What’s new – Volume 1 Release 1.6

Overview

Specification update overview

- Volume 1, Release 1.6, published July 15, 2022
- The specification defines InfiniBand and RoCE
- Available to IBTA Members

- 2074 pages
- 83 comments submitted and included
- New features added by both the LWG and the MgtWG

What's new in Vol1 Release 1.6

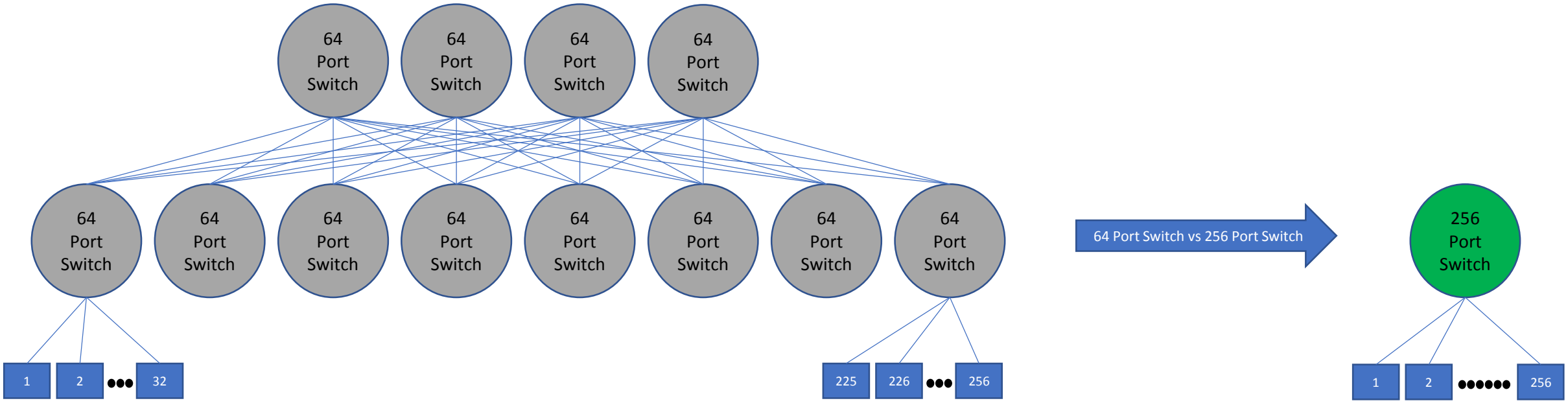
IBTA - Management Working Group

Support For Large Radix Switches

- 1.6 Spec Version
  - Support for class version 2 directed-route and LID-route SMPs
  - Updated the directed-route algorithm to be transparent and backward-compatible
  - Added multiple diagrams to describe how the directed-route algorithm supports large radix switches
  - Defined NodeInfo for class version 2
- Next Steps
  - Define class version 2 – SwitchInfo and other required management attributes

Support For Large Radix Switches

256 HCAs non blocking topology



	64 Port Switch	256 Port Switch
Cables	512	256
Switches	12	1
Max # hops	3	1

Next Generation Speed

- Spec 1.7 is expected to support XDR speed ~200Gb/s per lane.
  - QSFP ➔ 800 Gb/s
  - QSFP-DD and OSFP ➔ 1600 Gb/s
- Coordinate with EWG to support signaling rate and physical layer requirements.

What's new in Vol1 Release 1.6

IBTA - Link Working Group

Extended OpCodes

- See section 9.3.9
  - Two new OpCodes for generalized transport function classes
    - Get Class (Read like)
    - Put Class (Write like)
  - Extended OpCode Extended Transport Header (EOETH)
    - New 16b OpCode space per transport function class
    - Additional fields used for transport management
    - Each OpCode may define new headers to follow the EOETH

MPE - VERIFY CHECK

- Utilizes GET class extended OpCode with VERIFY CHECK OpCode specified in the EOETH Extended OpCode
- Provide native transport additions to allow a requestor to supply a Requestor Hash Value and ask the Responder to verify it calculates the same hash value
- Pipelined operation: Responder breaks the connection if the hash check fails allowing other writes, flushes, and verifies to be queued behind this operation
- The hash algorithm to be utilized is pre-determined by the requestor and responder at memory registration time and not specified in the VERIFY CHECK transport

