

Resctrl Microconference

1. **resctrl Schema Descriptions** – Reinette Chatre
2. **Region Aware MBA/MBM** – Tony Luck
3. **Upcoming QoS Features on AMD for Linux** – Babu Moger
4. **Difficulties Mapping MPAM onto resctrl's ABI** – James Morse
5. **MBA/MBM on CPU-less Memory Node** – Fenghua Yu

resctrl schema descriptions

Reinette Chatre, Dave Martin, Tony Luck, Chen Yu



Join the conversation

- History:
 - Re: Region aware RDT options for resctrl
 - https://lore.kernel.org/lkml/Z_mB-gmQe_LR4FWP@agluck-desk3
 - Re: [PATCH] fs/resctrl,x86/resctrl: Factor mba rounding to be per-arch
 - <https://lore.kernel.org/lkml/aNFliMZTTUiXyZzd@e133380.arm.com>
- Latest:
 - [RFC] fs/resctrl: Generic schema description
 - <https://lore.kernel.org/lkml/aPtFMffLV1l%2FRB0L@e133380.arm.com>

Motivation

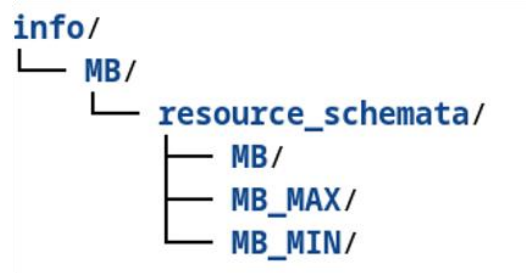
"For Memory bandwidth resource, by default the user controls the resource by indicating the percentage of total memory bandwidth."

But ...

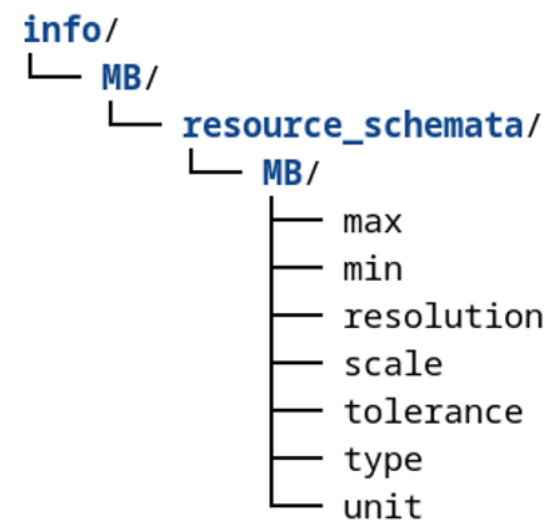
- Need to support new control values:
 - Smaller than 1% steps for memory bandwidth: 1 to 511, up to 16-bit fraction,...
- Need to support multiple controls per resource:
 - Minimum and maximum bandwidth limits.

Current proposal

- Separate resource from controls in `/sys/fs/resctrl/info/`.
- New controls: Support multiple controls per resource:

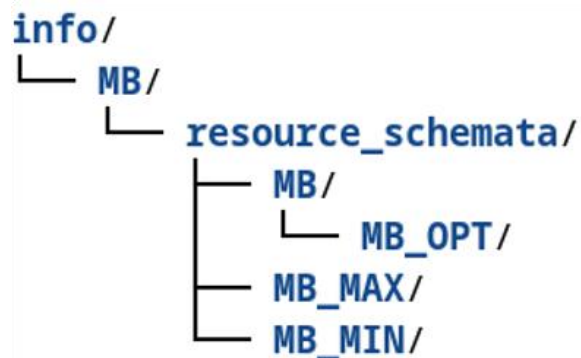


- New control values: Each control has a "type" that indicates its documented properties.
- Example:
 - User writes control value "C" to schemata file: $\text{min} \leq C \leq \text{max}$
 - Resource allocation = $C * \text{scale} / \text{resolution} * \text{unit}$

