

Intel® XPU Manager Overview

October 2022



Notices and Disclaimers

Notices & Disclaimers

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

In-Band vs. Out-of-Band Intel® XPU Management

In-Band (IB)

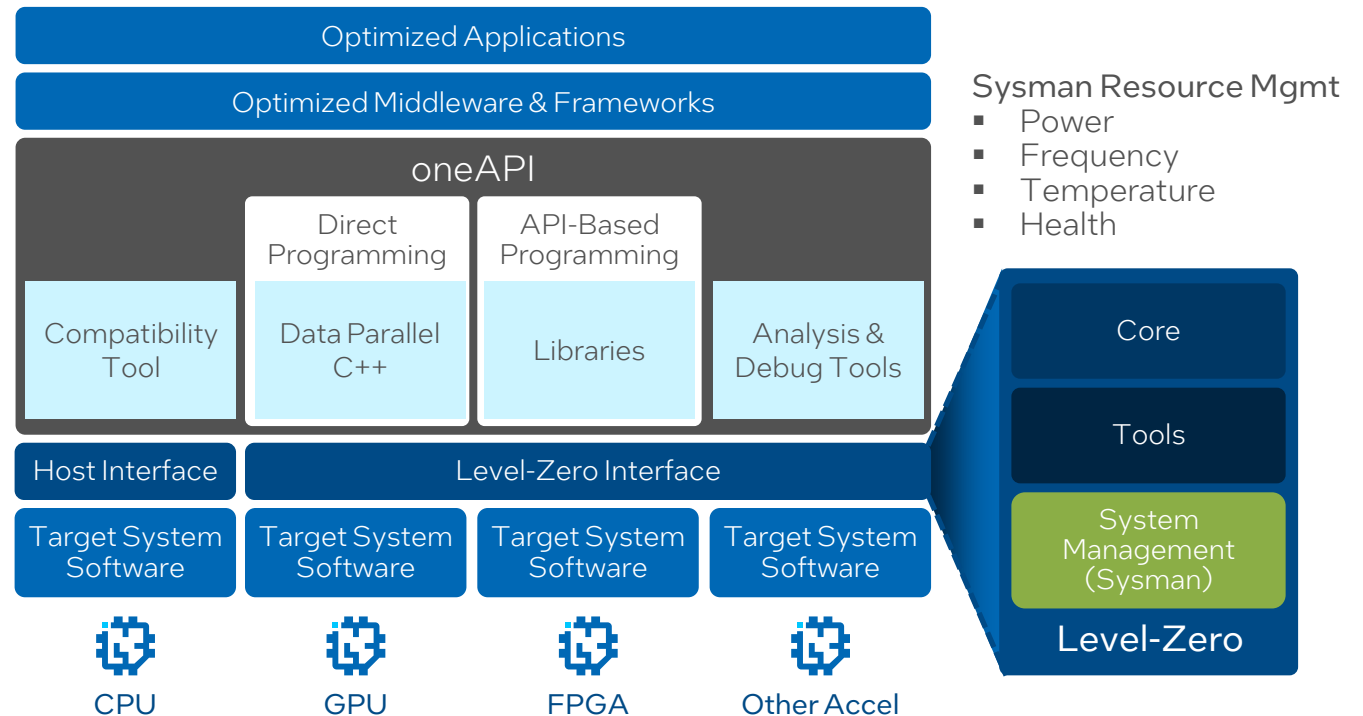
- Does not require platform support
- Requires a software agent to be set up and running on each node
- Provides more details and capabilities (e.g. GPU grouping, job scheduling, diagnostics)
- Suitable for users with more knowledge of what is running on specific servers and have login access to servers

Out-of-Band (OOB)

- Requires platform support
- Does not require any additional software components or agents
- Provides basic monitoring and control and capabilities (e.g. power, thermals, utilization)
- Suitable for users with no knowledge of or access to what is running on a collection of different servers

oneAPI and the Level-Zero APIs

- **oneAPI** is a cross-industry, open, standards-based unified programming model that delivers a common developer experience across accelerator architectures
- The oneAPI **Level-Zero APIs** provide a low-level hardware interface to support cross-architecture programming
- The Level-Zero **Sysman APIs** provide in-band access to resource management features for each accelerator device



Intel® XPU Manager Product Suite

A free and open-source suite of solutions built on top of the oneAPI Level Zero interface for monitoring and managing Intel data center XPUs.

- **Intel XPU System Management Interface (SMI)**
A command line utility for local XPU management.
- **Intel XPU Manager**
A full-fledged solution with a daemon for aggregate telemetry collection, RESTful APIs for remote XPU management, and a local library for 3rd party solutions integration, and more.