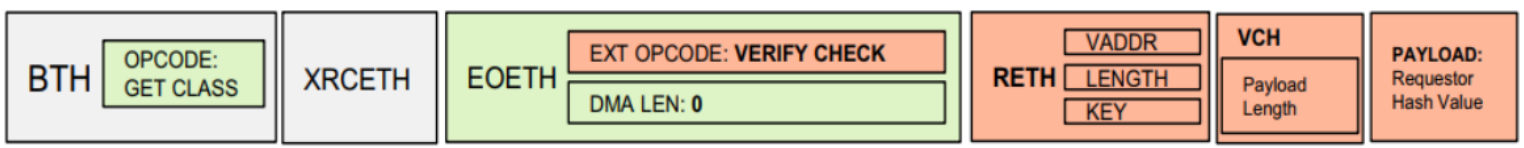


Figure 373: XRC VERIFY CHECK request packet format

Figure 374: RC VERIFY CHECK response packet format

MPE - VERIFY CHECK

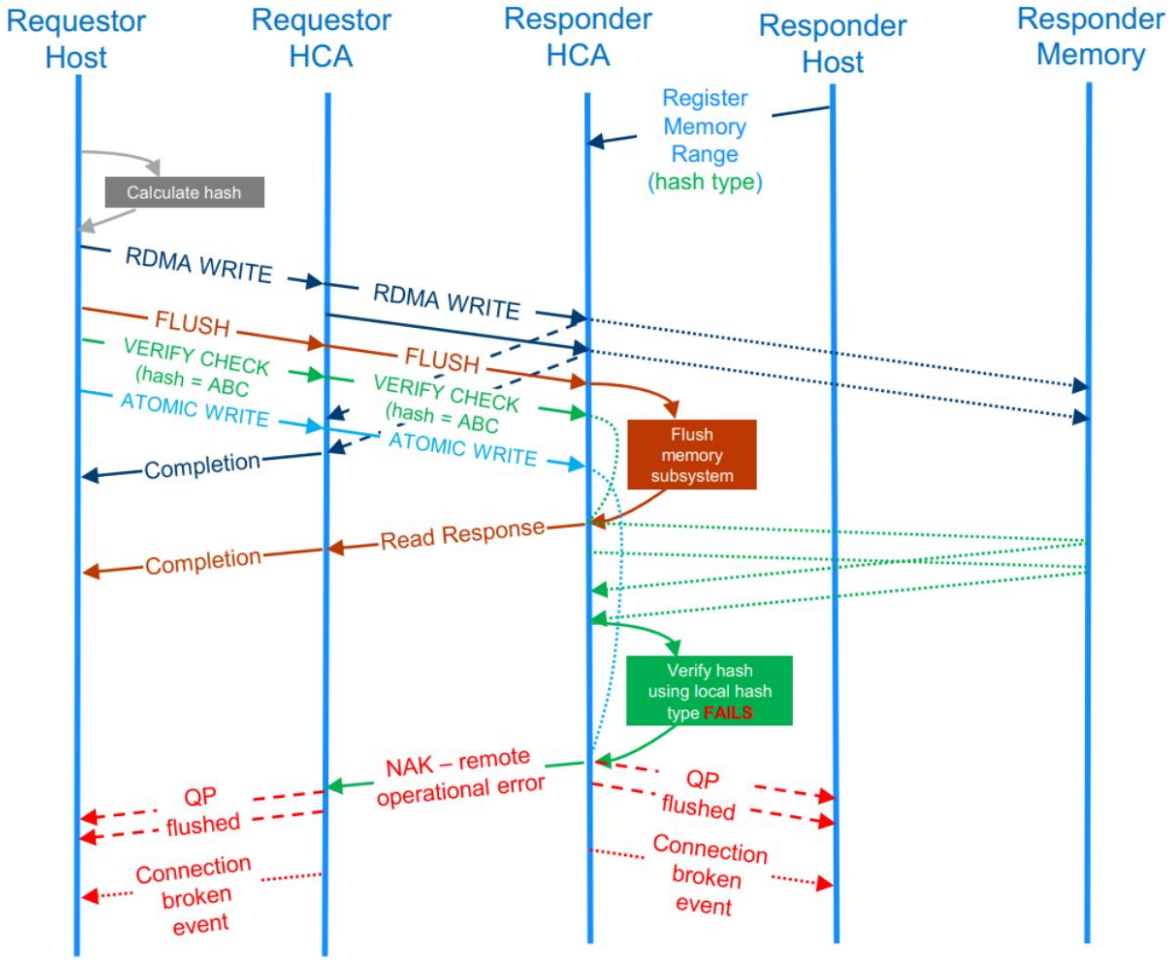


Figure 378: Transport Flow for VERIFY CHECK – Responder fails to verify requestor calculated hash

MPE - VERIFY COMPUTE

- Utilizes GET class extended OpCode with VERIFY COMPUTE opcode specified in the EOETH Extended Opcode
- Provide native transport additions to allow a requestor to request the responder to calculate a hash over a specified address range
- Responder returns its calculated hash result back to the requestor
- The hash algorithm to be utilized is pre-determined by the requestor and responder at memory registration time and not specified in the VERIFY COMPUTE transport

