

# **Efficient sharing of physical** devices between KVM guests and host

29-Oct-2012

**Bharat Bhushan** Vakul Garg Ruchika Gupta



he Energy Efficient Solutions logo, mobileGT, PowerQUICC, QorlQ, StarCore and Symphony



are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. BeeKit, BeeStack, ColdFire+, CoreNet, Flexis, Kinetis, MXC, Platform in a Package, Processor Expert, QorlQ Qonverge, Qorivva, QUICC Engine, SMARTMOS, TurboLink, VortiQa and Xtrinsic are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property

#### Agenda

- Introduction
- Hardware Queuing Mechanism
- Sharing Network interface
- Sharing Hardware Accelerator
- Performance Data
- Under Investigation





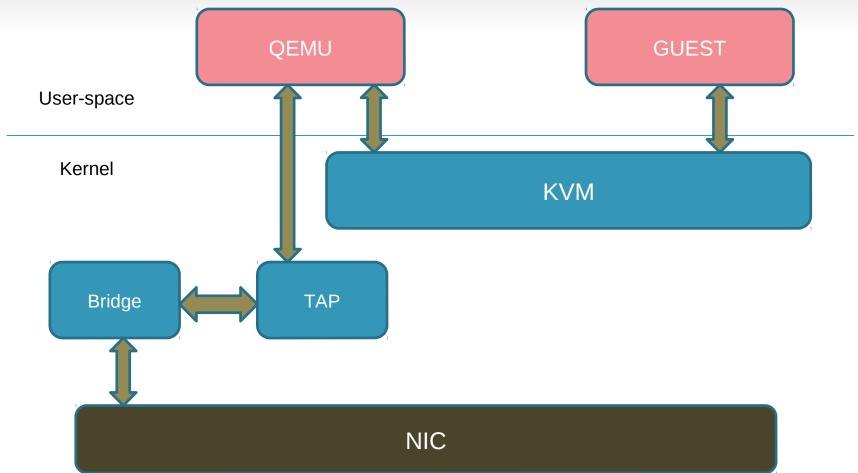


# Existing Para-Virtualized Mechanisms for Sharing Network Interface Card





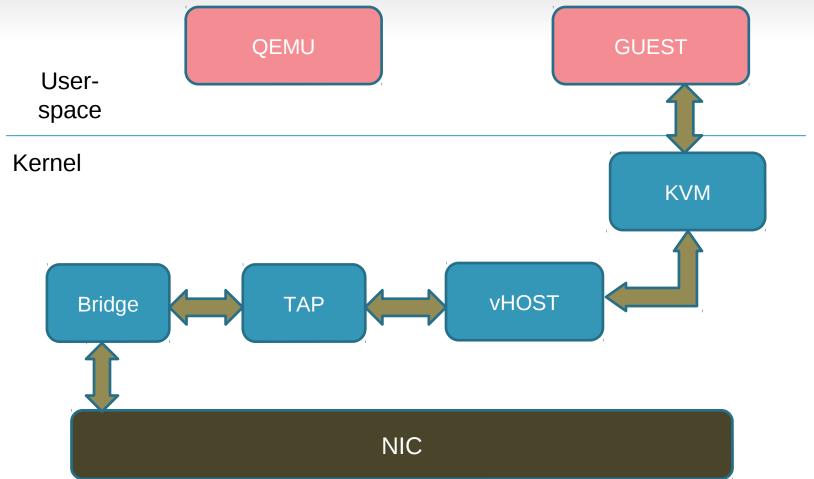
