CHINA
OpenStack Days

CHINA RUNS ON OPENSTACK

CHINA OpenStack Days

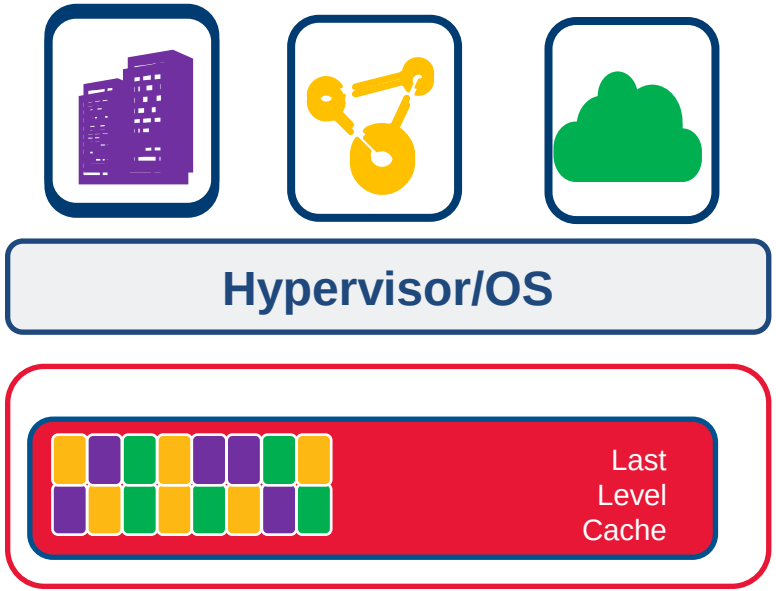
Manage Hidden/Dark Resources in OpenStack

Eli Qiao (乔立勇)
Senior Software Engineer, Open Source Technology Center, Intel

Agenda

- Hidden/Dark Resources
- Resource Manager Daemon (RMD)
- RMD Integration with OpenStack
- Summary/call for action

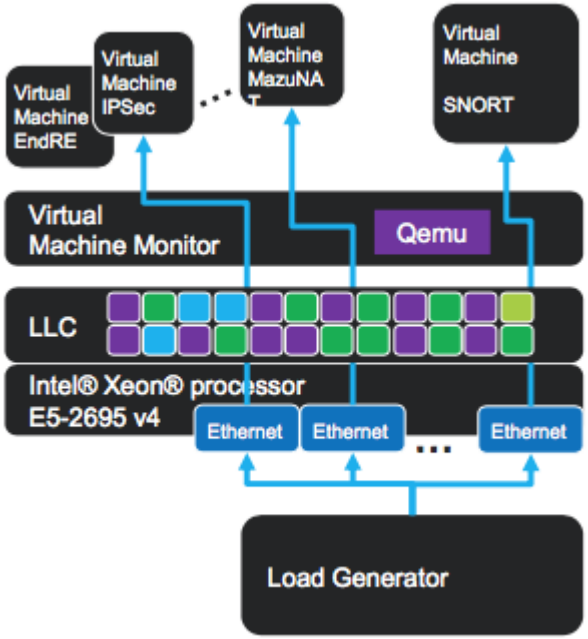
Noisy Neighbors: Cache as a Resource



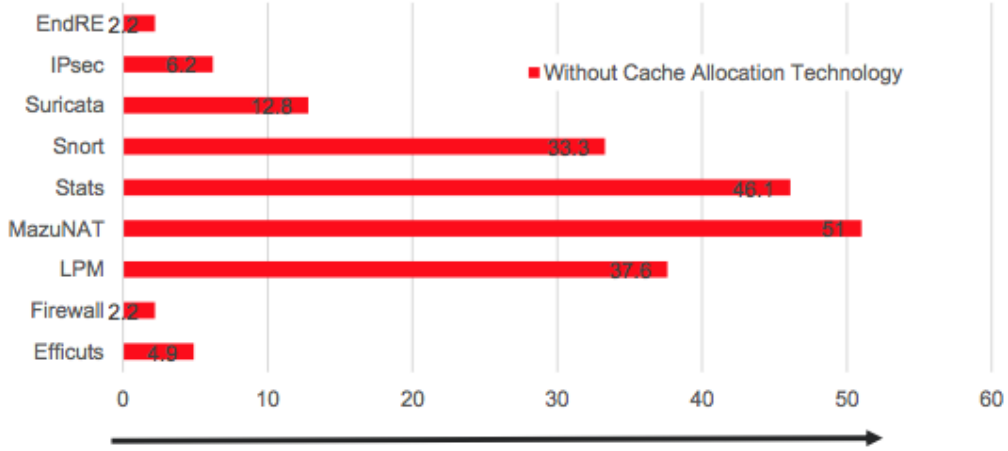
- LLC是CPU上的共享资源，系统会尽可能使用它们
- 然而，某些应用会过多地征用LLC，导致其他应用性能下降
- 正常情况下，应用切换会导致的LLC换出

LLC竞争会导致高优先级负载性能下降

Performance degradation in NFV



- VNF is isolated on cores.
- Package 64 bytes, 10万流量, 均匀分布
- LLC 竞争导致51% 的吞吐率下降



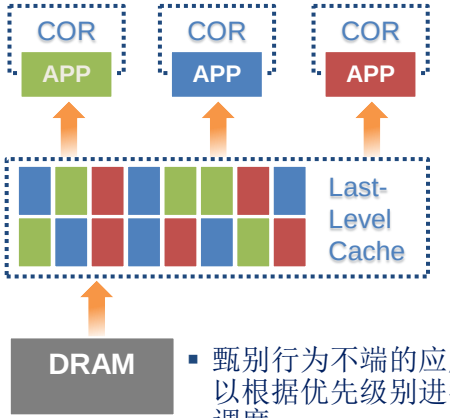
max.% throughput degradation, normalized

http://span.cs.berkeley.edu

Resource Director Technology

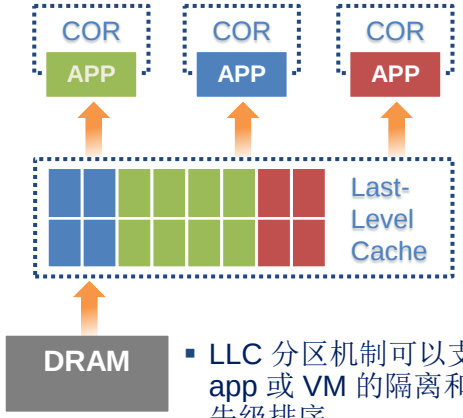
Intel® Resource Director Technology (Intel® RDT)

Cache Monitoring Technology (CMT)



