capacity values presented by this command are a target goal or request to platform firmware. The actual capacity values are subject to change due to rounding and alignment requirements. If the goal request is invalid or not possible it may be rejected by platform firmware.
-------------	---

NOTE	Deleting the PCD can be used as a way to prepare individual PMem modules for provisioning. See the delete -pcd command.
-------------	---

WARNING	This command may result in data loss. Data should be backed up to other storage before executing this command.
----------------	---

NOTE	Changing a memory allocation goal modifies how the platform firmware maps persistent memory in the System Physical Address space (SPA) which may result in data loss or inaccessible data, but does not explicitly delete or modify user data found in persistent memory.
-------------	---

OPTIONS

-f

-force

Reconfiguring PMem modules is a destructive operation which requires confirmation from the user. This option suppresses the confirmation. This option can also be used to recover/override corrupted Platform Configuration Data (PCD). The force flag will also suppress the security enabled warning as well as all other warning prompts.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml". The "nvmxml" format implies the "-force" flag.

-u (B|MB|MiB|GB|GiB|TB|TiB)

-units (B|MB|MiB|GB|GiB|TB|TiB)

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS

-dimm [DimmIDs]

Creates a memory allocation goal on specific PMem modules by optionally supplying one or more comma-separated PMem module identifiers. This list must meet one of the following criteria:

1. Target all PMem modules on a given socket.
 - For example, all PMem modules located within a single socket may be specified together using the dimm target.
2. Target all unconfigured PMem modules.
 - For example, when PMem modules have had their PCD deleted, then these PMem modules may all be specified together using the dimm target.
3. Target PMem modules for 100% MemoryMode with all unspecified PMem modules configured for MemoryMode only.
 - For example, when existing PMem modules are 100% MM, then any newly added PMem modules may all be specified together using the dimm target and configured as 100% MM to match the existing configuration.
4. Target PMem modules for App Direct Not-Interleaved with all unspecified PMem modules configured for App Direct Not-Interleaved only
 - For example, when existing PMem modules are 100% App Direct Not-

Interleaved, then any newly added PMem modules may all be specified together using the dimm target and configured as 100% App Direct Not-Interleaved to match the existing configuration.

NOTE

If the dimm target is not specified, the default is to configure all manageable PMem modules on all sockets.

-socket [SocketIds]

Loads the memory allocation goal onto all manageable PMem modules on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to load the memory allocation goal onto all manageable PMem modules on all sockets.

PROPERTIES**MemoryMode**

Percentage of the total capacity to use in Memory Mode (0-100). Default = 0.

PersistentMemoryType

If MemoryMode is not 100%, the type of persistent memory to create.

- "AppDirect": (Default) Create App Direct capacity utilizing hardware interleaving across the requested PMem modules if applicable given the specified target.
- "AppDirectNotInterleaved": Create App Direct capacity that is not interleaved any other PMem modules.

NamespaceLabelVersion

The version of the namespace label storage area (LSA) index block

- "1.2": (Default) Defined in UEFI 2.7a - sections 13.19
- "1.1": Legacy 1.1 namespace label support

Reserved

Reserve a percentage (0-100) of the requested PMem module App Direct capacity that will not be mapped into the system physical address space and will be presented as Reserved Capacity with Show Device and Show Memory Resources Commands.

EXAMPLES

Configures all the PMem module capacity in Memory Mode.

```
ipmctl create -goal MemoryMode=100
```

Configures all the PMem module capacity as App Direct.


```
ipmctl create -goal PersistentMemoryType=AppDirect
```

Configures the capacity on each PMem module with 20% of the capacity in Memory Mode and the remaining as App Direct capacity that does not use hardware interleaving.

```
ipmctl create -goal MemoryMode=20 PersistentMemoryType=AppDirectNotInterleaved
```

Configures the PMem module capacity across the entire system with 25% of the capacity in Memory Mode, 25% reserved and the remaining 50% as App Direct. Configures the PMem module capacity across the entire system with 25% of the capacity in Memory Mode and the remaining 75% as App Direct.

```
ipmctl create -goal MemoryMode=25 PersistentMemoryType=AppDirect Reserved=25
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software and must all have the same SKU.
- SKU based maximum total mapped memory is enforced. See section [\[Maximum Mapped Memory Limiting\]](#).
- If no PMem modules are specified, all PMem modules in a non-functional or unmanageable state will not be included in the goal's creation.
- If the goal request is for an interleaved set, PMem modules in population violation will not be included in the goal's creation.
- Existing memory allocation goals that have not been applied and any namespaces associated with the requested PMem modules must be deleted before running this command.
- Goal requests may not be applied by platform firmware (BIOS) if the PMem module is in security enabled, locked state.

NOTE

It is recommended to disable security prior to reboot if requesting a new goal.

NOTE

A goal request may be initiated even if a target PMem module is in security state enabled, but care must be taken to ensure the PMem module is in either unlocked or disabled security state prior to the platform firmware (BIOS) provisioning flow following a reboot. In addition, a warning will be presented to the user: 'WARNING: Goal will not be applied unless security is disabled prior to platform firmware (BIOS) provisioning!'

- Changing the memory configuration is a destructive operation which results in loss of data stored in the persistent memory region. Therefore, data should be backed up to other storage before executing this command. Targets may be limited to individual PMem modules or sockets, but all PMem modules on affected sockets must be configured when the command finishes. If the selected targets make this impossible, the command will be rejected. Refer to **Show System Capabilities** for a list of BIOS supported modes.
- Some requests are dependent on BIOS and/or platform configuration. For details, refer to the *Intel® Optane™ Persistent Memory Software Memory Allocation Rules*, document number 564194. For example:
- Provisioning PMem modules for Memory Mode while BIOS is configured for 1LM only will result in unused capacity.
- Provisioning PMem modules for Memory Mode while not all iMCs have at least one PMem module will result in unused capacity.
- PMem modules in population violation can be targeted for goal creation only if PersistentMemoryType=AppDirectNotInterleaved and no volatile memory is requested (100% ADx1).

RETURN DATA

Minor adjustments (up to 10%) in the requested capacities are sometimes necessary to align properly according to the platform rules. There are also some situations that require additional confirmation from the user because they may result in a non-optimal configuration (i.e., reduced performance). These are described below:

The requested goal may result in a non-optimal configuration due to the population of PMem modules in the system.

Memory Mode capacity requested but the population of DRAM DIMMs and PMem modules in the system may result in reduced performance (i.e., the ratio of DRAM and PMem modules is not balanced, DRAM and PMem modules are not on the same channel or not all the same size).

The requested goal may result in a non-optimal configuration due to the population of PMem modules in the system.

App Direct capacity requested but the population of PMem modules in the system may result in reduced performance (i.e., PMem modules are not the same size or populated asymmetrically across the socket).

The requested goal will result in App Direct capacity which is not supported by the host software.

App Direct capacity requested but App Direct is not supported by the currently installed host software.

The requested goal will result in Memory Mode capacity that is unusable with the currently selected platform BIOS volatile mode.

Memory Mode capacity requested by the platform BIOS is currently set to 1LM Mode.

The requested goal was adjusted more than 10% to find a valid configuration.

> 10% adjustment from the requested goal

The amount of mapped memory was limited based on the SKU resulting in unmapped capacity.

Mapped memory was limited based on the CPU SKU.

Therefore, before making any changes to the configuration, a prompt is displayed showing the memory allocation goals that will be created on each PMem module as documented in the command Section [Show Memory Allocation Goal](#), along with any additional confirmation messages. The force option can be used to override this confirmation and proceed directly with creating the goals.

```
The following configuration will be applied:
SocketID DimmID MemorySize AppDirect1Size AppDirect2Size (Refer to
the command Section <<Show Memory Allocation Goal>>)
[Additional Confirmation Messages (see above)] Do you want to
continue?
```

2.2.2. Show Memory Allocation Goal

NAME

ipmctl-show-goal - Shows the memory allocation goal on one or more PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -goal [TARGETS] [PROPERTIES]
```

DESCRIPTION

Shows the memory allocation goal on one or more PMem modules. Once the goal is successfully applied by the platform firmware (BIOS), it is no longer displayed. Use the command [Show Memory Resources](#) to view the system-wide memory resources or the command [Show Persistent Memory](#) for detailed persistent memory information.

OPTIONS

-a

-all

Shows all attributes.

NOTE | The all and display options are exclusive and may not be used together.

-d (attributes)

-display (attributes)

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

NOTE | The all and display options are exclusive and may not be used together.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE | The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text | nvmlxml)

-output (text | nvmlxml)

Changes the output format. One of: "text" (default) or "nvmlxml".

-u (B | MB | MiB | GB | GiB | TB | TiB)

-units (B | MB | MiB | GB | GiB | TB | TiB)

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS

-dimm [DimmIDs]

Restricts output to specific PMem modules by supplying one or more comma separated PMem module identifiers. The default is to display all manageable PMem modules with

memory allocation goals.

-socket [SocketIds]

Restricts output to the PMem modules on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to display all manageable PMem modules on all sockets with memory allocation goals.

EXAMPLES

Shows the default memory allocation goal attributes for each PMem module.

```
ipmctl show -goal
```

Shows all the memory allocation goal attributes for the PMem modules on socket 1.

```
ipmctl show -a -goal -socket 1
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

The default behavior is to display a table with the default attributes for each PMem module; applying options changes the output to a more detailed format.

SocketID

(Default) The processor socket identifier where the PMem module is installed.

DimmID

(Default) The PMem module identifier

MemorySize

(Default) The PMem module capacity that will be configured in Memory Mode.

AppDirect1Size

(Default) The PMem module capacity that will be configured as the first App Direct interleave set if applicable.

AppDirect1Index

Unique identifier of the first App Direct interleave set.

- N/A: If no App Direct interleave set

- Numeric value if App Direct Interleave set is present.

AppDirect1Settings

The settings for the first App Direct Interleave set in the format: x(Way) [- (Size) iMC] [x (Size) Channel]

AppDirect2Size

(Default) The PMem module capacity that will be configured as the second App Direct Interleave set if applicable.

AppDirect2Index

Unique identifier of the second App Direct Interleave set.

- N/A: If no App Direct Interleave set
- Numeric value if App Direct Interleave set is present.

AppDirect2Settings

The settings for the second App Direct Interleave set in the format: x(Way) [- (Size) iMC] [x (Size) Channel]

Status

The status of the memory allocation goal. One of:

- Unknown: The status cannot be determined.
- New: A reboot is required for the memory allocation goal to be processed by the BIOS.
- Failed - Bad request: The BIOS failed to process the memory allocation goal because it was invalid.
- Failed - Not enough resources: There were not enough resources for the BIOS to process the memory allocation goal.
- Failed - Firmware error: The BIOS failed to process the memory allocation goal due to a firmware error.
- Failed - Unknown: The BIOS failed to process the memory allocation goal due to an unknown error.

SAMPLE OUTPUT

If a new memory allocation goal has been created, a prompt to reboot will be presented.

```
A reboot is required to process new memory allocation goals.
```

2.2.3. Dump Memory Allocation Settings

NAME

ipmctl-dump-goal - Stores the current configured memory allocation settings to a file

SYNOPSIS

```
ipmctl dump [OPTIONS] -destination (path) -system -config
```

DESCRIPTION

Store the currently configured memory allocation settings for all PMem modules in the system to a file in order to replicate the configuration elsewhere. Apply the stored memory allocation settings using the command Section [Load Memory Allocation Goal](#).

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text | nvmlxml)

-output (text | nvmlxml)

Changes the output format. One of: "text" (default) or "nvmlxml".

EXAMPLES

Stores the memory allocation settings from all the PMem modules into the file "config.txt".

