Status Update About COLO FT

www.huawei.com

Hailiang Zhang (Huawei)
Randy Han (Huawei)

HUAWEI TECHNOLOGIES CO., LTD.

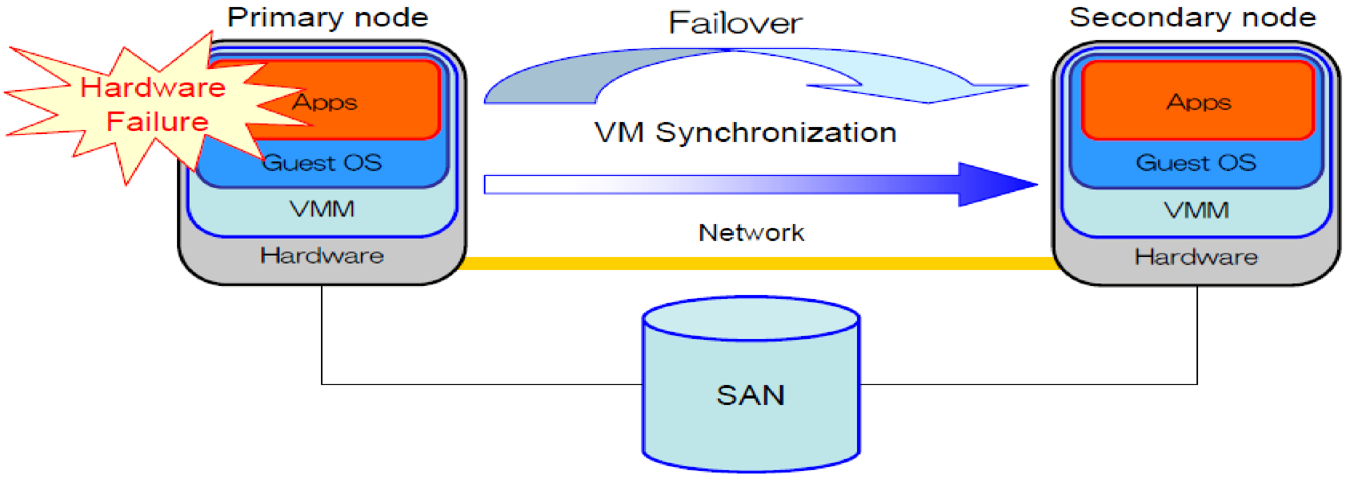
Agenda

- Introduce COarse-grain LOck-stepping
- COLO Design and Technology Details
- Current Status Of COLO In KVM
- Further Work About COLO

Non-Stop Service with VM Replication

Virtual Machine (VM) replication

➢ A software solution for business continuity and disaster recovery through application-agnostic hardware fault tolerance by replicating the state of primary VM (PVM) to secondary VM (SVM) on different physical node.



Existing VM Replication Approaches

- **Replication Per Instruction: Lock-stepping**
  - Execute in parallel for deterministic instructions
  - Lock and step for un-deterministic instructions
- **Replication Per Epoch: Continuous Checkpoint**
  - Secondary VM is synchronized with Primary VM per epoch
  - Output is buffered within an epoch

Problems

- **Lock-stepping**
  - Excessive replication overhead
    - ⑩memory access in an MP-guest is un-deterministic
- **Continuous Checkpoint**
  - Extra network latency
  - Excessive VM checkpoint overhead

What Is COLO

- **VM and Clients model**
  - VM and Clients are a system of networked request-response system
  - Clients only care about the response from the VM
- **COarse-grain LOck-stepping VMs (COLO)**
  - PVM and SVM execute in parallel
  - Duplicates client’s request stream to SVM
  - Compare the output packets from PVM and SVM
  - Synchronize SVM state with PVM when their responses (network packets) are not identical