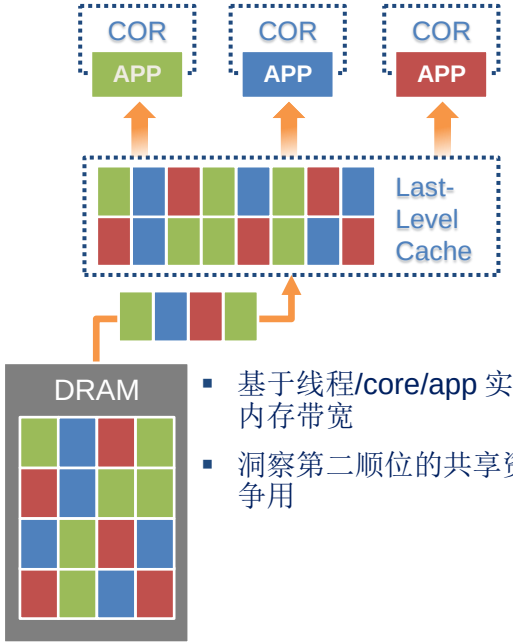
- 甄别行为不端的应用，可以根据优先级别进行重新调度

Cache Allocation Technology (CAT)



- LLC 分区机制可以支持 app 或 VM 的隔离和优先级排序
- 增加行为不端的应用的确定性

Memory Bandwidth Monitoring (MBM)



- 基于线程/core/app 实现监控内存带宽
- 洞察第二顺位的共享资源的争用

http://www.intel.com/content/www/us/en/architecture-and-technology/resource-director-technology.html

With RDT?