Simplify
Administration &
configuration

Maximize
Reliability &
Uptime

Improve
Utilization

Perform
Firmware
Updates

For more info visit:

<https://github.com/intel/xpumanager/>



Intel® XPU Manager Executive Summary and FAQ

- **What is Intel® XPU Manager?**
 - A free open-source solution for monitoring, managing, and updating Intel data center GPUs. It is designed to simplify administration, maximize reliability & uptime, improve utilization, and perform firmware updates.
- **What are the key features of Intel XPU Manager**
 - Monitoring GPU utilization and health, getting job-level statistics, running comprehensive diagnostics, controlling power, policy management, firmware updating, and more.
- **Who is it targeted for?**
 - End users, and 3rd party solutions including open source, ISV and OEM solutions
- **How can it be used?**
 - Intel XPU Manager can be used standalone through its CLI to manage Intel data center GPUs locally, or through its RESTful APIs to manage them remotely. It will also have a library that can be integrated into 3rd party solutions
- **Which 3rd party solutions will integrate Intel XPU Manager?**
 - We are working with a variety of ISVs and open-source projects that develop cluster and workload managers, resource schedulers, and monitoring solutions to integrate Intel XPU Manager to support Intel data center GPUs.
- **When will it be released, and which products will it support?**
 - The Intel XPU Manager landing page can be found [here](#)
 - Intel XPU Manager v1.0 Gold release was published publicly on GitHub [here](#)
 - It is also included as part of the Intel data center GPU SW tools package on Registration center [here](#)
 - A docker container image was released publicly on DockerHub [here](#)
 - The latest internal releases can be found on the Intel sandbox GitHub [here](#)
 - Version 1.0 supports Intel Data Center GPU variants.

Intel® XPU Manager vs. Intel® oneAPI Level-Zero APIs

Feature	Sysman	Intel® XPU Manager
Interface	Local Library API	CLI/Remote HTTPS/Local SSH/Local Library
GPU Telemetry	Per-GPU Real-Time Data	Aggregated System-Level Data (Multi-GPU)
GPU Control	GPU Level	Server/Group/GPU Levels
GPU Policies	No	Yes
GPU Health & Diagnostics	No	Yes

Overall Product Feature Comparison

	Intel® XPU Manager	Intel® XPU SMI
Architecture	Daemon based	Daemon-less
Interfaces	CLI, Remote HTTPS, Local SSH, Local Library	CLI
Discovery, Inventory, Topology	Yes	Yes
GPU Grouping	Yes	No
GPU statistics/metrics	Yes (aggregated system-level)	Yes (real-time per GPU)
GPU Configuration	Yes	Yes*
GPU Policies	Yes (per Group or GPU)	No
GPU Health & Diagnostics	Yes (per Group or GPU)	Yes (per GPU)
Firmware updating	Yes (per GPU)	Yes (per GPU)
Supported OSes	Linux: Ubuntu 20.04.3, RHEL 8.4, CentOS 8 Stream, SLES 15 SP3 Windows: Win Server 2022 (limited features)	Linux: Ubuntu 20.04.3, RHEL 8.4, CentOS 8 Stream, SLES 15 SP3
Frameworks	Prometheus exporter, Docker container, Icinga plugin	N/A

Intel® XPU Manager Value

Simplifies GPU Administration

- Detailed Inventory & topology
- Customizable groupings
- Comprehensive settings
- Integratable into other solutions

Maximizes GPU Reliability & Uptime

- Comprehensive health monitoring
- Aggregated telemetry
- Real-time alerting
- Multi-level diagnostics & stress tests