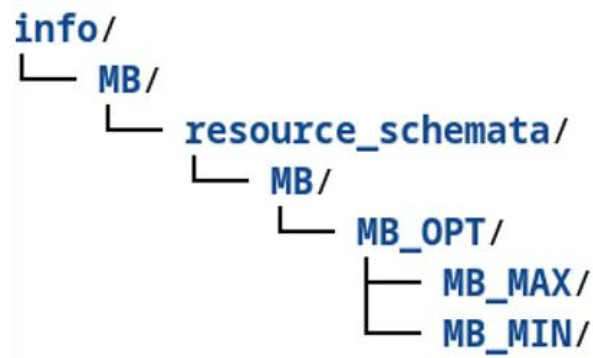


resctrl controls vs. hardware controls

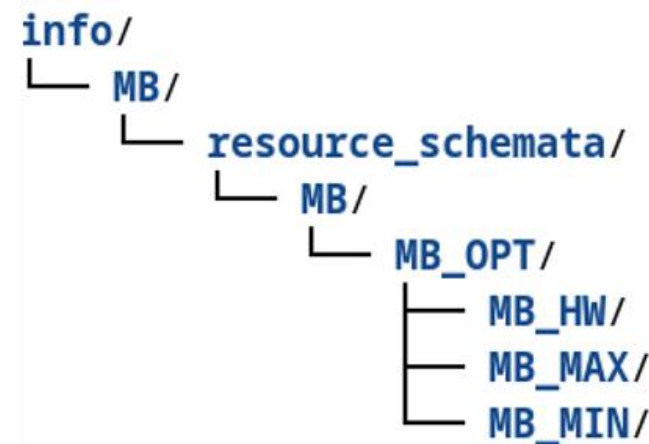
MB:0=100;1=100
MB_OPT:0=511;1=511
MB_MAX:0=511;1=511
MB_MIN:0=1;1=1



MB:0=100;1=100
MB_OPT:0=511;1=511
MB_MAX:0=511;1=511
MB_MIN:0=485;1=485



MB:0=100;1=100
MB_OPT:0=511;1=511
MB_HW:0=511;1=511
MB_MAX:0=511;1=511
MB_MIN:0=1;1=1



Open: User space interactions with schemata

Starting schemata:

```
MB:0=100;1=100  
MB_OPT:0=511;1=511
```

read-modify-write

```
cat << EOF > schemata  
MB:0=50;1=100  
MB_OPT:0=511;1=511  
EOF
```

vs

only write changes:

```
echo "MB:0=50" > schemata
```

- Is this a problem?
 - *"When writing you only need to specify those values which you wish to change."*
- Could a schema prefix help?

Starting schemata:

```
MB:0=100;1=100  
# MB_OPT:0=511;1=511
```

read-modify-write

```
cat << EOF > schemata  
MB:0=50;1=100  
# MB_OPT:0=511;1=511  
EOF
```

Open: maintaining backward compatibility when region aware

```
MB:0=100;1=100  
MB_REGION0_OPT:0=511;1=511  
MB_REGION1_OPT:0=511;1=511  
MB_REGION2_OPT:0=511;1=511  
MB_REGION3_OPT:0=511;1=511
```



```
echo "MB_REGION0_OPT:0=200" > schemata
```



```
MB:0=?;1=100  
MB_REGION0_OPT:0=200;1=511  
MB_REGION1_OPT:0=511;1=511  
MB_REGION2_OPT:0=511;1=511  
MB_REGION3_OPT:0=511;1=511
```


Join the conversation

- History:
 - Re: Region aware RDT options for resctrl
 - https://lore.kernel.org/lkml/Z_mB-gmQe_LR4FWP@agluck-desk3
 - Re: [PATCH] fs/resctrl,x86/resctrl: Factor mba rounding to be per-arch
 - <https://lore.kernel.org/lkml/aNFliMZTTUiXyZzd@e133380.arm.com>
- Latest:
 - [RFC] fs/resctrl: Generic schema description
 - <https://lore.kernel.org/lkml/aPtfMFfLV1l%2FRB0L@e133380.arm.com>



東京 2025

LINUX PLUMBERS CONFERENCE

TOKYO, JAPAN / DECEMBER 11-13, 2025



Region Aware MBM/MBA

Tony Luck

Linux Plumbers Conference

December 13th, 2025

What is a “region”

