Live Migration with SR-IOV Pass-through

Weidong Han <hanweidong@huawei.com>

August, 2015

www.huawei.com

HUAWEI TECHNOLOGIES CO., LTD.

Agenda

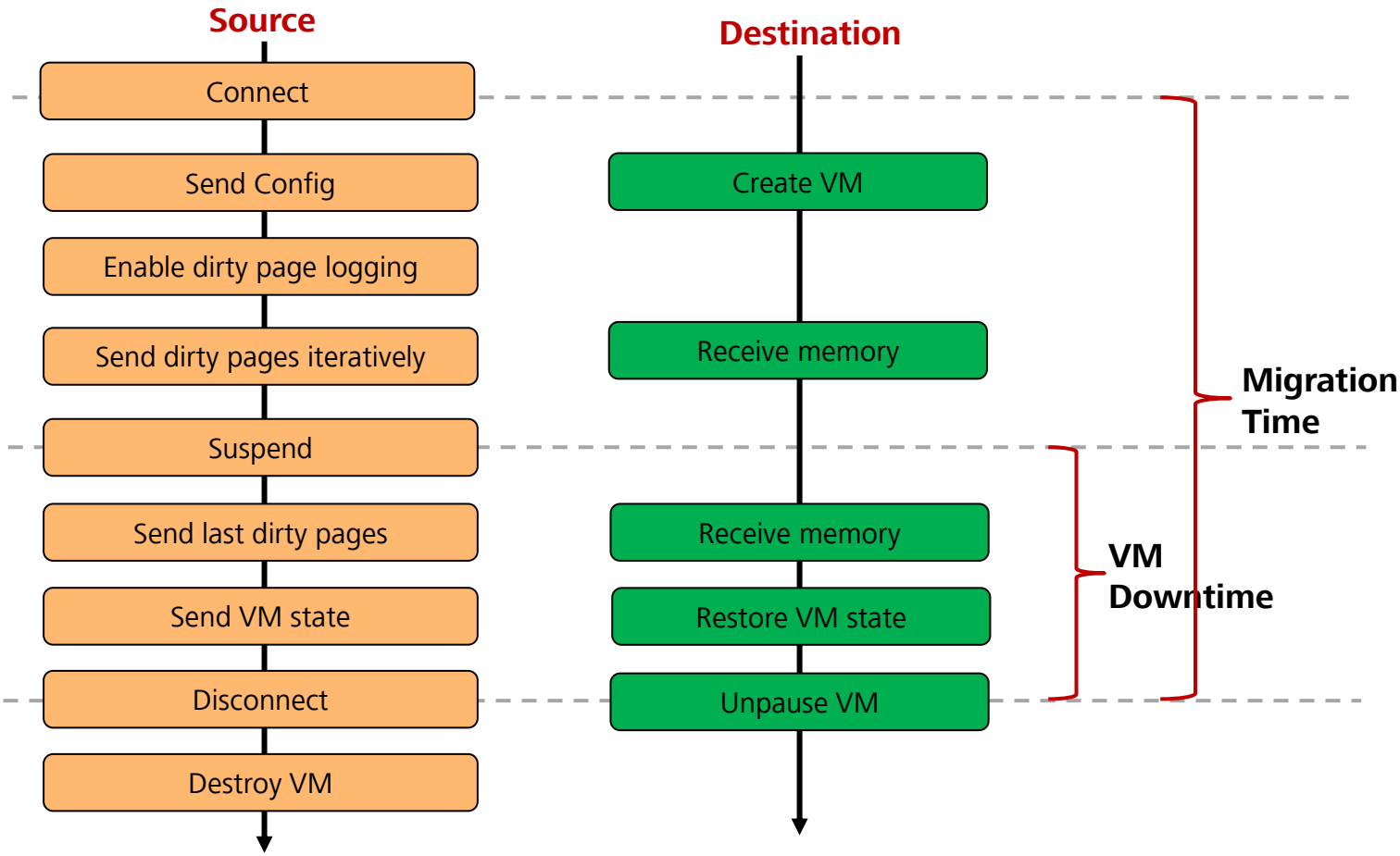
- Background
- Prototype
- Evaluation
- Summary