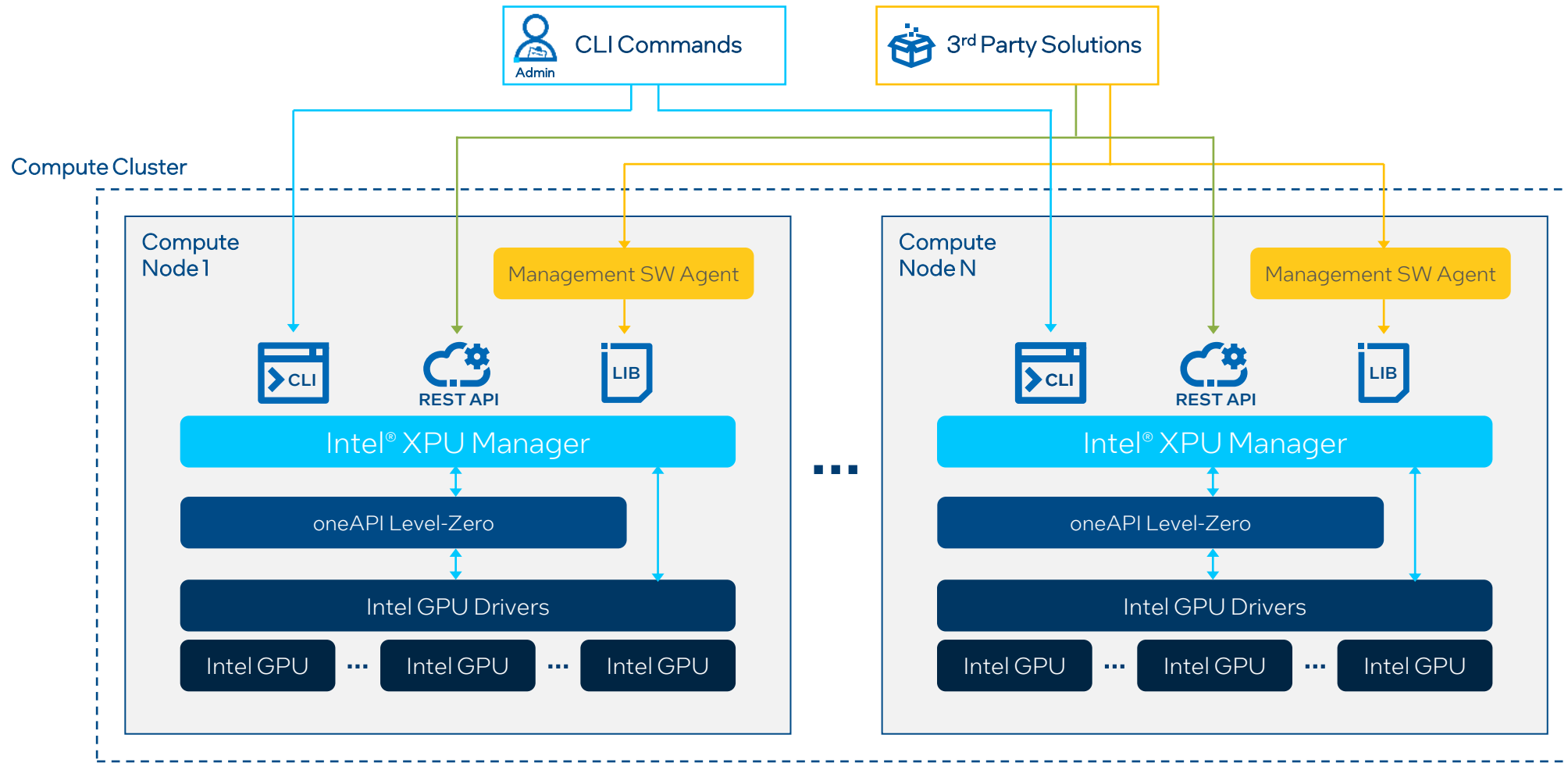
Improves GPU Utilization

- Fine-grained GPU statistics
- Different performance metrics
- Customizable policies and thresholds
- Accurate power & clock management

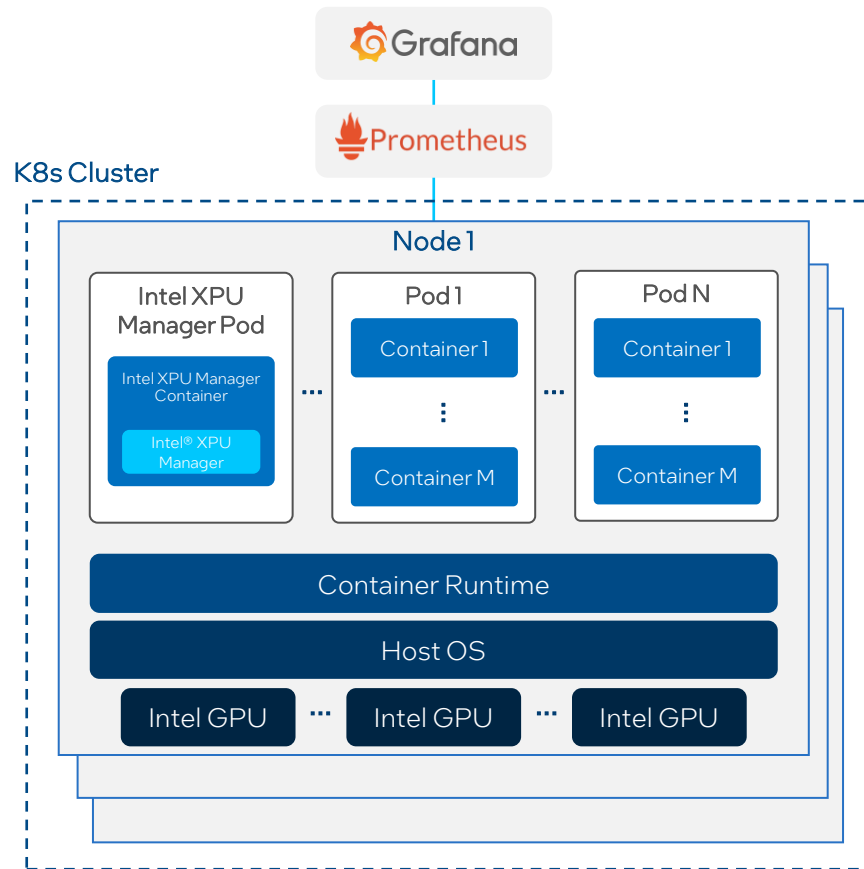
Performs Firmware Updates

- Reliable updating for GPU components.
- Simplifies firmware updating for multiple GPUs
- OS level firmware updating
- Supports most major operating systems

Intel® XPU Manager Architecture and Integration



Intel® XPU Manager Kubernetes and Grafana Support



- A docker image published on Dockerhub will allow users to get Intel data center GPU telemetry from within a Kubernetes cluster in a format that is compatible to Prometheus which can then be exposed graphically in Grafana.

