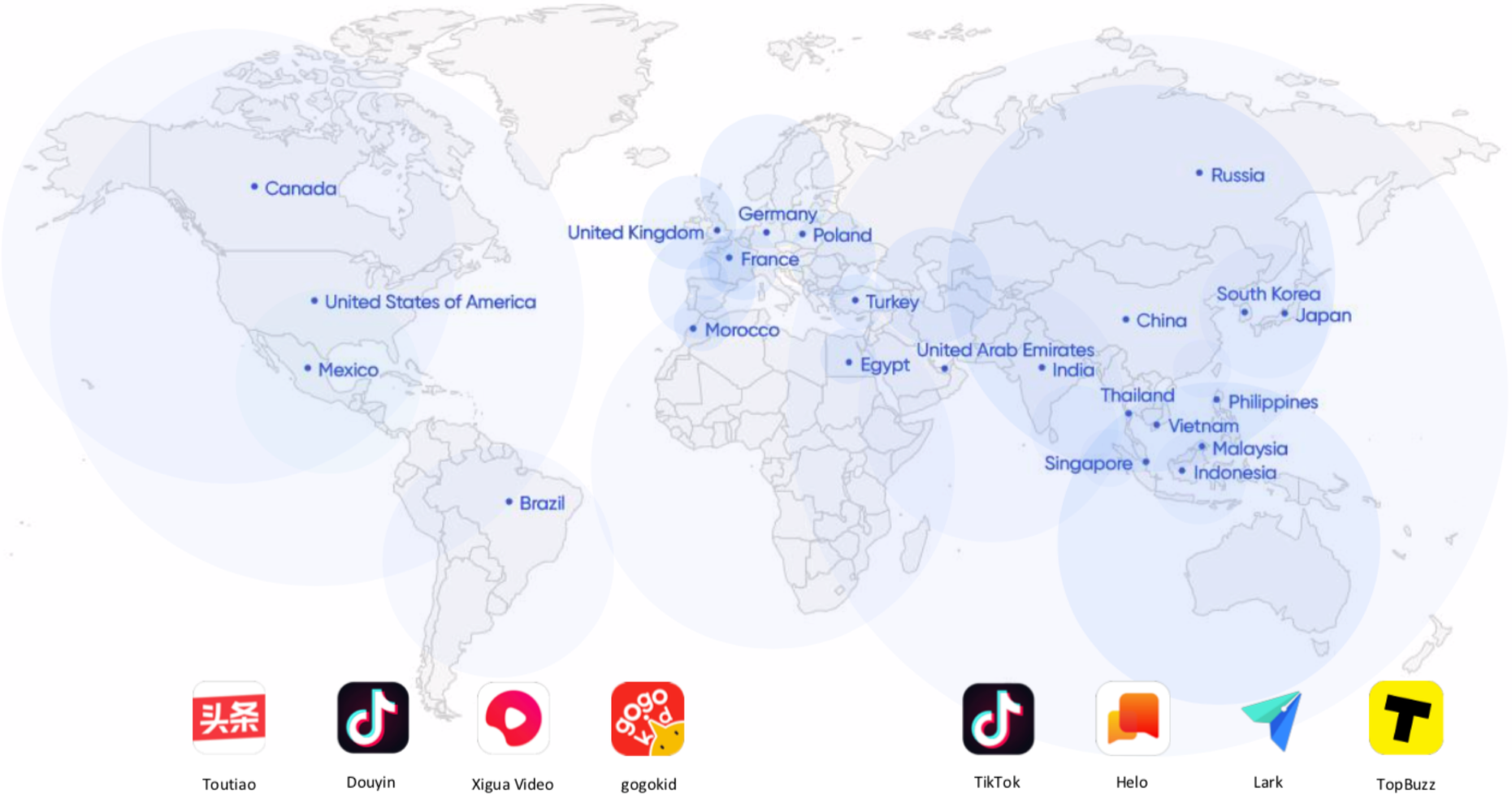
How KVM-based Hybrid Deployment Powers ByteDance's Biggest Day Ever

Lu Ye & Zhenwei Pi

ByteDance

Global Footprint

ByteDance has 50+ offices in over 30 countries and regions.
Products in 150 markets and 75 languages



Agenda

- Background
- Why virtualization
- Improvements
- Achievements

Background

ByteDance

Background

Challenges for Infra team to support Spring Festival Gala Events

- All resources are used out
- Request for millions of cores in short term

Background

CPU/Mem usage is low on object storage servers.

Hybrid deployment is the answer.

