LAG AND HARDWARE OFFLOAD TO SUPPORT RDMA AND IO VIRTUALIZED INTERFACES

Vivek Kashyap, Anjali Singhai Jain, Piotr Uminski (Intel)

Linux Plumbers Conference 2019, Lisbon

Legal Disclaimers

Intel technologies features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com].

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps. Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice Revision #20110804

Intel, the Intel logo, Intel. Experience What's Inside, the Intel. Experience What's Inside logo, Intel Xeon Phi, and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States or other countries.

*Other names and brands may be claimed as the property of others.

Agenda

Current Link Aggregation limitations

Link Aggregation for RDMA

Seamless Link Aggregation for Virtual Machines

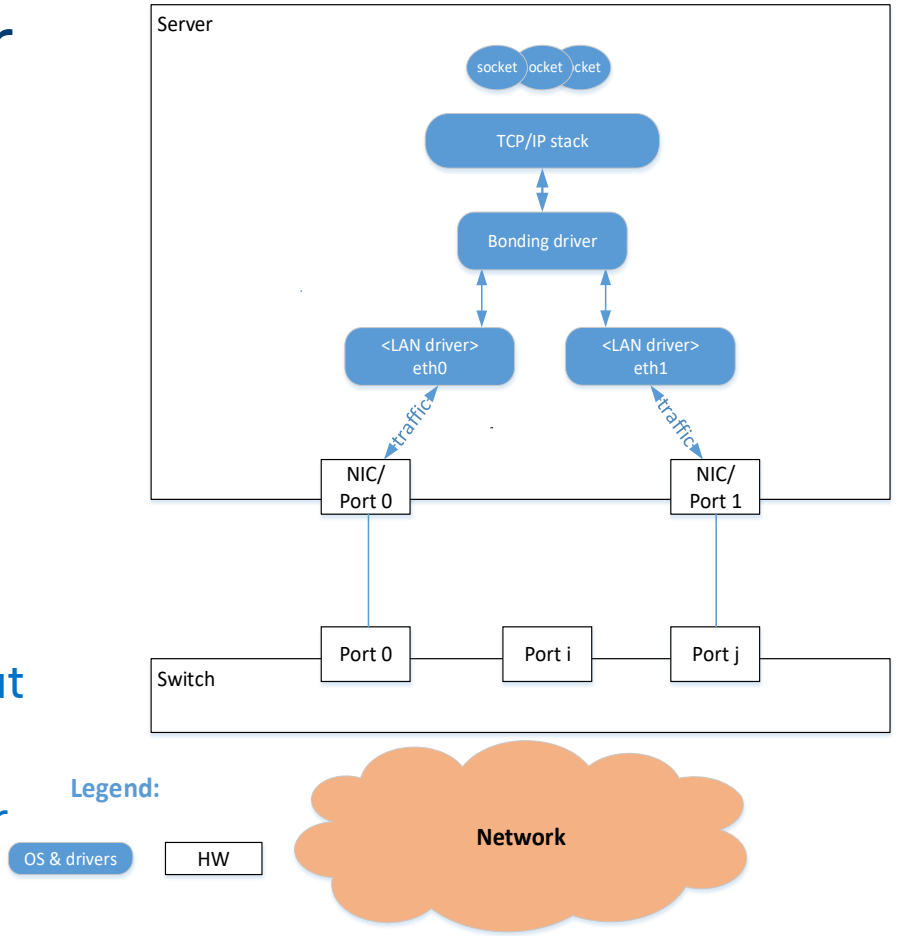
Discussion

LAG using bonding driver

Link Aggregation (LAG) offers link-level redundancy and performance improvements by using multiple links

Implemented by bonding driver

- SW driver between LAN drivers and the rest of network stack
- Can use ports from one or more NICs
- LAN driver does not need to know about bonding
  - Notifications send by bonding driver allows to build LAG-aware drivers