- System physical address space is divided by memory type and topology into “ranges”
- Each range is tagged with two “region” numbers
 - One for access by CPUs local to that range
 - Another for access by remote CPUs
- E.g. the range from 0 to 2GiB is generally DDR memory on the first NUMA node in the system
 - Region tag for access by CPUs on node0 may be “0”
 - Region tag for access by CPUs on other nodes may be “1”

Region tags are assigned by firmware

- Expectation for first region aware systems is:

Region #	Usage
0	Local DDR
1	Remote DDR
2	Local CXL _[1]
3	Remote CXL

[1] Firmware will tag the address ranges reserved for CXL hot plug

Memory Bandwidth Monitoring

- As with previous generations there are separate sets of counters for each L3 cache instance
- Instead of “local” and “total” counters, there are counters for each region
- File names in resctrl are up for discussion, but conceptually there will_[1] be:
 - `/sys/fs/resctrl/mon_data/mon_L3_00/mbm_local_bytes`
 - `/sys/fs/resctrl/mon_data/mon_L3_00/mbm_remote_bytes`
 - `/sys/fs/resctrl/mon_data/mon_L3_00/smbm_local_bytes`
 - `/sys/fs/resctrl/mon_data/mon_L3_00/smbm_remote_bytes`

[1] Resctrl may skip files if regions do not exist, e.g. on a DDR-only system

Memory Bandwidth Allocation

- Controls are per-L3 cache, per-region
- Format for the schemata file is up for discussion, but conceptually like this^[1]. `$ grep MB schemata`

```
MB_LOCAL:0=100;1=100
```

```
MB_REMOTE:0=100;1=100
```

```
SMBA_LOCAL:0=100;1=100
```


```
SMBA_REMOTE:0=100;1=100
```

- Additional wrinkles there are “minimum” and “maximum” controls, and the hardware supports finer granularity controls

[1] Resctrl may skip entries if regions do not exist, e.g. on a DDR-only system

More information

- Intel® Resource Director Technology (Intel® RDT) Architecture Specification
- <https://cdrdv2.intel.com/v1/dl/getContent/789566>



Upcoming QoS features on AMD for Linux

Babu Moger

Upcoming QoS features on AMD for Linux

❑ Key enhancements

- Increased Assignable Bandwidth Monitoring Counters from 32 to 64
- Global Memory Bandwidth Allocation (GMBA)
- Privilege-Level Zero Association (PLZA)

Global Memory Bandwidth Allocation (GMBA)

- ❑ Problem:
 - The MBA feature enables users to set bandwidth limits individually for each domain.
 - This approach can cause bandwidth to be either underutilized or overutilized when applications operate across multiple domains.
- ❑ Solution:
 - GMBA provides user options to set the ceiling for group of domains. The group is called GMBA control domain.
 - Bandwidth will be competitively shared across multiple domains.
- ❑ Linux resctrl changes
 - Add a new GMBA/SMBA resource type
- ❑ Example Usage
 - To set the global limit of 8GB for domains 0 and 2.
`#echo "GMBA:0=8,2=8" > /sys/fs/resctrl/group1/schemata`
The domain 0 and 2 in this case is called GMBA control domain.

Privilege-Level Zero Association (PLZA)

❑ Problem:

- If a CLOS is aggressively throttled, and it moves into Kernel mode, the kernel operations are also aggressively throttled.
- This can lead to performance bottlenecks, hinder forward progress, and cause delays under heavy system load.

❑ Solution:

- Privilege-Level Zero Association (PLZA) allows the user to specify a COS and/or RMID associated with execution in Privilege-Level Zero.
- When enabled, on a HW thread, when that thread enters Privilege-Level Zero, transactions associated with that thread will be automatically associated with the PLZA COS and/or RMID.
- Otherwise, the HW thread will be associated with the COS and RMID identified by PQR_ASSOC.

❑ Linux resctrl changes

- Add support to create PLZA resctrl group.
- Provide options to enable or disable or monitor group.