```
ipmctl dump -destination config.txt -system -config
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- Only memory allocation settings for manageable PMem modules that have been successfully applied by the BIOS are stored in the file. Unconfigured PMem modules are not included, nor are memory allocation goals that have not been applied.

RETURN DATA

The CLI will indicate the overall status of the operation when complete. If a failure occurs when dumping the memory allocation from multiple PMem modules, the process will stop and the output file will be removed.

The output file is formatted as an ASCII file with one row per PMem module containing the following comma-separated values.

SocketID

Identifier for the socket the PMem module is associated with.

DimmHandle

PMem module device handle.

Capacity

Total capacity of the PMem module in GiB.

MemorySize

Capacity of the PMem module allocated as Memory Mode in GiB.

AppDirect1Size

Capacity of the PMem module allocated for the first App Direct interleave set in GiB.

AppDirect1Format

Bit mask representing the interleave format of the first App Direct interleave set.

AppDirect1Mirrored

State indicating if the first App Direct interleave set is mirrored.

- 0 - The first App Direct interleave set is not mirrored.
- 1 - The first App Direct interleave set is mirrored.

AppDirect1Index

Unique index of the first App Direct interleave set.

AppDirect2Size

Capacity of the PMem module allocated for the second App Direct interleave set in GiB.

AppDirect2Format

Bit mask representing the interleave format of the second App Direct interleave set.

AppDirect2Mirrored

State indicating if the second App Direct interleave set is mirrored.

- 0 - The second App Direct interleave set is not mirrored.
- 1 - The second App Direct interleave set is mirrored.

AppDirect2Index

Unique index of the second App Direct interleave set.

SAMPLE OUTPUT

Successfully dumped system configuration to file: config.csv

config.csv contents:

```
#SocketID,DimmHandle,Capacity,MemorySize,AppDirect1Size,AppDirect
1Format,AppDirect
1Mirrored,AppDirect1Index,AppDirect2Size,AppDirect2Format,AppDire
ct2Mirrored,AppDirect2Index
1,4385,64,64,0,0,0,0,0,0,0,0
1,4401,64,64,0,0,0,0,0,0,0,0
1,4417,64,64,0,0,0,0,0,0,0,0
1,4433,64,64,0,0,0,0,0,0,0,0
1,4449,64,64,0,0,0,0,0,0,0,0
```

2.2.4. Load Memory Allocation Goal

NAME

ipmctl-load-goal - Creates a memory allocation goal request from a file

SYNOPSIS

```
ipmctl load [OPTIONS] -source (path) -goal [TARGETS]
```

DESCRIPTION

Creates a memory allocation goal request from a file onto one or more PMem modules.

NOTE	Deleting the PCD can be used as a way to prepare individual PMem modules for provisioning. See the delete -pcd command.
-------------	---

WARNING	This command may result in data loss. Data should be backed up to other storage before executing this command.
----------------	---

NOTE	Changing a memory allocation goal modifies how the platform firmware maps persistent memory in the System Physical Address space (SPA) which may result in data loss or inaccessible data, but does not explicitly delete or modify user data found in persistent memory.
-------------	---

OPTIONS

-f

-force

Reconfiguring PMem modules is a destructive operation which requires confirmation from the user. This option suppresses the confirmation. The force flag will also suppress the security enabled warning as well as all other warning prompts.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE	The -ddrt and -smbus options are mutually exclusive and may not be used together.
-------------	---

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml". The "nvmxml" format implies the "-force" flag.

-u (B|MB|MiB|GB|GiB|TB|TiB)**-units (B|MB|MiB|GB|GiB|TB|TiB)**

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS**-dimm [DimmIDs]**

Loads the memory allocation goal onto specific PMem modules by supplying one or more comma separated PMem module identifiers. This list must include all unconfigured PMem modules on the affected sockets. The default is to load the memory allocation goal onto all manageable PMem modules.

-socket [SocketIds]

Loads the memory allocation goal onto all manageable PMem modules on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to load the memory allocation goal onto all manageable PMem modules on all sockets.

EXAMPLES

Loads the configuration settings stored in "config.txt" onto all the PMem modules in the system as a memory allocation goal to be applied by the BIOS on the next reboot.

```
ipmctl load -source config.txt -goal
```

Loads the configuration settings stored in "config.txt" onto a specified set of PMem modules as a memory allocation goal to be applied by the BIOS on the next reboot.

```
ipmctl load -source config.txt -goal -dimm 1,2,3
```

Loads the configuration settings stored in "config.txt" onto all manageable PMem modules on sockets 1 and 2 as a memory allocation goal to be applied by the BIOS on the next reboot.

```
ipmctl load -source config.txt -goal -socket 1,2
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software and must all have the same SKU.
- SKU based maximum total mapped memory is enforced. See section [\[Maximum Mapped Memory Limiting\]](#).
- Existing memory allocation goals that have not been applied and any namespaces associated with the requested PMem modules must be deleted before running this command.
- Goal requests may not be applied by platform firmware (BIOS) if the PMem module is in security enabled, locked state.

NOTE

It is recommended to disable security prior to reboot if requesting a new goal.

NOTE

A goal request may be initiated even if a target PMem module is in security state enabled, but care must be taken to ensure the PMem module is in either unlocked or disabled security state prior to the platform firmware (BIOS) provisioning flow following a reboot. In addition, a warning will be presented to the user: 'WARNING: Goal will not be applied unless security is disabled prior to platform firmware (BIOS) provisioning!'

- Changing the memory configuration is a destructive operation which results in loss of data stored in the persistent memory region. Therefore, data should be backed up to other storage before executing this command. Targets may be limited to individual PMem modules or sockets, but all PMem modules on affected sockets must be configured when the command finishes. If the selected targets make this impossible, the command will be rejected. Refer to **Show System Capabilities** for a list of BIOS supported modes.
- Some requests are dependent on BIOS and/or platform configuration. For details, refer

to the *Intel® Optane™ Persistent Memory Software Memory Allocation Rules*, document number 564194. For example:

- Provisioning PMem modules for Memory Mode while BIOS is configured for 1LM only will result in unused capacity.
- Provisioning PMem modules for Memory Mode while not all iMCs have at least one PMem module will result in unused capacity.

RETURN DATA

If successful, the CLI will display the memory allocation goal stored on each PMem module as documented in the command Section [Show Memory Allocation Goal](#). If a failure occurs, an error code and message will be displayed. If a failure occurs when configuring multiple PMem modules, the process will exit and remove the memory allocation goal from any PMem modules that succeeded prior to the failure.

2.2.5. Delete Memory Allocation Goal

NAME

ipmctl-delete-goal - Deletes the memory allocation goal from PMem modules

SYNOPSIS

```
ipmctl delete [OPTIONS] -goal [TARGETS]
```

DESCRIPTION

Deletes the memory allocation goal from one or more PMem modules. This command only deletes a memory allocation goal request that has not yet been processed by platform firmware (BIOS).

If the PMem module target is used and the specified PMem modules do not include all PMem modules that are part of a memory allocation goal request the result will be PMem modules (those not included) that have a broken request which will be rejected by platform firmware (BIOS) upon the next reboot.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS**-dimm [DimmIDs]**

Deletes the memory allocation goal from specific PMem modules by optionally supplying one or more comma-separated PMem module identifiers. The default is to delete the memory allocation goals from all manageable PMem modules.

-socket [SocketIds]

Deletes the memory allocation goal from the PMem modules on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to delete the memory allocation goals from manageable PMem modules on all sockets.

EXAMPLES

Deletes the memory allocation goal from all PMem modules on all sockets.

```
ipmctl delete -goal
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software and unlocked

if security is enabled.

- Given socket and all specified PMem modules must contain a memory allocation goal.

RETURN DATA

For each PMem module, the CLI will indicate the status of the operation. If a failure occurs when deleting the memory allocation goals from multiple PMem modules, the process will output a failure message for those PMem modules that did not succeed and a success message for those that did.

SAMPLE OUTPUT

```
Delete memory allocation goal from PMem module (DimmID): Success
```

```
Delete memory allocation goal from PMem module (DimmID): Error (Code) -  
(Description)
```

2.3. Instrumentation

2.3.1. Show Sensor

NAME

ipmctl-show-sensor - Shows sensor data for one or more PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -sensor [SENSORS] [TARGETS]
```

DESCRIPTION

Shows sensor data for one or more PMem modules. This data includes PMem module health, temperatures, percentage remaining, up-time and more.

OPTIONS

-a

-all

Shows all attributes.

NOTE

The all and display options are exclusive and may not be used together.

-d (attributes)**-display (attributes)**

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

NOTE	The all and display options are exclusive and may not be used together.
-------------	---

-h**-help**

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE	The -ddrt and -smbus options are mutually exclusive and may not be used together.
-------------	---

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

SENSORS

Health

(Default) The current PMem module health as reported in the SMART log. Attributes include:

- DimmID
- Type
- CurrentValue

MediaTemperature

(Default) The current PMem module media temperature in Celsius. Attributes include:

- DimmID
- Type
- CurrentValue
- AlarmThreshold

- AlarmEnabled
- ThrottlingStopThreshold
- ThrottlingStartThreshold
- ShutdownThreshold
- MaxTemperature

ControllerTemperature

(Default) The current PMem module controller temperature in Celsius. Attributes include:

- DimmID
- Type
- CurrentValue
- AlarmThreshold
- AlarmEnabled
- ThrottlingStopThreshold
- ThrottlingStartThreshold
- ShutdownThreshold
- MaxTemperature

PercentageRemaining

(Default) Remaining PMem module's life as a percentage value of factory expected life span. Attributes include:

- DimmID
- Type
- CurrentValue
- AlarmThreshold
- AlarmEnabled

LatchedDirtyShutdownCount

(Default) The number of shutdowns without notification over the lifetime of the PMem module. Attributes include:

- DimmID
- Type
- CurrentValue

UnlatchedDirtyShutdownCount

(Default) The number of shutdowns without notification over the lifetime of the PMem

module. This counter is the same as LatchedDirtyShutdownCount except it will always be incremented on a dirty shutdown, even if Latch System Shutdown Status was not enabled. Attributes include:

- DimmID
- Type
- CurrentValue

PowerOnTime

(Default) The total power-on time over the lifetime of the PMem module. Attributes include:

- DimmID
- Type
- CurrentValue

UpTime

(Default) The total power-on time since the last power cycle of the PMem module. Attributes include:

- DimmID
- Type
- CurrentValue

PowerCycles

(Default) The number of power cycles over the lifetime of the PMem module. Attributes include:

- DimmID
- Type
- CurrentValue

FwErrorCount

(Default) The total number of firmware error log entries. Attributes include:

- DimmID
- Type
- CurrentValue

TARGETS

-dimm [DimmIDs]

Restricts output to the sensors on specific PMem modules by supplying the DIMM target and one or more comma-separated PMem module identifiers. The default is to

display sensors for all manageable PMem modules.

EXAMPLES

Gets all sensor information for all PMem modules.

```
ipmctl show -sensor
```

Shows the media temperature sensor for the specified PMem module.

```
ipmctl show -sensor MediaTemperature -dimm 0x0001
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

This command displays a table with a row for each sensor on each PMem module. Applying a specific PMem module or sensor type target limits the number of rows. Applying options can be used to expand or restrict the output.

NOTE

Supported thresholds for a given sensor are listed as a part of the return data when using the -all or -display flag. Only the alarm threshold is settable.

DimmID

(Default) The PMem module identifier

Type

(Default) The sensor type. Refer to the sensor table above.