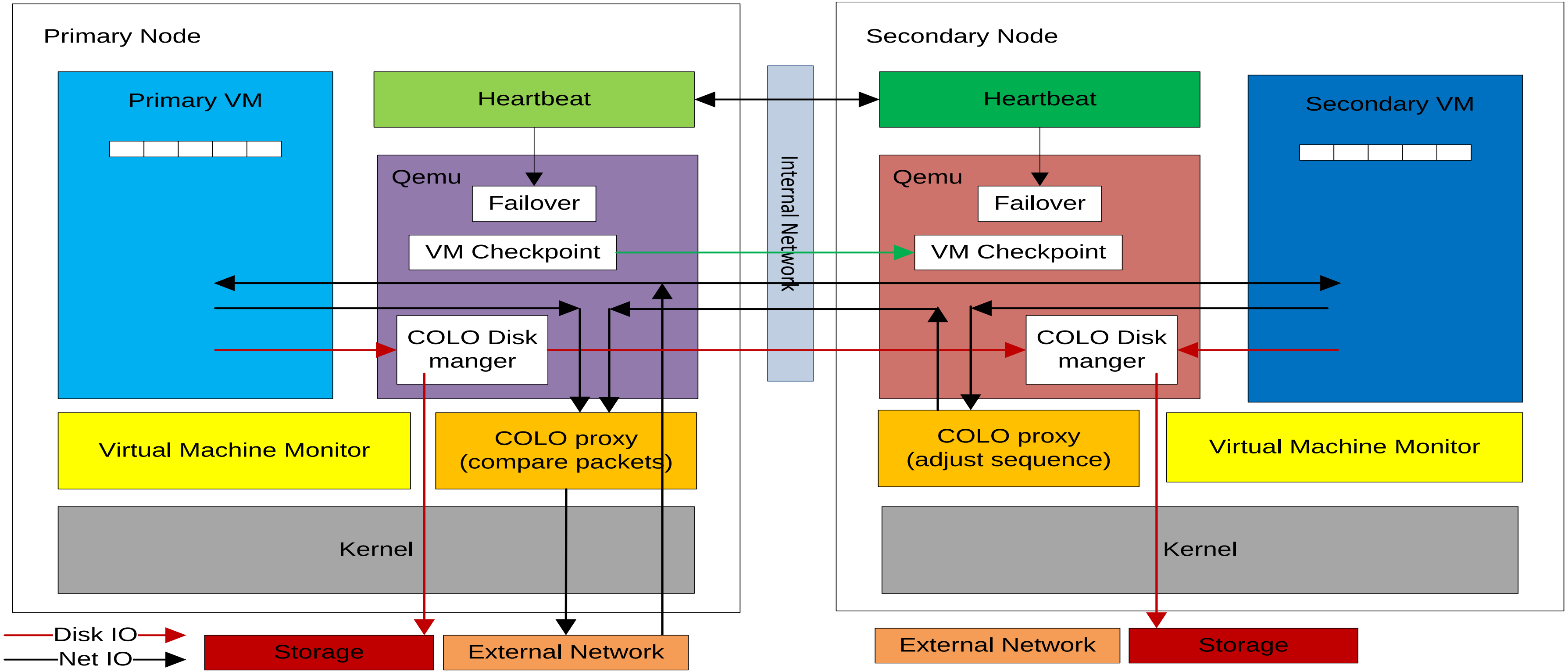
Why Is COLO Better

- **Comparing with Continuous VM checkpoint**
  - No buffering-introduced latency
  - Less checkpoint frequency
    - On demand vs periodic
- **Comparing with lock-stepping**
  - Eliminate excessive overhead of un-deterministic instruction execution due to MP-guest memory access

Agenda

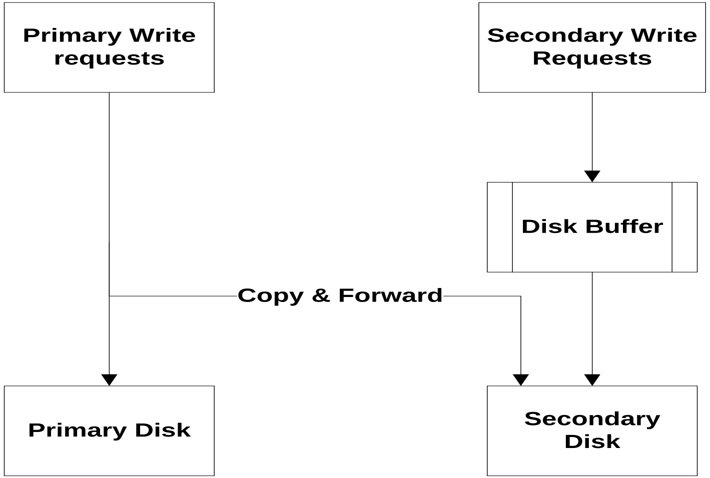
- □ Introduce COarse-grain LOck-stepping
- □ COLO Design and Technology Details
- □ Current Status Of COLO In KVM
- □ Further Work About COLO