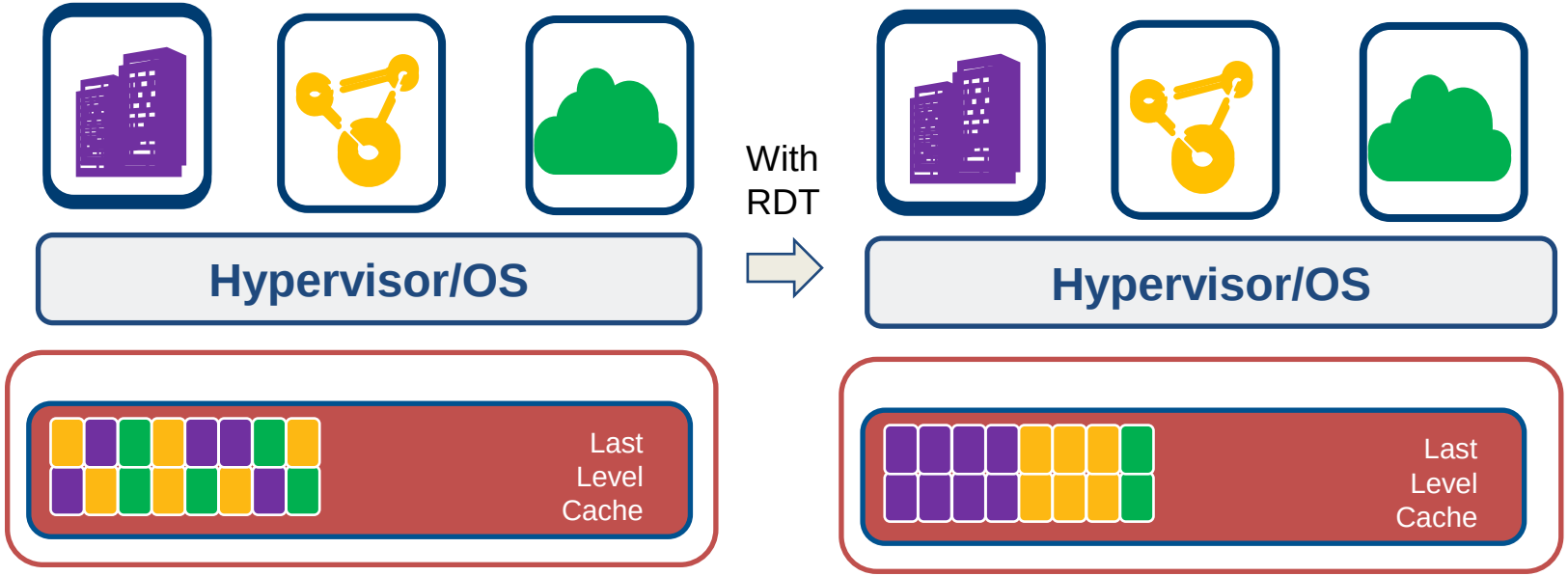
1. 限制
   “Noisy Neighbors”
   - 监禁noisy

2. 实时QoS
   - 监控和调节

3. 保证云的性能
   - SLA

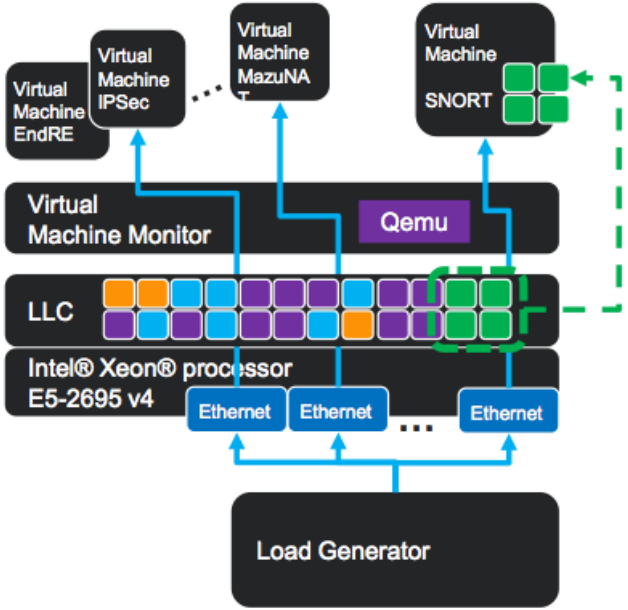
4. 通过隔离保证安全
   - 避免cache 竞争引发的饥饿，DDos

With RDT

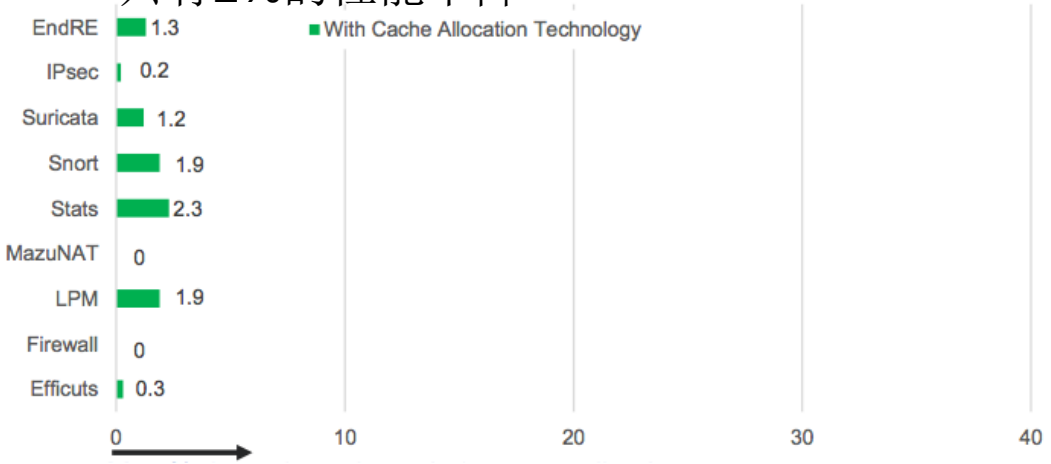


使用RDT/CAT 技术保证高优先级应用使用隔离的cache

Performance w/ CAT



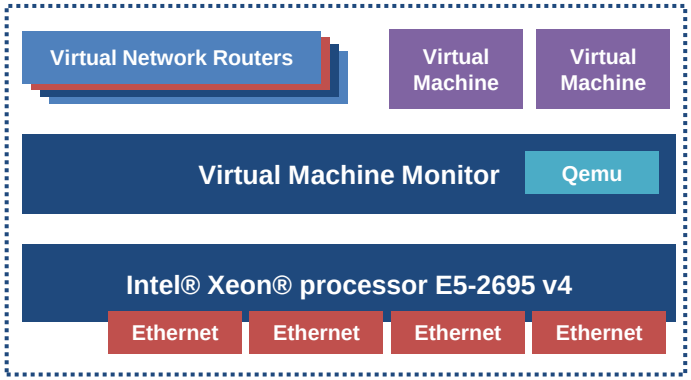
- VNF is isolated on cores.
- Package 64 bytes, 10万流量, 均匀分布
- 使用CAT技术为SNORT分配2路cache, 只有2%的性能下降



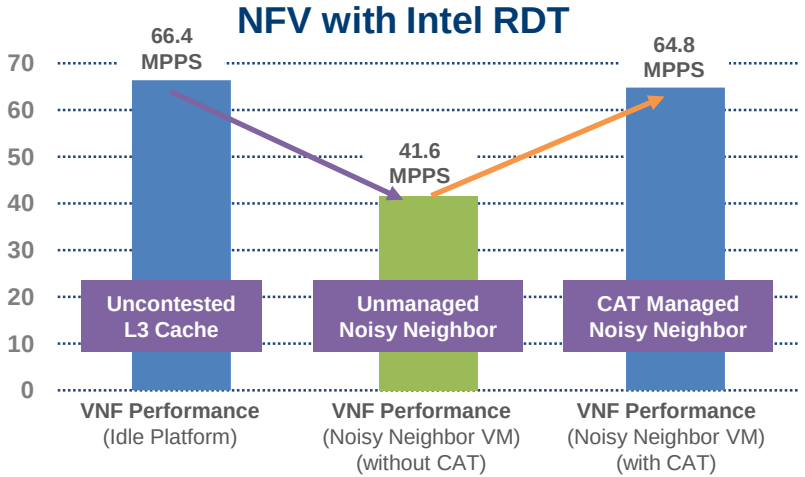
max.% throughput degradation, normalized

http://span.cs.berkeley.edu

Example: Network Function Virtualization (NFV)



Packet processing pipeline replicates
4x Network Function Virtualization

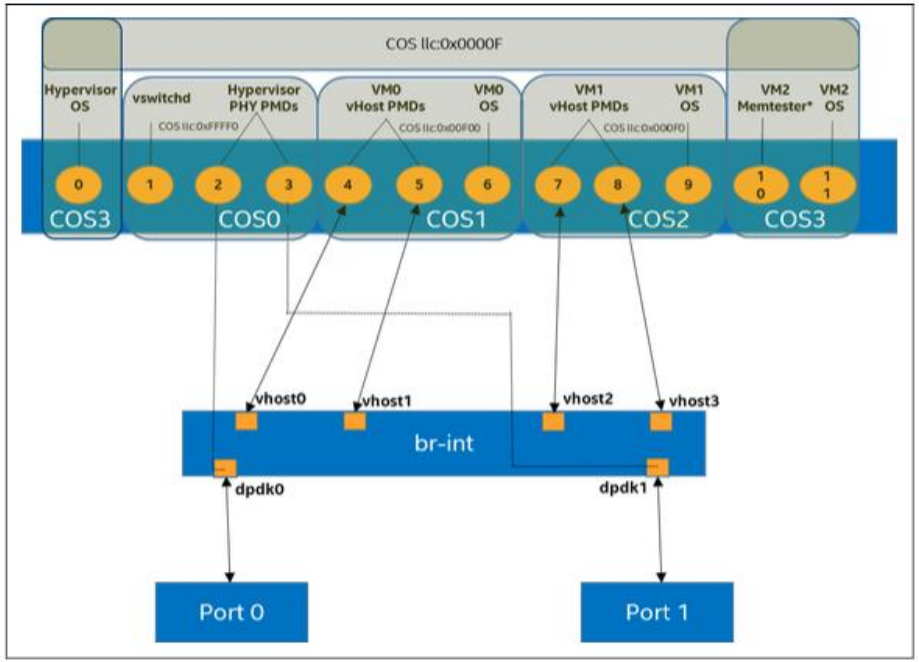


Average Latency is reduced from **36usec to 7usec** after isolation of the **noisy neighbors**

Prioritizing Apps: 38% performance degradation **restored** by utilizing CAT

RMD, (Resource Management Daemon), RDT Orchestration

RDT for NFV



See [2]

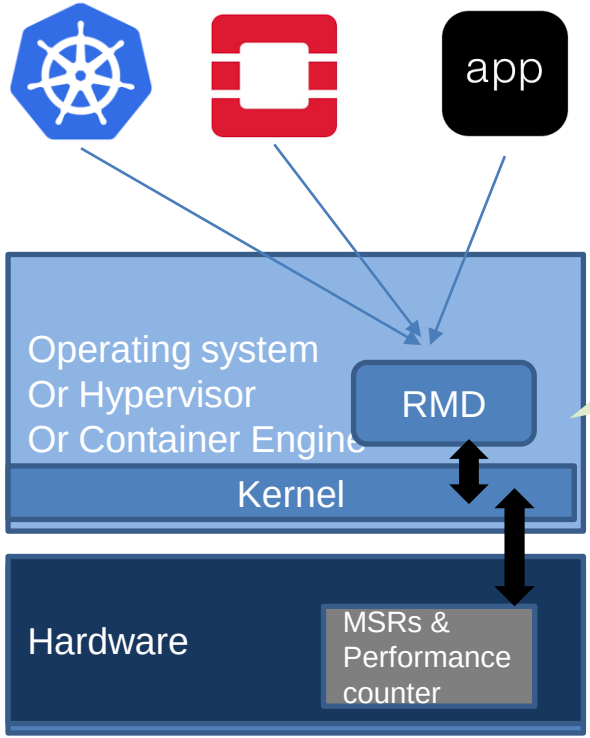
OVS + DPDK
COS: class of service

COS	CPU	MASK	APPs
COS0	1-3	FFFF0	vswitchd PMDs
COS1	4-6	00F00	VM0
COS2	7-9	000F0	VM1
CoS3	0,11-12	0000F	VM2(memtester) OS

Why RMD ?