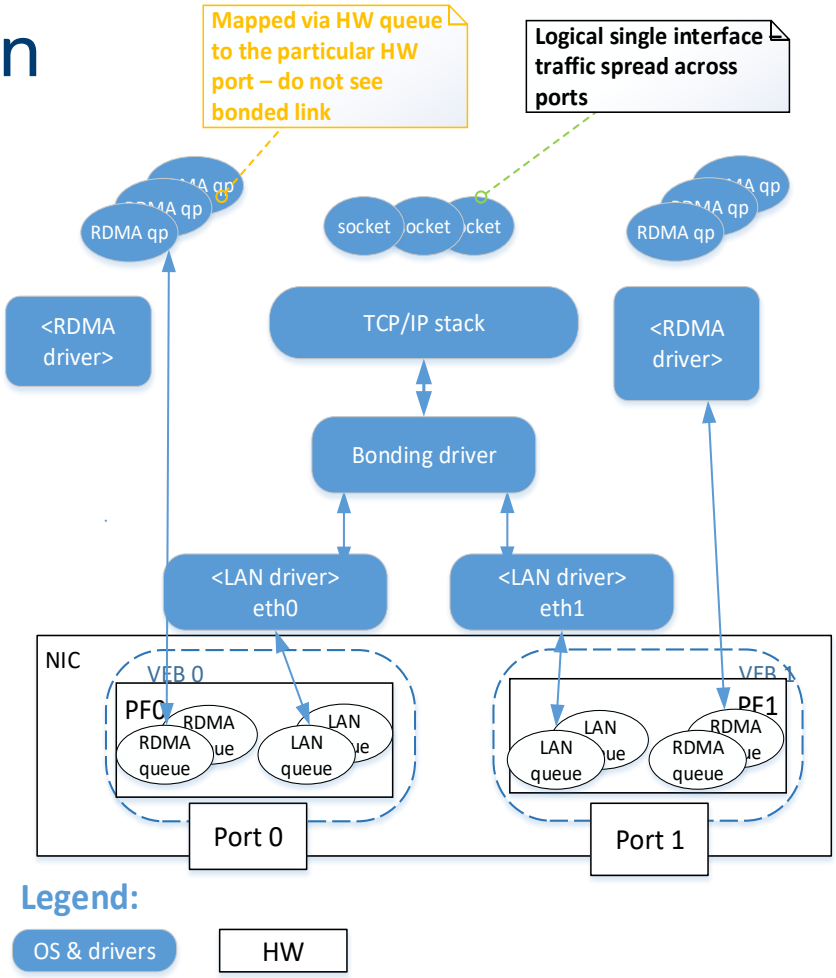
Problems with a legacy solution

HW RDMA does not work with LAG

- RDMA queues are not aware of a bonded link
- Different path for RDMA traffic and regular LAN traffic
- Consequences:
- RDMA-based storage solutions do not tolerate single link errors
- Cannot easily boost RDMA performance by SW-based active-active

SR-IOV LAN virtual functions (VFs) do not work with LAG

- VF maps via Physical Function to the selected port
- Infrastructure detail exposed to the VM
- Consequence: to obtain link redundancy or performance boost, VM must be aware of bonding interface



Proposed solution

Implement active-backup LAG in a NIC driver

- Combined SW/FW solution

Address HW RDMA and VFs

Generic concept but details are NIC-specific

- No changes in generic kernel code
- No changes in NIC hardware
- Small changes in the NIC firmware

RDMA LAG:
Before failover

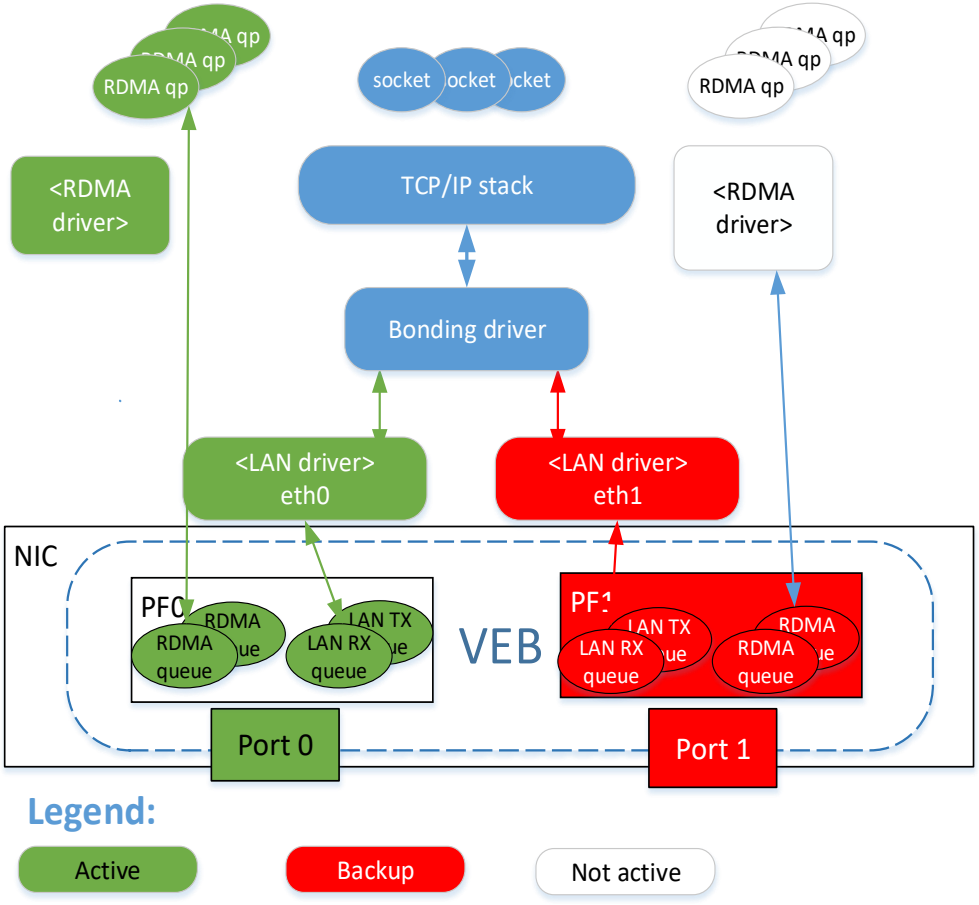
Separate PCIe Physical Functions (PFs) handle separate NIC ports
- LAN PF driver is aware of RDMA driver
- Control queue allocations

RDMA queues allocated from “active” PF
- Application directly uses HW queues
- Backup PF not used to allocate RDMA queues

LAN traffic handled via bonding driver
- LAN PF drivers aware of other bonding members and bonding state from netdev notifications

Single Virtual Ethernet Bridge (VEB) configured on RX
- Detailed rules to direct the traffic