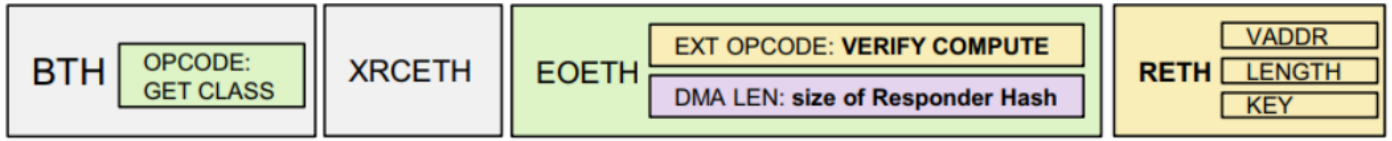


Figure 369: XRC VERIFY COMPUTE request packet format

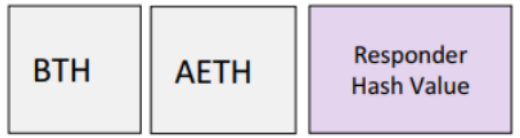


Figure 370: RC VERIFY COMPUTE response packet format

MPE - VERIFY COMPUTE

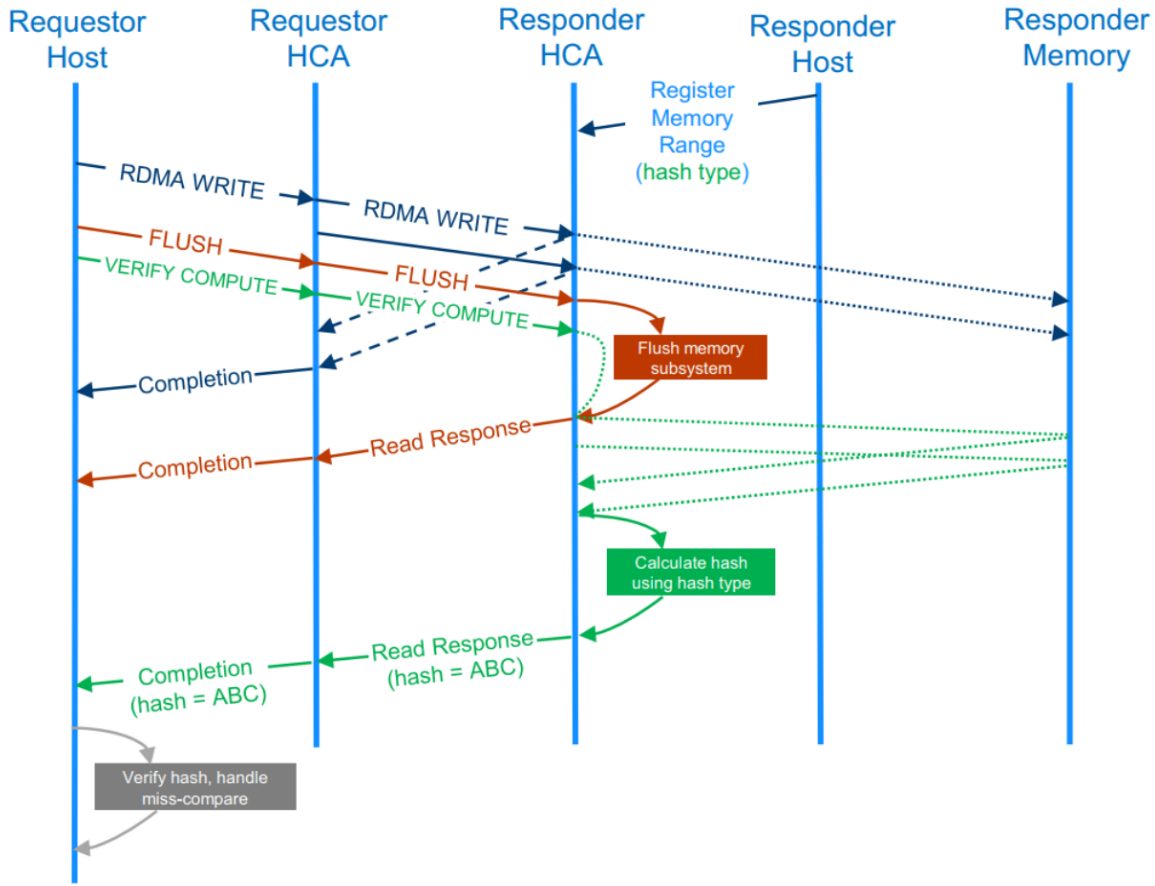


Figure 375: Transport Flow for VERIFY COMPUTE – requestor verifies responder calculated hash

For more information

https://www.infinibandta.org/ibta-specification/

- RDMA vendors:
  - Implement MPE in your InfiniBand and RoCE adapter(s)
- RDMA users:
  - Enhance your application(s) and ULP(s) to leverage MPE