CurrentValue

(Default) The current reading followed by the units of measurement (e.g., 57 °C or 25%)

AlarmThreshold

The threshold value at which an alarm for the respective sensor will be triggered.

AlarmEnabled

Current state of the alarm threshold, where applicable. One of:

- 0: Disabled

- 1: Enabled
- N/A

ThrottlingStopThreshold

The threshold value at which firmware will stop throttling to reduce overall thermals and keep the PMem module within safe operating temperatures.

ThrottlingStartThreshold

The threshold value at which firmware will start throttling to reduce overall thermals and keep the PMem module within safe operating temperatures.

ShutdownThreshold

The threshold value at which device shutdown will occur.

MaxTemperature

The highest temperature reported in degrees Celsius for a given media or controller sensor. This value is persistent through Power Loss and is read-only.

2.3.2. Change Sensor Settings

NAME

ipmctl-change-sensor - Changes the threshold or enabled state for PMem modules sensors

SYNOPSIS

```
ipmctl set [OPTIONS] -sensor (SENSORS) [TARGETS]  
AlarmThreshold=(temperature) AlarmEnabled=(0|1)
```

DESCRIPTION

Changes the alarm threshold or enabled state for one or more PMem modules sensors. Use the command Show Sensor to view the current settings.

OPTIONS

-f

-force

Changing the sensor settings is a potentially destructive operation which requires confirmation from the user for each PMem module. This option suppresses the confirmation.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

SENSORS

MediaTemperature

The PMem module media temperature in Celsius.

ControllerTemperature

The PMem module controller temperature in Celsius.

PercentageRemaining

Remaining PMem module's life as a percentage value of factory expected life span.

TARGETS

-dimm [DimmIDs]

Update specified sensors on specific PMem modules by optionally supplying the PMem module target and one or more comma-separated PMem module identifiers. The default is to update the specified sensors for all manageable PMem modules.

PROPERTIES

AlarmThreshold

The threshold value at which an alarm for the respective sensor will be triggered. The upper (for temperatures) or lower (for percentage remaining) alarm threshold of the sensor. Temperatures may be specified to a precision of 1 degree Celsius.

Table 1. Allowed AlarmThreshold

Sensor	Allowed AlarmThreshold values	Units
MediaTemperature	0-2047	Celsius
ControllerTemperature	0-2047	Celsius

Sensor	Allowed AlarmThreshold values	Units
PercentageRemaining	1-99	%

AlarmEnabled

Enable or disable the alarm threshold, where applicable. One of:

- "0": Disable
- "1": Enable

EXAMPLES

Changes the media temperature alarm threshold to 51C on the specified PMem module and enables the alarm.

```
ipmctl set -sensor MediaTemperature -dimm 0x0001 AlarmThreshold=51
AlarmEnabled=1
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

For each PMem module, the CLI will indicate the status of the operation. If a failure occurs when modifying multiple PMem modules, the process will exit and not continue modifying the remaining PMem modules.

SAMPLE OUTPUT

```
Modify (Sensor) settings on DIMM (DimmID): Success
```

```
Modify (Sensor) settings on DIMM (DimmID): Error (Code) -
(Description)
```

2.3.3. Show Device Performance

NAME

ipmctl-show-performance - Shows performance metrics for one or more PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -performance [METRICS] [TARGETS]
```

DESCRIPTION

Shows performance metrics for one or more PMem modules.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

METRICS

Restricts output to a specific performance metric by supplying the metric name. See RETURN DATA for more information. One of:

- MediaReads
- MediaWrites
- ReadRequests
- WriteRequests
- TotalMediaReads
- TotalMediaWrites

- TotalReadRequests
- TotalWriteRequests

The default is to display all performance metrics.

TARGETS

-dimm [DimmIDs]

Restricts output to the performance metrics for specific PMem module by supplying one or more comma separated PMem module identifiers. The default is to display performance metrics for all manageable PMem module.

EXAMPLES

Shows all performance metrics for all PMem modules in the server.

```
ipmctl show -dimm -performance
```

Shows the number of 64 byte reads since last AC cycle for all PMem modules in the server.

```
ipmctl show -dimm -performance MediaReads
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

This command displays a table of the specified metrics for each specified PMem module. Applying a specific PMem module target limits the rows in the table. Applying a specific metric name target limits the columns in the table.

DimmID

The PMem module identifier

MediaReads

Number of 64 byte reads from media on the PMem module since last AC cycle.

MediaWrites

Number of 64 byte writes to media on the PMem module since last AC cycle.

ReadRequests

Number of DDRT read transactions the PMem module has serviced since last AC cycle.

WriteRequests

Number of DDRT write transactions the PMem module has serviced since last AC cycle.

TotalMediaReads

Number of 64 byte reads from media on the PMem module over its lifetime.

TotalMediaWrites

Number of 64 byte writes to media on the PMem module over its lifetime.

TotalReadRequest

Number of DDRT read transactions the PMem module has serviced over its lifetime.

TotalWriteRequest

Number of DDRT write transactions the PMem module has serviced over its lifetime.

2.4. Persistent Memory Provisioning

During the initial [Provisioning](#) of the PMem modules it is possible to allocate some or all of the available PMem module capacity as persistent memory. If this was done, the result is one or more regions of persistent memory. The number of persistent memory regions depends on the platform configuration (e.g., is it a 2 socket or a 4 socket server) and the type of initial provisioning request.

Persistent memory provisioning is the process by which the capacity in available regions is partitioned and made available as a persistent memory device or "namespace" in the file system. Due to a number of factors there may be restrictions on the type, size and number of namespaces that can be created from a given region. The output of the command [Show Persistent Memory](#) includes information about the persistent memory regions. FreeCapacity indicates the available capacity of a given region that can be used to create one ore more new namespaces. A FreeCapacity value of zero indicates that it is not possible to create a namespace from that region.

NOTE

If PMem module is locked, then namespaces cannot be detected or displayed.

NOTE

Namespace management is not supported via ipmctl CLI in the Operating System (OS) environment. Native OS utilities must be used for namespace management. See the *Intel® Optane™ Persistent Memory Help and User's Guide* (document number 576369) for additional details on using the native OS utilities.

2.4.1. Show Persistent Memory

NAME

ipmctl-show-region - Retrieves a list of persistent memory regions.

SYNOPSIS

```
ipmctl show [OPTIONS] -region [TARGETS]
```

DESCRIPTION

Retrieves a list of persistent memory regions of PMem module capacity.

OPTIONS

-a

-all

Shows all attributes.

NOTE | The all and display options are exclusive and may not be used together.

-d (attributes)

-display (attributes)

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

NOTE | The all and display options are exclusive and may not be used together.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE | The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-nfit

Used to specify NFIT table as the source instead of PCD (default) for the current invocation of ipmctl.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

-u (B|MB|MiB|GB|GiB|TB|TiB)**-units (B|MB|MiB|GB|GiB|TB|TiB)**

Changes the units that capacities are displayed in for this command. One of: bytes (B), megabytes (MB), mebibytes (MiB), gigabytes (GB), gibibytes (GiB), terabytes (TB) or tebibytes (TiB).

TARGETS**-region [RegionIDs]**

Restricts output to specific persistent memory regions by providing one or more comma separated region identifiers. The default is to display the persistent memory regions across all manageable PMem modules.

-socket [SocketIDs]

Restricts output to the persistent memory regions on specific sockets by supplying the socket target and one or more comma-separated socket identifiers. The default is to display all sockets.

EXAMPLES

Shows all attributes of all persistent memory regions in the server.

```
ipmctl show -a -region
```

Shows all attributes for the specified persistent memory region.

```
ipmctl show -a -region 1
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- All the underlying PMem modules should be unlocked to accurately reflect the available capacities. The specified PMem modules must be manageable by the host software.

RETURN DATA

The default behavior is to display a table with the default attributes listed below; applying options changes the output to a more detailed format.

ISetID

(Default) The region unique identifier. Also known as interleave set cookie.

PersistentMemoryType

(Default) A comma-separated list of the underlying types of persistent memory capacity in the region. One or more of:

- AppDirect: App Direct capacity interleaved across two or more PMem modules that is fully mapped into the system physical address space.
- AppDirectNotInterleaved: App Direct capacity wholly contained on a single PMem modules that is fully mapped into the system physical address space.

Capacity

(Default) Total usable capacity, both allocated and unallocated

FreeCapacity

(Default) Remaining usable capacity

SocketID

(Default) Socket ID to which the region belongs

HealthState

The rolled up health of the underlying PMem modules. One of:

- Unknown: The region health cannot be determined.
- Healthy: All underlying PMem module persistent memory capacity is available.
- Pending: A new memory allocation goal has been created but not applied. Reboot or delete any existing memory allocation goals before creating namespaces on the region.
- Error: There is an issue with some or all of the underlying PMem module capacity because the interleave set has failed. One of: + PMem module missing (serial number of PMem module from the Platform Config Data not found in the PMem

module list) + PMem module not configured + Failure to retrieve AppDirect I/O structures from NFIT (system physical address missing) + Interleave set or PMem module region initialization failure (out of memory) + Locked: One or more of the of the underlying Pmem modules are locked.

DimmID

A list of all the PMem modules that are part of this reg.

2.5. Support and Maintenance

2.5.1. Help

NAME

ipmctl-help - Shows help for the supported commands

SYNOPSIS

```
ipmctl help [OPTIONS]
```

DESCRIPTION

Shows help for the supported commands.

OPTIONS

-h

-help

Displays help for the command.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

EXAMPLES

Lists all supported commands

```
ipmctl help
```

RETURN DATA

The default behavior is to display an introduction to CLI followed by a list of the

supported commands. To display detailed help for a specific command, use the help option with that specific command.

SAMPLE OUTPUT

```
ipmctl help
Commands:
(command synopsis)
(command syntax)
\...
(command synopsis)
(command syntax)
```

2.5.2. Version

NAME

ipmctl-version - Shows the PMem module host software versions

SYNOPSIS

```
ipmctl version [OPTIONS]
```

DESCRIPTION

Shows the PMem module host software versions.

OPTIONS

-h

-help

Displays help for the command.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

EXAMPLES

Displays the available version information for the PMem module software components.

```
ipmctl version
```

RETURN DATA

By default returns the following inventory information.

Component

The name of the software component. One of:

- [Product Name] Software Version: The PMem module management software version

Version

The current version of the software component if found or an error if not.

NOTE

If the software version is incompatible, the version will be followed by an error message indicating such. If PMem modules are found with a FIS implementation higher than supported by the SW version, this command will print a warning.

2.5.3. Show Device Firmware

NAME

ipmctl-show-firmware - Shows detailed information about the firmware

SYNOPSIS

```
ipmctl show [OPTIONS] -firmware [TARGETS]
```

DESCRIPTION

Shows detailed information about the firmware on one or more PMem modules.

OPTIONS

-a

-all

Shows all attributes.

NOTE

The all and display options are exclusive and may not be used together.

-d (attributes)

-display (attributes)

Filters the returned attributes by explicitly specifying a comma-separated list of any of the attributes defined in the Return Data section.

NOTE The all and display options are exclusive and may not be used together.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS

-dimm [DimmIDs]

Restricts output to the firmware information for specific PMem modules by supplying one or more comma-separated PMem module identifiers. The default is to display the firmware information for all manageable PMem modules.

