vms Software

LAN Performance Improvements

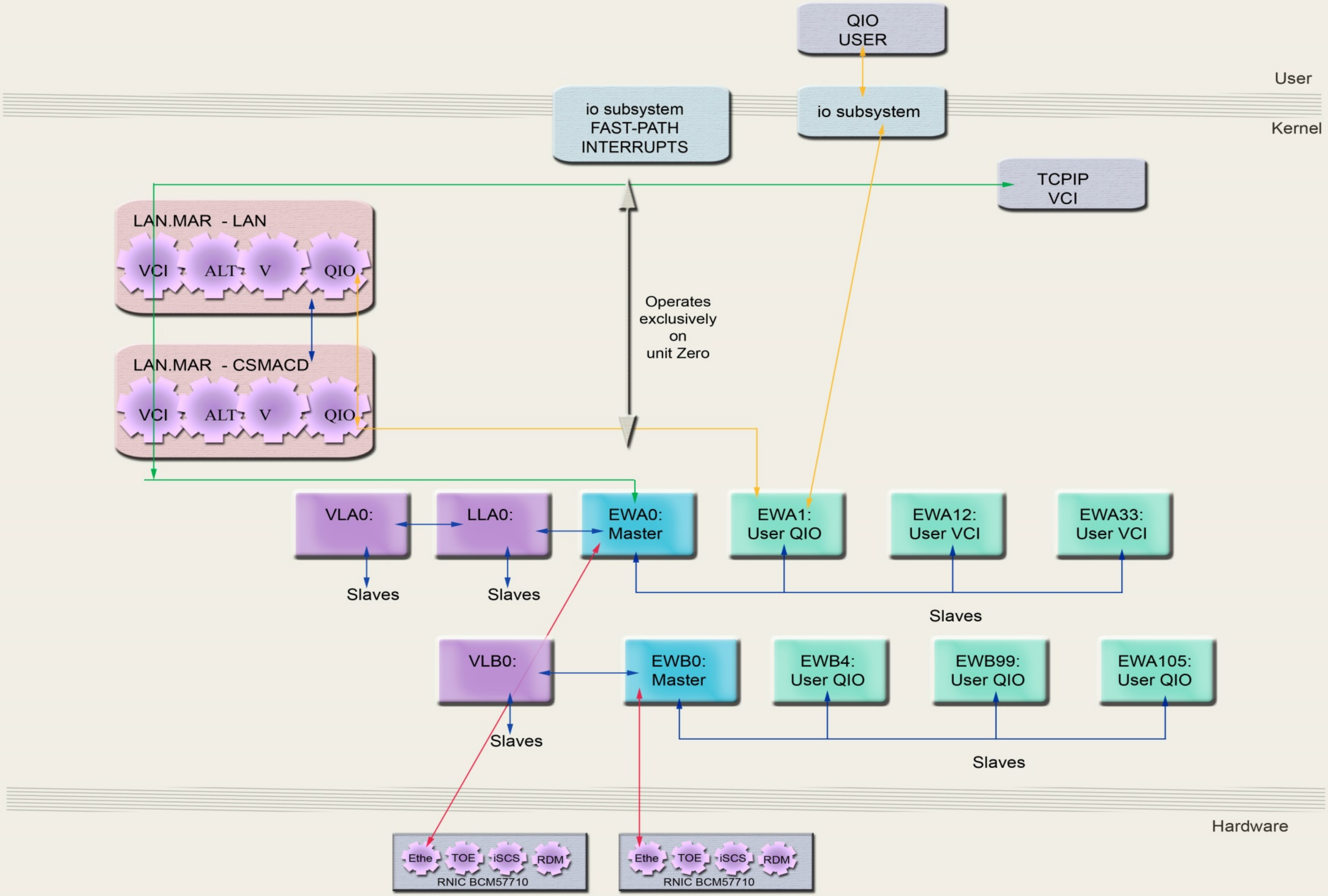
October 20, 2015

LAN Performance Improvements

This information contains forward looking statements and is provided solely for your convenience. While the information herein is based on our current best estimates, such information is subject to change without notice.