Architecture Of COLO

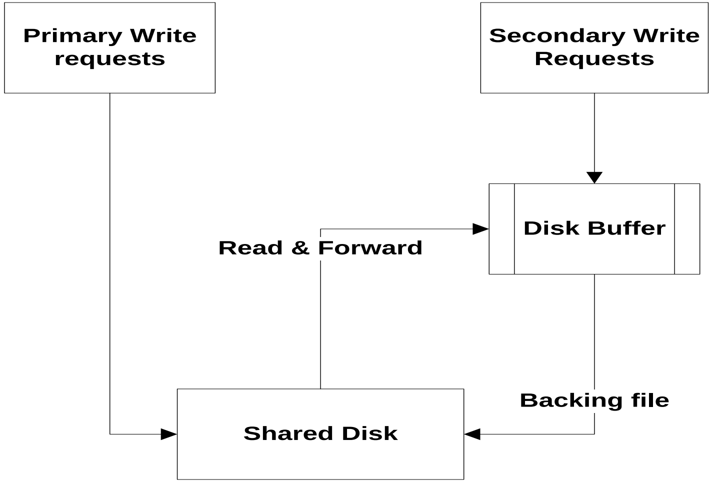


COarse-grain LOck-stepping Virtual Machine for Non-stop Service

How Block Replication Work



Non-Shared disk workflow



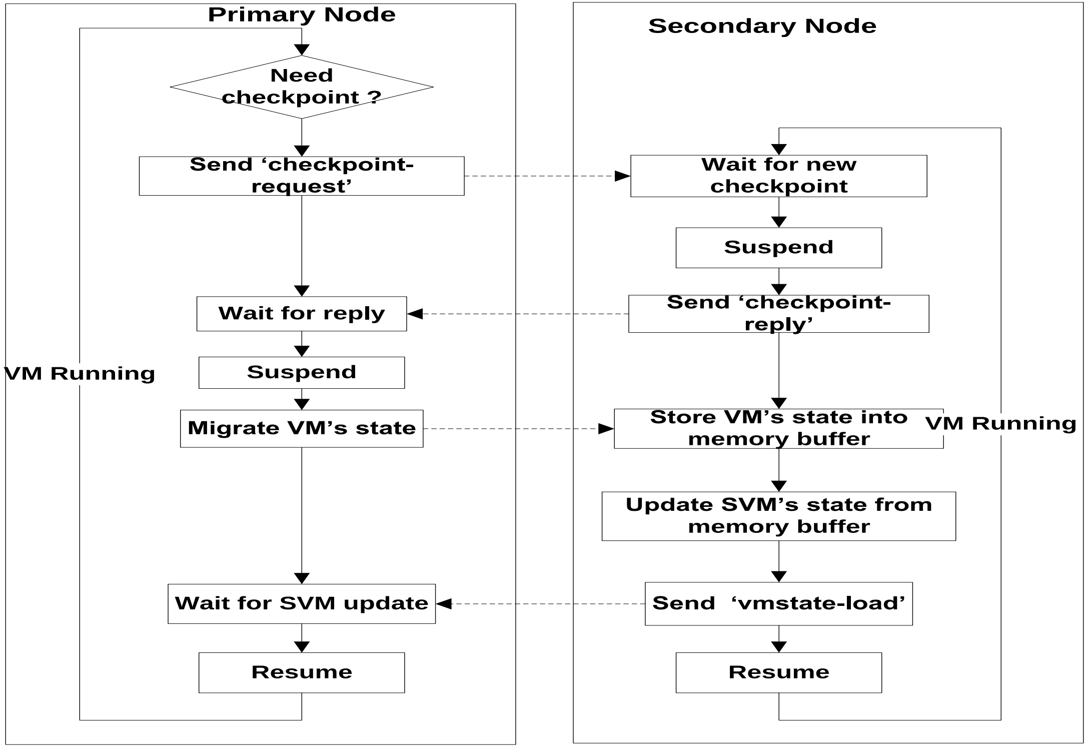
Shared disk workflow

From SVM’s point of view: Its storage is disk-buffer whose parent backing file is Secondary Disk (Or Shared Disk)