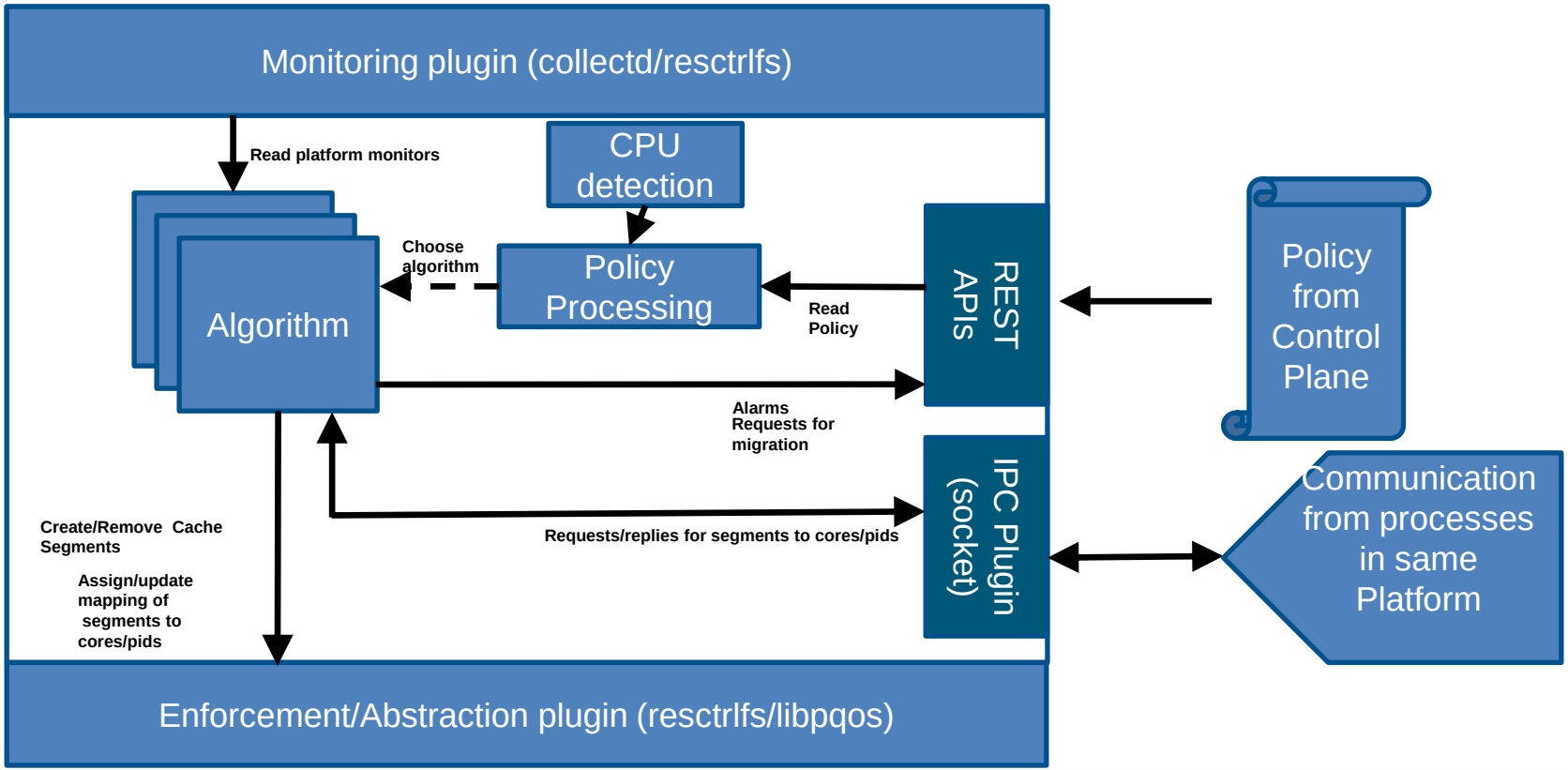
1. 使用复杂度(mask)
2. 实时调节
3. 统一平台 (cache size, bandwidth, numa)
4. 快速调节 (local policy)
5. 统一接口RDT

RMD



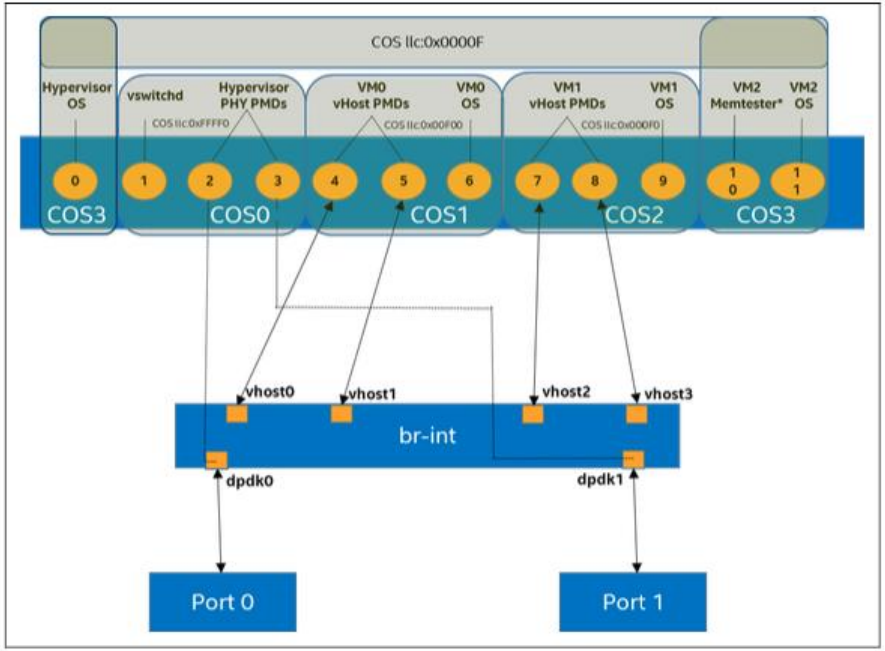
1. 易用: policy, hidden mask
2. Monitor & tuning
3. 隐藏平台
4. 本地响应
5. 统一平台入口，便于集成 OS/K8S/APP

RMD Architecture

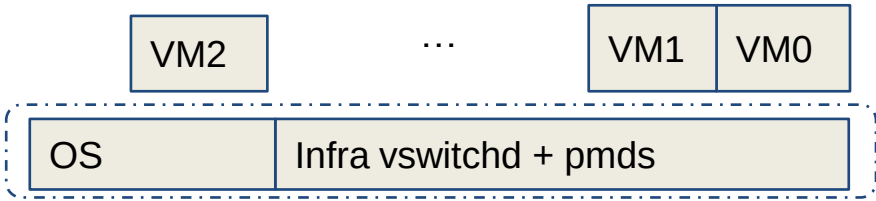


- DEMO(2 mins)