Feature List for v1.0

Features	
Administration	<ul style="list-style-type: none">▪ GPU discovery – model, frequency, memory capacity, firmware version, location, etc.▪ GPU groups – create logical groups of GPUs, built-in group for 2 SOCs on the same M3 card▪ GPU topology▪ FW Updating
Monitoring	<ul style="list-style-type: none">▪ GPU health – memory health, temperature health, fabric port health, etc.▪ GPU statistics – utilization, power, temperature, performance metrics, fabric speed, etc.▪ GPU telemetry – collect utilization, power, energy consumption, frequency and temperature for a period of time
Reliability	<ul style="list-style-type: none">▪ GPU diagnostics – run GPU test suites to check computing performance and memory throughput▪ GPU provisioning – firmware updates▪ GPU policies – cap GPU when the temperature is high, etc.
Configure/Control	<ul style="list-style-type: none">▪ GPU settings – set power limits, GPU frequency ranges, performance factor (compute/media), control fabric port, ECC on/off etc.
Interface	<ul style="list-style-type: none">▪ Remote RESTful APIs and local command line interface
Supported OSes	<ul style="list-style-type: none">• Ubuntu, RHEL, SLES, CentOS• Windows 2022 with Limited support (GPU device info, telemetry and settings)
Frameworks	<ul style="list-style-type: none">• Prometheus exporter, Container support, Icinga plugin

GPU Discovery

Intel XPU Manager can find GPUs on a node and display their details including:

- Basic GPU model info
- Memory info
- Tile and EU info
- Driver and firmware info

```
intel@dms-coyotepass:~$ /opt/xpum/bin/xpumcli discovery
+-----+-----+
| Device ID | Device Information |
+-----+-----+
| 0         | Device Name: Intel(R) Graphics [0x56c1] |
|           | Vendor Name: Intel(R) Corporation      |
|           | UUID: 01000000-0000-0000-0000-0000000510000 |
|           | PCI BDF Address: 0000:51:00.0          |
+-----+-----+
| 1         | Device Name: Intel(R) Graphics [0x56c1] |
|           | Vendor Name: Intel(R) Corporation      |
|           | UUID: 01000000-0000-0000-0000-0000000560000 |
|           | PCI BDF Address: 0000:56:00.0          |
+-----+-----+
```

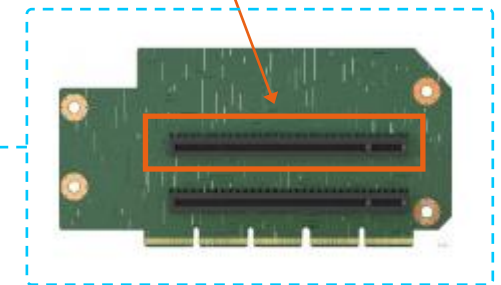
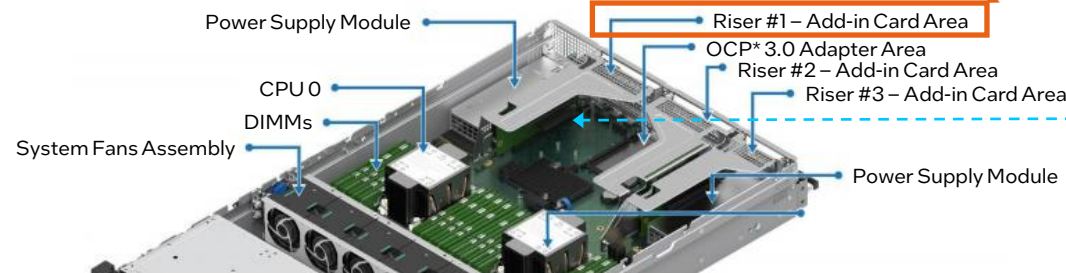
```
+-----+-----+
| Device ID | Device Information |
+-----+-----+
| 0         | Device Type: GPU |
|           | Device Name: Intel(R) Graphics [0x56c1] |
|           | Vendor Name: Intel(R) Corporation      |
|           | UUID: 01000000-0000-0000-0000-0000000510000 |
|           | Serial Number: unknown |
|           | Core Clock Rate: 1950 MHz |
|           | Stepping: B1 |
|           | |
|           | Driver Version: 16997184 |
|           | Firmware Name: GSC |
|           | Firmware Version: DG02_2.2237 |
|           | Firmware Name: GSC_DATA |
|           | Firmware Version: 101.170.0 |
|           | |
|           | PCI BDF Address: 0000:51:00.0 |
|           | PCI Slot: J37 - Riser 1, Slot 1 |
|           | PCIe Generation: 4 |
|           | PCIe Max Link Width: 8 |
|           | |
|           | Memory Physical Size: 6088.00 MiB |
|           | Max Mem Alloc Size: 4095.99 MiB |
|           | Number of Memory Channels: 2 |
|           | Memory Bus Width: 128 |
|           | Max Hardware Contexts: 65536 |
|           | Max Command Queue Priority: 0 |
|           | |
|           | Number of EUs: 128 |
|           | Number of Tiles: 1 |
|           | Number of Slices: 1 |
|           | Number of Sub Slices per Slice: 8 |
|           | Number of Threads per EU: 8 |
|           | Physical EU SIMD Width: 8 |
|           | Number of Media Engines: 2 |
|           | Number of Media Enhancement Engines: 2 |
+-----+-----+
```

GPU Location

- When a GPU has some issues, it could be difficult to locate it physically because usually servers have multiple GPUs
- Intel XPU Manager can show the PCIe slot details for each GPU to provide end users with an easy way to physically locate it

Device ID	Device Information
0	<p>Device Type: GPU Device Name: Intel(R) Graphics [0x56c1] Vendor Name: Intel(R) Corporation UUID: 01000000-0000-0000-0000-000000051000 Serial Number: unknown Core Clock Rate: 1950 MHz Stepping: B1</p> <p>Driver Version: 16997184 Firmware Name: GSC Firmware Version: DG02_2.2237 Firmware Name: GSC_DATA Firmware Version: 101.170.0</p> <p>PCI BDF Address: 0000:51:00.0 PCI Slot: J37 - Riser 1, Slot 1</p> <p>PCIe Generation: 4 PCIe Max Link Width: 8</p>