Background

High isolation level of various resource is required:

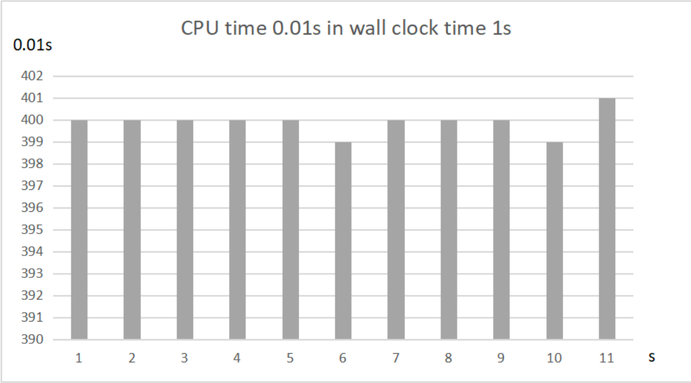
- Scheduler
- Memory
- I/O
- Network
- ...

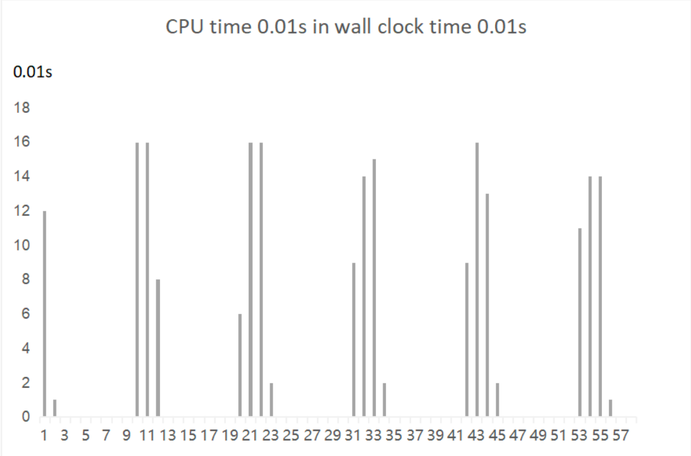
Why virtualization

ByteDance

Scheduler

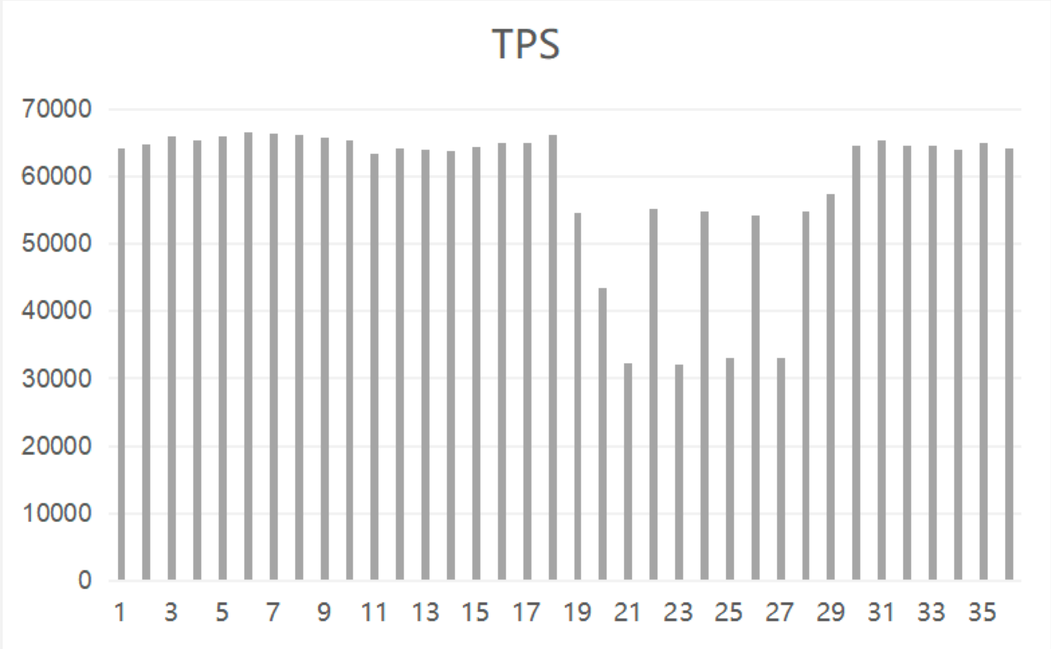
Problems with CPU quota in cgroup:





16 threads run in 4 CPU

The CPU time in 1s/0.01s

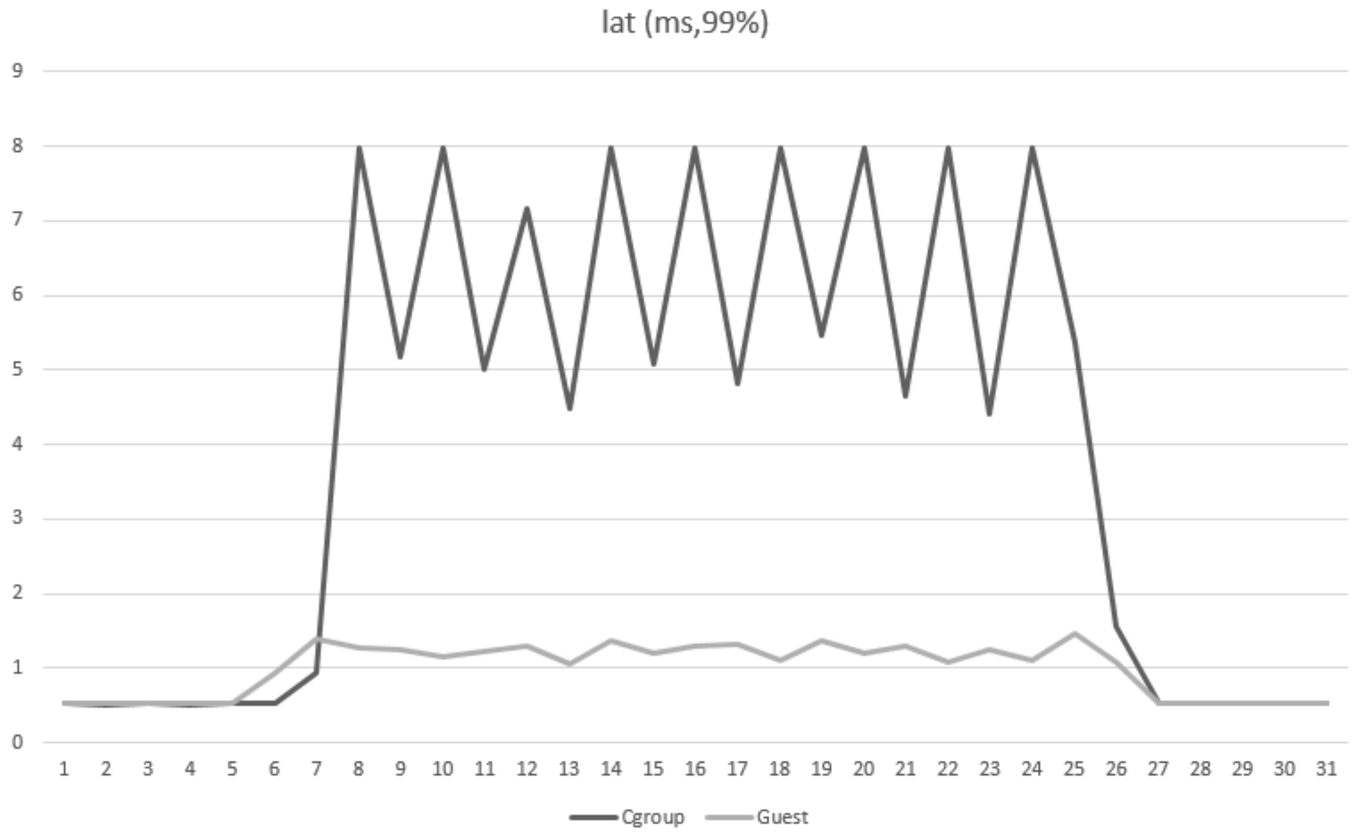


MySQL performance drop

Scheduler

Problems with CPU quota in cgroup:

While guest workloads increase intensively, host service latency comparation using virtualization vs cgroup