Basic functionality *is not what it used to be*

- DECnet and SCS Cluster over LAN. The center of the known Universe.
- Performance was OK with SCS. Single channel is at line rate.
- Did not perform well with standard size frames.
- The VMS Z device naming! 26 devices are just fine!
- Around 1Gig memory (per system) for all NICs.
- Not a 100% full duplex stack.
- Single RX ring + single TX Ring.
- TCP/IP, what’s that? Never implemented:
  - Checksum Offload
  - LSO TSO RSS
  - RDMA, iSCSI &TOE



QIO USER

io subsystem

io subsystem FAST-PATH INTERRUPTS

User

Kernel

TCPIP VCI

LAN.MAR - LAN

VCI ALT V QIO

LAN.MAR - CSMACD

VCI ALT V QIO

Operates exclusively on unit Zero

VLA0:

LLA0:

EWA0: Master

EWA1: User QIO

EWA12: User VCI

EWA33: User VCI

Slaves

Slaves

Slaves

VLB0:

EWB0: Master

EWB4: User QIO

EWB99: User QIO

EWA105: User QIO

Slaves

Slaves

Hardware

Ethe TOE iSCS RDM

RNIC BCM57710

Ethe TOE iSCS RDM

RNIC BCM57710

Current Situation

- LAN.MAR – LAN Common
  - Macro32 code. Designed centered in QIO model.
  - Supports legacy software & HW features.
  - IO path supports multiple software layers; QIO, VCI, DECnet V, AltStart, Network Manager, VCI management.
  - Amazingly sturdy despise its complexity.

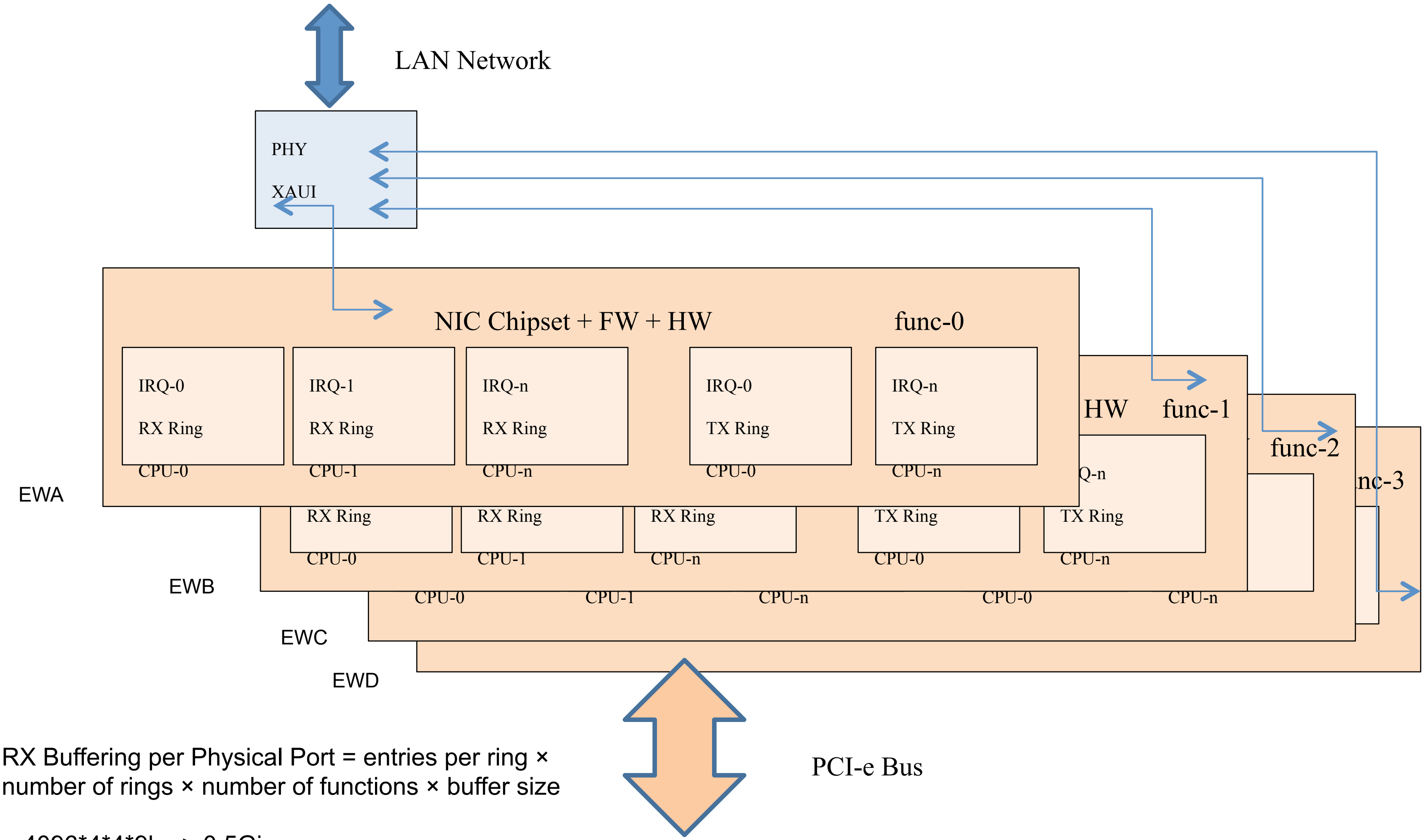
- VCI
  - Very well performed, can achieve line rate with single queue and jumbo frames.
  - Hard to implement. Applications needs to be very well debugged before deployment.
  - Used by TCP/IP, Clusters, LAST, DECnet V, LAT, AM

Current Situation

- IO-Subsystem dependencies
  - Completely centered in Unit Zero.
  - Unit Zero is used for interrupt delivery, FAST PATH.
  - Multi-queue difficult to implement
  - QIO users are strongly tied to slave units
  - VCI users bypass most of QIO interface for data transfer purpose. They are weakly tied to slave units.