Intel® Server System M50CYP2UR Family

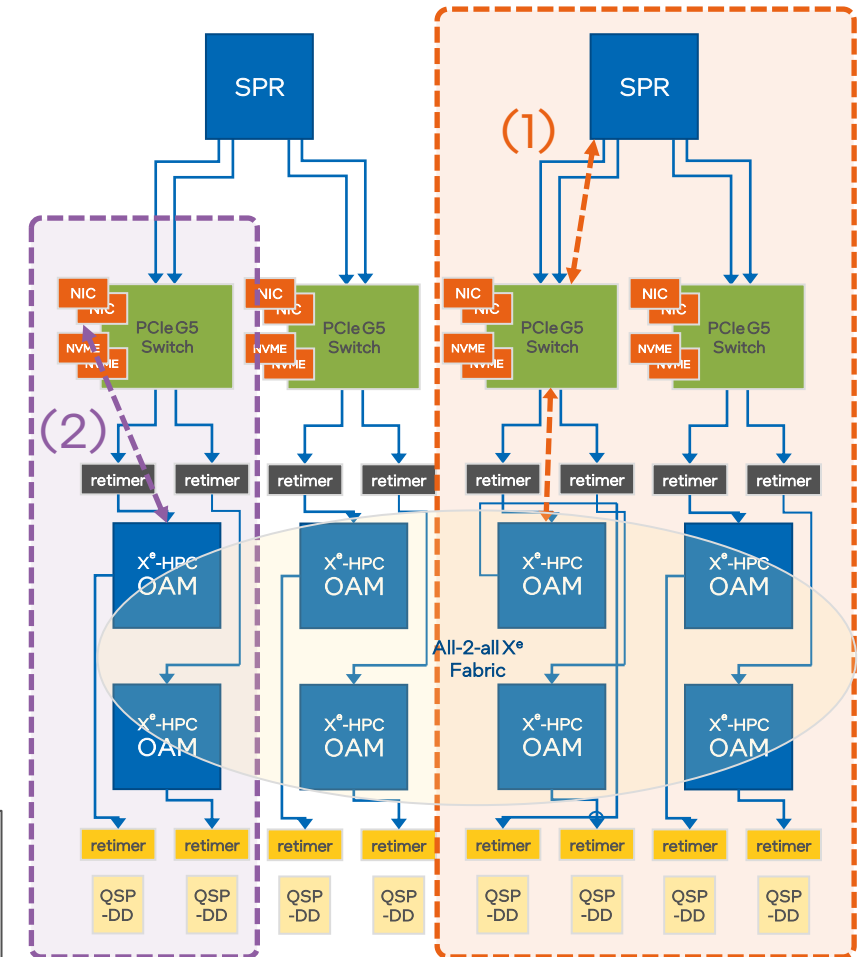


GPU Topology

- Intel XPU Manager can provide the local GPU-to-CPU and GPU-to-PCIe switch topology info
- This can be used by locality-aware schedulers to improve workload performance by:
 - Allocating workloads on CPU/GPUs which are directly connected
 - Having GPUs exchange network packets through

```

intel@dms-coyotepass:~$ /opt/xpum/bin/xpumcli topology -m
GPU 0/0      GPU 0/0      GPU 1/0      CPU Affinity
GPU 1/0      S            NODE        0-17,36-53
GPU 1/0      NODE        S           0-17,36-53
intel@dms-coyotepass:~$ /opt/xpum/bin/xpumcli topology -d 0
+-----+-----+
| Device ID | Topology Information |
+-----+-----+
| 0         | Local CPU List: 0-17,36-53
|           | Local CPUs: 00,003ffff0,0003ffff
|           | PCIe Switch Count: 1
|           | PCIe Switch: /sys/devices/pci0000:4a/0000:4a:02.0/0000:4b:00.0/0000:4c:00.0/
|           | 0000:4d:00.0/0000:4e:08.0/0000:4f:00.0/0000:50:01.0
+-----+-----+
    
```



GPU Grouping

- Intel XPU Manager can create logical groups of GPUs on the same node
- Users can do the following:
 - Create/remove groups
 - Add/delete GPUs to groups
 - Display the GPUs in one group
 - Move GPUs between groups
 - Show the telemetry per group

List devices in groups

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli group -l
```

Group ID	Group Properties
1	Group Name: ALL_GPUs Device IDs: [0,1]
2	Group Name: card-1 Device IDs: [1]

Create Group

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli group -c -n "ALL_GPUs"
```

Group ID	Group Properties
1	Group Name: ALL_GPUs Device IDs: []

Add a device to a group

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli group -g 1 -a -d0
Successfully add device [0] to group 1
```

Group ID	Group Properties
1	Group Name: ALL_GPUs Device IDs: [0]

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli group -g 1 -a -d1
Successfully add device [1] to group 1
```

Group ID	Group Properties
1	Group Name: ALL_GPUs Device IDs: [0,1]

