INFINIBAND VIRTUALIZATION

Liran Liss

InfiniBand Trade Association

March, 2017
AGENDA

- InfiniBand Virtualization concepts
- VPort LIDs
- Packet relay
- VPort PortState
- Verbs
- Subnet management
- Subnet administration
- Performance management
Enable an HCA to expose multiple transport endpoints – Virtual HCAs (VHCAs)
- How VHCAs are presented to software is outside the scope of the specification
- Examples include multiple PCI functions, virtual functions, and logical HCAs

VHCAs have independent transport resources
- PDs, CQs, QPs, SRQs, MRs, AHs, etc.

VHCAs are connected through Virtual Ports (VPorts)
- Introduced through a new port capability
- Share physical port
- VPorts do not change the physical topology
  - Efficient use of fabric resources
  - Scalable
- VPorts are identified by unique GIDs
  - May optionally be assigned unique LIDs
**VPort properties**
- GID Table
- P_Key Table
- Logical PortState (independent from physical PortState)
- Capability Mask
- P_KeyViolations counter
- Q_KeyViolations counter
- Local port number (with respect to VHCA port numbering)
- LID (if a unique LID was requested for this VPort)
- Profile
- SL mask

**Remaining VPort attributes are shared from physical port**
- LID, LMC (applies only to physical port LID), SL2VL, VL arbitration, etc.
**INFINIBAND VIRTUALIZATION (CONT.)**

- **VPort properties are visible to subnet management**
  - Partition tables, GID tables, link state, etc.

- **The Subnet Manager (SM) manages VPorts similar to physical ports**
  - Enables connectivity only when configuration is allowed and consistent
  - Determines partitioning
  - Services path queries
    - Controls paths and QoS levels

- **VPorts are indistinguishable from physical ports to peers**
INFINIBAND VIRTUALIZATION (CONT.)

- The first VPort, VPort0, is privileged
  - Identical to a physical port when Virtualization is disabled
    - Links of other VPorts are forced down
  - Represents physical port when Virtualization is enabled
    - Handles privileged traffic
    - Default for traffic that doesn’t target other VPorts

- Virtualization must be explicitly enabled on each Node by a virtualization-aware SM
  - Ensures that VHCAs cannot initiate unauthorized traffic

<table>
<thead>
<tr>
<th></th>
<th>VPort0</th>
<th>VPortN; N&gt;0</th>
</tr>
</thead>
<tbody>
<tr>
<td>GID table</td>
<td>Mirrors physical port</td>
<td>Independent</td>
</tr>
<tr>
<td>P_Key table</td>
<td>Mirrors physical port</td>
<td>Independent</td>
</tr>
<tr>
<td>Capabilities</td>
<td>Mirrors physical port</td>
<td>Independent</td>
</tr>
<tr>
<td>SMP traffic</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Raw Ethertype traffic</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Raw IPv6 traffic</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>GMP traffic</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
- **VPorts typically share the physical port LID**
  - Efficient use of LID space and fabric resources

- **A VPort may request a unique LID**
  - E.g., for transparent migration of virtual machines (assuming other path attributes are stable)
  - The SM will activate the VPort only after assigning a unique LID

- **Additional VPorts may share the unique LID of a given VPort**
  - Association visible to SM
## PACKET RELAY

<table>
<thead>
<tr>
<th><strong>Unicast</strong></th>
<th><strong>Loopback</strong></th>
<th><strong>Multicast</strong></th>
</tr>
</thead>
</table>
| Relay packet as follows:  
  • If GRH is present  
    Forward to VPort for which  
    DLID = VPort LID and  
    DGID matches VPort GID table  
  • Forward to default VPort for DLID  
    (Even if GRH is present and DGID did not match any VPort)  
  • Drop packet (DLID miss)  | Loopback within physical port unchanged  
  Loopback within VPort as follows:  
  • Unconditionally if loopback indicator is set  
  • If GRH is present and both DGID and DLID match  
  • GRH is not present and this VPort is the default for the DLID | Any QP attached to MGID |
**VPORT PORTSTATE**

- **VPorts have an independent PortState**
  - SM / Host control individual VPorts
  - Intra-VPort traffic even if physical port is down

- **CA indicates when a VPort is enabled**
  - Policy out of the scope of the specification
    - E.g., Always enabled, follow physical link status
  - Determines the VPort's Virtual Link (VirtLink)

- **SM activates VPort**
  - Modifies PortState
VERBS

- **OpenHCA**
  - Returns a handle to a VHCA
  - Regardless of whether Virtualization was enabled by the SM (!)