- QIO Interface
  - Painfully slow but very secure in terms of protecting the integrity of the system.

“Modern” NICs



LAN Network

PHY

XAUI

NIC Chipset + FW + HW

func-0

IRQ-0 RX Ring CPU-0

IRQ-1 RX Ring CPU-1

IRQ-n RX Ring CPU-n

IRQ-0 TX Ring CPU-0

IRQ-n TX Ring CPU-n

HW func-1

func-2

nc-3

EWA

EWB

EWC

EWD

PCI-e Bus

RX Buffering per Physical Port = entries per ring × number of rings × number of functions × buffer size

= 4096*4*4*9k ~> 0.5Gig

“Modern” NICs

- Multiple queues/rings
  - With MSI-X and system FW/HW support you can affinitize a queue to a specific CPU.
  - Vendor specific: In most cases you can configure 8 TX queues and 8 RX queues.
  - Ring/queue size in most cases is not fixed. It can be very large. It can be sized at compilation and/or startup time.

- Multiple PCIe functions – HP Flex10
  - NIC FW/HW is capable of incorporating more than one NIC for a single port.
  - Each function has is own and separate PCI configuration space and bandwidth assignment.
  - All functions share the same PHY.
  - In 10G devices there is support for 4 functions per port

Why Change
What's to gain

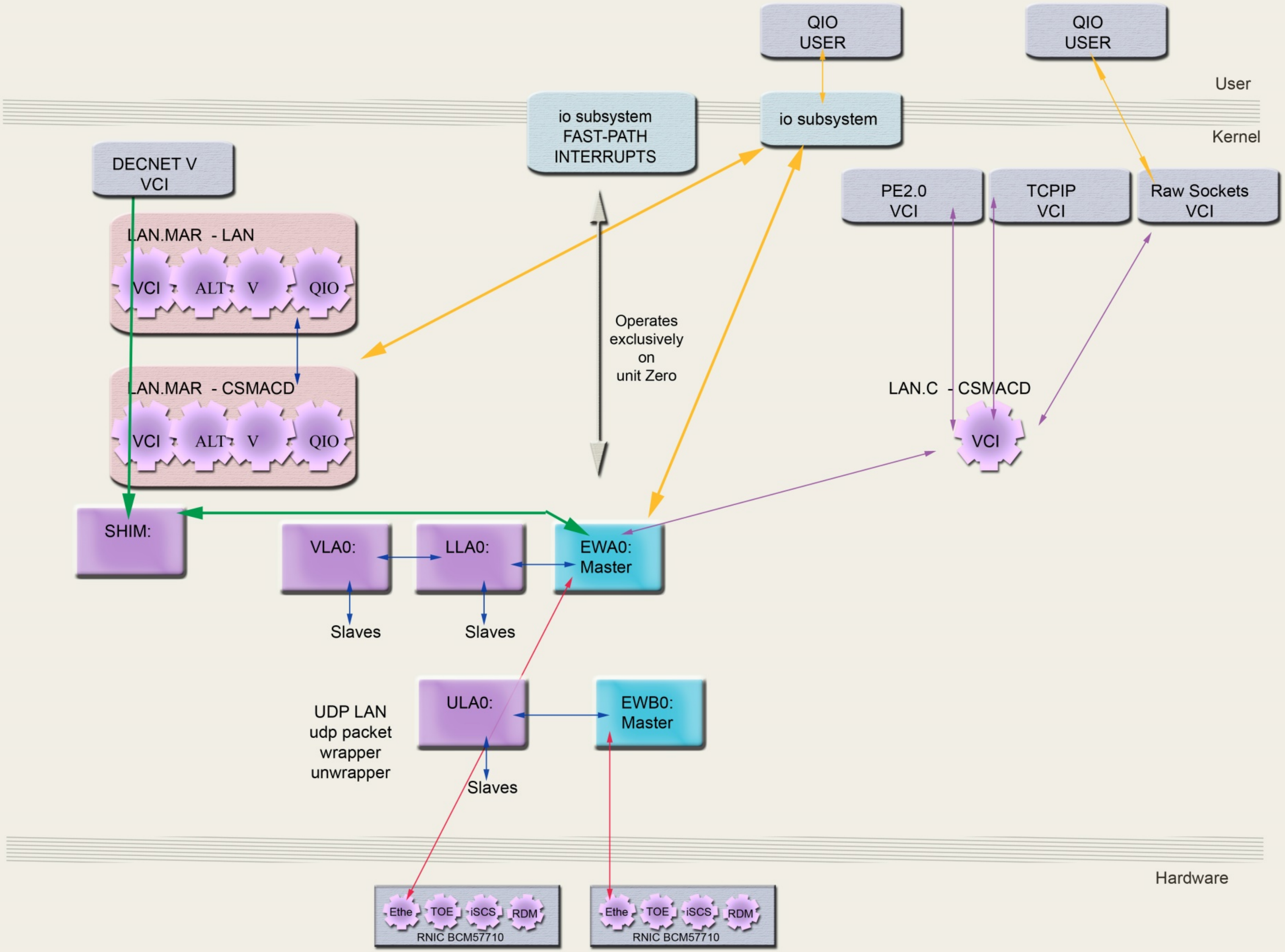
- We simply cannot afford to avoid implementing what today is considered basic functionality.
- We need to give TCP/IP the importance it has gained throughout the industry.
- Performance.
  - LAN/DRIVER: We are fine with jumbos ~ 99%. We are at 85% line rate with standard frames. Latency (Unknown)
  - TCP/IP: 12.5% of line rate!
  - SCS: 90+% with single channel. Poor when using multiple channels or two way streams.
- Flexibility, Maintainability, Extensibility, Performance