Delete a device from a group

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli group -g 1 -r -d0
Successfully remove device [0] from group 1
```

Group ID	Group Properties
1	Group Name: ALL_GPUs Device IDs: [1]

GPU Telemetry

- Intel XPU Manager can collect the following GPU telemetry for a certain period:
 - GPU Utilization (%)
 - GPU Power (W)
 - Energy Counter (millijoule)
 - GPU Frequency (MHz)
 - GPU Temperature (°C)

- The sampling period of raw data is configurable
- Raw telemetry can be exported to help improve workload performance in a fine-grained way

```
dcm@dcm-test:/opt/xpum/bin$ sudo ./xpumcli dump -d 0 -m 24,25,27
Timestamp, DeviceId, Decoder Engine 0 (%), Decoder Engine 1 (%), Encoder Engine 0 (%),
Encoder Engine 1 (%), Media Enhancement Engine 0 (%), Media Enhancement Engine 1 (%)
2022-07-08T00:55:20.000Z, 0, 93.14, 93.36, 93.14, 93.36, 0.00, 0.00
2022-07-08T00:55:20.000Z, 0, 93.14, 93.36, 93.14, 93.36, 0.00, 0.00
2022-07-08T00:55:21.000Z, 0, 88.78, 81.60, 88.79, 81.60, 0.00, 0.00
2022-07-08T00:55:22.000Z, 0, 93.31, 92.74, 93.31, 92.74, 0.00, 0.00
2022-07-08T00:55:23.000Z, 0, 90.70, 89.67, 90.70, 89.67, 0.00, 0.00
2022-07-08T00:55:24.000Z, 0, 92.28, 84.99, 92.28, 85.00, 0.00, 0.00
2022-07-08T00:55:25.000Z, 0, 92.30, 88.50, 92.30, 88.47, 0.00, 0.00
2022-07-08T00:55:26.000Z, 0, 91.06, 82.74, 91.06, 82.74, 0.00, 0.00
2022-07-08T00:55:27.000Z, 0, 93.41, 92.92, 93.41, 92.95, 0.00, 0.00
2022-07-08T00:55:28.000Z, 0, 88.62, 79.87, 88.62, 79.87, 0.00, 0.00
2022-07-08T00:55:29.000Z, 0, 93.62, 93.10, 93.62, 93.10, 0.00, 0.00
```

GPU Statistics

- Intel XPU Manager can provide the following statistics for each GPU or group of GPUs:

GPU Utilization (%)	Memory Read (KB/s)
GPU Power (W)	Memory Written (KB/s)
Energy (J)	EU Active (%)
GPU Frequency (MHz)	EU Stall (%)
GPU Temperature (Celsius)	EU Idle (%)
Memory Used (MB)	PCIe throughput (CPU<->GPU) (kB/s)
Per-engine utilization (%)	

- EU Active, the normalized sum of all cycles on all EUs that were spent actively executing instructions.
- EU Stall, the normalized sum of all cycles on all EUs during which the EUs were stalled.
- EU Idle, the normalized sum of all cycles on all cores when no threads were scheduled on a core.

```

/opt/xpum/bin/xpumcli stats -d0
+-----+-----+
| Device ID           | 0 |
+-----+-----+
| Start Time          | 2022-07-12T01:46:38.683Z |
| End Time            | 2022-07-12T01:46:57.334Z |
| Elapsed Time (Second) | 18 |
| Energy Consumed (J) | Tile 0: 263.13 |
| GPU Utilization (%) | Tile 0: 3 |
| EU Array Active (%) | Tile 0: |
| EU Array Stall (%)  | Tile 0: |
| EU Array Idle (%)   | Tile 0: |
+-----+-----+
| Reset               | Tile 0: 0, total: 0 |
| Programming Errors  | Tile 0: 0, total: 0 |
| Driver Errors       | Tile 0: 0, total: 0 |
| Cache Errors Correctable | Tile 0: 0, total: 0 |
| Cache Errors Uncorrectable | Tile 0: 0, total: 0 |
| Mem Errors Correctable | Tile 0: 0, total: 0 |
| Mem Errors Uncorrectable | Tile 0: 0, total: 0 |
+-----+-----+
| GPU Power (W)       | Tile 0: avg: 15, min: 15, max: 15, current: 15 |
+-----+-----+
| GPU Frequency (MHz) | Tile 0: avg: 624, min: 0, max: 1950, current: 0 |
+-----+-----+
| GPU Core Temperature | Tile 0: avg: 38, min: 38, max: 38, current: 38 |
| (Celsius Degree)    | |
+-----+-----+
| GPU Memory Used (MiB) | Tile 0: avg: 23, min: 23, max: 23, current: 23 |
+-----+-----+

```

GPU Health

- Intel XPU Manager can report the threshold-based health status for each GPU or group:
 - GPU Core Temperature
 - GPU Memory Temperature
 - GPU Power
 - GPU Memory
 - Xe Link Port

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli health -d0
```

Device ID	0
1. GPU Core Temperature	Status: OK Description: All temperature sensors are healthy. Throttle Threshold: 105 Celsius Degree Shutdown Threshold: 130 Celsius Degree Custom Threshold: none
2. GPU Memory Temperature	Status: OK Description: All temperature sensors are healthy. Throttle Threshold: 85 Celsius Degree Shutdown Threshold: 100 Celsius Degree Custom Threshold: none
3. GPU Power	Status: OK Description: All power domains are healthy. Throttle Threshold: 37 watts Custom Threshold: none
4. GPU Memory	Status: Unknown Description: The memory health cannot be determined.
5. Xe Link Port	Status: Unknown Description: The device has no Xe Link capability.

