Table 1 Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Release Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1a</td>
<td>6/30/2003</td>
<td>Errata to Release 1.1</td>
</tr>
</tbody>
</table>

LEGAL DISCLAIMER

"This IBTA document is provided "AS IS" and without any warranty of any kind, including, without limitation, any express or implied warranty of non-infringement, merchantability or fitness for a particular purpose.

In no event shall IBTA or any member of IBTA be liable for any direct, indirect, special, exemplary, punitive, or consequential damages, including, without limitation, lost profits, even if advised of the possibility of such damages."
CHAPTER 1: INTRODUCTION ERRATA

Page: 59  Line: 40  Section: 1.1.5.2  Ref: 3646
Replace Original Text   hexadecimal
With New Text   hexadecimal

Page: 60  Line: 8  Section: 1.1.5.2  Ref: 3647
Replace Original Text   hexadecimal
With New Text   hexadecimal
CHAPTER 2: GLOSSARY ERRATA

Page: 73  Line: 3  Section: 2  Ref: 3648

Replace Original Text   Infiniband
With New Text           InfiniBand
CHAPTER 3: ARCHITECTURAL OVERVIEW ERRATA

Page: 79 Line: 18, 20, 40 Section: 3.1.1 Ref: 3739
Replace Original Text  IBT
With New Text  IBA

Page: 80 Line: 24 Section: 3.3.2.1 Ref: 3661
Replace Original Text  IBA is architected as ...
With New Text  IBA is architected as ...

Page: 90 Line: 23 Section: 3.4.5.3 Ref: 3617, 3640
Replace Original Text
- Subnet Administration (SubnAdm) - this is a service provided by the SM that allows nodes to access information about the subnet to discover other nodes and services, to resolve paths, and to register its services.
- Performance Management (Perf) - monitors and reports well-defined performance counters
- Device Management (DevMgt) - provides for management of I/O devices behind TCAs.
- Baseboard Management (BM) - provides for chassis management using IB-ML as defined in Volume 2.
- SNMP Tunneling (SNMP) - provides SNMP functionality by defining the method for sending and receiving SNMP messages.
- Vendor Defined (Vendor) - allows private extensions that a device vendor may use to remotely configure and manage its devices.
- Communication Management (CommMgt) - Provides for connection establishment and other communication management functions between endnodes.
- Device Management (DevMgt) - Provides I/O resource management

With New Text
- Subnet Administration (SA) - this is a service provided by the SM that allows a node to access information about the subnet, to discover other nodes and services, to resolve paths, and to register its services.
- Performance Management - monitors and reports well-defined performance counters
- Baseboard Management - provides for chassis management using IB-ML as defined in Volume 2.
- SNMP Tunneling - provides SNMP functionality by defining the method for sending and receiving SNMP messages.
- Vendor Defined - allows private extensions that a device vendor may use to remotely configure and manage its devices.
• Communication Management (ComMgt) - Provides for connection establishment and other communication management functions between end nodes.
• Device Management (DevMgt) - Provides I/O resource management

Page: 99 Line: 3 Section: 3 Ref: 3679
Replace Original Text loosing
With New Text losing

Page: 105 Line: 39 Section: 3 Ref: 3679
Replace Original Text the a
With New Text the

Page: 111 Line: 30 Section: 3 Ref: 3679
Replace Original Text domains
With New Text domain

Page: 122 Line: 40 Section: 3 Ref: 3679
Replace Original Text is shares
With New Text it shares

Page: 123 Line: 32 Section: 3 Ref: 3679
Replace Original Text SMI (QP1)
With New Text GSI (QP1)

Page: 124 Line: 1 Section: 3 Ref: 3679
Replace Original Text manger
With New Text manager