Basic functionality
What we will be

- Design cannot be centered on a specific application. Except for accommodating TCPIP HW features.
- Drivers perform at line rate with standard frames.
- 100% Full Duplex – Spinlock re-work
- RSS/Multiple queue/ CPU MQ affinity/MSI-X Interrupts.
- The VMS Z naming! 26 no more! How about 17576!
- Expanded Memory usage per function / per NIC
  - S2 space for just about everything in LAN
  - Zero Non Paged Pool usage is the goal
  - 64bit drivers and LAN subsystem, long pointers!
  - We are pushing for 64b TQEs/Fork and Fork-Wait
- RX Interrupt handling & polling & interrupt mitigation.
- Guest Virtualization support. Emulated/Direct-IO/SRIOV.
- Driver loading parameters per PCI device type.

Basic functionality
What we will be
TCIP

- Must achieve 90%+ of line rate using standard frames in OpenVMS 9.0 VCI 2.0.
- TCP/IP Offloading capabilities on every new driver.
- Hardware LSO & TSO on every capable NIC.
- Software LSO & TSO.



QIO
USER

QIO
USER

User

Kernel

io subsystem
FAST-PATH
INTERRUPTS

io subsystem

DECNET V
VCI

PE2.0
VCI

TCPIP
VCI

Raw Sockets
VCI

LAN.MAR - LAN

VCI ALT V QIO

LAN.MAR - CSMACD

VCI ALT V QIO

Operates
exclusively
on
unit Zero

LAN.C - CSMACD

VCI

SHIM:

VLA0:

LLA0:

EWA0:
Master

Slaves

Slaves

UDP LAN
udp packet
wrapper
unwrapper

ULA0:

EWB0:
Master

Slaves

Hardware

Ethe TOE iSCS RDM
RNIC BCM57710

Ethe TOE iSCS RDM
RNIC BCM57710

New situation

- LAN.MAR – LAN Common
  - Will stay as it is now.
  - Some drivers will not be moved to VCI2.
  - Provide QIO shim to bridge VCI1 to VCI2.

- VCI 2.0
  - Separate module to service VCI2 requests only.
  - Has to provide CPU Affinity at device IPL.
  - New LAN Failover and VLAN drivers
  - UDP wrapper driver to wrap legacy protocols in UDP packets.
  - QOS driver.
  - Delay Box.
  - Initial VCI2 users: Test tool (customer visible), PE and TCP/IP
  - User startup/connection designed closer to socket style rather than QIO.

Basic functionality
The Sky is the limit!

- VCI 2.0 Current Status
  - Basic VCI/DRIVER/User Connection done
  - Basic TX & RX, 8-TX 8-RX queues with MSI-X vector interrupts and S2 buffering using Broadcom 10G NIC. No RX locking. TX lock. S2 memory management lock

- VCI 2.x
  - Add all the VMS virtual devices we need
  - Kernel Raw Sockets/User raw sockets?
  - Finish QIO RX/TX

- VCI 3.0
  - Reloadable drivers
  - User Raw Sockets
  - Investigate at least one of the Big Offloading Items
    - iSCSI or RDMA or TOE
  - Investigate Infiniband Kernel support & RDMA

- VCI 4.0

For more information, please contact us at:

RnD@vmssoftware.com

VMS Software, Inc. • 580 Main Street • Bolton MA 01740 • +1 978 451 0110