EXAMPLES

Shows the firmware information for all PMem modules in the server.

```
ipmctl show -dimm -firmware
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

The default behavior is to display a table with the default attributes listed below; the options can be used to expand or restrict the output.

DimmID

(Default) The PMem module identifier

ActiveFWVersion

(Default) The BCD-formatted revision of the active firmware in the format PN.RN.SV.bbbb where:

- PN = 2-digit product number
- RN = 2-digit revision number
- SV = 2-digit security version number
- bbbb = 4-digit build version

StagedFWVersion

(Default) The BCD-formatted revision of the firmware staged for execution on the next power cycle in the format PN.RN.SV.bbbb where:

- PN = 2-digit product number
- RN = 2-digit revision number
- SV = 2-digit security version number
- bbbb = 4-digit build version

StagedFWActivatable

The state of whether the staged firmware is activatable or not, where:

- 0 = Not activatable, reboot is required
- 1 = Activatable

FWUpdateStatus

The status of the last firmware update operation. One of:

- Unknown
- Staged successfully
- Update loaded successfully
- Update failed to load, fell back to previous firmware

FWImageMaxSize

The maximum size of a firmware image.

QuiesceRequired

The state of whether activating the firmware requires the host to quiesce traffic prior to calling the command.

- 0 = No traffic quiesce required prior to activate

- 1 = Traffic quiesce required prior to activate

ActivationTime

The estimated activation time, in ms, required for activating the firmware.

2.5.4. Update Firmware

NAME

ipmctl-update-firmware - Updates the firmware on one or more PMem modules

SYNOPSIS

```
ipmctl load [OPTIONS] -source (path) -dimm [TARGETS]
```

DESCRIPTION

Updates the firmware on one or more PMem modules. On the next power cycle, the firmware will become active.

NOTE	If Address Range Scrub (ARS) is in progress on any target PMem module, an attempt will be made to abort ARS and then proceed with the firmware update.
NOTE	A power cycle reboot is required to activate the updated firmware image and is recommended to ensure ARS runs to completion.

OPTIONS

-x

-examine

Verifies the target PMem modules are compatible and ready to receive the firmware image specified in the source option. Returns the firmware image version.

-f

-force

Downgrading the firmware to an older version is a potentially destructive operation which requires confirmation from the user for each PMem module. This option suppresses the confirmation when attempting to downgrade.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-source

Specifies the firmware image binary to upload to the PMem module.

-recover**--DEPRECATED--**

This flag is no longer necessary to run firmware update on PMem modules where the DDRT link is not trained. These untrained PMem modules are now automatically included when the command is run without the '-recover' option.. However, this flag is still maintained for backwards compatibility.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS**-dimm [DimmIDs]**

Updates the firmware on specific PMem modules by supplying one or more comma-separated PMem module identifiers. However, this is not recommended as it may put the system in an undesirable state. The default is to update all manageable PMem modules.

EXAMPLES

Updates the firmware on all PMem modules in the system to the image in sourcefile.bin on the next power cycle.

```
ipmctl load -source sourcefile.bin -dimm
```

Checks the firmware image in sourcefile.bin and retrieves the version.

```
ipmctl load -examine -source sourcefile.bin -dimm
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

Firmware version (PN.RN.SV.bbbb) updates are supported as follows:

- The product number (PN) cannot be changed.
- The revision number (RN) can be upgraded when PN is the same.
- The security revision number (SV) can be upgraded when PN.RN is the same. In some configurations it can also be downgraded when PN.RN is the same; use the examine option to determine if the security revision number can be downgraded.
- The build number (bbbb) can be upgraded or downgraded. However, if the firmware API version in the firmware image is lower than is supported by the host software and would make the PMem module become unmanageable, the downgrade is not supported.

NOTE

Once a firmware image is staged for execution, a power cycle is required before another firmware image of the same type (production or debug) can be staged for execution using this command.

RETURN DATA

When the examine option is provided, the firmware image is checked and the version number and firmware type is provided. The firmware will either be valid for the PMem module, a downgrade or invalid meaning it cannot be used for that PMem module.

SAMPLE OUTPUT

```
(file path): MM.mm.hh.bbbb  
Load FW on PMem module (DimmID): (Valid|Downgrade) [(with  
confirmation or the force option)]
```

If the firmware is being downgraded and the force option is not provided, the user will be prompted to confirm the downgrade for each PMem module. Otherwise, for each PMem

module, the CLI will indicate the status of the operation.

```
Downgrade firmware on PMem module (DimmID)? (y or [n]) Downgrade firmware  
on PMem module (DimmID)? (y or [n])
```

If a failure occurs when updating multiple PMem modules, the process will continue attempting to update the remaining PMem modules requested. The firmware will not become active until the next power cycle. Use the command Section [Show Device Firmware](#) to view more detailed information about the active and staged firmware.

```
Load FW on PMem module (DimmID): Success, a power cycle is required to  
activate the FW.
```

```
Load FW on PMem module (DimmID): Error (Code) - (Description)
```

2.5.5. Modify Device

NAME

ipmctl-modify-device - Changes the configurable settings on one or more PMem modules

SYNOPSIS

```
ipmctl set [OPTIONS] -dimm [TARGETS] [PROPERTIES]
```

DESCRIPTION

Changes the configurable settings on one or more PMem modules.

OPTIONS

-f

-force

Changing PMem module settings is a potentially destructive operation which requires confirmation from the user for each PMem module. This option suppresses the confirmation.

-h

-help

Displays help for the command.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS

-dimm (DimmIDs)

Modifies specific PMem modules by supplying one or more comma-separated PMem module identifiers. However, this is not recommended as it may put the system in an undesirable state. The default is to modify all manageable PMem modules.

PROPERTIES

AveragePowerReportingTimeConstantMultiplier

The value that is used to multiply the base time value used for average power consumption measurements. Can be set to a value between 0 and 32. The default value is 0.

AveragePowerReportingTimeConstant

The value, in milliseconds, used to determine the time constant for reporting the average power consumption measurements. Can be set to a value between 100 and 12000, by increments of 100. The default value is 1000.

EXAMPLES

Sets average power reporting time constant multiplier to 1 on all manageable PMem modules.

```
set -dimm AveragePowerReportingTimeConstant=1500
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

For each PMem module, the CLI will indicate the status of the operation. If a failure occurs when modifying multiple PMem modules, the process will exit and not continue modifying the remaining PMem modules.

SAMPLE OUTPUT

```
Modify PMem module (DimmID): Success  
Modify PMem module (DimmID): Error (Code) - (Description)
```

2.5.6. Show Preferences

NAME

ipmctl-show-preferences - Displays a list of the PMem module software user preferences

SYNOPSIS

```
ipmctl show [OPTIONS] -preferences
```

DESCRIPTION

Displays a list of the PMem module software user preferences and their current values.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

EXAMPLES

Displays the current values for all the user preferences.

```
ipmctl show -preferences
```

RETURN DATA

CLI_DEFAULT_DIMM_ID

The default display of PMem module identifiers. One of:

- **UID:** Use the DimmUID attribute as defined in the command Show Device.
- **HANDLE:** Use the DimmHandle attribute as defined in the command Show Device. This is the default.

CLI_DEFAULT_SIZE

The default display of capacities in the CLI. One of:

- **AUTO:** Automatically choose the best format for each capacity in binary multiples of bytes (i.e., B, MiB, GiB or TiB). This is the default.
- **AUTO_10:** Automatically choose the best format for each capacity in decimal multiples of bytes (i.e., B, MB, GB or TB).
- **B:** Displays all capacities in bytes.
- **MB:** Displays all capacities in megabytes.
- **MiB:** Displays all capacities in mebibytes.
- **GB:** Displays all capacities in gigabytes.
- **GiB:** Displays all capacities in gibibytes.
- **TB:** Displays all capacities in terabytes.
- **TiB:** Displays all capacities in tebibytes.

APPDIRECT_SETTINGS

The interleave settings to use when creating App Direct capacity in the format: (IMCSize_ChannelSize). The default is "RECOMMENDED" which uses the BIOS recommended App Direct settings returned by the command Show System Capabilities.

DBG_LOG_LEVEL

Whether debug logging is enabled in the PMem module host software. These logs pertain to the operation of the command-line tool only and do not reflect any logging functionality of the PMem module. One of:

- **0:** Logging is disabled. This is the default.
- **1:** Log Errors.
- **2:** Log Warnings, Errors.
- **3:** Log Informational, Warnings, Errors.
- **4:** Log Verbose, Informational, Warnings, Errors.

2.5.7. Change Preferences

NAME

ipmctl-change-preferences - Modifies one or more user preferences

SYNOPSIS

```
ipmctl set [OPTIONS] -preferences [PROPERTIES]
```

DESCRIPTION

Modifies one or more user preferences in the PMem module software.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

PROPERTIES

CLI_DEFAULT_DIMM_ID

The default display of PMem module identifiers. One of:

- "UID": Use the DimmUID attribute as defined in the section **Show Device**.
- "HANDLE": Use the DimmHandle attribute as defined in section **Show Device**. This is the default

CLI_DEFAULT_SIZE

The default display of capacities in the CLI. One of:

- "AUTO": Automatically choose the best format for each capacity in binary multiples of bytes (i.e., B, MiB, GiB or TiB). This is the default.
- "AUTO_10": Automatically choose the best format for each capacity in decimal multiples of bytes (i.e., B, MB, GB or TB).
- "B": Displays all capacities in bytes.
- "MB": Displays all capacities in megabytes.
- "MiB": Displays all capacities in mebibytes.
- "GB": Displays all capacities in gigabytes.
- "GiB": Displays all capacities in gibibytes.
- "TB": Displays all capacities in terabytes.
- "TiB": Displays all capacities in tebibytes.

APPDIRECT_SETTINGS

The interleave settings to use when creating App Direct capacity in the format: (IMCSize ChannelSize). Must be one of the BIOS supported App Direct settings returned by the command **Show System Capabilities**.

NOTE	ByOne is not a valid setting for this preference. The default is "RECOMMENDED" which uses the BIOS recommended App Direct settings.
NOTE	The same interleave settings are used for all the App Direct capacity in the system. Therefore, if any App Direct capacity already exists, this preference cannot be changed.

DBG_LOG_LEVEL

Whether debug logging is enabled in the PMem module host software. These logs pertain to the operation of the command-line tool only and do not reflect any logging functionality of the PMem module. One of:

- "0": Logging is disabled. This is the default.
- "1": Log Errors.
- "2": Log Warnings, Errors.
- "3": Log Informational, Warnings, Errors.
- "4": Log Verbose, Informational, Warnings, Errors.

EXAMPLES

Use DimmUID as the default PMem module identifier, and display all capacities in bytes.

```
ipmctl set -preferences CLI_DEFAULT_DIMM_ID=UID CLI_DEFAULT_SIZE=B
```

RETURN DATA

Returns the status of the operation.

SAMPLE OUTPUT

```
Set (Property)=(Value): Success|Error (Code)-(Description)
```

2.5.8. Dump Support Data

NAME

ipmctl-dump-support-data - Dumps a support snapshot to a file

SYNOPSIS

```
ipmctl dump [OPTIONS] -destination (file_prefix) [-dict (filename)] -support
```

DESCRIPTION

Creates a support snapshot and dump support data to a file for off-line analysis by support personnel. Support data may include system logs, error logs, trace logs, platform configuration, sensor information and diagnostic results.

This command may take significantly longer than other commands to complete, because it combines and executes multiple individual commands.

Commands executed:

- version
- show -memoryresources
- show -a -system -capabilities
- show -a -topology
- start -diagnostic
- show -system

Commands executed per PMem module:

- show -a -dimm
- show -a -sensor -dimm

- show -pcd -dimm
- show -error media -dimm
- show -error thermal -dimm

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmlxml)

-output (text|nvmlxml)

Changes the output format of the command execution (the output file content will remain text). One of: "text" (default) or "nvmlxml".

TARGET

-destination (filename)

This command creates a text file with a name starting with the given filename option and dumps the platform support information into it. In addition, this command also outputs the debug log information in separate files. Refer to ***Dump Debug Log*** for more details.

-dict (filename)

Optional file path to the dictionary file. If supplied, the command will create both the binary debug log and a text file with decoded log data with the file prefix specified by -destination. This option is used only to dump the debug log information.

EXAMPLES

Creates a text file named dumpfile_platform_support_info.txt and stores the platform supported data in that file. Also, dumps the debug log info in the related files that start with the file name dumpfile_*. Refer to ***Dump Debug Log*** for more info on the output files.