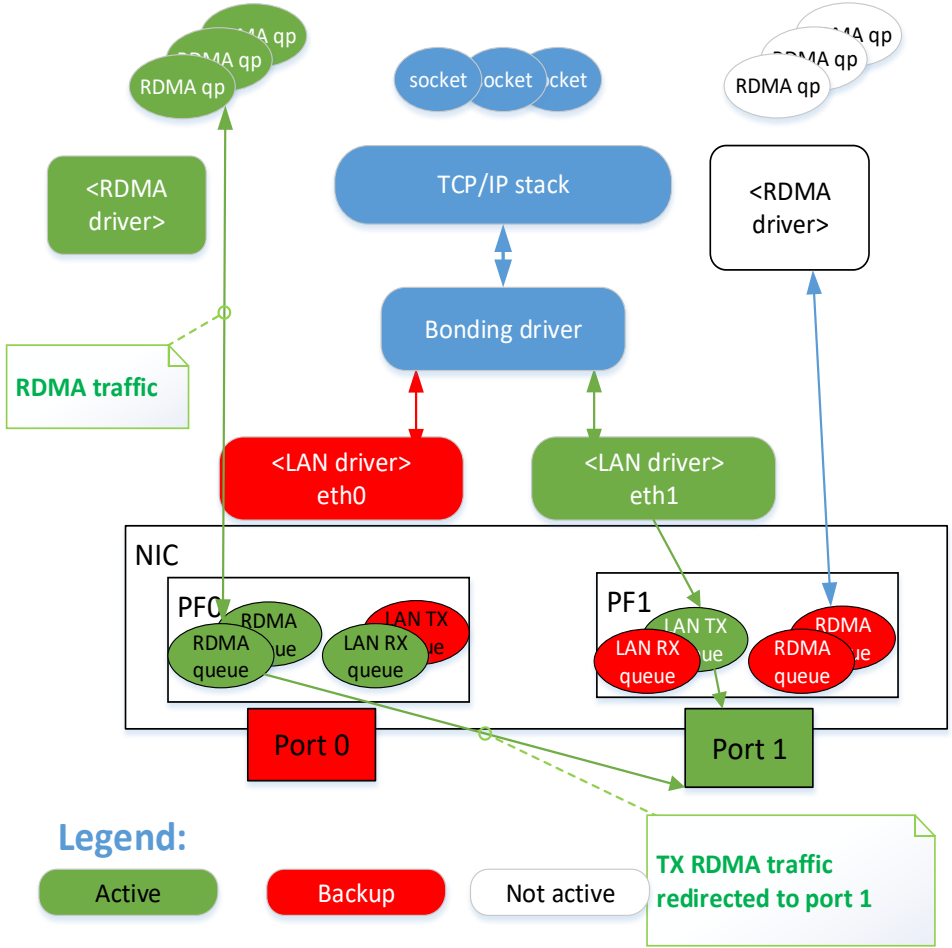
Management & statistics
- Management via NIC drivers
- Statistic read by bonding driver from NIC driver
  - NIC diver expose statistics of right HW queues



RDMA LAG:
TX path after failover

RDMA TX queues are moved to the new active port

- Not visible by the application
  - The application still uses the same queues
- Traffic destructed only for a short time
- Controlled by the LAN driver using existing firmware commands
  - Reprogram TX scheduler to send RDMA traffic over the new port



RDMA LAG:
RX path after failover

Virtual Ethernet Bridge (VEB) on RX reconfigured

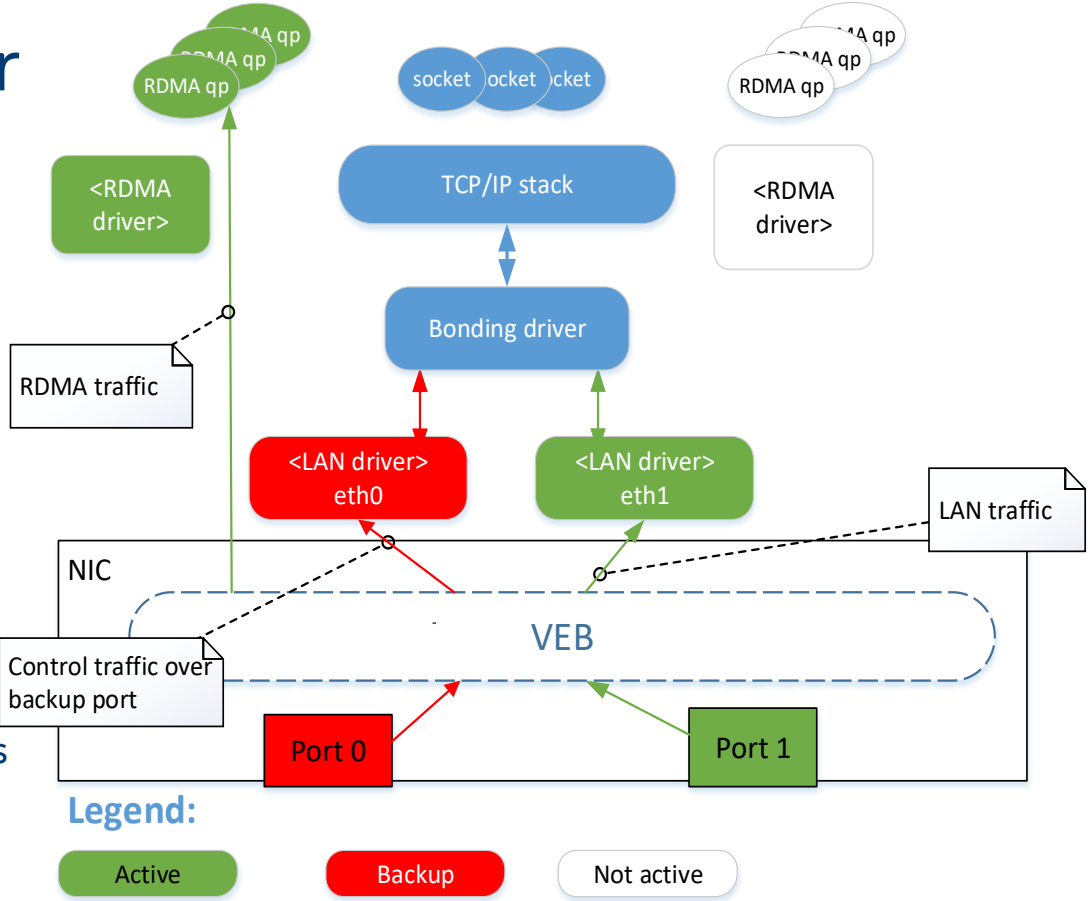
- Traffic from a new active port redirected to old active queues
- Control traffic from the new backup port go to old backup queues
  - LLDP, LACP