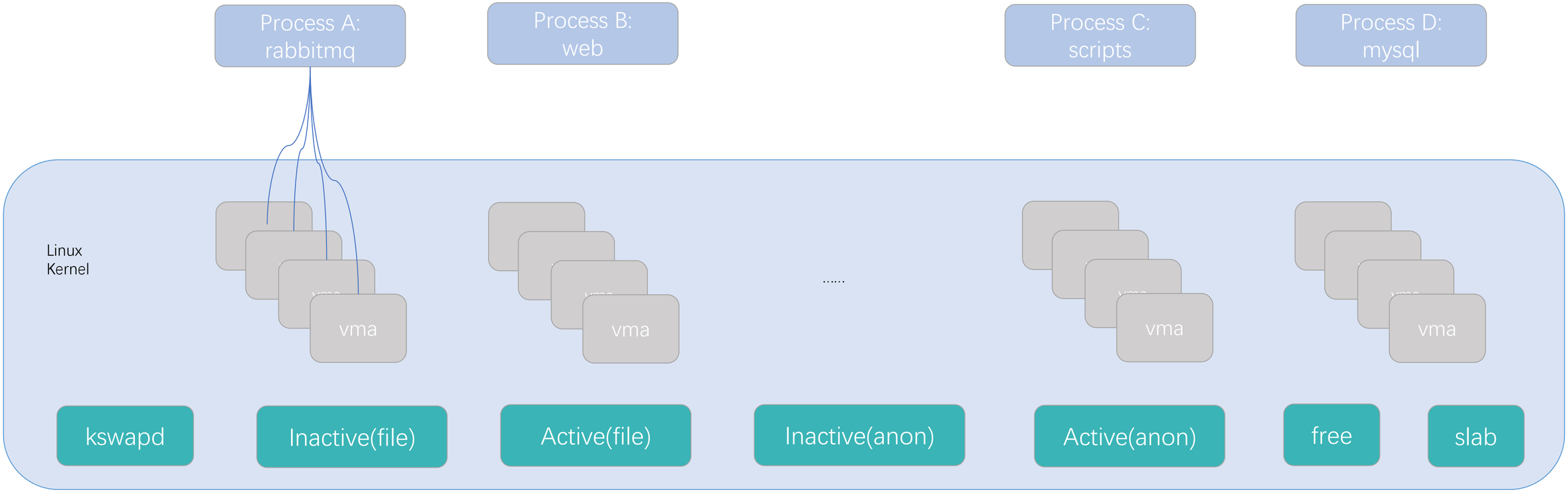


ByteDance

Memory

Problem with memory management without virtualization

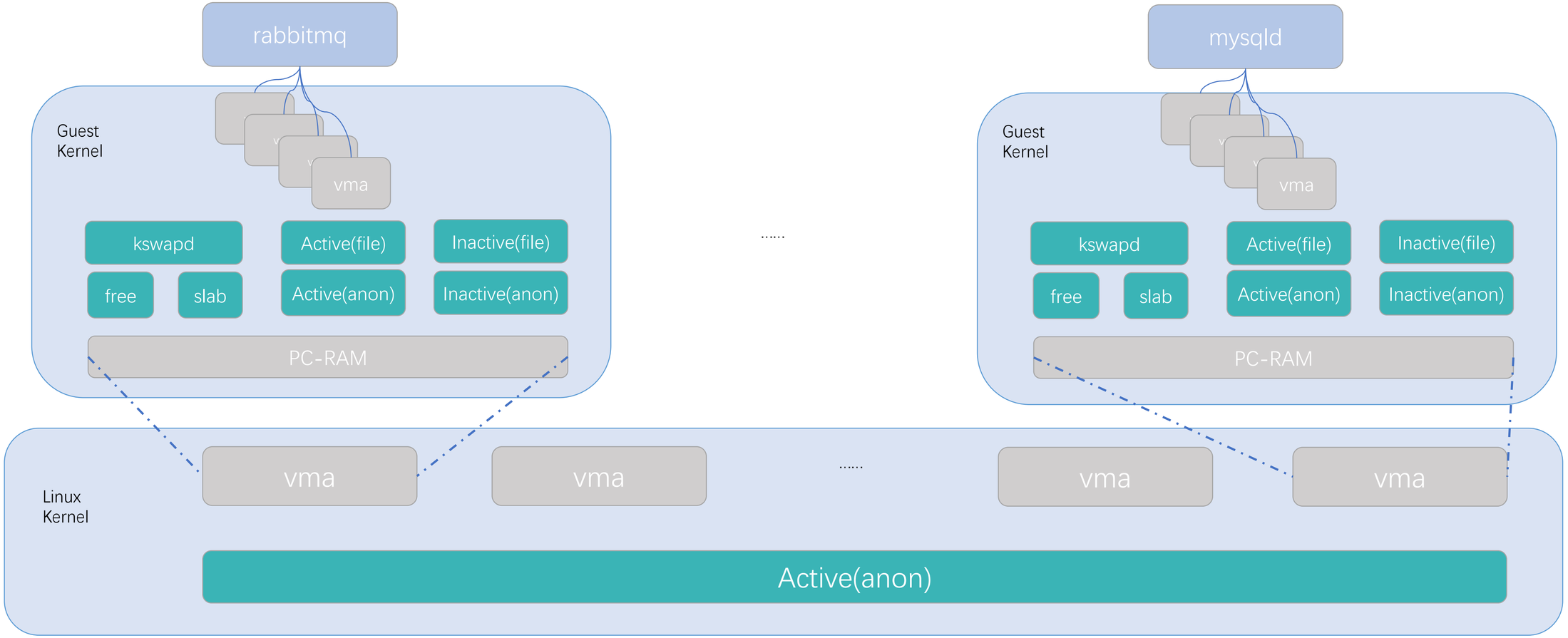
- shared page cache
- kswapd scan and reclaim memory system wide