RMD For NFV

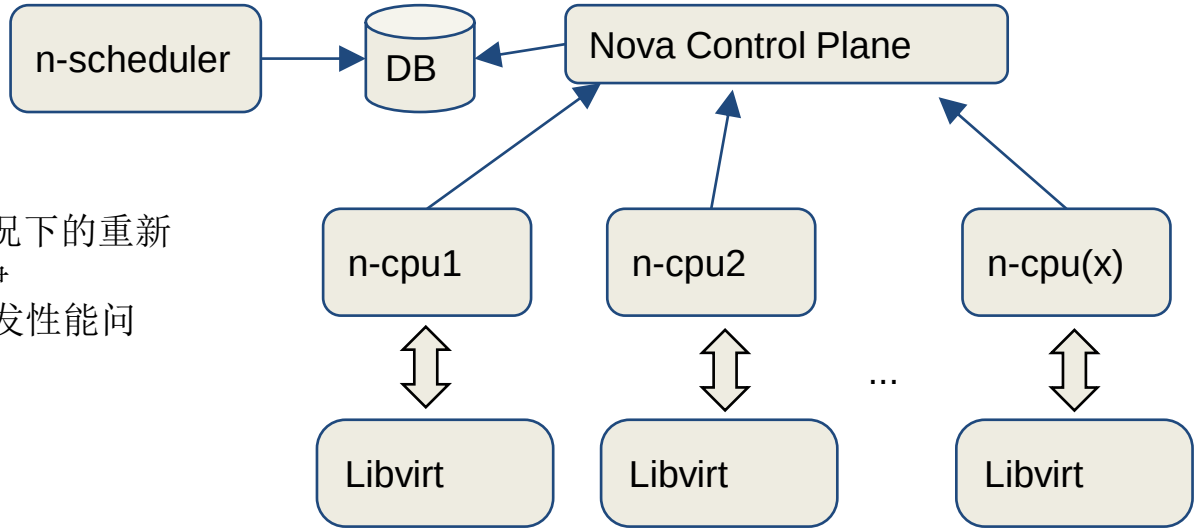


1. VNFs are isolated with each other
2. And shared with a infra group (vswitchd, PHY PMDs)



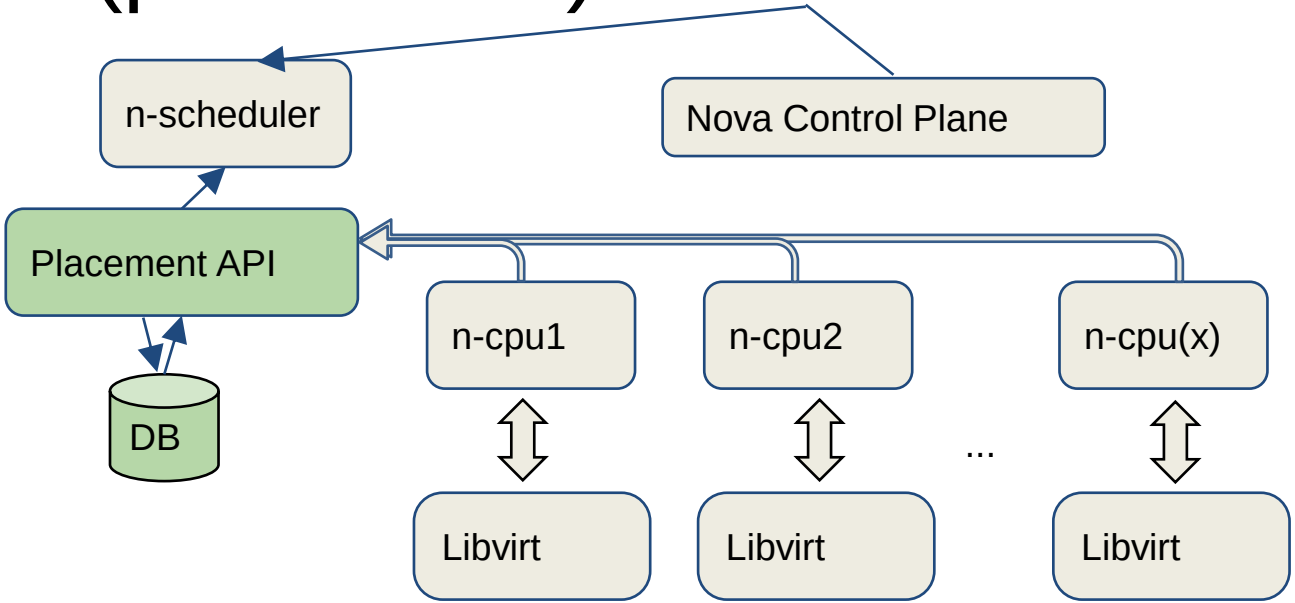
Integration with OpenStack

How Nova works (legacy)



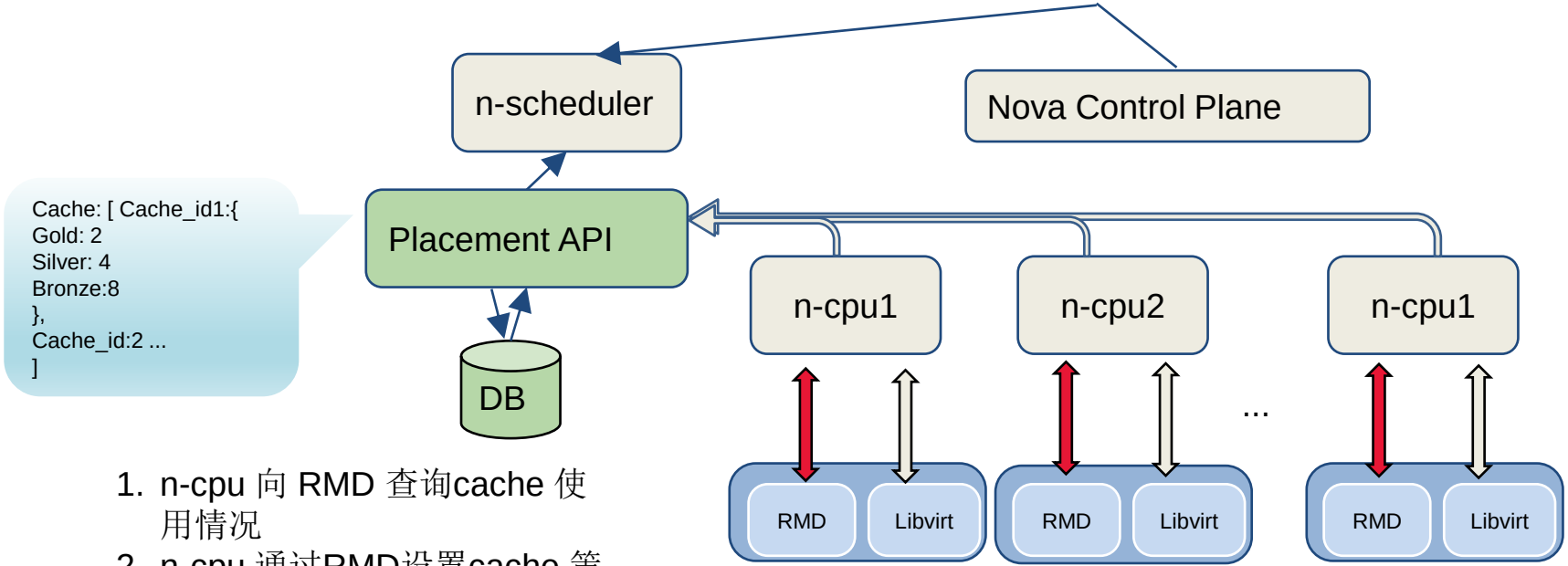
1. 多scheduler情况下的重新调度引发的竞争
2. 频繁访问DB引发性能问题
3. 扩展新资源