```
ipmctl dump -destination filename -dict nlog_dict.1.1.0.0000.txt -support
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

Returns the status of the operation.

SAMPLE OUTPUT

On success:

```
Dump support data successfully written to filename_platform_support_info.txt.
```

2.6. Debug

2.6.1. Start Diagnostic

NAME

ipmctl-start-diagnostic - Starts a diagnostic test

For a complete list of diagnostic test results, refer to addendum section [Diagnostic Events](#).

SYNOPSIS

```
ipmctl start [OPTIONS] -diagnostic [TARGETS]
```

DESCRIPTION

Starts a diagnostic test.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS**-diagnostic [Quick|Config|Security|FW]**

Start a specific test by supplying its name. By default all tests are run. One of:

- "Quick" - This test verifies that the PMem module host mailbox is accessible and that basic health indicators can be read and are currently reporting acceptable values.
- "Config" - This test verifies that the BIOS platform configuration matches the installed hardware and the platform configuration conforms to best known practices.
- "Security" - This test verifies that all PMem modules have a consistent security state. It is a best practice to enable security on all PMem modules rather than just some.
- "FW" - This test verifies that all PMem modules of a given model have consistent FW installed and other FW modifiable attributes are set in accordance with best practices.

Note that the test does not have a means of verifying that the installed FW is the optimal version for a given PMem module model just that it has been consistently applied across the system.

-dimm [DimmIDS]

Starts a diagnostic test on specific PMem modules by optionally supplying one or more comma-separated PMem module identifiers. The default is to start the specified tests on all manageable PMem modules. Only valid for the Quick diagnostic test.

EXAMPLES

Starts all diagnostics.

```
ipmctl start -diagnostic
```

Starts the quick check diagnostic on PMem module 0x0001.

```
ipmctl start -diagnostic Quick -dimm 0x0001
```

LIMITATIONS

If a PMem module is unmanageable, then Quick test will report the reason, while Config, Security and FW tests will skip unmanageable PMem modules.

RETURN DATA

Each diagnostic generates one or more log messages. A successful test generates a single log message per PMem module indicating that no errors were found. A failed test might generate multiple log messages each highlighting a specific error with all the relevant details. Each log contains the following information.

Test

The test name along with overall execution result. One of:

- "Quick"
- "Config"
- "Security"
- "FW"

State

The collective result state for each test. One of:

- "Ok"
- "Warning"
- "Failed"
- "Aborted"

Message

The message indicates the status of the test. One of:

- "Ok"
- "Failed"

SubTestName

The subtest name for given Test.

Test Name	Valid SubTest Names
Quick	<ul style="list-style-type: none"> • Manageability • Boot status • Health
Config	<ul style="list-style-type: none"> • PMem module specs • Duplicate PMem module • System Capability • Namespace LSA • PCD
Security	<ul style="list-style-type: none"> • Encryption status • Inconsistency
FW	<ul style="list-style-type: none"> • FW Consistency • Viral Policy • Threshold check • System Time

State

The severity of the error for each sub-test displayed with SubTestName. One of:

- "Ok"
- "Warning"
- "Failed"
- "Aborted"

2.6.2. Show Error Log

NAME

ipmctl-show-error-log - Shows thermal or media errors on the specified PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -error (Thermal|Media) [TARGETS]
[PROPERTIES]
```

DESCRIPTION

Shows thermal or media errors on the specified PMem modules.

OPTIONS

-a

-all

Shows all attributes.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS

-dimm [DimmIDs]

Filter output to specific PMem modules by optionally supplying one or more comma-separated PMem module identifiers. The default is to include all manageable PMem modules.

PROPERTIES

SequenceNumber

Error log entries are stored with a sequence number starting with 1 and rolling over back to 1 after 65535. Limit the error log entries returned by providing a sequence number. Only errors with a sequence number equal to or higher than provided will be returned. The default is 1.

Level

Severity level of errors to be fetched. One of:

- "High": High severity errors (Default)
- "Low": Low severity errors

Count

Max number of error entries to be fetched and printed. The default is 8 for media errors and 16 for thermal errors.

EXAMPLES

Show all high thermal error log entries

```
ipmctl show -error Thermal Level=High
```

Show all low media error log entries

```
ipmctl show -error Media Level=Low
```

LIMITATIONS

To successfully execute this command, the specified PMem modules must be manageable by the host software. In addition, for PMem modules with firmware version 2.3 and above, errors due to AIT operations are no longer exposed.

RETURN DATA

Prints errors of the specified type for the specified PMem modules. If no errors are found, the following message will be printed:

```
No errors found on PMem module (DimmID)
```

SAMPLE OUTPUT

```
DimmID | System Timestamp | Error Type
```

```
0x0001 | 01/01/1970 00:00:20 | 0x04 - Locked/Illegal Access
0x0001 | 01/01/1970 00:00:21 | 0x04 - Locked/Illegal Access
Show error executed successfully
```

```
Media Error occurred on Dimm (DimmID): System
Timestamp : 1527266471 DPA : 0x000014c0 PDA :
0x00000600 Range : 4B Error Type : 4 - Locked/Illegal
Access Error Flags : DPA Valid Transaction Type : 11 -
CSR Write Sequence Number : 2
```

2.6.3. Dump Debug Log

NAME

ipmctl-dump-debug-log - Dumps encoded firmware debug logs from specified PMem modules and optionally decodes to human readable text.

SYNOPSIS

```
ipmctl dump [OPTIONS] -destination (file_prefix) [-dict (file)] -debug
[TARGETS] [PROPERTIES]
```

DESCRIPTION

Dumps encoded firmware debug logs from specified PMem modules and optionally decodes to human readable text using a dictionary file.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

TARGET

-destination (file_prefix)

The command will create files that use the given filename as a prefix and append the PMem module UID, PMem module handle, debug log source, and the appropriate file type (.bin for encoded logs, .txt for decoded logs) onto the end.

```
file_prefix_Uid_Handle_logsource.[bin,txt]
```

-dict (path)

Optional file path to the dictionary file. If supplied, the command will create both the binary debug log and a text file with decoded log data with the file prefix specified by -destination. Firmware dictionaries are lookup tables that match firmware codes to their text descriptions. When referenced, the dictionary will be used to substitute the provided codes with text, making the logs more human readable. Dictionaries are included in every firmware release under the format nlog_dict.X.X.X.XXXX.txt.

-dimm [DimmIDs]

Dumps the debug logs from the specified PMem modules.

EXAMPLES

Dumps and decodes the debug log from PMem module 0x0001 and 0x0011 using the dictionary file.

```
ipmctl dump -destination file_prefix -dict nlog_dict.txt -debug -dimm  
0x0001,0x0011
```

LIMITATIONS

To successfully execute this command, the specified PMem modules must be manageable by the host software.

RETURN DATA

Dumps the encoded and optionally decoded contents of all 3 firmware debug log sources for the specified PMem modules. Output file names are generated based on the -destination parameter above.

SAMPLE OUTPUT

```
Dumped media FW debug logs to file (file_prefix_8089-A1-1816-  
00000016_0x0001_media.bin)  
Decoded 456 records to file (file_prefix_8089-A1-1816-  
00000016_0x0001_media.txt)  
No spi FW debug logs found
```

2.6.4. Show ACPI Tables

NAME

ipmctl-show-acpi - Shows the system ACPI tables related to the PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -system
```

DESCRIPTION

Shows the system ACPI tables related to the PMem modules in the system.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

TARGETS

-system [NFIT|PCAT|PMTT]

The system ACPI tables to display. By default NFIT, PCAT and PMTT tables are displayed. One of:

- "NFIT" - The NVDIMM Firmware Interface Table
- "PCAT" - The Platform Capabilities Table
- "PMTT" - The Platform Memory Topology Table

Refer to the ACPI specification for detailed information about the ACPI tables.

EXAMPLES

Show the ACPI NFIT

```
ipmctl show -system NFIT
```

RETURN DATA

Returns the formatted data from the requested ACPI tables and their sub-tables. Refer to the ACPI specification for detailed information about the format of the ACPI tables.

NOTE

All data is presented in ACPI little endian format.

2.6.5. Show Device Platform Configuration Data

NAME

ipmctl-show-pcd - Shows the platform configuration data for one or more PMem modules

SYNOPSIS

```
ipmctl show [OPTIONS] -pcd [TARGETS]
```

DESCRIPTION

Shows the platform configuration data for one or more PMem modules.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

TARGETS

-dimm [DimmIDs]

Restricts output to the platform configuration data on specific PMem modules by supplying one or more comma-separated PMem module identifiers. The default is to display the platform configuration data for all manageable PMem modules.

-pcd [Config|LSA]

Restricts output to a specific partition of the platform configuration data. The default is to display both. One of:

- Config - Configuration management information
- LSA - Namespace label storage area

EXAMPLES

Shows the configuration information from the platform configuration data for all manageable PMem modules.

```
ipmctl show -dimm -pcd
```

Shows the configuration information from the platform configuration data for PMem module 0x1.

```
ipmctl show -dimm -pcd Config
```

LIMITATIONS

The specified PMem modules must be manageable by the host software.

RETURN DATA

Returns the formatted data from the requested platform configuration data for the specified PMem modules for debugging and troubleshooting purposes.

2.6.6. Delete Device Platform Configuration Data

NAME

ipmctl-delete-pcd - Clears select partition data from the PCD

SYNOPSIS


```
ipmctl delete [OPTIONS] -pcd [TARGETS]
```

DESCRIPTION

When Config is specified, the Current, Input, and Output Data Size and Start Offset values in the Configuration header are set to zero, making those tables invalid.

NOTE When Config is specified, only PCD partition 1 is modified. If the platform is rebooted prior to creating a new goal on any targeted PMem modules, UEFI platform firmware will detect the missing tables and, if possible, restore previous config using the PCD partition 0 tables.

NOTE This action can be useful when moving PMem modules from one system to another, as goal creation rules may restrict provisioning PMem modules with an existing configuration. Deleting the PCD can be used as a way to prepare a PMem module for provisioning with the create -goal or load -goal commands by clearing existing configuration metadata. This allows the PMem module to be provisioned in isolation using the create-goal command with the -dimm option. Once the PCD has been deleted the desired goal should be created before rebooting.

WARNING **This command may result in data loss. Data should be backed up to other storage before executing this command. Because of data dependencies, other commands may be affected until the system has been rebooted.**

OPTIONS

-f

-force

Deleting the PCD data is a destructive operation which requires confirmation from the user for each PMem module. This option suppresses the confirmation.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text|nvmxml)**-output (text|nvmxml)**

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS**-dimm [DimmIDs]**

Deletes the PCD data on specific PMem modules by supplying one or more comma-separated PMem module identifiers. The default is to delete the PCD data for all manageable PMem modules.

-pcd [Config]

Clears the configuration management information.