Memory

page cache is fully isolated

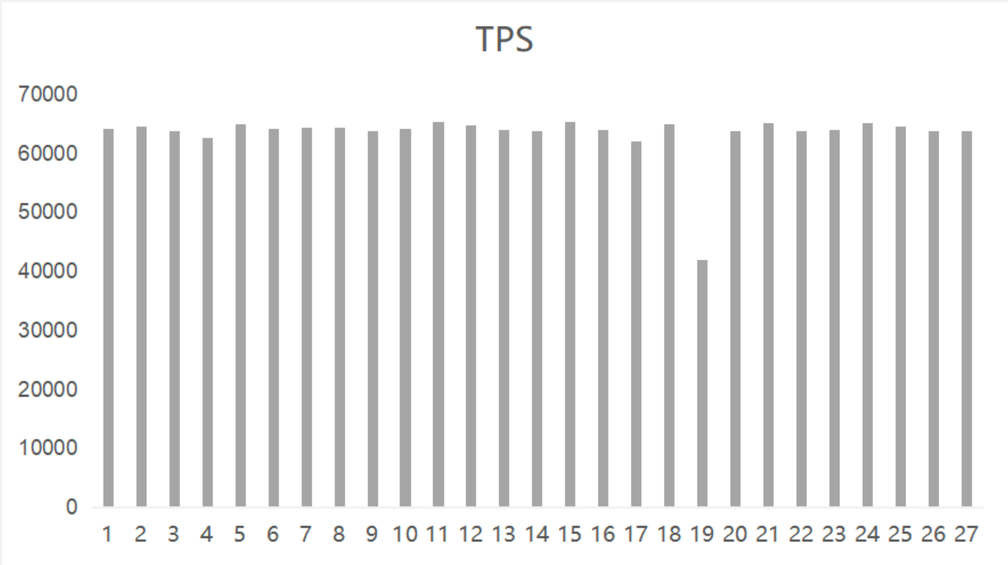
kswapd is limited in per VM



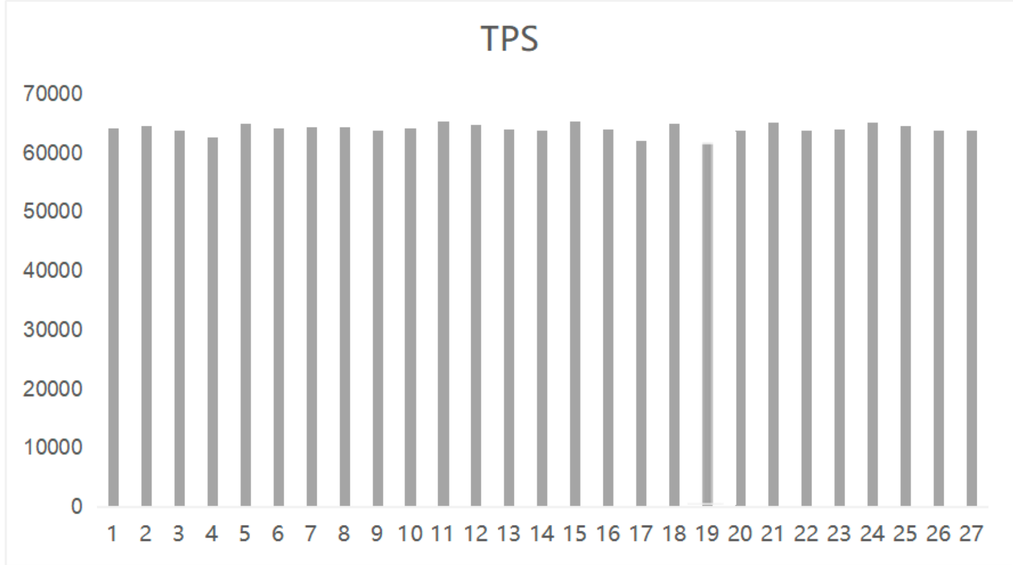
IO

Problem: implicit critical resource race, eg: ext-4 journal.

Case: TPS drop in mysql while delete a huge file on mysql server



Cgroup



Virtualization

Improvements

ByteDance

kvm-utils for trouble shooting

Key point:

find VM-exit reasons without kernel/kmod upgrading

benchmark performance for main VM-exit reasons

Our solution:

base on kprobe

microbenchmark for PIO, MMIO, TIMER, IPI …

kvm-utils: open source soon

kvm-utils for trouble shooting

```
VM EXIT REASON STATISTIC
        TOTAL EXITS : 159551
                EXIT_REASON_EXCEPTION_NMI : 10
            EXIT_REASON_EXTERNAL_INTERRUPT : 8352
                        EXIT_REASON_CPUID : 1221
                          EXIT_REASON_HLT : 48889
                       EXIT_REASON_VMCALL : 27
            EXIT_REASON_IO_INSTRUCTION : 24
                  EXIT_REASON_MSR_READ : 38
                 EXIT_REASON_MSR_WRITE : 73877
                EXIT_REASON_EPT_MISCONFIG : 11
            EXIT_REASON_PREEMPTION_TIMER : 9142
```

```
WRMSR STATISTIC
        [MSR_IA32_APICBASE] 0
        [MSR_IA32_TSC_ADJUST] 0
        [MSR_IA32_TSCDEADLINE] 28583
        [MSR_IA32_MISC_ENABLE] 0
        [MSR_IA32_MCG_STATUS] 0
        [MSR_IA32_MCG_CTL] 0
        [MSR_IA32_MCG_EXT_CTL] 0
        [MSR_IA32_SMBASE] 0
        [MSR_PLATFORM_INFO] 0
        [MSR_MISC_FEATURES_ENABLES] 0
        [MSR_KVM_WALL_CLOCK] 0
        [MSR_KVM_SYSTEM_TIME] 0
        [MSR_CORE_PERF_FIXED_CTR0] 0
        [MSR_CORE_PERF_FIXED_CTR1] 0
        [MSR_CORE_PERF_FIXED_CTR2] 0
        [MSR_CORE_PERF_FIXED_CTR_CTRL] 0
        [MSR_CORE_PERF_GLOBAL_STATUS] 0
        [MSR_CORE_PERF_GLOBAL_CTRL] 0
        [MSR_CORE_PERF_GLOBAL_OVF_CTRL] 0
        [MSR_OTHERS] 0
APIC STATISTIC
        [APIC_TASKPRI] 0
        [APIC_EOI] 0
        [APIC_LDR] 0
        [APIC_DFR] 0
        [APIC_SPIV] 0
        [APIC_ICR] 60008
        [APIC_ICR2] 0
        [APIC_LVT0] 0
        [APIC_LVTT] 0
        [APIC_SELF_IPI] 0
        [APIC_OTHERS] 0
```

kvm-utils for trouble shooting

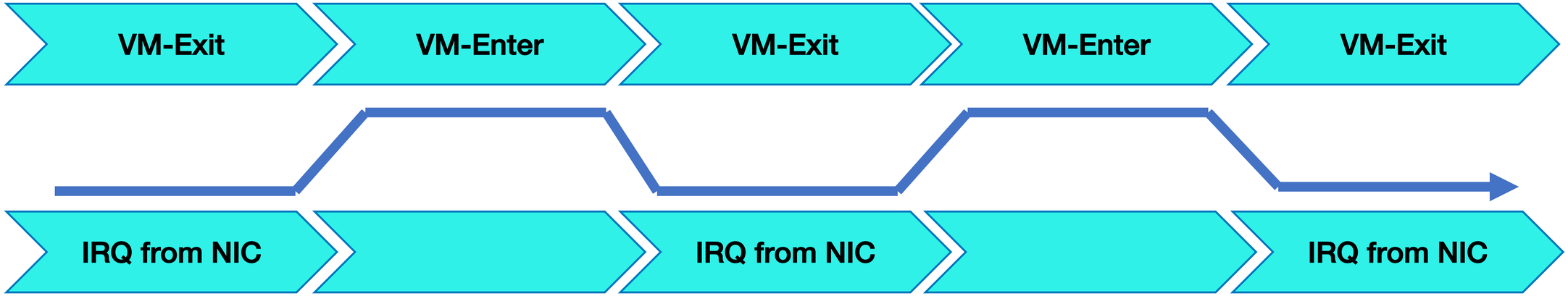
Case 1:

PIO VM-exit increased intensively. It turned out guest OS clock source is set as acpi-pm wrongly.

Case 2:

wrmsr TSC DEADLINE vm-exit over 1M/s, the reason was that tcp congestion control bbr writes timer a lot.

Problem: too many VM-exits by HLT/PI wakeup



Guest 36 vCPUs / Host 48 CPUs
8 queues / NIC(VF) binding vCPU 0~7
150K ~ 500K interrupts/s

nohlt_list for system side

Key point:

reduce VM-exits by HLT/PI wakeup
avoid performance drop by HT polling
balance host/guest CPU consumption

“nohlt_list” kernel parameter:

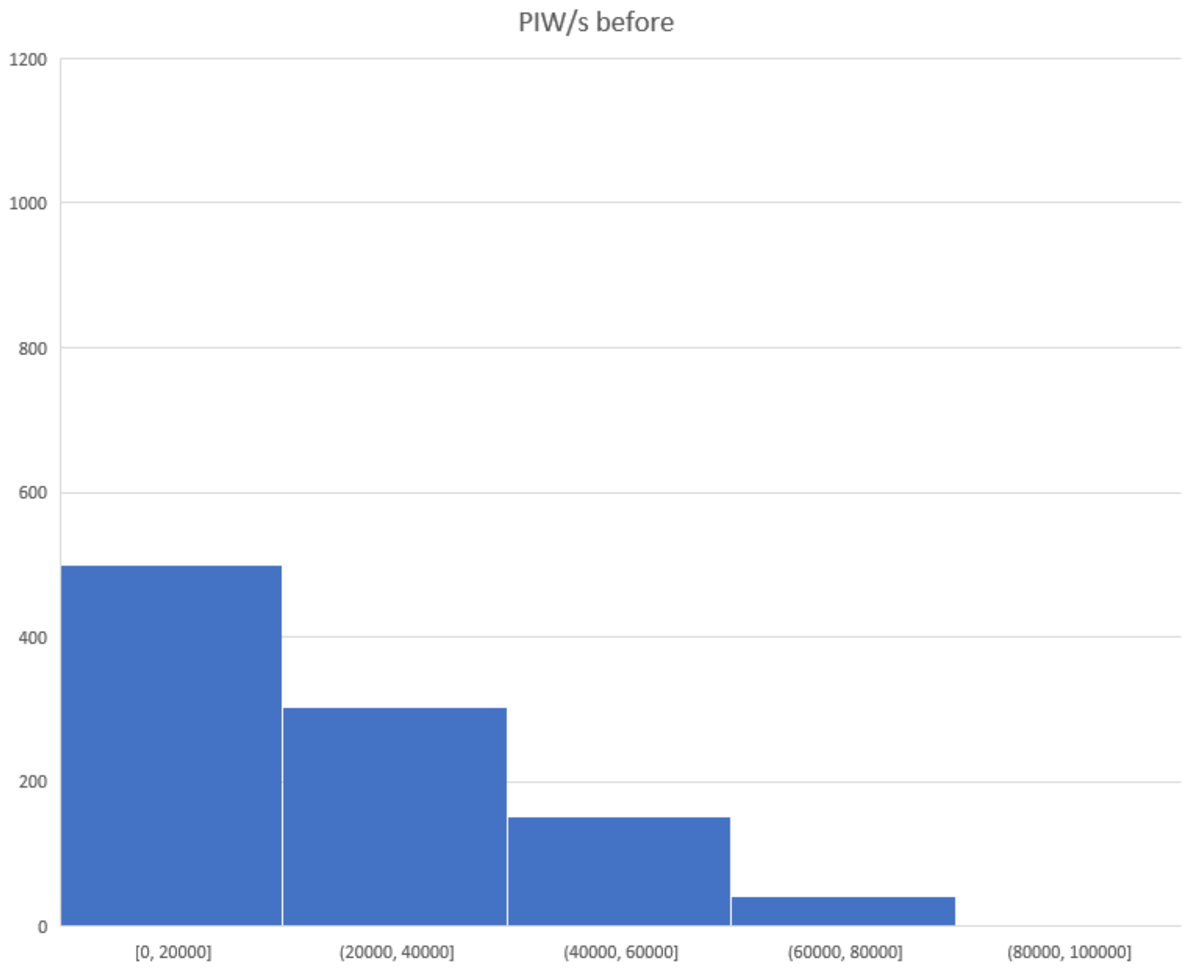
allowing the specified CPU(s) to run in polling mode.

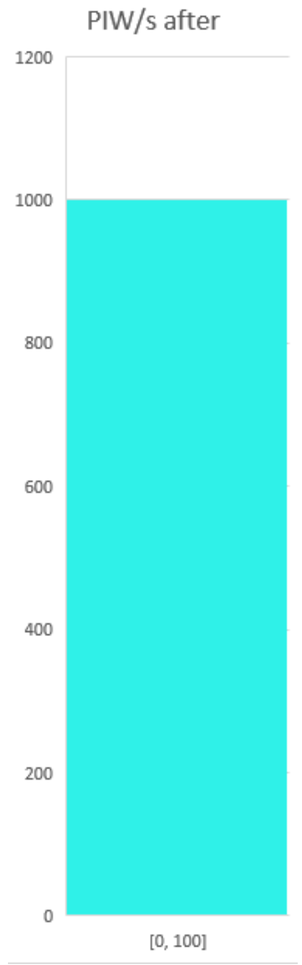
eg: linux ... irqaffinity=0-7 nohlt_list=0-7

Patch: https://lkml.org/lkml/2019/5/22/164

nohlt_list for system side

Random 1K online servers PIW/s distribution





Problem: performance drop amplified in VM

Key point:

reduce VM-exit by collecting IPC by vPMU(wrmsr/rdpmc)
reduce VM-exit by TLB shootdown
compat with bare metal