# **Sharing NIC Using Bridge (virtio)**







# **Sharing NIC Using Bridge (vHOST)**

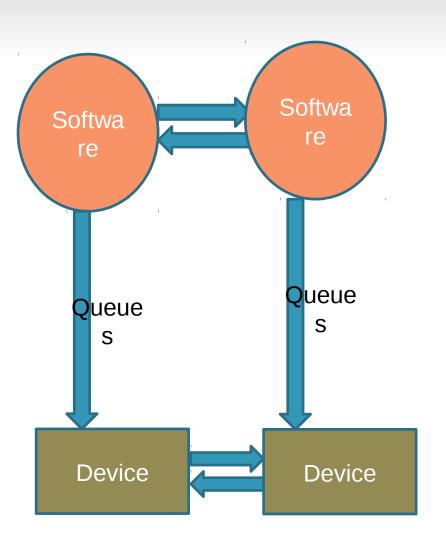






## **Partitionable Hardware Queue Architecture**

- Hardware queues are data carriers
- Can be used for communication between
  - software entities
  - software and hardware device
  - two devices
- There are millions of queues in system
- Queues can be partitioned.



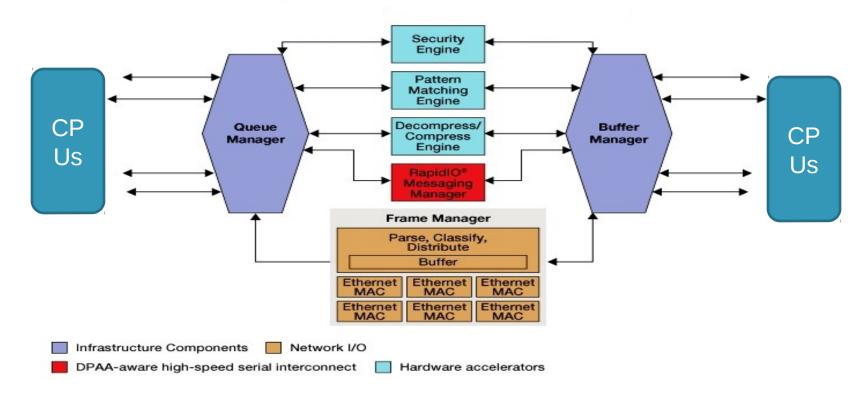




#### **DPAA Implements Hardware Queue**

QorlQ Device (P4080, P5020, P2040, T4240 etc) Implements DPAA architecture.

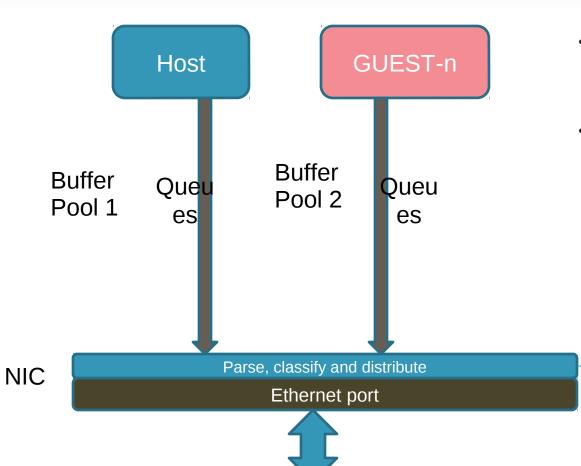
Data Path Acceleration Architecture (DPAA)







### **Sharing NIC Using Queues**



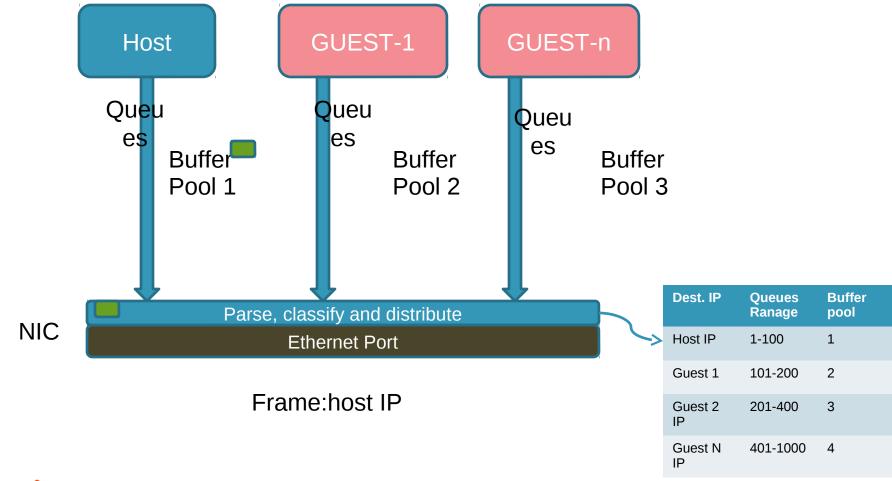
- Hardware managed buffers
- NIC:
  - Ethernet port interface
  - Parse and classify packet
- Classification Rules