Background

- **VM live migration is one of the most important feature of virtualization**

- **SR-IOV migration is required**
  - NIC becomes more powerful: 10Gbit - > 40Gbit -> 100Gbit

Live Migration Algorithm



Challenges

- **How to migrate hardware state of the assigned device?**
  - Some registers of existing NICs are not writable

- **Bonding driver (VF and virtio-net) in VM**
  - Performance is not consistent
  - CPU consumption is not consistent
  - Hot plugging device increases downtime

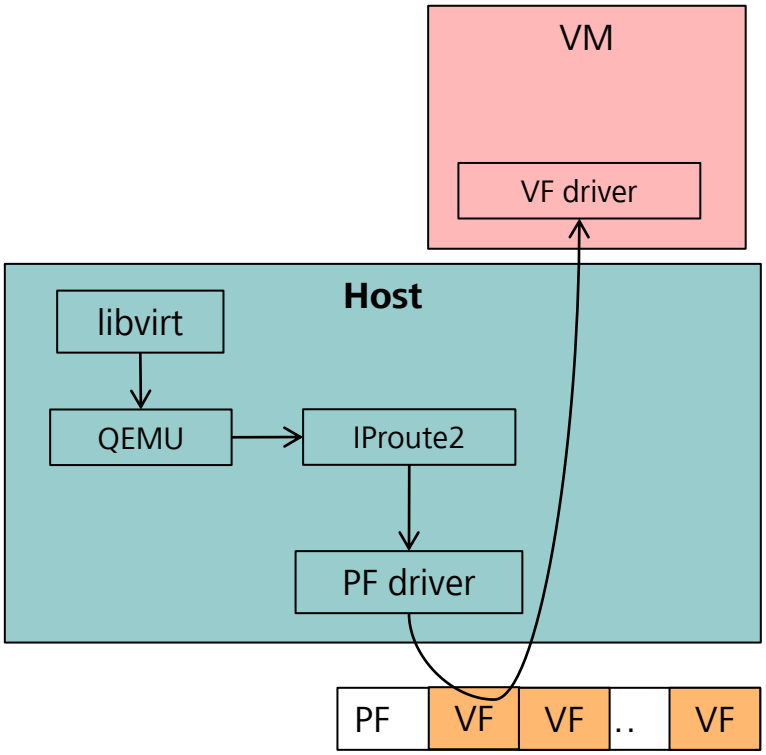
Agenda

- Background
- Prototype
- Evaluation
- Summary

Ideally, Hardware can help

- **I/O registers are readable and writable**
- **NIC Driver provides suspend and resume functions**
  - Suspend: save hardware state
  - Resume: restore hardware state