Our solution:

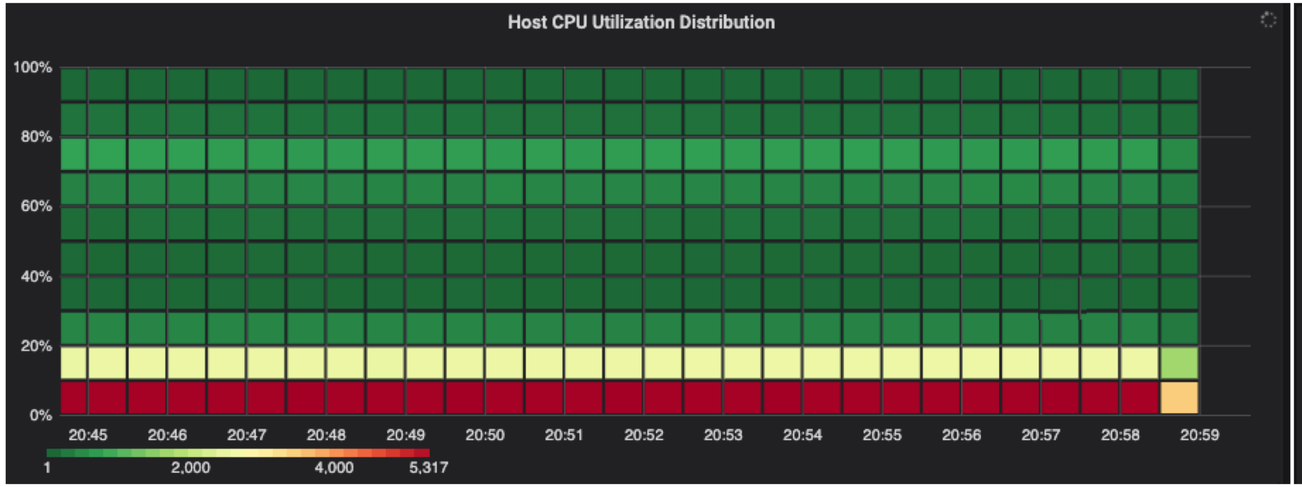
adjust software used in guest OS, eg: atop, jemalloc

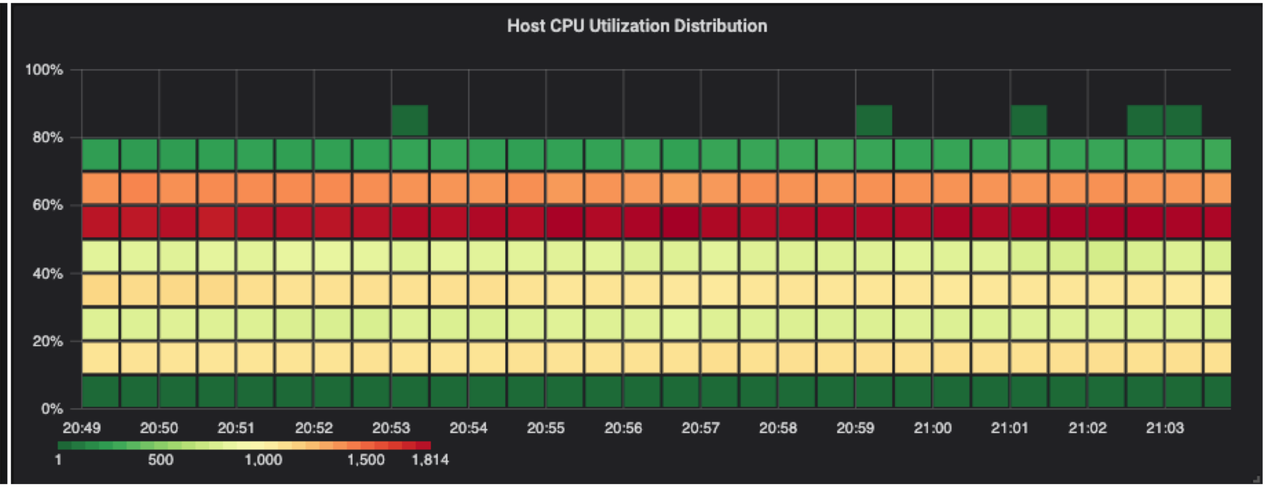
Patch:
https://github.com/Atoptool/atop/commit/16abcac132eec4755373aa673389e67219488844

Achievement

ByteDance

Host CPU utilization distribution





Before hybrid deployment → After hybrid deployment

Dataset: 7000 hosts

ByteDance

Thank You

ByteDance

ByteDance