How Nova work with resources(placement)



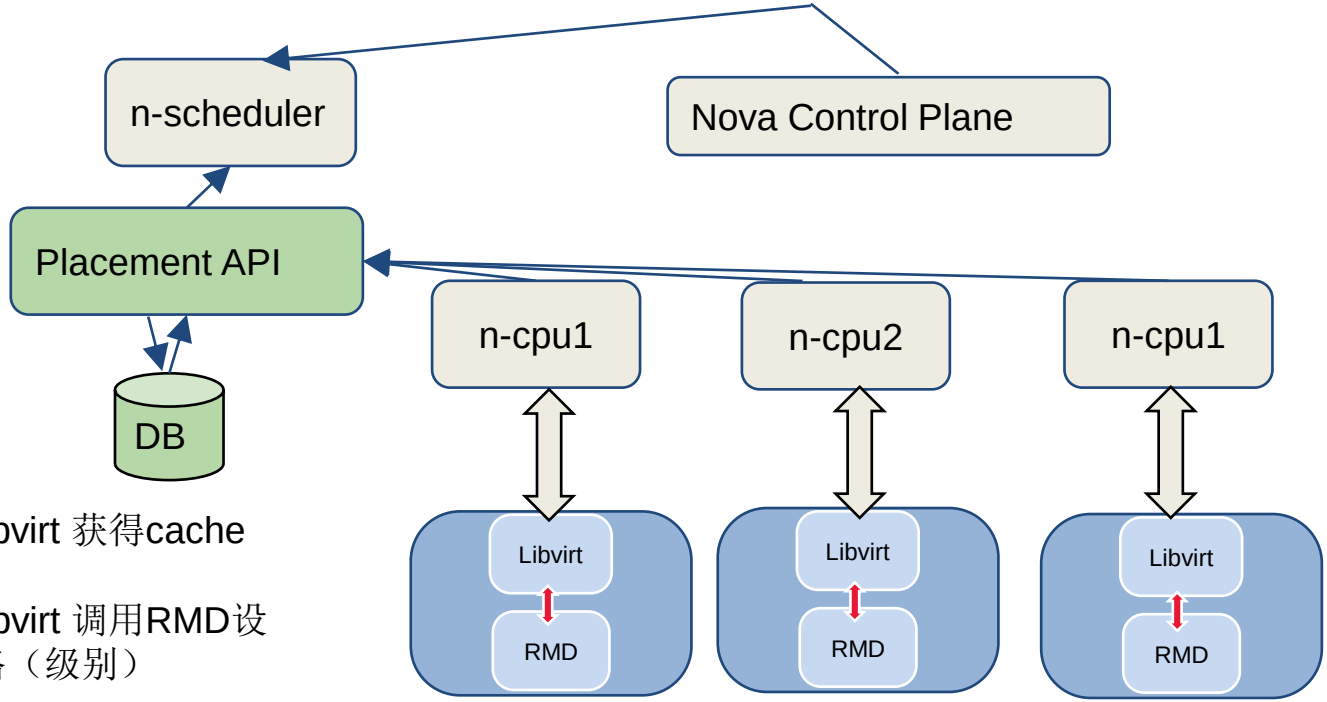
1. 减小竞争
2. 便于扩展支持新资源

Solution A: Nova Direct



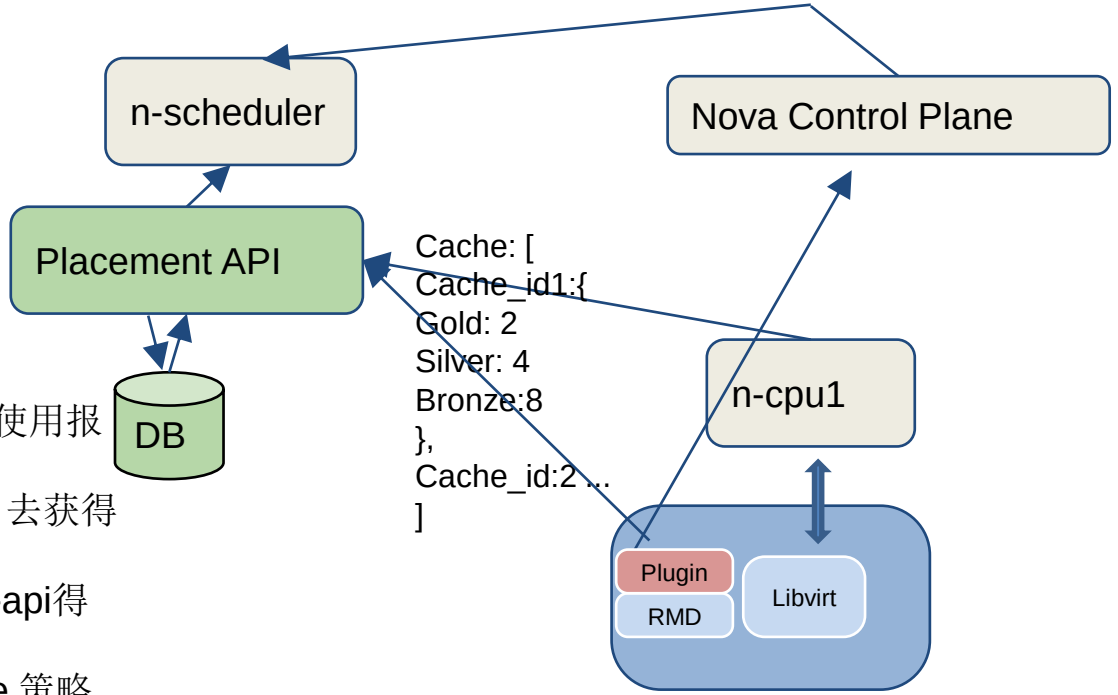
1. n-cpu 向 RMD 查询cache 使用情况
2. n-cpu 通过RMD设置cache 策略（级别）

Solution B: via Libvirt



1. n-cpu 查询 libvirt 获得cache 使用情况
2. n-cpu 通过libvirt 调用RMD设置cache 策略（级别）

Solution C: (out-of-band)