EXAMPLES

Clears the Cin, Cout, Ccur tables from all manageable PMem modules

```
delete -dimm -pcd Config
```

LIMITATIONS

The specified PMem modules must be manageable by the host software, and if data-at-rest security is enabled, the PMem modules must be unlocked. Any existing namespaces associated with the requested PMem modules should be deleted before running this command.

RETURN DATA

For each PMem module, the CLI will indicate the status of the operation. If a failure occurs when deleting the platform configuration data from multiple PMem modules, the process will continue deleting the remaining PMem modules.

2.6.7. Inject Error

NAME

ipmctl-inject-error - Injects an error or clears a previously injected error

SYNOPSIS

```
ipmctl set [OPTIONS] -dimm [TARGETS] [PROPERTIES]
```

DESCRIPTION

Injects an error or clears a previously injected error on one or more PMem module for testing purposes.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmlxml)

-output (text|nvmlxml)

Changes the output format. One of: "text" (default) or "nvmlxml".

TARGETS

-dimm [DimmIDs]

Injects or clears an error on specific PMem modules by supplying one or more comma separated PMem module identifiers. The default is to inject the error on all manageable PMem modules.

PROPERTIES

This command only supports setting or clearing one type of error at a time.

Clear

- "1": Clears a previously injected error. This property must be combined with one of the other properties indicating the previously injected error to clear.

Temperature

Injects an artificial media temperature in degrees Celsius into the PMem module. The firmware that is monitoring the temperature of the PMem module will then be alerted and take necessary precautions to preserve the PMem module. The value is injected immediately and will override the firmware from reading the actual media temperature of the device and use this value instead which may cause adverse reactions by the firmware and result in an alert or log.

NOTE

The injected temperature value will remain until the next reboot or until it is cleared. The media temperature is an artificial temperature and will not cause harm to the part. Although firmware actions due to improper temperature injections may cause adverse effects on the PMem module. If the Critical Shutdown Temperature, or higher, is passed in, this may cause the PMem module firmware to perform a shutdown in order to preserve the part and data. The temperature value will be ignored on clear.

Poison

The physical address to poison.

Poison is not possible for any address in the PM region if the PM region is locked. Injected poison errors are only triggered on a subsequent read of the poisoned address in which case an error log will be generated by the firmware, but no alerts will be sent. This command can be used to clear non-injected poison errors. The data will be zero'd after clearing. There is no requirement to enable error injection prior to request to clear poison errors.

The caller is responsible for keeping a list of injected poison errors, in order to properly clear the injected errors afterwards. Simply disabling injection does not clear injected poison errors. Injected poison errors are persistent across power cycles and system resets.

NOTE

System firmware (BIOS) will not read from any Intel® Optane™ PMem device addresses that are known to be poisoned. For any poisoned address, the first read may result in a hang/fault, but system firmware (BIOS) will mark this address as poisoned so subsequent attempts to read poisoned addresses will be rejected with an error. The result of such an error may prevent booting from a namespace that has poisoned data.

NOTE

The address must be 256 byte aligned (e.g., 0x10000000, 0x10000100, 0x10000200...).

PoisonType

The type of memory to poison. One of:

- "PatrolScrub": Injects a poison error at the specified address simulating an error found during a patrol scrub operation indifferent to how the memory is currently allocated, This is the default.
- "MemoryMode": Injects a poison error at the specified address currently allocated in Memory Mode.
- "AppDirect": Injects a poison error at the specified address currently allocated as App Direct.

NOTE

If the address to poison is not currently allocated as the specified memory type, an error is returned.

PackageSparing

- "1": Triggers an artificial package sparing. If package sparing is enabled and the PMem module still has spares remaining, this will cause the firmware to report that there are no spares remaining.

PercentageRemaining

Injects an artificial percentage remaining value into the PMem module. This will cause the firmware to take appropriate action based on the value and if necessary generate an error log, an alert, and update the health status.

FatalMediaError

- "1": Injects a fake media fatal error which will cause the firmware to generate an error log and an alert.

NOTE

When media fatal error is injected, BSR Media Disabled status bit will be set indicating media error, until the fatal error is cleared using disable trigger input parameter to clear this injected fatal error.

NOTE

Injecting a Fatal Media error is unsupported on Windows. Contact Microsoft for assistance in performing this action.

DirtyShutdown

- "1": Injects an ADR failure resulting in dirty shutdown upon reboot.

EXAMPLES

Sets the media temperature on all manageable PMem modules to 50 degrees Celsius.

```
ipmctl set -dimm Temperature=50
```

Clears the injected media temperature on all manageable PMem modules.

```
ipmctl set -dimm Clear=1 Temperature=1
```

Poison address 0x10000200 on PMem module 1234.

```
ipmctl set -dimm 1234 Poison=0x10000200
```

Clears the injected poison of address 0x10000200 on PMem module 1234.

```
ipmctl set -dimm 1234 Poison=0x10000200 Clear=1
```

Triggers an artificial package sparing on all manageable PMem modules.

```
ipmctl set -dimm PackageSparing=1
```

Sets the life remaining percentage on all manageable PMem modules to 10%.

```
ipmctl set -dimm PercentageRemaining=10
```

Clears the injected remaining life percentage on all manageable PMem modules. The value of PercentageRemaining is irrelevant.

```
ipmctl set -dimm PercentageRemaining=10 Clear=1
```

Triggers an artificial ADR failure on all manageable PMem modules resulting in a dirty shutdown on each PMem module on the next reboot.

```
ipmctl set -dimm DirtyShutdown=1
```

LIMITATIONS

This command is available only when error injection is enabled on the PMem modules in the BIOS. To successfully execute this command, the specified PMem modules must be manageable by the host software.

RETURN DATA

For each PMem module, the CLI will indicate the status of the operation. If a failure occurs when injecting an error on multiple PMem modules, the process will continue with the remaining PMem modules.

SAMPLE OUTPUT

```
Set temperature on PMem module (DimmID): Success|Error (Code) -  
(Description)  
Clear injected temperature on PMem module (DimmID): Success|Error  
(Code) - (Description)
```

```
Poison address (Address) on PMem module (DimmID): Success|Error  
(Code) - (Description)  
Clear injected poison of address (Address) on PMem module  
(DimmID): Success|Error (Code) - (Description)
```

```
Trigger package sparing on PMem module (DimmID): Success|Error  
(Code) - (Description)  
Clear injected package sparing on PMem module (DimmID):  
Success|Error (Code) - (Description)
```

```
Trigger a spare capacity alarm on PMem module (DimmID):  
Success|Error (Code) - (Description)  
Clear injected spare capacity alarm on PMem module (DimmID):  
Success|Error (Code) - (Description)
```

```
Create a media fatal error on PMem module (DimmID): Success|Error  
(Code) - (Description)  
Clear injected media fatal error on PMem module (DimmID):  
Success|Error (Code) - (Description)
```

2.6.8. Show Command Access Policy

NAME

ipmctl-show-cap - Shows the current Command Access Policy restrictions.

SYNOPSIS

```
ipmctl show [OPTIONS] -cap [TARGETS]
```

DESCRIPTION

Shows the current Command Access Policy restrictions. This is a list of Opcode:SubOpcode and the corresponding mailboxes the commands are restricted, if any. Each command may be restricted to None, BIOS only, SMBus only, or BIOS and SMBus only. None indicates that Host, BIOS, and SMBus mailbox access is allowed.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS

-dimm [DimmIDs]

Restricts output to specific PMem modules by supplying one or more comma separated PMem module identifiers. The default is to display all manageable PMem modules.

EXAMPLES

Lists restrictions for all PMem modules installed in the system

```
ipmctl show -cap
```

Lists restrictions for PMem module 0x1001


```
ipmctl show -dimm 0x1001 -cap
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

The default behavior is to return a table which indicates the restrictions enforced by Command Access Policy (CAP).

DimmID

The default display of PMem module identifiers. One of:

- UID: Use the DimmUID attribute as defined in the command Show Device.
- HANDLE: Use the DimmHandle attribute as defined in the command Show Device.
This is the default.

Opcode

The Opcode for a command.

SubOpcode

The SubOpcode for a command.

Restriction

Text describing which mailboxes are restricted for Opcode:SubOcode combination.
One of:

- None
- BIOS only
- SMBus only
- BIOS and SMBus only
- Unsupported

2.6.9. Show Command Effect Log

NAME

ipmctl-show-cel - Shows the current PMem module firmware Command Effect Log.

SYNOPSIS

```
ipmctl show [OPTIONS] -cel [TARGETS]
```

DESCRIPTION

Retrieves the PMem module command effect log and presents the response as a list of DimmID, Opcode, SubOpcode and PMem module firmware command effects.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

-lpmb

Used to specify large transport payload size for the current invocation of ipmctl.

-spmb

Used to specify small transport payload size for the current invocation of ipmctl.

NOTE

The -lpmb and -spmb options are mutually exclusive and may not be used together.

-o (text|nvmxml)

-output (text|nvmxml)

Changes the output format. One of: "text" (default) or "nvmxml".

TARGETS

-dimm [DimmIDs]

Restricts output to specific PMem modules by supplying one or more comma separated PMem module identifiers. The default is to display all manageable PMem modules.

EXAMPLES

Lists command effects for all PMem modules installed in the system

```
ipmctl show -cel
```

Lists command effects for PMem module 0x1001

```
ipmctl show -dimm 0x1001 -cel
```

LIMITATIONS

In order to successfully execute this command:

- The caller must have the appropriate privileges.
- The specified PMem modules must be manageable by the host software.

RETURN DATA

The default behavior is to return a table which lists the command effects per Opcode and SubOpcode.

DimmID

The default display of PMem module identifiers. One of:

- UID: Use the DimmUID attribute as defined in the command Show Device.
- HANDLE: Use the DimmHandle attribute as defined in the command Show Device.
This is the default.

Opcode

The Opcode for a command.

SubOpcode

The SubOpcode for a command.

CE Description

A comma separated list that includes one or more of:

- NE: No Effect
- SSC: Security State Change
- DCC: DIMM Configuration Change after reboot
- IDCC: Immediate DIMM Configuration Change
- QIO: Quiesce All IO

- IDDC: Immediate DIMM Data Change
- TM: Test Mode
- DM: Debug Mode
- IDPC: Immediate DIMM Policy Change

2.7. Playback and Record

Playback and Record (PBR) is a capability included to enable efficient reproduction and debug of issues a user may encounter. The capability is designed to capture the current state of the platform as it relates to PMem modules, and all interactions with the PMem module firmware. This data can then be stored in a file and sent to the development team for rapid reproduction and debug.

The PBR file contains the following:

- ACPI tables: NFIT, PCAT and PMTT
- SMBIOS tables
- Raw firmware command response data

Theory of operation: Recording

1. Start a recording session (start -session).
2. Execute all commands to be included in session.
3. Save the recording to a file (dump -session).
4. Stop the recording session (stop -session).
5. Send PBR files to support personnel for analysis.

Example Recording Sequence

To record CLI commands you begin by starting a recording session.

```
# ipmctl start -session -mode record  
  
Setting to record mode.
```

All commands executed from this point forward will be added to the recording session.

```
# ipmctl show -dimm 1
Warning - Executing in recording mode!