LAN drivers reconfigured

- Packets received by the old active queues passed via the new active netdevice

RDMA drivers not changed

- Traffic received on the same queues as before failover
- **[Configuration, statistics]**



Sharing resources between PFs

Separate network ports are managed by separate PCIe Physical Functions (PFs)

Each queue belongs to a given PF

- To redirect TX traffic, queue must be scheduled on the port belongs to another PF

New mechanism to enable sharing resources between PFs on the same NIC

- For security, all PFs involved must agree

A mechanism to move TX queues between ports

- Existing operations of a scheduler modified to be used for move RDMA queues and VF queues between ports

VF LAG:
Before failover

SR-IOV pass-trough mode

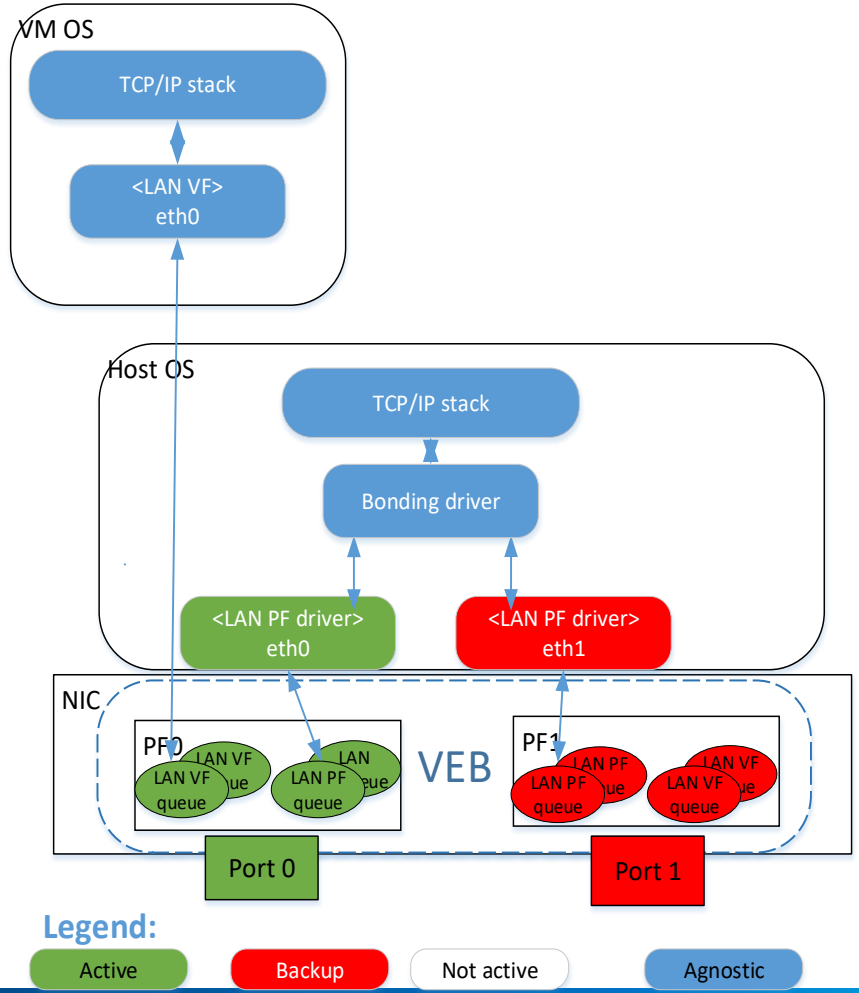
- A VM use VFs and HW-specific VF driver

VFs allocated from “active” PF

- Application queues available via VFs
- Backup PF not used to allocate VFs

Single Virtual Ethernet Bridge (VEB) configured on RX as for RDMA

- Host LAN traffic handled as for RDMA case



VF LAG:
After a failover

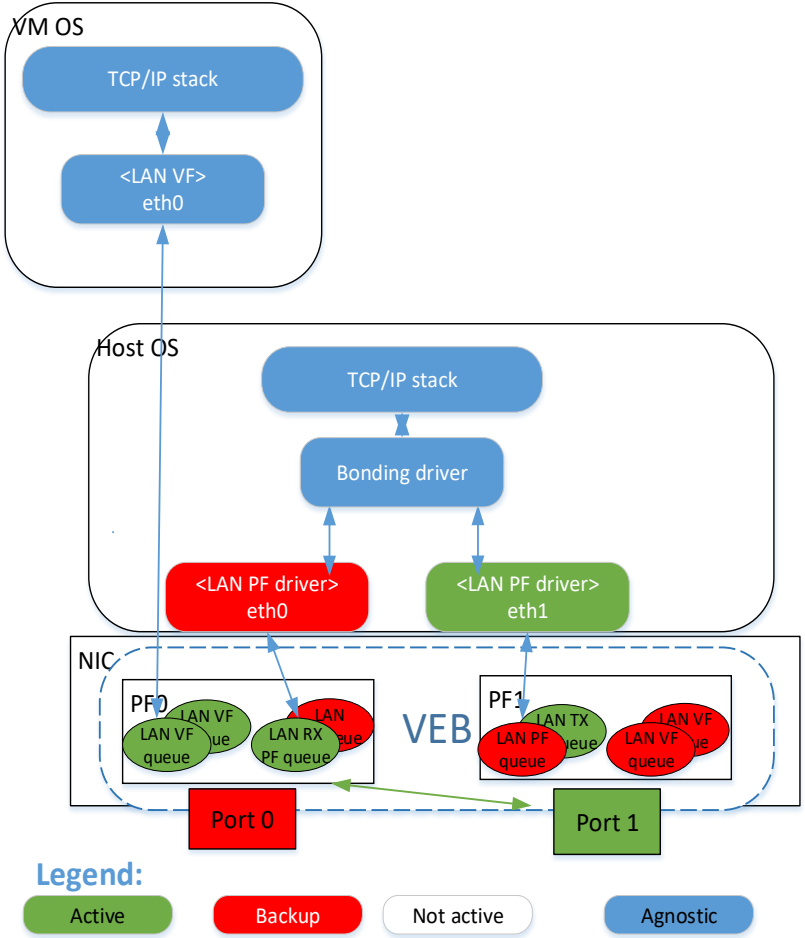
Performed similar to RDMA

TX direction:

- VF TX queues moved to a scheduler tree on the new active port

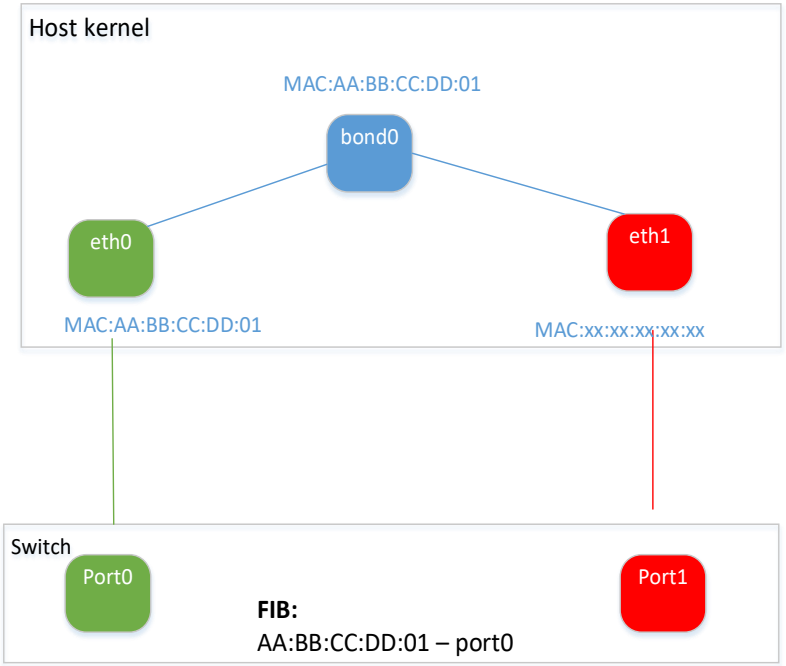
RX direction:

- VF RX queues still used
- All traffic from new active port redirect to old queues
  - Except control traffic – LLDP, LACP