1. RMD 插件将cache 使用报告给placement
2. RMD插件轮询libvirt 去获得instance信息
3. RMD插件查询nova-api得到cache 策略
4. 调用RMD设置cache 策略

Summary

- Cloud Admin 定义cache policy
- Placement-api 管理资源
- RMD提供细粒度资源的高层次的抽象管理.
- NFV 会得益于RMD/RDT

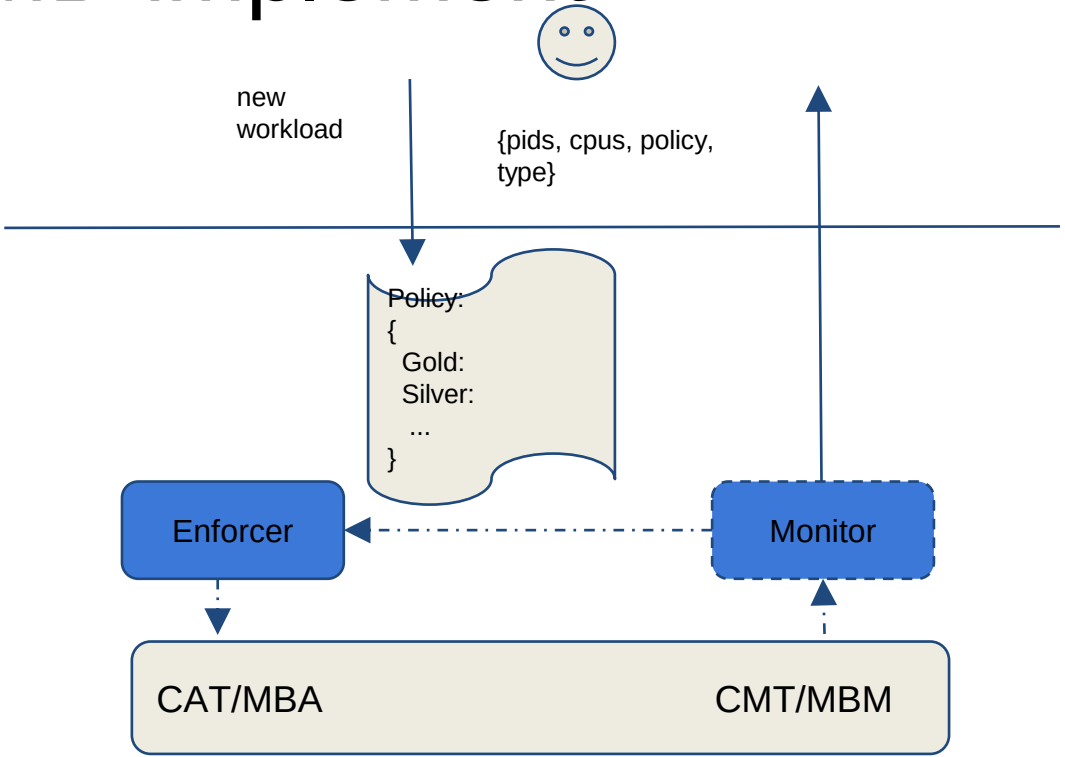
1. Intel RDT: http://www.intel.com/content/www/us/en/architecture-and-technology/resource-director-technology.html
2. NFV with RDT https://builders.intel.com/docs/networkbuilders/deterministic_network_functions_virtualization_with_Intel_Resource_Director_Technology.pdf

Intel RDT: http://www.intel.com/content/www/us/en/architecture-and-technology/resource-director-technology.html
Intel-cmt-cat: https://github.com/01org/intel-cmt-cat
Sdm: http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html

Thanks !
Q & A

Qiao, Liyong
liyong.qiao@intel.com
OTC OpenStack Core Team Intel

RMD implement



curl -H "Content-Type: application/json" --request POST --data '{"core_ids":["2-5"], "policy": "bronze"}' http://127.0.0.1:8888/v1/workloads

- Back up

https://www.juniper.net/assets/us/en/local/pdf/solutionbriefs/3510607-en.pdf

Intel® Resource Director Technology (Intel® RDT)

Building on a rich and growing portfolio of technologies embedded in Intel silicon

Cache Monitoring Tech (CMT)

- Per-thread L3 Occupancy Monitoring

Cache Allocation Tech (CAT)

- Per-thread L3 Occupancy Control
- New Code/Data Prioritization (CDP) extension

Cache

Monitoring

Allocation

Memory

Memory BW Monitoring (MBM)

- Per-thread Memory Bandwidth Monitoring
- Leverages RMID infrastructure

Memory Bandwidth Allocation (MBA)

- Per-core Bandwidth Control – more precisely control “noisy neighbors”
- New on Purley

Key Technologies for Improved Visibility and Performance Determinism

CMT/MBM

- CMT
  - Cache Monitoring Technology
- MBM
  - Memory Bandwidth Monitoring Technology

- RMID (Resource Monitor ID)
  - Track Resource (Cache, MB) usage per thread/group

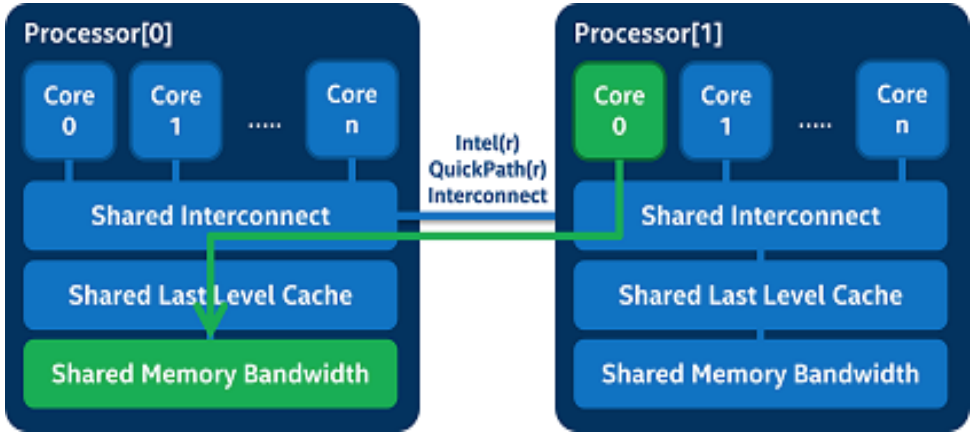
CMT

- CMT usage:
  - Long-term dynamic application **profiling** for aberration detection and software tuning/optimization
  - Precise and accurate **cache sensitivity measurement** without the need for simulators
  - Cache **contention detection** and measurement (including finding cache-starved applications or VMs amongst a large set of co-running applications/VMs)
  - Monitoring performance to **SLAs**
  - Optimal insertion of new applications on a cluster
  - Charging/bill-back
  - Administration: data can be aggregated and provided back to datacenter administrators to gauge the level of efficiency within the datacenter

MBM

MBML: Local memory bandwidth
MBMT: Total memory bandwidth

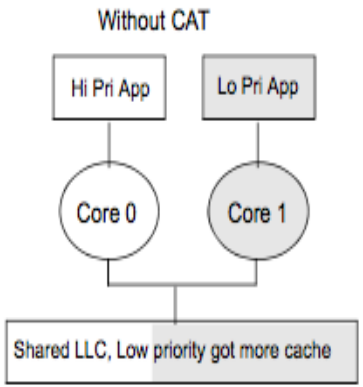
MBM provides real-time data on bandwidth usage per thread, application, VM, container or any combination Longer-term, these metrics can be provided to higher-level software including orchestration frameworks, enabling automated characterization and **resource-aware scheduling** usages.