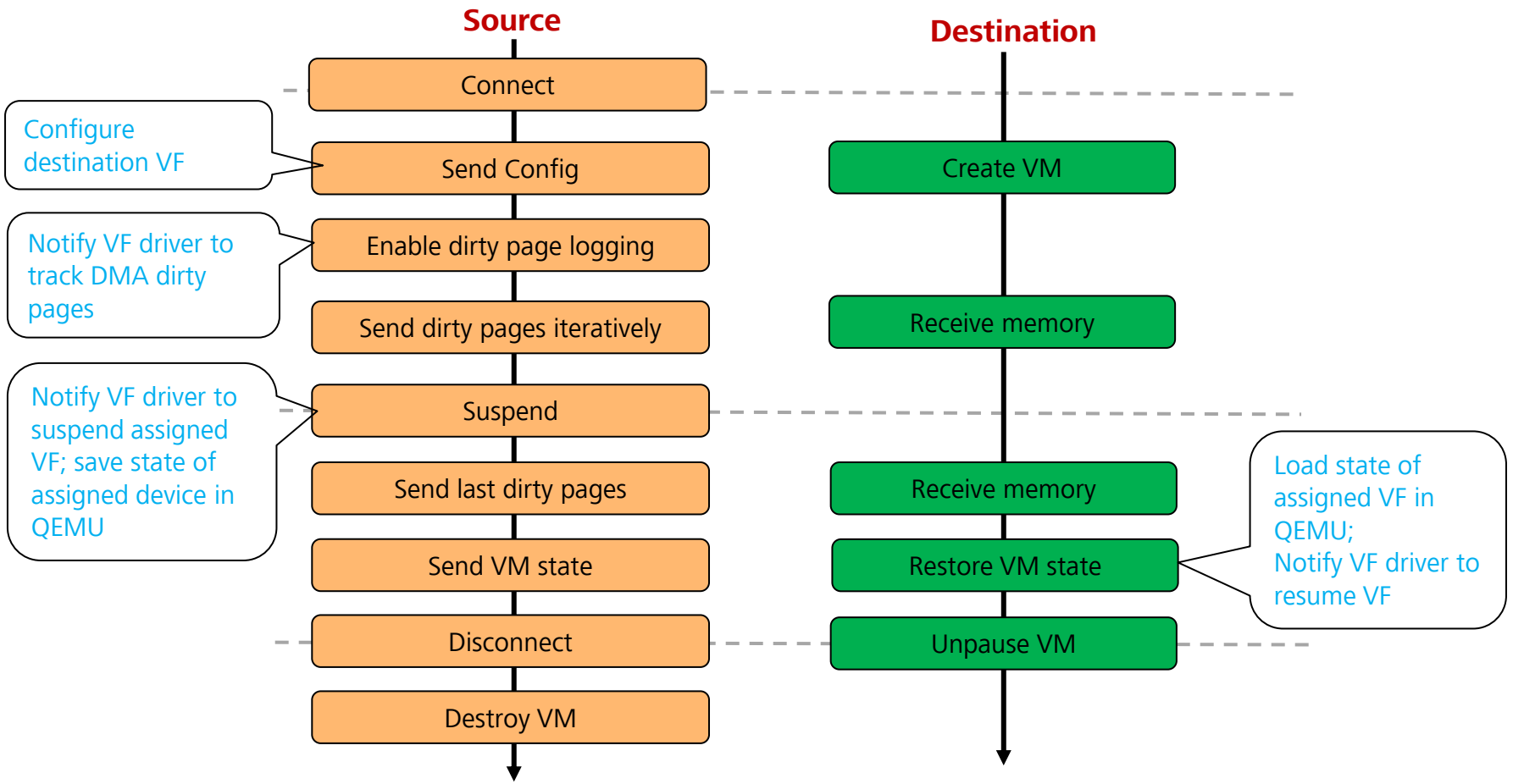
Prototype Overview



- **Libvirt**
  - Migration check, prepare VM config
- **QEMU**
  - Implement savevm handlers (save and load) for assigned device
  - Use IProute2 command to notify SR-IOV driver for migration
- **Iproute2**
  - Add commands: migrate, cancelmigration, suspend, resume.
- **PF driver**
  - Notify VF driver for migration operations
- **VF driver**
  - DMA dirty page logging
  - Suspend and resume VF state

Note: based on a Huawei NIC prototype

Live Migration Algorithm with SR-IOV Pass-through



Iproute2 Migration Commands

- **Iproute2 can set VF state from kernel 3.12**
  - #ip link set <pf> vf <vf_index> state auto|enable|disable
- **Extend iproute2 VF state set commands**
  - #ip link set <pf> vf <vf_index> state auto|enable|disable| migrate|cancelmigration|suspend|resume
- **PF driver receives migration commands from iproute2, and passes them to VF driver via mailbox**

DMA Dirty Pages Logging

- Memory access by DMA can not be tracked by page table (e.g EPT)
- VF driver uses dummy writes (read and write a byte at the same address) to make it dirty, then the memory can be tracked
- It almost doesn’t impact the performance

VF State Migration

- **VF suspend**
  - VF driver saves internal hardware states, and down interface
  - QEMU saves states of assigned VF via registered savevm handlers

- **VF resume**
  - QEMU restores states of assigned VF via registered savevm handlers
  - VF driver restores internal hardware states, up interface, and sends ARP.