Enable ARM Memory System Resource Partitioning and Monitoring (MPAM) on CPU-less Memory Node

Fenghua Yu, NVIDIA | Linux Plumbers Conference Tokyo, Japan Dec 13, 2025

CPU-less Memory Node

- A traditional NUMA node has CPUs, cache, and memory
 - A local CPU accesses MPAM registers to control or monitor L2/L3 cache and memory bandwidth on the node
- A CPU-less memory node only has memory
 - No CPU
 - No L2/L3 cache
- Why CPU-less memory node?
 - Resolve ratios and over-provisioning of CPU and memory
 - Resolve imbalanced scaling of CPU and memory
 - Disaggregate memory from host processors
 - Better utilization of CPU and memory
 - Reduced TCO
- Example: CXL Type 3 memory expansion and pooling

Discovery of CPU-less Memory Node for MPAM

- MPAM ACPI table describes Memory Sub-system Component (MSC)
 - Corresponds to a NUMA node.
- MPAM resource node in MSC
 - Locator type (1 byte): Memory (01)
 - Locator: Proximity domain (8 bytes)
- Linked device ID (8 bytes) in MSC
 - 0: there is no device linked to the MSC -> no CPU is affiliated to the MSC node.
- Enumerate linked device ID in the MSC and “memory” locator type in the resource node
 - An MSC (NUMA node) with only “memory” locator but without linked device ID (no CPU) is detected
 - The CPU-less node is located by its proximity domain -> NUMA node ID

Access MBA/MBM Registers on CPU-less Memory Node

- MMIO base address and size are specified in MPAM ACPI MSC table
 - IO remapped and stored in kernel `msc->mapped_hwpage`
 - The MMIO memory range can be accessed by any host CPU: `cpumask_set(&msc->accessibility, cpu_possible_mask)`
- On any CPU, MBA/MBM registers can read/write by `msc->mapped_hwpage + register offset`
 - Enumerate MBA or MBM features
 - Configure MBA bandwidth
 - Read MBM local/total bytes

New Schemata Line for CPU-less MBA

Proposed resctrl Changes for MBA

- **Current memory bandwidth allocation is per L3 to allocate memory bandwidth between L3 and memory**
 - MBA line in resctrl “schemata” file: MB: <L3 cache id 0>=<bw>; <L3 cache id 1>=<bw>
- **No L3 cache on CPU-less node**
 - The current MBA line cannot be used
- **Need to create a new line for MBA on CPU-less node**
 - **MBN: <numa node id 0>=<bw>; <numa node id 1>=<bw>**
 - NUMA node id is stored in ris->nid for each resource node
- schemata example on two traditional nodes and one CPU-less node
 - MB: 1=20; 2=40 ← allocate 20% bw on L3 cache id 1 and 40% bw on L3 cache id 2
 - MBN: 5=80 ← allocate 80% bw on CPU-less NUMA node 5

New Monitoring Event Paths for CPU-less MBM

Proposed resctrl Changes for MBM

- **Current MBM events in resctrl are per L3 to monitor MB between L3 and memory**
 - `<partition>/mon_data/mon_L3_<cache id>/mbm_local_bytes`: local memory rw bytes
 - `<partition>/mon_data/mon_L3_<cache id>/mbm_total_bytes`: local and remote memory rw bytes
- **No L3 cache on CPU-less node**
 - The current local and total bytes cannot be used
- **Need to monitor the total bytes per NUMA node on CPU-less node:**
 - `<partition>/mon_data/cpu_less_<numa node id>/mbm_total_bytes`: remote r/w bytes on this CPU-less node

Discussion on the Proposed resctrl API Changes

- **Valid and generic schemata change of new “MBN” line for CPU-less MBA?**

MBN: <numa node id 0>=<bw>; <numa node id 1>=<bw>

- **Valid and generic total bytes per NUMA node id changes?**

<partition>/mon_data/cpu_less_<numa node id>/mbm_total_bytes: remote r/w bytes on this CPU-less node

References

- ARM MPAM specification: <https://developer.arm.com/documentation/ih0099/latest/>
- ACPI for ARM MPAM: <https://developer.arm.com/documentation/den0065/latest/>
- MPAM base driver: <https://git.kernel.org/pub/scm/linux/kernel/git/morse/linux.git/>
- Resctrl kernel document: Documentation/filesystems/resctrl.rst