DimmID | Capacity      | LockState | HealthState | FWVersion
=====
0x0001 | 253.734 GiB | Disabled  | Healthy     | 02.01.00.1034

# ipmctl show -dimm -firmware
Warning - Executing in recording mode!

DimmID | ActiveFWVersion | StagedFWVersion
=====
0x0001 | 02.01.00.1034   | N/A
0x0101 | 02.01.00.1034   | N/A
0x1001 | 02.01.00.1034   | N/A
0x1101 | 02.01.00.1034   | N/A

# ipmctl show -dimm 1 -sensor
Warning - Executing in recording mode!

DimmID | Type                                | CurrentValue
=====
0x0001 | Health                              | Healthy
0x0001 | MediaTemperature                     | 38C
0x0001 | ControllerTemperature                | 40C
0x0001 | PercentageRemaining                  | 100%
0x0001 | LatchedDirtyShutdownCount           | 4
0x0001 | PowerOnTime                          | 10661690s
0x0001 | UpTime                              | 4138492s
0x0001 | PowerCycles                          | 46
0x0001 | FwErrorCount                        | 2
0x0001 | UnlatchedDirtyShutdownCount          | 26
```

To preserve the session for later playback, dump the session to a file.

```
# ipmctl dump -destination myrecording.pbr -session

Warning - Executing in recording mode!

Successfully dumped 101405 bytes to file.
```

Remember to stop the session when you are done recording.

NOTE

Stopping a session frees all recording data saved, which is why there is a verify prompt. To skip this verify prompt, use the `-force` option.

NOTE

Session related commands are ignored by the recording/playback mechanism.

```
# ipmctl stop -session

Warning - Executing in recording mode!

Stopping a session will free all recording content.

Do you want to continue? [y/n] y

Stopped PBR session.
```

Theory of operation: Playback

1. Load the recorded session (load -session).
2. Start playback to execute commands all at once or individually (start -session).
3. Debug as necessary.
4. Stop the playback session (stop -session).

Example Playback Sequence

Load the existing PBR file.

```
# ipmctl load -listing myrecording.pbr -session
Successfully loaded 35175 bytes to session buffer.
```

To examine the recorded command sequence, show the session.

```
# ipmctl show -session
TagID | Args
=====
0x0*  | show -dimm 1
0x1   | show -dimm -firmware
0x2   | show -dimm 1 -sensor
```

During playback, all the commands can be run at once or individually.

To run all the commands at once, use playback mode.

```
# ipmctl start -session -mode playback
DimmID | Capacity      | LockState | HealthState | FWVersion
=====
0x0001 | 253.734 GiB | Disabled  | Healthy     | 02.01.00.1034

DimmID | ActiveFWVersion | StagedFWVersion
=====
0x0001 | 02.01.00.1034   | N/A
0x0101 | 02.01.00.1034   | N/A
0x1001 | 02.01.00.1034   | N/A
0x1101 | 02.01.00.1034   | N/A

DimmID | Type                                | CurrentValue
=====
0x0001 | Health                              | Healthy
0x0001 | MediaTemperature                    | 38C
0x0001 | ControllerTemperature               | 40C
0x0001 | PercentageRemaining                 | 100%
0x0001 | LatchedDirtyShutdownCount          | 4
0x0001 | PowerOnTime                         | 10661690s
0x0001 | UpTime                             | 4138492s
0x0001 | PowerCycles                        | 46
0x0001 | FwErrorCount                       | 2
0x0001 | UnlatchedDirtyShutdownCount        | 26
```

To run the commands individually (one at a time), use `playback_manual` mode.

This requires invoking the commands in the correct order - the same order they were recorded. To see which command is next, use `'show -session'` and note the asterisk (*) denotes the command that will be executed next.

To set the next command to be executed, use the `-tag` option. In this example, the command associated with tag 1 will be set as next.

```
# ipmctl start -session -mode playback_manual -tag 1
Warning - Executing in playback mode!
```

Setting to playback_manual mode.

```
# ipmctl show -session
Warning - Executing in playback mode!
```

TagID	Args
0x0	show -dimm 1
0x1*	show -dimm -firmware
0x2	show -dimm 1 -sensor

Now the command 'show -dimm -firmware' can be run and the next command to be executed will advance to tag 2.

```
# ipmctl show -dimm -firmware
Warning - Executing in playback mode!
```

DimmID	ActiveFWVersion	StagedFWVersion
0x0001	02.01.00.1034	N/A
0x0101	02.01.00.1034	N/A
0x1001	02.01.00.1034	N/A
0x1101	02.01.00.1034	N/A

```
# ipmctl show -session
Warning - Executing in playback mode!
```

TagID	Args
0x0	show -dimm 1
0x1	show -dimm -firmware
0x2*	show -dimm 1 -sensor

When done with the playback session, use 'stop -session' to disable the playback mode and resume normal operation.

```
# ipmctl stop -session
Warning - Executing in playback mode!
```

Stopping a session will free all recording content.
Do you want to continue? [y/n] y
Stopped PBR session.

2.7.1. Start Session

NAME

ipmctl-start-session - Starts a recording or playback session.

SYNOPSIS

```
ipmctl      start      [OPTIONS]      -session      -mode  
(record|playback|playback_manual) [TARGETS]
```

DESCRIPTION

Starts a recording or playback session. The recording session records the platform's ACPI NFIT, PCAT, PMTT tables, SMBIOS tables, and FIS mailbox transactions that occur during the recording session. The normal use-case would be to start a recording session, execute commands (e.g., `create -goal`, `show -sensors`, etc.) to be recorded, dump the recorded session using the [Dump Session](#) command, followed by stopping the session using the [Stop Session](#) command. The "dumped" session can then be loaded and "played" back on any platform that can execute the ipmctl tool.

The playback session has two modes: 'playback' and 'playback_manual'. The 'playback' mode will automatically execute all commands that were previously recorded. The 'playback_manual' mode allows commands to be executed one at a time in a manual fashion. If a tag is given, the playback will begin at the command that corresponds with the tagID. Note, the [Show Session](#) command displays the order and commands to execute, where the '*' denotes which command to execute next.

OPTIONS

-f

-force

Do not warn the user that starting a new session terminates an active recording session resulting in deleting recorded content.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

TARGET**-session**

Specifies to start a session.

-mode (record|playback|playback_manual)

The session modes supported. One of:

- "record" - records data associated with command execution
- "playback" - automatically executes commands previously recorded
- "playback_manual" - enables manual execution of commands previously recorded

-tag [tagID]

Specifies the starting command by tagID. Only available with "playback" and "playback_manual" mode.

EXAMPLES

Start a recording session.

```
ipmctl start -session -mode record
```

Automatically execute commands in a session.

```
ipmctl start -session -mode playback
```

Allow for manual execution of commands in playback mode

```
ipmctl start -session -mode playback_manual
```

LIMITATIONS

Recordings should be played back on the same IPMCTL version that created the recording. Recordings taken in UEFI should be played back in the UEFI environment (simulated or real). Recordings taken in an OS are binary compatible with other OS versions of IPMCTL (i.e., recording taken in Linux can be played back in Windows).

RETURN DATA

In 'playback' mode the output will be a concatenation of the output from each played back command.

2.7.2. Dump Session

NAME

ipmctl-dump-session - Dumps content captured during a recording session.

SYNOPSIS

```
ipmctl dump [OPTIONS] -destination (path) -session
```

DESCRIPTION

Dumps content captured during a recording session. Captured content includes ACPI, SMBIOS tables, and FIS requests and responses. The resulting session file that is generated can be used to playback commands recorded on a real platform in a simulated environment, making it possible to debug issues offline.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

TARGET

-destination (path)

An absolute or relative path including the filename, where the contents of the active session will be copied to.

-session

Specifies to dump the contents associated with an active recording session.

EXAMPLES

Dump the contents associated with the current active recording session to

/tmp/session.pbr.

```
ipmctl dump -destination /tmp/session.pbr -session
```

LIMITATIONS

To successfully execute this command, there must be an active recording session.

RETURN DATA

The resulting file includes, NFIT, PCAT, PMTT and SMBIOS tables that are used by IPMCTL to determine the PMem module topology. PMem module data that is transferred to/from PMem modules over the mailbox interface.

SAMPLE OUTPUT

```
Successfully dumped 1060619 bytes to file.
```

2.7.3. Load Session

NAME

ipmctl-load-session - Loads content captured during a recording session.

SYNOPSIS

```
ipmctl load [OPTIONS] -source (path) -session
```

DESCRIPTION

Loads content captured during a recording session into internal memory buffers. Captured content includes ACPI, SMBIOS tables, and FIS requests and responses. A loaded session can be executed using the 'start -session' command.

OPTIONS

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The `-ddrt` and `-smbus` options are mutually exclusive and may not be used together.

TARGET**-source (path)**

An absolute or relative path including the filename.

-session

Specifies that the source file contains recording data.

EXAMPLES

Load the contents of a previously recorded session from `/tmp/session.pbr`.

```
ipmctl load -source /tmp/session.pbr -session
```

SAMPLE OUTPUT

```
Successfully loaded 1060619 bytes to session buffer.
```

2.7.4. Show Session

NAME

`ipmctl-show-session` - Displays the playback and record (PBR) command history of a session.

SYNOPSIS

```
ipmctl show [OPTIONS] -session
```

DESCRIPTION

Displays the command history of a session. Also displays which command to execute next during 'playback_manual' mode (see 'start -session').

OPTIONS**-h****-help**

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of `ipmctl`.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

TARGET

-session

Specifies to show the command history of loaded/active session.

EXAMPLES

Show the contents of the loaded/active session.

```
ipmctl show -session
```

LIMITATIONS

A session must be loaded or active prior to executing this command. A session may be loaded via the 'load -session' command or made active via the 'start -session' command.

SAMPLE OUTPUT

```
TagID | RC | Args
```

```
0x0* | 0 | show -sensor
0x1 | 0 | show -dimm
```

2.7.5. Stop Session

NAME

ipmctl-stop-session - Stops the active playback or recording session.

SYNOPSIS

```
ipmctl stop [OPTIONS] -session
```

DESCRIPTION

Stops the active playback or recording session.

OPTIONS

-f

-force

Do not warn the user that stopping a new session terminates an active recording session resulting in deleting recorded content.

-h

-help

Displays help for the command.

-ddrt

Used to specify DDRT as the desired transport protocol for the current invocation of ipmctl.

-smbus

Used to specify SMBUS as the desired transport protocol for the current invocation of ipmctl.

NOTE

The -ddrt and -smbus options are mutually exclusive and may not be used together.

TARGET

-session

Specifies to stop a session.

EXAMPLES

Stop the current session.