Agenda

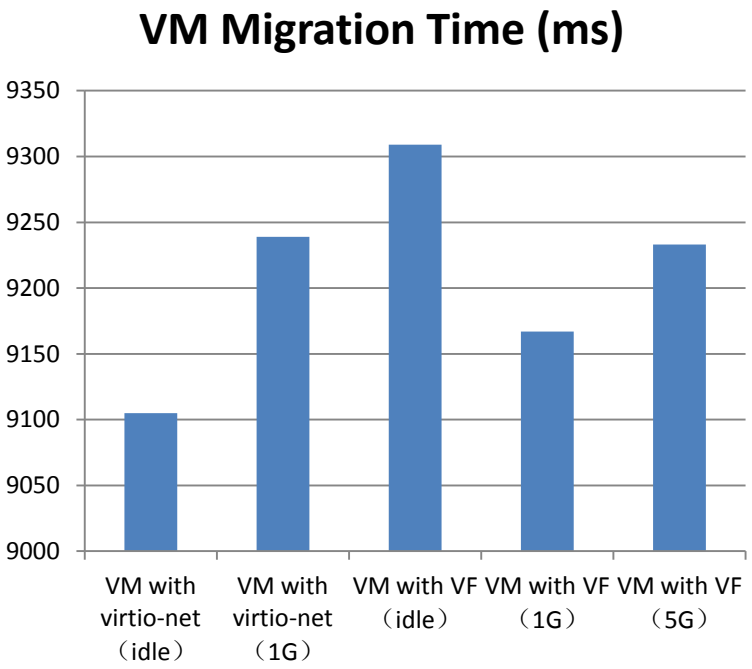
- Background
- Prototype
- Evaluation
- Summary

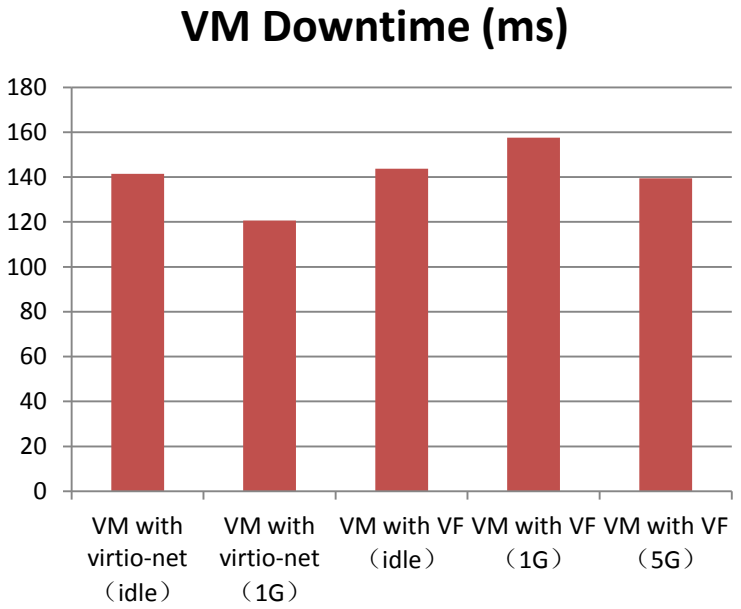
Test Environment

- **Host**
  - CPU: Huawei RH2288v2 (Xeon CPU E5-2620 v2@2.1Ghz)
  - NIC:
    - Huawei smart NIC prototype (for pass-through)
    - Broadcom Corporation NetXtreme BCM5719 Gigabit (VM data transfer for migration)
  - Storage: Huawei OceanSpace S5500T, through IPSAN