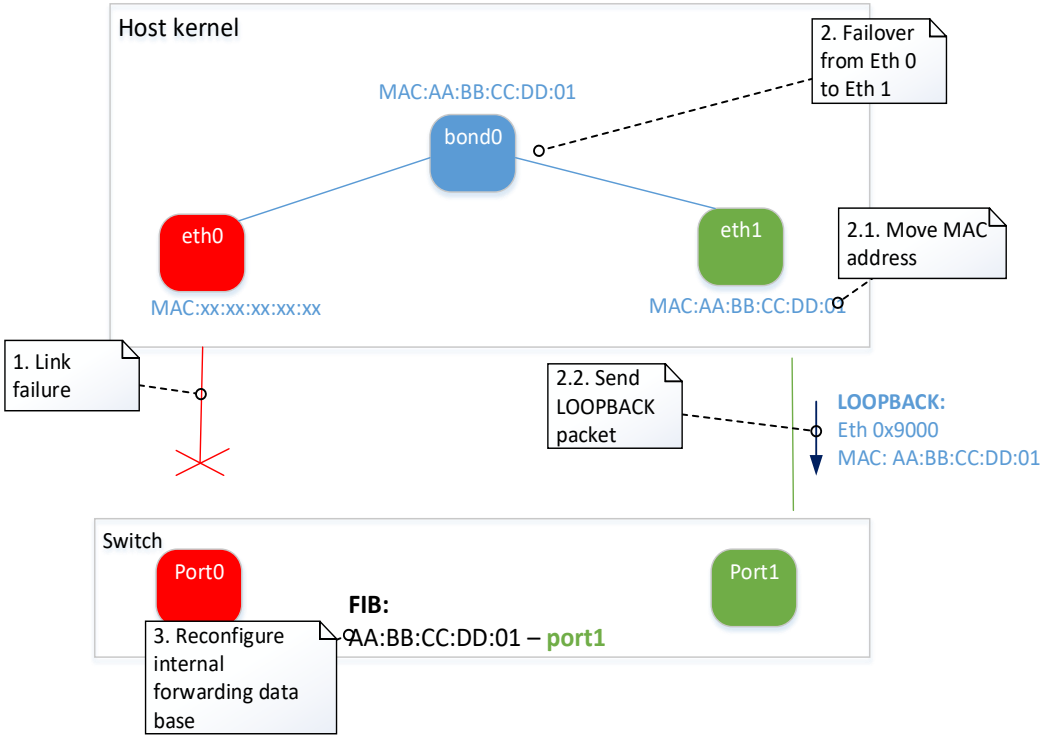


Host-switch synchronization during handover

Before fail-over



After fail-over



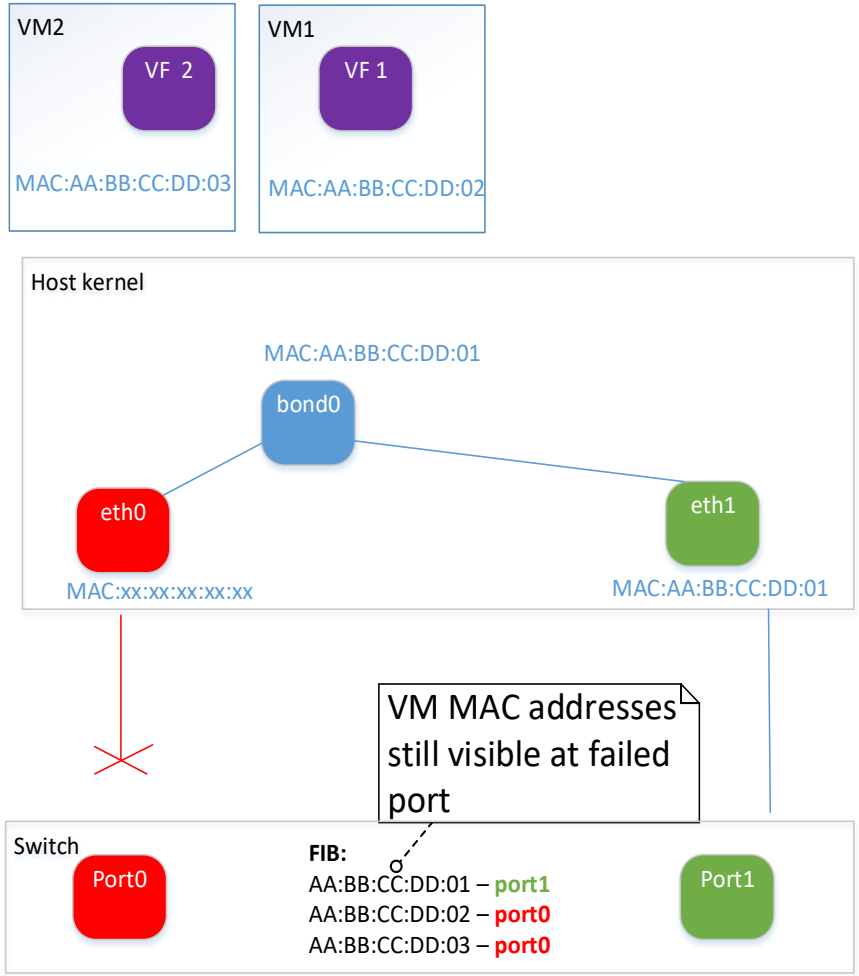
LAG for VMs: problem with the switch synchronization

Bonding driver notifies the Ethernet switch about MAC address assignment to port

- Only for bare metal LAN

Bonding driver is not aware of VMs

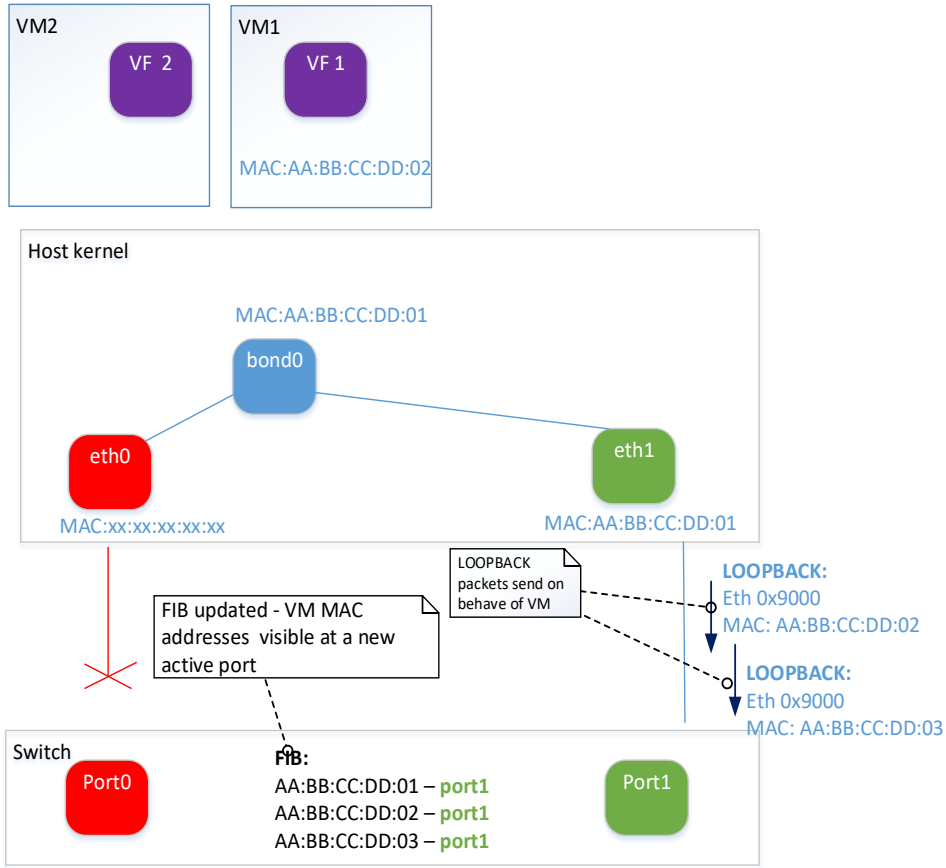
- It cannot communicate the changes to the switch
- Switch FIB is no updated – VMs are not available