Checkpoint: Disk buffer will be emptied to achieve block replication

Failover: Disk buffer will be written back to the ‘parent’ disk

VM State Checkpointing



Execution and Checkpoint Flow in COLO

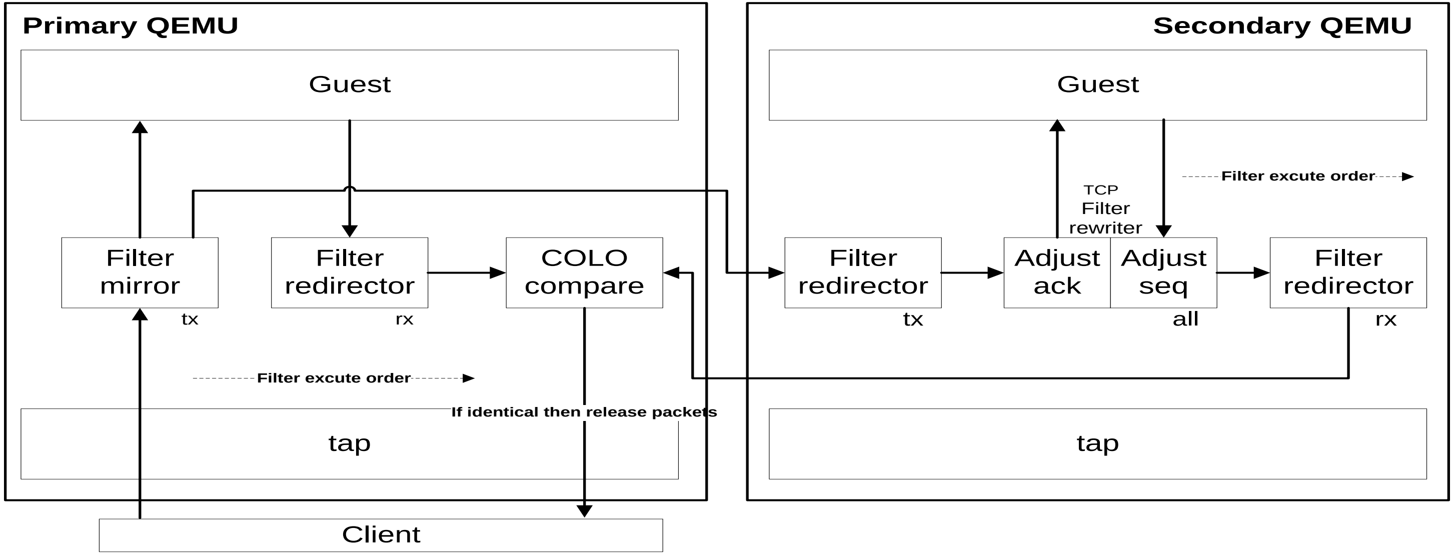
- ➢ Based on live migration
- ➢ PVM’s memory/device data be stored in extra memory-buffer of SVM before be synchronized to SVM

COLO Proxy Design

Scheme:

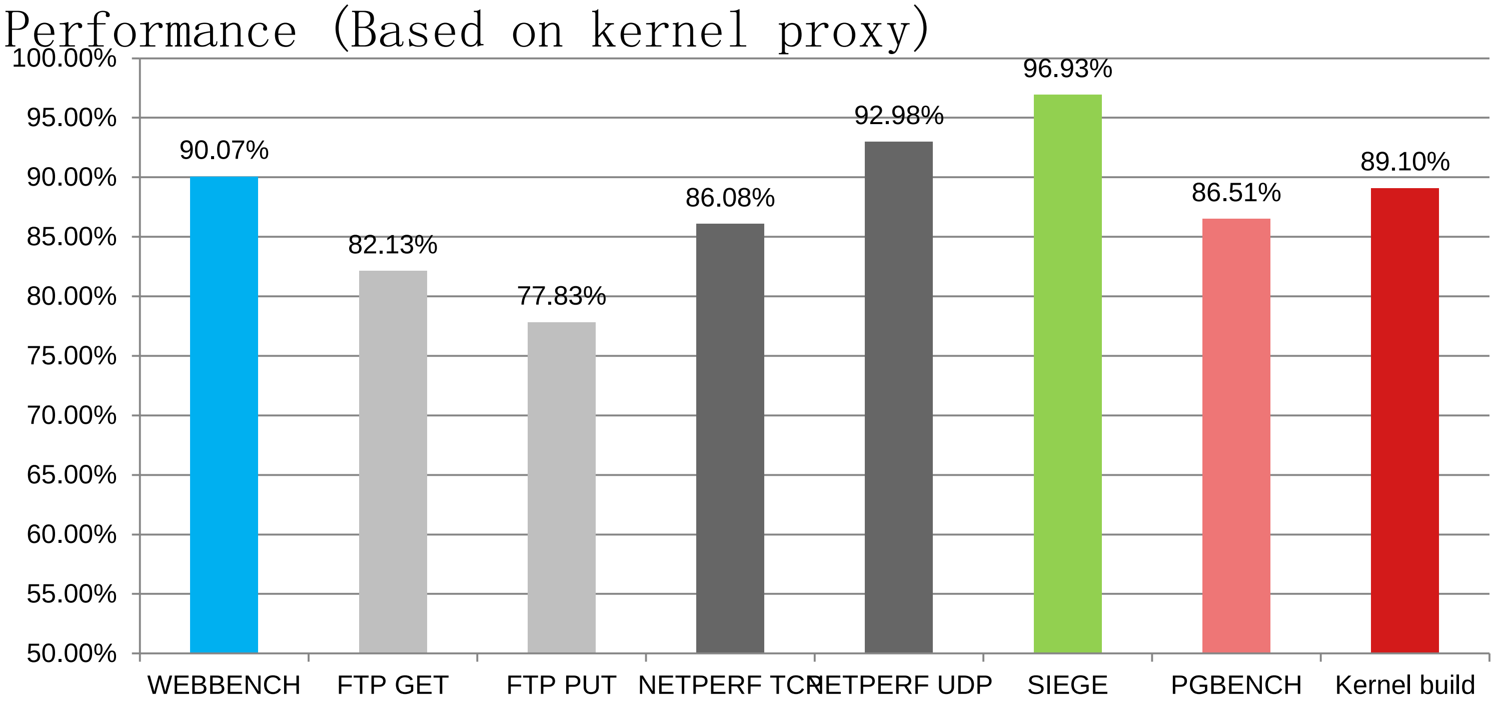
- ~~Kernel scheme~~:
  - Based on kernel TCP/IP stack and netfilter component
  - Can support vhost-net, virtio, e1000, rtl8139, etc
  - Better performance but less flexible (Need modify netfilter/iptables and kernel)
- Userspace scheme:
  - Totally realized in QEMU
  - Based on QEMU’s netfilter components and SLIRP component
  - Not support vhost-net, but e1000, rtl8139
  - More flexible

Proxy Design (Userspace scheme)



- **Filter mirror:** copy and forward client's packets to SVM
- **Filter redirector:** redirect net packets
- **COLO compare:** compare PVM's and SVM's net packets;
- **Filter rewriter:** adjust tcp packets' ack and tcp packets' seq

COLO Performance In KVM



The experimental data is normalized to the native system

Agenda

- Introduce COarse-grain LOck-stepping
- COLO Design and Technology Details
- Current Status Of COLO In KVM
- Further Work About COLO

Status of COLO In KVM

COLO Framework:

- Include VM state checkpoint process, failover process
- Patch set v18 had been post

COLO block replication:

- Only including non-shared storage replication scheme
- Already been merged to branch https://github.com/stefanha/qemu/commits/block-next

COLO proxy:

- netfilter base/buffer/mirror plugins have been merged
- Userspace packets compare is WIP and v11 version has been posted

Agenda

- Introduce COarse-grain LOck-stepping
- COLO Design and Technology Details
- Current Status Of COLO In KVM
- Further Work About COLO

TODO

- Continuous VM replication development
- Support shared storage
- Optimize performance
- Reduce VM’s downtime while do checkpoint
- Improve storage and network performance
- Implement the heartbeat component
- Support COLO in libvirt

Thank you

www.huawei.com