GPU Diagnostics

- Intel XPU Manager can run 3 levels of diagnostic tests to check GPU HW & SW components:
 - Level-Zero/GPU driver existence
 - GPU core performance
 - GPU memory throughput
 - GPU media engine performance
 - GPU PCIe throughput
 - GPU memory allocation

Device ID	0
Level	2
Result	Pass
Items	6
Software Env Variables	Result: Pass Message: Pass to check environment variables.
Software Library	Result: Pass Message: Pass to check libraries.
Software Permission	Result: Pass Message: Pass to check permission.
Software Exclusive	Result: Pass Message: Pass to check the software exclusive.
Integration PCIe	Result: Pass Message: Pass to check PCIe bandwidth. Its bandwidth is 10.347 GBPS.
Media Codec	Result: Pass Message: Pass to check Media codec.

GPU Settings

- Intel XPU Manager can configure the following GPU or tile settings:
 - Power limit
 - Frequency range
 - Scheduler mode
 - Standby mode
 - Performance factor
 - Xe link port
 - Xe link port beaconing
 - Memory ECC (on/off)

```
root@dms-coyotepass:/home/intel# /opt/xpum/bin/xpumcli config -d 0
[2022-07-11 19:14:53.508] [W] [90889-90936] XPUM finished
```

Device Type	Device ID/Tile ID	Configuration
GPU	0	Power Limit (w): 22 Valid Range: 1 to 0 Power Average Window (ms): 1212118352 Valid Range: 1 to 124
GPU	0/0	GPU Min Frequency (MHz): 1950 GPU Max Frequency (MHz): 1950 Valid Options: 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000, 1050, 1100, 1150, 1200, 1250, 1300, 1350, 1400, 1450, 1500, 1550, 1600, 1650, 1700, 1750, 1800, 1850, 1900, 1950 Standby Mode: default Valid Options: default, never Scheduler Mode: timeslice Timeout (us): Interval (us): 5000 Yield Timeout (us): 640000 Engine Type: compute Performance Factor: Engine Type: media Performance Factor: 50 Xe Link ports: Up: Down: Beaconing On: Beaconing Off: Memory Ecc: Available: true Configurable: true Current: disabled Pending: disabled Action: none

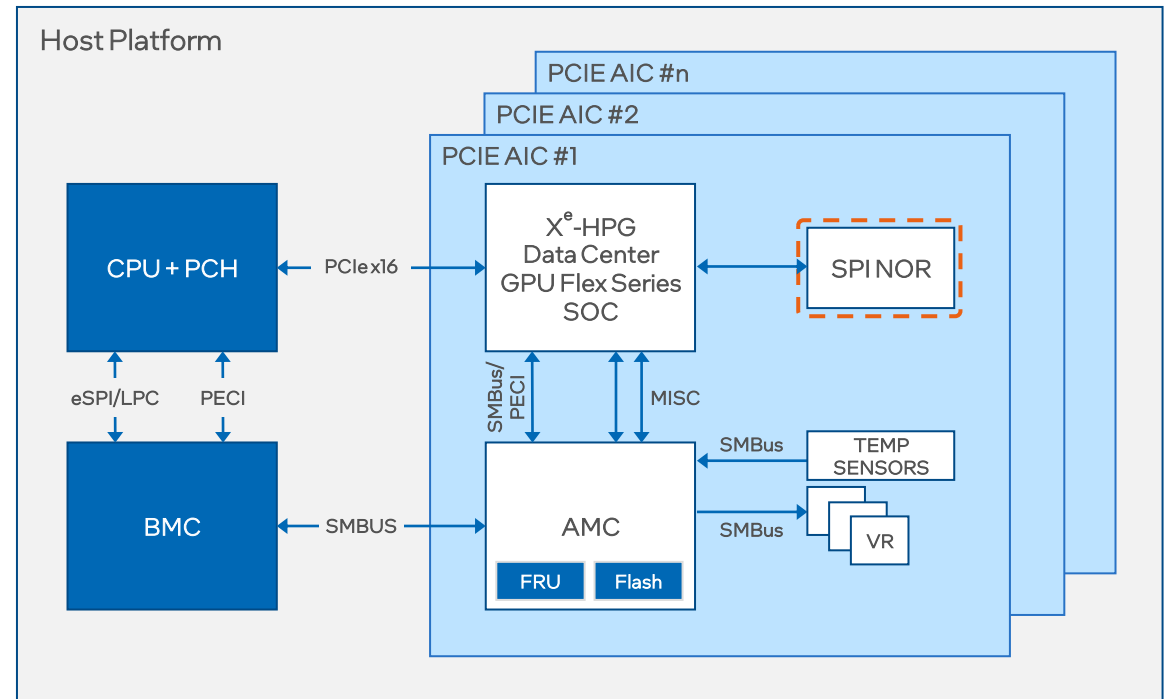
GPU Policies

Condition	Threshold	Action
GPU Temperature	Settable	<ul style="list-style-type: none">▪ Notification▪ Throttle GPU and notification
GPU Memory Temperature	Settable	<ul style="list-style-type: none">▪ Notification
GPU Power	Settable	<ul style="list-style-type: none">▪ Notification
GPU Reset	When event occurs	<ul style="list-style-type: none">▪ Notification
GPU Missing	When event occurs	<ul style="list-style-type: none">▪ Notification
Programming Error	When event occurs	<ul style="list-style-type: none">▪ Notification
Driver Error	When event occurs	<ul style="list-style-type: none">▪ Notification
Cache Correctable Errors	Settable	<ul style="list-style-type: none">▪ Notification
Cache Uncorrectable Errors	When event occurs	<ul style="list-style-type: none">▪ Notification▪ Reset GPU and notification