	Dest. IP	Queues Ranage	Buffer pool
>	Host IP	1-100	1
	Guest-1	101-200	2
	Guest -2 IP	201-400	3
	Guest -n IP	401-1000	4





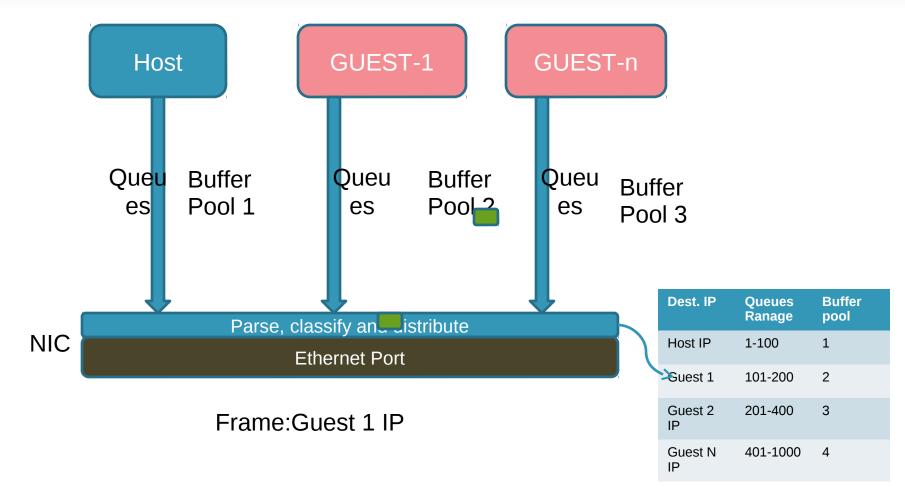
#### **Packet Flow: Receive for Host**







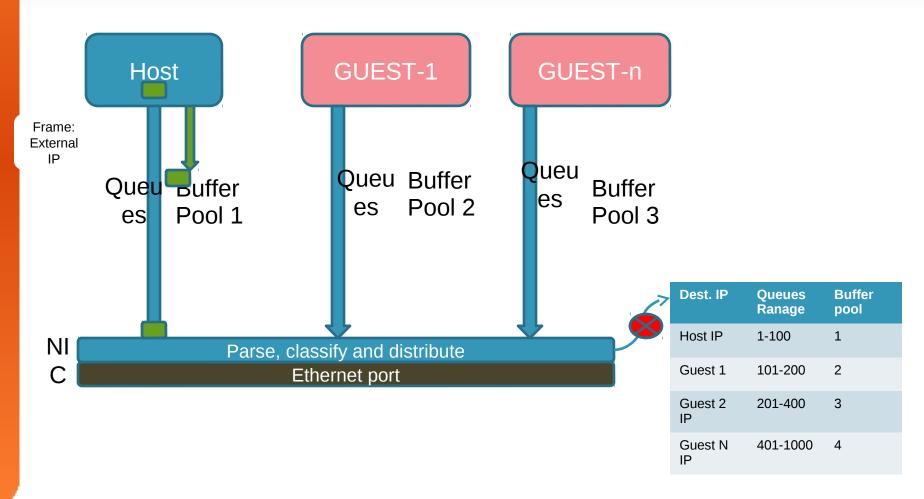
#### **Packet Flow: Receive for Guest**







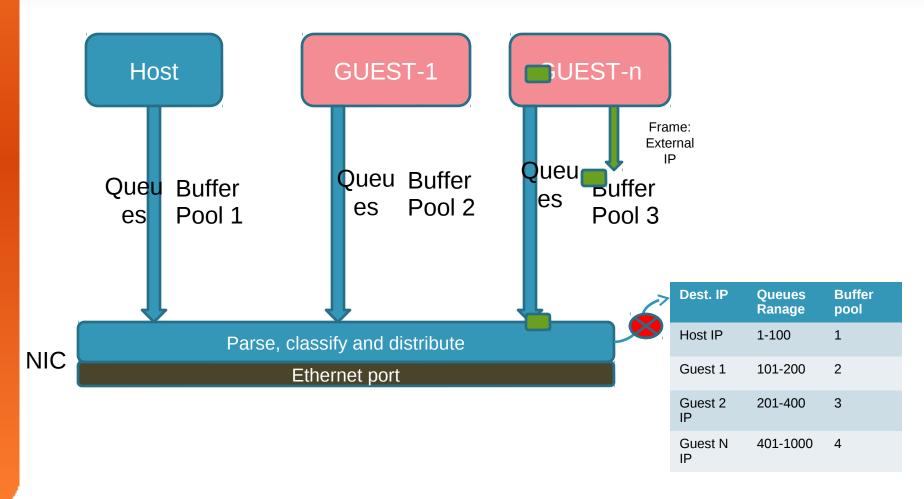
## **Packet Flow: Transmit from Host**







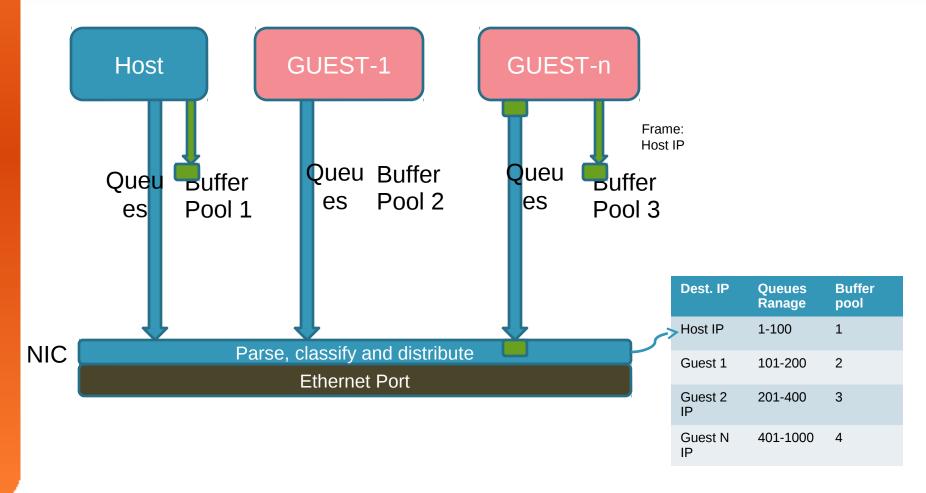
## **Packet Flow: Transmit from Guest**







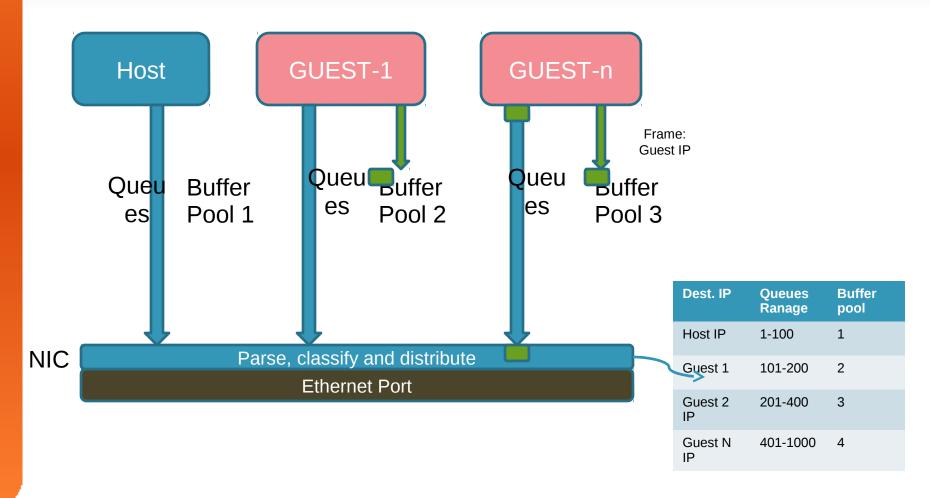
#### **Packet Flow: Guest and Host**







#### **Packet Flow: Between Guest**

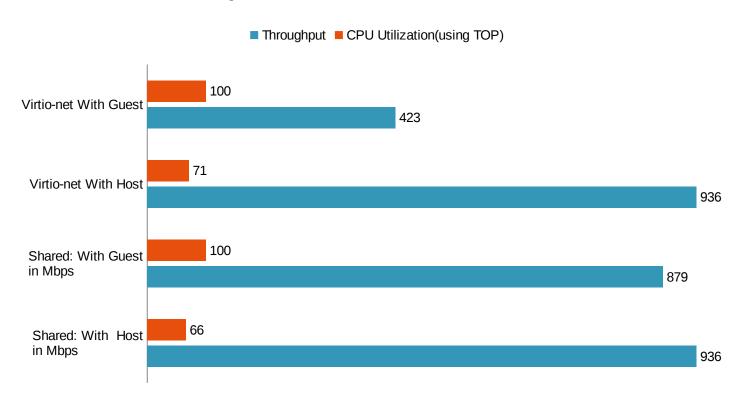






