RDMA In Virtual Environments (v1.0)

Aaron Blasius, ESXi Product Manager
Bhavesh Davda, Office of CTO
VMware
RDMA for hypervisor services
vMotion/RDMA Performance

Total vMotion Time (seconds)

Precopy bandwidth (Pages/sec)

% Core Utilization used by vMotion

Time (s)

Destination CPU Utilization

Source CPU Utilization

0:00 0:05 0:11 0:16 0:21 0:26 0:31 0:36 0:41 0:46 0:51 0:56 1:01 1:06 1:11 1:16

0 10 20 30 40 50

% Core Utilization used by vMotion

Time (s)

0 5 10 15 20 25 30 35 40 45 50

TCP/IP

RDMA

14.18 Gbps

10.84 Gbps

36 % faster

30% Higher

432,757.73

330,813.66

70.63

45.31

36 %

30%

Higher

92% Lower

84% Lower

92% Lower

84% Lower
Hypervisor level RDMA Requirements

- Integrate OFED Kernel Space Mid-Layer and Provider components into ESXi hypervisor
- Add RDMA Verbs support for hypervisor services: vMotion, FT, vSAN, vRDMA, iSCSI
- Create RDMA Device layer for hardware drivers to plug into
- Work in progress
Options to offer RDMA to vSphere Virtual Machines

- Full-function VM DirectPath (passthrough) \( \geq \) ESXi 4.0
- SR-IOV VF VM DirectPath (passthrough) \( \geq \) ESXi 5.1
- SoftRoCE over 10GbE in VM DirectPath mode
- SoftRoCE over paravirtual Ethernet vNIC over 10GbE uplink
- SoftRoCE over paravirtual Ethernet vNIC between VMs
- Paravirtual RDMA HCA (vRDMA) offered to VM

Prototyping/Future
SR-IOV VF VM DirectPath

- Single-Root IO Virtualization (SR-IOV): PCI-SIG standard
- Physical (IB/RoCE/iWARP) HCA can be shared between VMs or by the ESXi hypervisor
  - Virtual Functions direct assigned to VMs
  - Physical Function controlled by hypervisor
- Still VM DirectPath, which is incompatible with many important vSphere features
Paravirtual RDMA HCA (vRDMA) offered to VM

- New paravirtualized device exposed to Virtual Machine
  - Implements “Verbs” interface
- Device emulated in ESXi hypervisor
  - Translates Verbs from Guest to Verbs to ESXi “OFED Stack”
  - Guest physical memory regions mapped to ESXi and passed down to physical RDMA HCA
  - Zero-copy DMA directly from/to guest physical memory
  - Completions/interrupts “proxied” by emulation
- “Holy Grail” of RDMA options for vSphere VMs
Guest Kernel vRDMA Driver

- Registers kernel verbs functionality with `ib_core` framework
- Re-uses response structures from `ib_user_verbs.h`
ESXi VMkernel vRDMA Backend

- vRDMA can return cached/dummy Device_Attri
- No need to send Query_Device_MAD everytime
vRDMA: VMs On Different Hosts
vRDMA: VMs On The Same Host

- Invoked for control path operations
- When dest VM is not on the same host

- Fast VM-VM (same host) communication
- Look at dest VM Vaddr
- Memcpy between VM buffers
vRDMA Prototype Status

- Prototyped by intern in CTO office in summer ‘12
- MAD Verbs functional
- Data path Verbs work-in-progress
- Estimated performance
  - 5 µs HRT for RDMA Write with polling completion
  - Back-of-envelope estimate based on VM-exit overheads and VMCI performance benchmarks
RDMA Use Cases

• Traditional Enterprise
• HPC
• Big Data
• Storage
• Messaging
• Other Scale Out Applications
• What’s your preferred method for obtaining device drivers?
  – Linux distribution OFED
  – Hardware vendor OFED
RDMA Adoption Trend

• What % of your organization’s applications leverage RDMA today?
• What % by 2016?
• Is RDMA driving your hardware purchasing decisions?
• Are any of your RDMA-based applications stateful?
Moving the enterprise to virtual RDMA

- Do you want to run RDMA-based applications in a VM?
- Would you use a virtual RDMA device in a VM?
  - vMotion, HA, Snapshots, DRS
- What is your tolerance for latency overhead?

Aaron Blasius
ablasius@vmware.com
Thank You