- **QueryHCA**
  - CA attributes pertain to VHCA resources
  - The following Port Attributes correspond to the associated VPort
    - PortState
    - P_Key and GID Tables
    - P_Key and Q_Key violation counters
    - CapabilityMask bits
  - Base LID refers to either physical port (Shared LID) or VPort (dedicated LID)
  - Indicators for
    - Is this a VPort
    - Is this VPort0
    - Must traffic use a GRH (e.g., for SA queries)
  - PortState of physical port

  Typically referenced only by kernel code or management applications
VERBS (CONT.)

- **ModifyHCA**
  - The following Port Attributes correspond to the associated VPort
    - Optional shutdown port indicator
    - Q_Key Violation counter reset bit
    - CapabilityMask bits
    - Optional InitType value (VPort0 only)

- **Transport resource management**
  - Apply to VHCA resources

- **Asynchronous events**
  - Affiliated events and errors delivered to corresponding VHCA
  - Unaffiliated asynchronous events and errors
    - PortActive/PortError indicate VPort PortState transitions
    - PortChange event issued when
      - VPort GID or P_Key tables change
      - Physical PortInfo fields change
    - ClientReregistration issued when triggered on either VPort or physical port
### Virtualization support
- Indicated by a PortInfo:CapabilityMask2 bit – IsVirtualizationSupported

### VirtualizationInfo Attribute
- Virtualization capabilities, e.g.,
  - Number of VPorts
  - Highest enabled index
- Enable Virtualization
- VPort state change indication

### VPortState Attribute
- Aggregates the PortState of up to 128 VPorts per block
- Block number passed in attribute modifier
**VPortInfo**
- VPortState, GUIDCap, Capability Mask, LocalPortNum
- P_Key and Q_Key violation counters
- LID, LIDRequired, LIDByVportIndex
- ProfileID
- SLMask

**VNodeInfo**
- Partition table size
- System, Node, and Port GUIDs
- Number of local ports on VHCA

**VNodeDescription Attribute**
- Format identical to NodeDescription

**VPortGUIDInfo**
- Format identical to GUIDInfo

**VPortPartitionTable**
- Format identical to P_KeyTable

VPort Index indicated by Attribute modifier
The following trap types are defined for VPorts

<table>
<thead>
<tr>
<th>Trap</th>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1144</td>
<td>VPort Local Change</td>
<td>Informational</td>
</tr>
<tr>
<td>1146</td>
<td>VPort State Change</td>
<td>Urgent</td>
</tr>
<tr>
<td>1257</td>
<td>VPort P_Key Violation</td>
<td>Security</td>
</tr>
<tr>
<td>1258</td>
<td>VPort Q_Key Violation</td>
<td>Security</td>
</tr>
</tbody>
</table>

Traps 1144 and 1146 aggregate changes for all VPorts
- SM must query the Port to detect which VPorts have changed their state

Traps 1257 and 1258 are VPort specific
- Notice DataDetails indicates VPort index
DYNAMIC PORTSTATE CHANGES

Example

- Enable VPort0
- Set VPort State Change Indication
- VPort State Change Trap
- Enable VPort2
- Enable VPort3
- Trap repress
- Reset VPort State Change Indication
- Scan VPort State (bulk of 128 VPorts)
- Activate VPorts 0, 2, 3

Activate VPorts 0, 2, 3
VPorts access the SA via MADs with GRH
- DGID must be refer to well-known SA GUID
  - SubPfx | 0x02000000000002

Partition checks apply to VPort P_Key tables

VPort LIDs/GIDs may be provided in the following existing Attributes
- InformInfo
- InformInfoRecord
- ServiceRecord
- PathRecord
- MCMemberRecord
- MultiPathRecord
New SA attributes

- VirtualizationInfoRecord
- VNodeRecord
- VPortInfoRecord
- VPortGUIDInfoRecord
- VPortPartitionTableRecord

RID extended to include VPort index
- **New PerfMgt ClassPortInfo capability**
  - CapabilityMask2.IsVirtualizationSupported

- **Provides per VPort counters**
  - Similar to PortCountersExtended Attribute
  - VPort PMA does not support standard mandatory PRM counters of physical ports
    - E.g., PortCounters, PortSamplesControl, and PortSamplesResult

- **Counters**
  - PortXmitData
  - PortRcvData
  - PortXmitPkts
  - PortRcvPkts
  - PortUnicastXmitPkts
  - PortUnicastRcvPkts
  - PortMultiCastXmitPkts
  - PortMultiCastRcvPkts
  - PortRelayErrors
    - Accounts for SL Mask and GRH violations
THANK YOU

Liran Liss
InfiniBand Trade Association