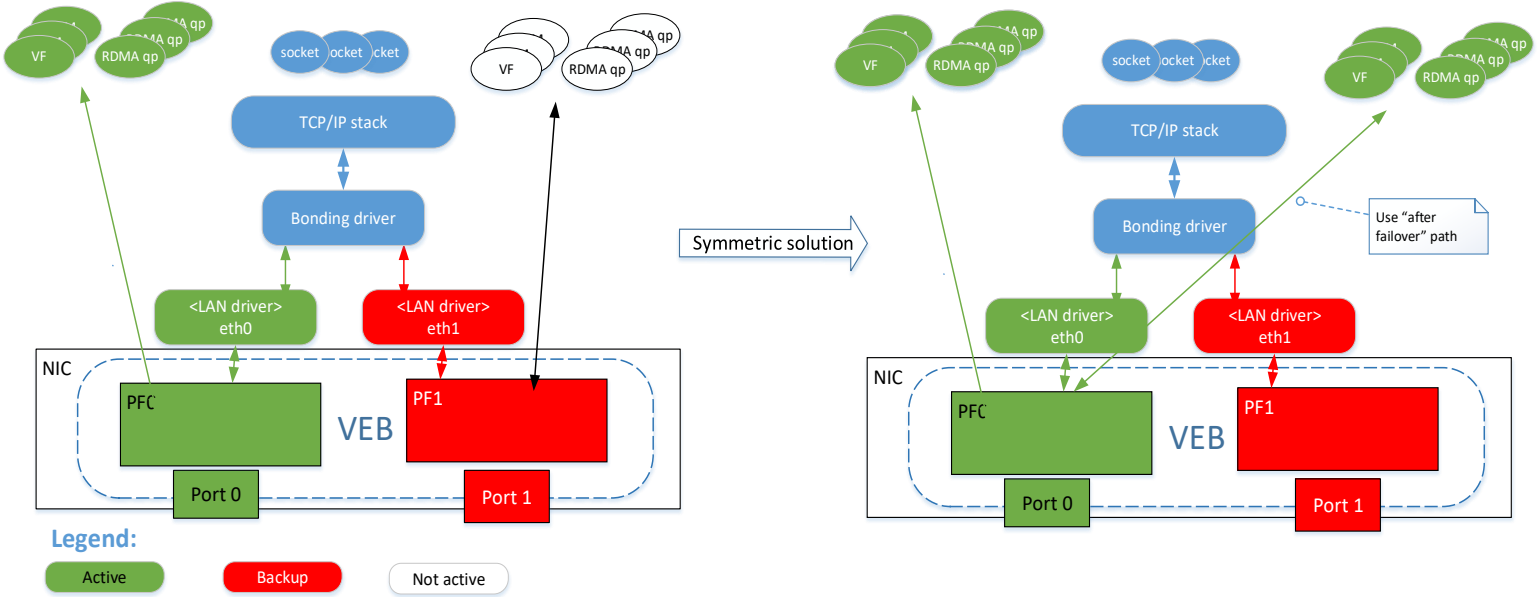
LAG for VMs: Ethernet switch FIB update

LAN driver notifies the Ethernet switch about VF MAC address assignment to a new port

- LAN PF driver knows all VFs
- The same LOOPBACK packet
- Switch FIB is updated – VMs are now available on the new port



Solution extension: use backup PF resources



Resources from backup PF can be also used when needed

- RDMA queues and VFs configured using "after failover" path
- Go back to "before failover" after actual failover

Conclusions and questions

Addressed problems:

- Active-backup for RDMA
- Seamless active-backup for VM

Remaining open:

- Active-active for RDMA and VMs

Looking for Your feedback about:

- Overall architecture
- Sending unsolicited LOOPBACK by the PF driver on behalf of VMs