- **VM**
  - SLES11 SP3 64bit, 4 CPU, 4GB Memory

Results

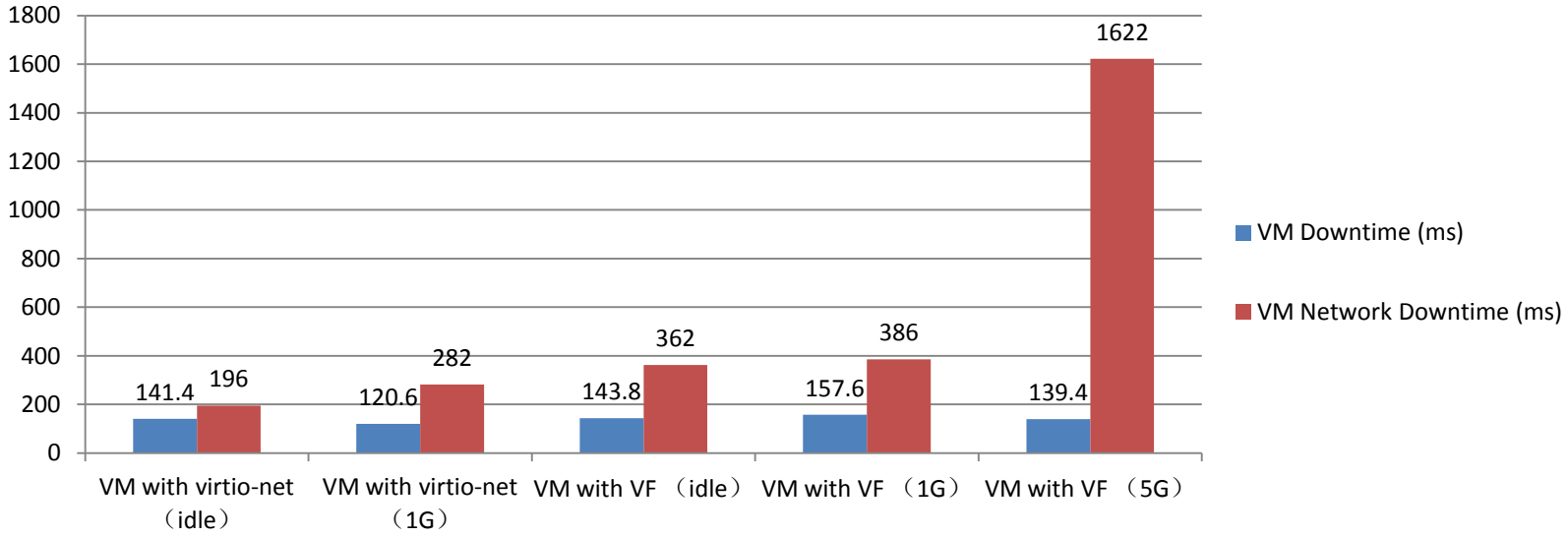




Note: tested with default qemu max_downtime set, here is not the minimal downtime

- **VM migration time and downtime impact of our prototype is little.**

Results (cont.)



- **Normally the network downtime of VM with VF is a bit of larger than VM with virtio-net**
  - Additional time of VF suspend and resume via VF driver: suspend time is about 5ms, resume time is about 20ms (need optimization)
  - The network downtime with 5G workload case is big (need fixing)

Agenda

- Background
- Prototype
- Evaluation
- Summary

Summary

- Demonstrate a prototype of SR-IOV migration with hardware and driver help
- The evaluation results show it basically performs well
- Need improvements
- Hope more future NICs will be friendly to live migration!

Thank you

www.huawei.com