GPU Firmware Display and Updates

- **FW Version Display:**
 - Intel XPU Manager will be able to display GPU SoC and AMC firmware versions
- **FW Updates:**
 - In-band SoC FW updates possible via Sysman APIs
 - In-band AMC FW updates - WIP

```
root@b49691a654ac:~/xpum# ./xpumcli fwflash -d 0 -t GSC -f /tmp/ATS_M75_128_B0_PVT_ES_017_gfx_fwupdate_SOC2.bin
Start flashing firmware:
Firmware name: GSC
Image path: /tmp/ATS_M75_128_B0_PVT_ES_017_gfx_fwupdate_SOC2.bin
Firmware flashed successfully!
```



GPU Metrics for Prometheus

Intel XPU Manager Metric	Definition	Prometheus Metrics	Corresponding oneAPI Level-Zero Methods
ENGINE_GROUP_COMPUTE_ALL_UTILIZATION	Avg utilization of all compute engines (in %), per GPU tile	XPU Manager_engine_group_ratio {type="compute"}	zesEngineGetActivity *
ENGINE_GROUP_MEDIA_ALL_UTILIZATION	Avg utilization of all media engines (in %), per GPU tile	XPU Manager_engine_group_ratio {type="media"}	zesEngineGetActivity *
ENGINE_GROUP_COPY_ALL_UTILIZATION	Avg utilization of all copy engines (in %), per GPU tile	XPU Manager_engine_group_ratio {type="copy"}	zesEngineGetActivity *
ENGINE_GROUP_RENDER_ALL_UTILIZATION	Avg utilization of all render engines (in %), per GPU tile	XPU Manager_engine_group_ratio {type="render"}	zesEngineGetActivity *
ENGINE_GROUP_3D_ALL_UTILIZATION	Avg utilization of all 3D engines (in %), per GPU tile	XPU Manager_engine_group_ratio {type="3d"}	zesEngineGetActivity *
GPU_FREQUENCY	Avg actual GPU frequency (in MHz), per GPU tile	XPU Manager_frequency_mhz { location="gpu", type="actual" }	zesFrequencyGetState
GPU_REQUEST_FREQUENCY	Avg request GPU frequency (in MHz), per GPU tile	XPU Manager_frequency_mhz { location="gpu", type="request" }	zesFrequencyGetState
MEMORY_READ	Total memory read bytes (in bytes), per GPU tile	XPU Manager_memory_read_bytes	zesMemoryGetBandwidth
MEMORY_WRITE	Total memory write bytes (in bytes), per GPU tile	XPU Manager_memory_write_bytes	zesMemoryGetBandwidth
MEMORY_USED	Used GPU memory (in bytes), per GPU tile	XPU Manager_memory_used_bytes	zesMemoryGetState
MEMORY_UTILIZATION	Used GPU memory / Total used GPU memory (in %), per GPU tile	XPU Manager_memory_ratio	zesMemoryGetState
ENERGY	Total GPU energy consumption since boot (in J), per GPU	XPU Manager_energy_joules	zesPowerGetEnergyCounter
POWER	Avg GPU power (in watts), per GPU and per card***	XPU Manager_power_watts	zesPowerGetEnergyCounter *
RAS_ERROR_CAT_RESET	Total number of GPU reset since boot, per GPU	XPU Manager_resets	zesRasGetState
GPU_TEMPERATURE	Avg GPU temperature (in Celsius degree), per tile	XPU Manager_temperatures_celsius { location="gpu" }	zesTemperatureGetState
GPU_UTILIZATION	Max utilization among all engine groups (in %), per GPU tile	XPU Manager_engine_ratio	zesEngineGetActivity *
MEMORY_BANDWIDTH	Avg memory throughput / max memory bandwidth (in %), per GPU tile	XPU Manager_memory_bandwidth_ratio	zesMemoryGetBandwidth *
RAS_ERROR_CAT_PROGRAMMING_ERRORS	Total number of GPU programming errors since boot, per GPU	XPU Manager_programming_errors	zesRasGetState
RAS_ERROR_CAT_DRIVER_ERRORS	Total number of GPU driver errors since boot, per GPU	XPU Manager_driver_errors	zesRasGetState
RAS_ERROR_CAT_CACHE_ERRORS_CORRECTABLE	Total number of GPU cache correctable errors since boot, per GPU	XPU Manager_cache_errors { type="correctable" }	zesRasGetState
RAS_ERROR_CAT_CACHE_ERRORS_UNCORRECTABLE	Total number of GPU cache uncorrectable errors since boot, per GPU	XPU Manager_cache_errors { type="uncorrectable" }	zesRasGetState

The Intel logo is centered on a solid blue background. It features the word "intel" in a white, lowercase, sans-serif font. A small blue square is positioned above the letter 'i'. To the right of the word "intel" is a registered trademark symbol (®).

intel®