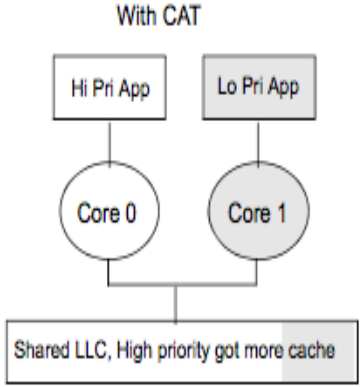


CAT

Cache Allocation Technology:
Allow high priority applications to reserve moure cache resources(i.e. cache space)

Restrict cache usage for low priority





CAT

Cache Allocation Technology

- COS: class of service
  Number limitation (16 per Socket/Package)
- CBM: capability bit mask
  Bit mask to indicate cache allocated

	M7	M6	M5	M4	M3	M2	M1	M0
COS0	A	A	A	A	A	A	A	A
COS1	A	A	A	A	A	A	A	A
COS2	A	A	A	A	A	A	A	A
COS3	A	A	A	A	A	A	A	A

Default Bitmask

	M7	M6	M5	M4	M3	M2	M1	M0
COS0	A	A	A	A	A	A	A	A
COS1					A	A	A	A
COS2							A	A
COS3								A

Overlapped Bitmask

	M7	M6	M5	M4	M3	M2	M1	M0
COS0	A	A	A	A				
COS1					A	A		
COS2							A	
COS3								A

Isolated Bitmask

CDP

CDP enables isolation and separate prioritization of code and data fetches to the L3 cache

Example of CAT-Only Usage - 16 bit Capacity Masks

COS0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
COS1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
COS2	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
COS3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Traditional CAT

Example of Code/Data Prioritization Usage - 16 bit Capacity Masks

COS0.Data	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
COS0.Code	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
COS1.Data	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
COS1.Code	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
Other COS.Data	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
Other COS.Code	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

CAT with CDP

Opens for CAT

- The allocation cache size important? - depends on platform (inclusive vs non-inclusive)
- Dynamic policy for CAT?
- Use case except NFV，public cloud

Interface:
Resctrl or Intel-cmt-cat ?
Solution:
RDTAgent

RDT Support Matrix

	CMT	MBM	L3 CAT	L3 CDP	L2 CAT
Intel(R) Xeon(R) processor E5 v3	Yes	No	Yes	No	No
Intel(R) Xeon(R) processor D	Yes	Yes	Yes	No	No
Intel(R) Xeon(R) processor E3 v4	No	No	Yes	No	No
Intel(R) Xeon(R) processor E5 v4	Yes	Yes	Yes	Yes	No
Intel(R) Atom(R) processor for Server C3000	No	No	No	No	Yes

CAT

- How does it work in linux (4.10 and later)?

mount -t resctrl resctrl /sys/fs/resctrl

1. create COS
2. change CBM for COS
3. associate COS <-> Tasks/CPU core

P. S. Another user space tool
Intel-cmt-cat:https://github.com/01org/intel-cmt-cat

root@s2600wt:/sys/fs/resctrl# mkdir foo

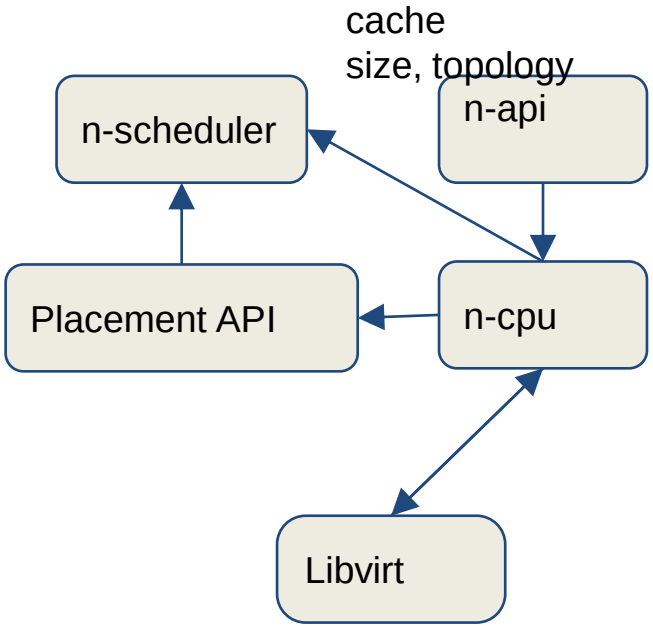
root@s2600wt:/sys/fs/resctrl# echo "L3:0=ff;1=ff" > foo/schemata

root@s2600wt:/sys/fs/resctrl# echo "16933" > foo/tasks

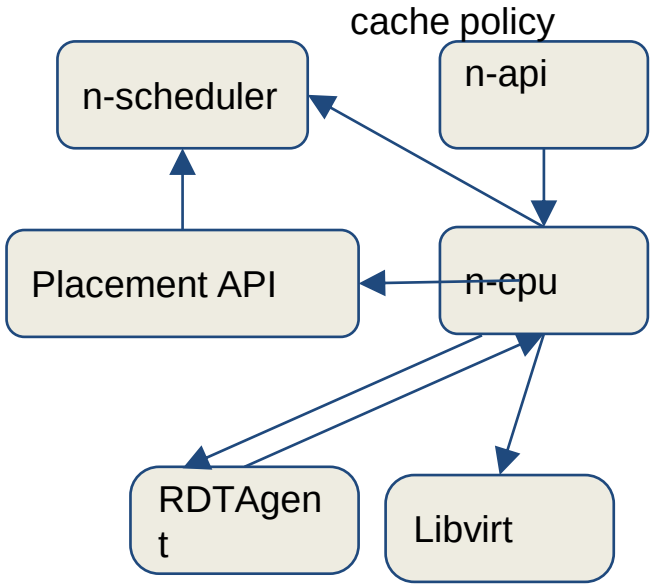
Upstream status

Feature	Status	Software
CMT/MBM(perf)	Released	Kernel 4.1, Libvirt 3.0.0
CAT (resctrl)	Released	Kernel 4.10, Libvirt (on-going)
CMT/MBM (resctrl)	On-going	Kernel 4.13
MBA (resctrl)	On-going	Kernel 4.12

How Nova work with resources



1 static allocation



1 policy based allocation