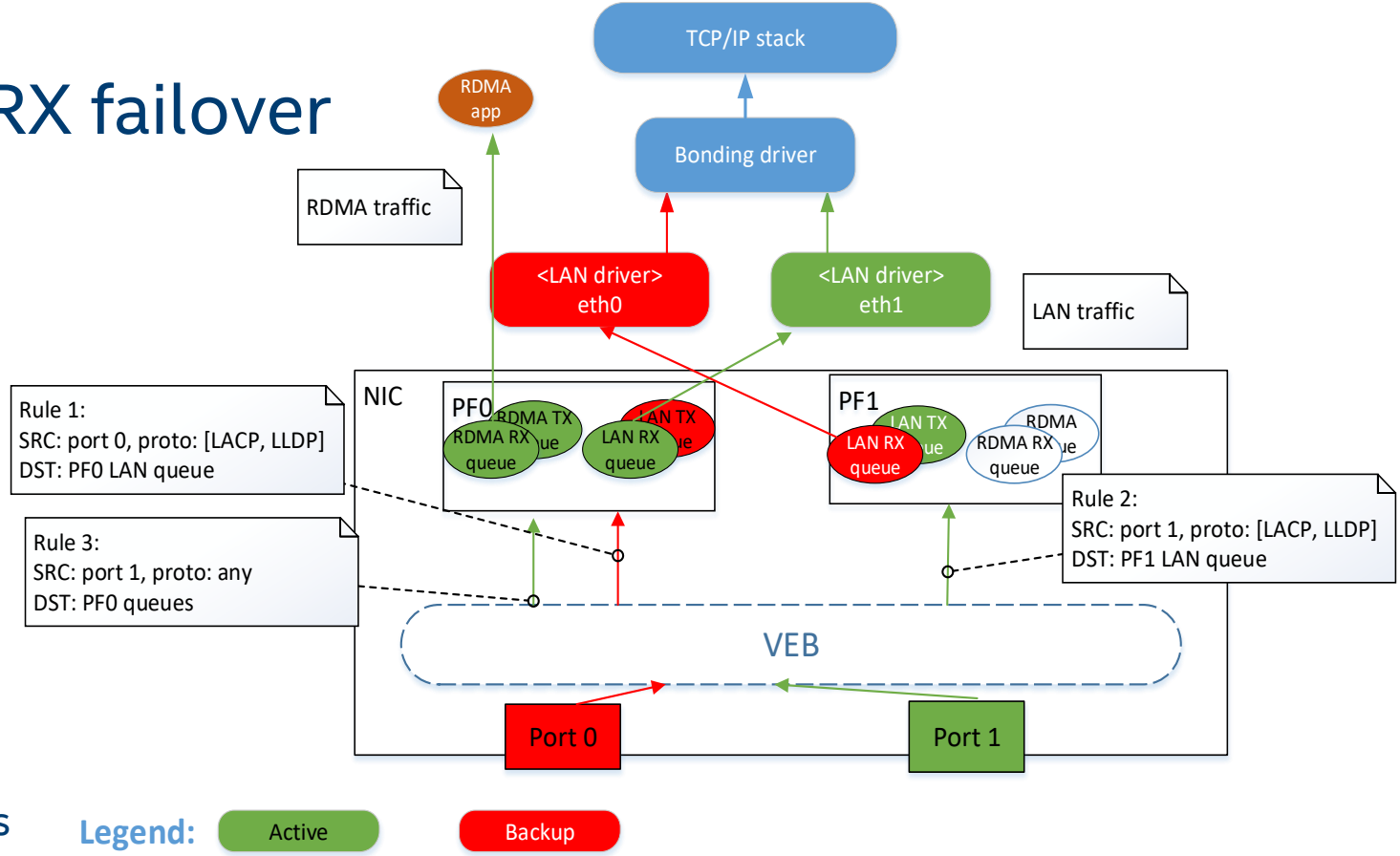
BACKUP

RDMA LAG:
Details of RX failover

Single VEB for both ports

RX rules to control packet flows

- Detailed control traffic rules
- Generic rules for the rest of the traffic
- SW control mapping RX queues to netdevs

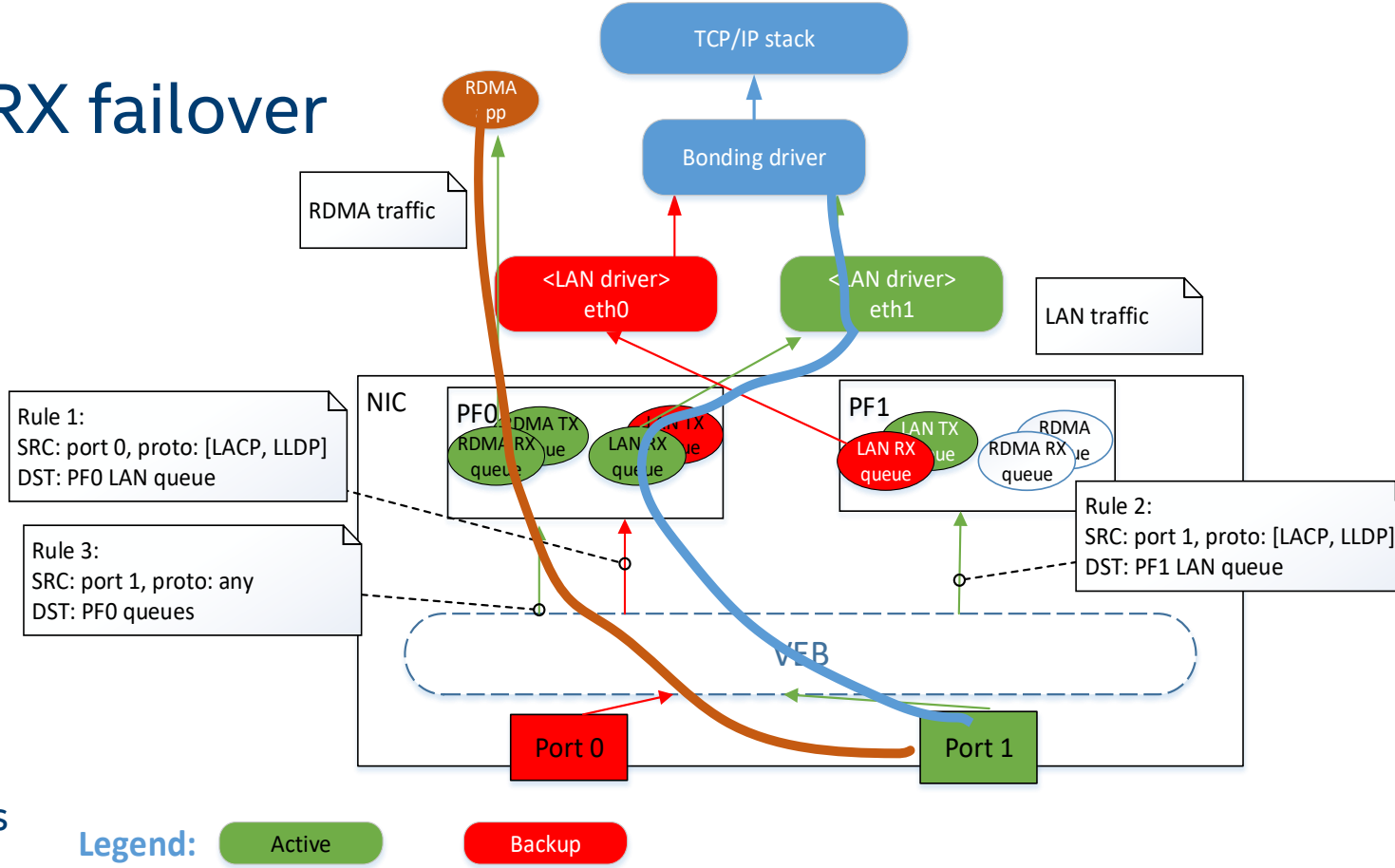


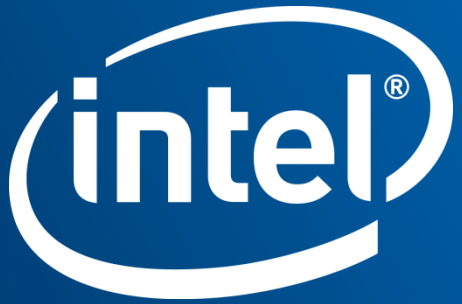
RDMA LAG:
Details of RX failover

Single VEB for both ports

RX rules to control packet flows

- Detailed control traffic rules
- Generic rules for the rest of the traffic
- SW control mapping RX queues to netdevs




intel®