```
ipmctl stop -session
```

LIMITATIONS

A session must be already started.

SAMPLE OUTPUT

```
Successfully dumped 1060619 bytes to file.
```

```
Warning - Executing in playback mode!
```

```
Stopping a session will free all recording content. Do  
you want to continue? [y/n] y Stopped PBR session.
```

3. Addendum

3.1. Diagnostic Events

Events are generated as a result of invoking the Start Diagnostics command in order to analyze the Intel® Optane™ Persistent Memory Module (PMem module) for potential issues.

Diagnostic events may fall into the following categories:

- Quick health diagnostic test event
- Platform configuration diagnostic test event
- Security diagnostic test event
- Firmware consistency and settings diagnostic test event

Each event includes the following pieces of information:

- The severity of the event that occurred. One of:
 - Informational (Info)
 - Warning (Warning)
 - Error (Failed)
 - Aborted (Aborted)
- A unique ID of the item (PMem module UUID, DimmID, NamespaceId, RegionId, etc.) the event refers to.
- A detailed description of the event in English.

The following sections list each of the possible events grouped by category of the event.

3.1.1. Quick Health Check Events

The quick health check diagnostic verifies that the Intel® Optane™ persistent memory modules host mailboxes are accessible and that basic health indicators can be read and are currently reporting acceptable values.

Table 2. Table Quick Health Check Events

Code	Severity	Message	Arguments
500	Info	The quick health check succeeded.	

Code	Severity	Message	Arguments
501	Warning	The quick health check detected that PMem module [1] is not manageable because subsystem vendor ID [2] is not supported. UID: [3]	<ol style="list-style-type: none"> PMem module Handle Subsystem Vendor ID PMem module UID
502	Warning	The quick health check detected that PMem module [1] is not manageable because subsystem device ID [2] is not supported. UID: [3]	<ol style="list-style-type: none"> PMem module Handle Subsystem Device ID PMem module UID
503	Warning	The quick health check detected that PMem module [1] is not manageable because firmware API version [2] is not supported. UID: [3]	<ol style="list-style-type: none"> PMem module Handle FW API version PMem module UID
504	Warning	The quick health check detected that PMem module [1] is reporting a bad health state [2]. UID: [3]	<ol style="list-style-type: none"> PMem module Handle Actual Health State PMem module UID
505	Warning	The quick health check detected that PMem module [1] is reporting a media temperature of [2] C which is above the alarm threshold [3] C. UID: [4]	<ol style="list-style-type: none"> PMem module Handle Actual Media Temperature Media Temperature Threshold PMem module UID
506	Warning	The quick health check detected that PMem module [1] is reporting percentage remaining at [2]% which is less than the alarm threshold [3]%. UID: [4]	<ol style="list-style-type: none"> PMem module Handle Actual Percentage Remaining Percentage Remaining Threshold PMem module UID
507	Warning	The quick health check detected that PMem module [1] is reporting reboot required. UID: [2]	<ol style="list-style-type: none"> PMem module Handle PMem module UID

Code	Severity	Message	Arguments
511	Warning	The quick health check detected that PMem module [1] is reporting a controller temperature of [2] C which is above the alarm threshold [3] C. UID: [4]	<ol style="list-style-type: none"> 1. PMem module Handle 2. Actual Controller Temperature 3. Controller Temperature Threshold 4. PMem module UID
513	Error	The quick health check detected that the boot status register of PMem module [1] is not readable. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
514	Error	The quick health check detected that the firmware on PMem module [1] is reporting that the media is not ready. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
515	Error	The quick health check detected that the firmware on PMem module [1] is reporting an error in the media. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
519	Error	The quick health check detected that PMem module [1] failed to initialize BIOS POST testing. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
520	Error	The quick health check detected that the firmware on PMem module [1] has not initialized successfully. The last known Major:Minor Checkpoint is [2]. UID: [3]	<ol style="list-style-type: none"> 1. PMem module Handle 2. Major checkpoint : Minor checkpoint in Boot Status Register 3. PMem module UID
523	Error	The quick health check detected that PMem module [1] is reporting a viral state. The PMem module is now read-only. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
529	Warning	The quick health check detected that PMem module [1] is reporting that it has no package spares available. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
530	Info	The quick health check detected that the firmware on PMem module [1] experienced an unsafe shutdown before its latest restart. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID

Code	Severity	Message	Arguments
533	Error	The quick health check detected that the firmware on PMem module [1] is reporting that the AIT DRAM is not ready. UID: [2]	1. PMem module Handle 2. PMem module UID
534	Error	The quick health check detected that the firmware on PMem module [1] is reporting that the media is disabled. UID: [2]	1. PMem module Handle 2. PMem module UID
535	Error	The quick health check detected that the firmware on PMem module [1] is reporting that the AIT DRAM is disabled. UID: [2]	1. PMem module Handle 2. PMem module UID
536	Error	The quick health check detected that the firmware on PMem module [1] failed to load successfully. UID: [2]	1. PMem module Handle 2. PMem module UID
538	Error	PMem module [1] is reporting that the DDRT IO Init is not complete. UID: [2]	1. PMem module Handle 2. PMem module UID
539	Error	PMem module [1] is reporting that the mailbox interface is not ready. UID: [2]	1. PMem module Handle 2. PMem module UID
540	Error	An internal error caused the quick health check to abort on PMem module [1]. UID: [2]	1. PMem module Handle 2. PMem module UID
541	Error	The quick health check detected that PMem module [1] is busy. UID: [2]	1. PMem module Handle 2. PMem module UID
542	Error	The quick health check detected that the platform FW did not map a region to SPA on PMem module [1]. ACPI NFIT NVPMem module State Flags Error Bit 6 Set. UID: [2]	1. PMem module Handle 2. PMem module UID
543	Error	The quick health check detected that PMem module [1] DDRT Training is not complete/failed. UID: [2]	1. PMem module Handle 2. PMem module UID
544	Error	PMem module [1] is reporting that the DDRT IO Init is not started. UID: [2]	1. PMem module Handle 2. PMem module UID

3.1.2. Platform Configuration Check Events

This diagnostic test group verifies that the BIOS platform configuration matches the installed hardware and the platform configuration conforms to best known practices.

Table 3. Table Platform Configuration Check Events

Code	Severity	Message	Arguments
600	Info	The platform configuration check succeeded.	
601	Info	The platform configuration check detected that there are no manageable PMem modules.	
606	Info	The platform configuration check detected that PMem module [1] is not configured. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
608	Error	The platform configuration check detected [1] PMem modules installed on the platform with the same serial number [2].	<ol style="list-style-type: none"> 1. Number of PMem modules with duplicate serial numbers. 2. The duplicate serial number
609	Info	The platform configuration check detected that PMem module [1] has a goal configuration that has not yet been applied. A system reboot is required for the new configuration to take effect. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
618	Error	The platform configuration check detected that a PMem module with physical ID [1] is present in the system but failed to initialize. UID: [2]	<ol style="list-style-type: none"> 1. PMem module handle in the SMBIOS table 2. PMem module UID
621	Error	The platform configuration check detected PCD contains invalid data on PMem module [1]. UID: [2]	<ol style="list-style-type: none"> 1. PMem module Handle 2. PMem module UID
622	Error	The platform configuration check was unable to retrieve the namespace information.	

Code	Severity	Message	Arguments
623	Warning	The platform configuration check detected that the BIOS settings do not currently allow memory provisioning from this software.	
624	Error	The platform configuration check detected that the BIOS could not apply the configuration goal on PMem module [1] because of errors in the goal data. The detailed status is COUT table status: [2] [3], Partition change table status: [4], Interleave change table 1 status: [5], Interleave change table 2 status: [6].	<ol style="list-style-type: none"> 1. PMem module Handle 2. Validation Status 3. Text error code corresponding to the status code 4. Partition Size Change Status 5. Interleave Change Status 6. Interleave Change Status
625	Error	The platform configuration check detected that the BIOS could not apply the configuration goal on PMem module [1] because the system has insufficient resources. The detailed status is COUT table status: [2] [3], Partition change table status: [4], Interleave change table 1 status: [5], Interleave change table 2 status: [6].	<ol style="list-style-type: none"> 1. PMem module Handle 2. Validation Status 3. Text error code corresponding to the status code 4. Partition Size Change Status 5. Interleave Change Status 6. Interleave Change Status

Code	Severity	Message	Arguments
626	Error	The platform configuration check detected that the BIOS could not apply the configuration goal on PMem module [1] because of a firmware error. The detailed status is COUT table status: [2] [3], Partition change table status: [4], Interleave change table 1 status: [5], Interleave change table 2 status: [6].	<ol style="list-style-type: none"> 1. PMem module Handle 2. Validation Status 3. Text error code corresponding to the status code 4. Partition Size Change Status 5. Interleave Change Status 6. Interleave Change Status
627	Error	The platform configuration check detected that the BIOS could not apply the configuration goal on PMem module [1] for an unknown reason. The detailed status is COUT table status: [2] [3], Partition change table status: [4], Interleave change table 1 status: [5], Interleave change table 2 status: [6].	<ol style="list-style-type: none"> 1. PMem module Handle 2. Validation Status 3. Text error code corresponding to the status code 4. Partition Size Change Status 5. Interleave Change Status 6. Interleave Change Status
628	Error	The platform configuration check detected that interleave set [1] is broken because the PMem modules were moved [2].	<ol style="list-style-type: none"> 1. Interleave set index ID 2. List of moved PMem modules.
629	Error	The platform configuration check detected that the platform does not support ADR and therefore data integrity is not guaranteed on the PMem modules.	
630	Error	An internal error caused the platform configuration check to abort.	

Code	Severity	Message	Arguments
631	Error	The platform configuration check detected that interleave set [1] is broken because the PMem module with UID: [2] is missing from location (Socket-Die-iMC-Channel-Slot) [3].	<ol style="list-style-type: none"> Interleave set index ID PMem module UID Location ID
632	Error	The platform configuration check detected that interleave set [1] is broken because the PMem module with UID: [2] is misplaced. It is currently in location (Socket-Die-iMC-Channel-Slot) [3] and should be moved to (Socket-Die-iMC-Channel-Slot) [4].	<ol style="list-style-type: none"> Interleave set index ID PMem module UID Location ID Location ID
633	Error	The platform configuration check detected that the BIOS could not fully map memory on PMem module [1] because of an error in current configuration. The detailed status is CCUR table status: [2] [3].	<ol style="list-style-type: none"> PMem module Handle Current Configuration Status Text error code corresponding to the status code

3.1.3. Security Check Events

The security check diagnostic test group verifies that all Intel® Optane™ persistent memory modules have a consistent security state.

Table 4. Table Security Check Events

Code	Severity	Message	Arguments
800	Info	The security check succeeded.	
801	Info	The security check detected that there are no manageable PMem modules.	
802	Warning	The security check detected that security settings are inconsistent [1].	<ol style="list-style-type: none"> A comma separated list of the number of PMem modules in each security state
804	Info	The security check detected that security is not supported on all PMem modules.	
805	Error	An internal error caused the security check to abort.	

3.1.4. Firmware Consistency and Settings Check Events

This test group verifies that all PMem modules of a given subsystem device ID have consistent FW installed and other FW modifiable attributes are set in accordance with best practices.

Table 5. Table Firmware Consistency and Settings Check Events

Code	Severity	Message	Arguments
900	Info	The firmware consistency and settings check succeeded.	
901	Info	The firmware consistency and settings check detected that there are no manageable PMem modules.	
902	Warning	The firmware consistency and settings check detected that firmware version on PMem modules [1] with subsystem device ID [2] is non-optimal, preferred version is [3].	<ol style="list-style-type: none"> 1. Comma separated list of PMem module UIDs 2. Subsystem device ID 3. Preferred firmware version
903	Warning	The firmware consistency and settings check detected that PMem module [1] is reporting a non-critical media temperature threshold of [2] C which is above the fatal threshold [3] C. UID: [4]	<ol style="list-style-type: none"> 1. PMem module Handle 2. Current media temperature threshold 3. Fatal media temperature threshold 4. PMem module UID
904	Warning	The firmware consistency and settings check detected that PMem module [1] is reporting a non-critical controller temperature threshold of [2] C which is above the fatal threshold [3] C. UID: [4]	<ol style="list-style-type: none"> 1. PMem module Handle 2. Current controller temperature threshold 3. Fatal controller temperature threshold 4. PMem module UID

Code	Severity	Message	Arguments
905	Warning	The firmware consistency and settings check detected that PMem module [1] is reporting a percentage remaining of [2]% which is below the recommended threshold [3]%. UID: [4]	<ol style="list-style-type: none"> 1. PMem module Handle 2. Current percentage remaining threshold 3. Recommended percentage remaining threshold 4. PMem module UID
906	Warning	The firmware consistency and settings check detected that PMem modules have inconsistent viral policy settings.	
910	Error	An internal error caused the firmware consistency and settings check to abort.	
911	Warning	The firmware consistency and settings check detected that PMem modules have inconsistent first fast refresh settings.	