#### **Performance Data**

#### Physical CPUs: 1Guest CPUs: 1



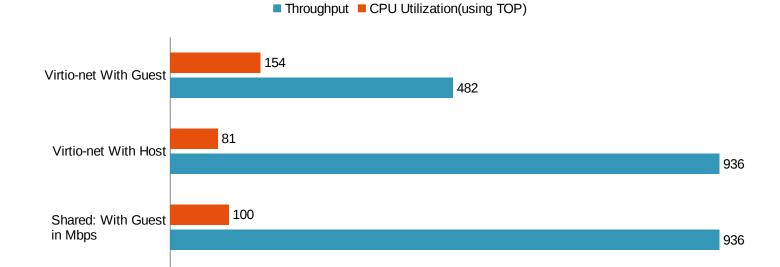




#### **Performance Data**

76

#### Physical CPUs: 2Guest CPUs: 1





Shared: With Host

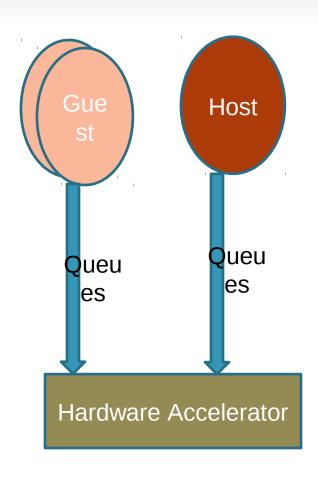
in Mbps

936



#### **Sharing the Hardware Accelerators**

- Guest and host will initialize there respective transmit and receive queues
- Guest/Host will transmit data on its TX queues
- After processing the hardware accelerator will place the data on Guest/Host RX queues
- Guest/Host will dequeue the data from there respective RX queue
- Host software help not required in poll mode
- In interrupt mode the interrupt goes via host, but data flow does not require host intervention







#### **Under Investigation**

- PoC Demonstrating Direct Assignment and sharing
- DPAA integration with VFIO
- Error handling
- Upstream
- Performance



#### References

- KVM
  - <a href="http://www.linux-kvm.org">http://www.linux-kvm.org</a>
- QEMU
  - http://www.qemu.org
- QorlQ Data Path Acceleration Architecture
  - http://www.freescale.com/webapp/sps/site/overview.jsp?code=QORIQ\_DPAA&fsrch=1&sr=1
  - P4080 Product Brief:

http://www.freescale.com/files/32bit/doc/prod brief/P4080PB.pdf