Page: 125 Line: 9 Section: 3.9.1 Ref: 3624
Replace Original Text The MAD contains a standard header containing information common for all classes. This includes data for the reliable multi-packet protocol (RMPP). RMPP is a protocol that permits management entities to exchange more data than will fit in a single MAD.
With New Text MADs begin with a standard header that carries information common to all classes. For classes that use the reliable multi-packet protocol (RMPP), this is immediately followed by an additional header for RMPP information.
Page: 125 Line: 40 Section: 3 Ref: 3679
Replace Original Text information where
With New Text information about where

Page: 125 Line: 27 Section: 3.9.2.1 Ref: 3739
Replace Original Text IBT
With New Text IBA

Page: 126 Line: 37 Section: 3.9.4 Ref: 3739
Replace Original Text IBT
With New Text IBA

Page: 128 Line: 7 Section: 3.9.5 Ref: 3739
Replace Original Text IBT
With New Text IBA
no errata
no errata
no errata
CHAPTER 7: LINK LAYER ERRATA

Page: 162 Line: 8 Section: 7.7.2.3 Ref: 3734

Replace Original Text
For switches, ActiveTrigger occurs upon reception of a non-VL15 packet which passes the VCRC check on any port of the switch. Note, that for switches, the port receiving the packet could be in either Active or Armed state, and that ActiveTrigger is generated for all ports on the switch that are in Armed state.

With New Text
For switches, ActiveTrigger occurs upon reception of a non-VL15 packet which passes the VCRC check on any port of the switch. Note that for switches, the port receiving the packet could be in either Active or Armed state, and that ActiveTrigger is generated for all ports (including enhanced port 0) on the switch that are in Armed state.

Page: 185 Line: 19 Section: 7 Ref: 3766

Replace Original Text
7.7.9 SOURCE LOCAL IDENTIFIER (SLID) - 16 BITS

With New Text
Note: this compliance statement defines the maximum size packet. The d_length_check specified in Section 7.4 Data Packet Check is done against PortInfo:MTUCap only, however.

Page: 205 Line: 26 Section: 7.10.2 Ref: 3672

Replace Original Text
i.e. it is impossible to know which end nodes are participating within a multicast group

With New Text
i.e. except as noted in Section 15.2.5.17.5 QUERYING A MULTICAST GROUP it is impossible to know which end nodes are participating within a multicast group
C8-19: The network layer shall silently discard, with the exception of adjusting any applicable management counters specified elsewhere in this specification, packets that meet any of the following conditions:

- Value of IPVer is not 6.
- The value of DGID does not equal one of the GID values assigned to the port that received the packet.

With New Text:

Any required checks on DGID are performed at the transport layer, see section 9.6.1.2 GRH Checks.

C8-19: This conformance statement is obsolete
CHAPTER 9: TRANSPORT LAYER ERRATA

Page: 258 Line: 35 Section: 9.9.6.1.1.2 Ref: 3687

Replace Original Text
page 258 line 35, first column:
"Request packet: QP is not in a Ready-to-Send state, Send-Queue-Drain state, or Ready-to-Receive state.

With New Text
"Request packet: QP is not in a Ready-to-Send state, Send-Queue-Drain state, Ready-to-Receive state, or Send-Queue-Error state."

Page: 259 Line: 7 to 26 Section: 9.9.6.1.1.3 Ref: 3754

Add New Text
Add the following informative text below page 259 line 21:
For further details of the process for matching the P_Key please see section 10.9.3 Partition Key Matching.

Page: 266 Line: 21 to 29 Section: 9.9.7 Ref: 3756

Replace Original Text
There are several mechanisms available to accomplish this such as:
5) Set the AckReq bit on the last packet of every message thus guaranteeing that the responder will generate the needed explicit response,
6) Set the AckReq bit on the last packet of the message for which an explicit response is desired,
7) If the AckReq bit was not set for the request for which an explicit response was desired, the requester can retry the request (with AckReq set) thus requiring the responder to return a response,
8) If the AckReq bit was not set for the request for which an explicit response was desired, the requester can send a NOP command (e.g. RDMA WRITE request with a length of zero) and set the AckReq bit. This strategy only works if the responder supports RDMA WRITES.

With New Text
There are several mechanisms available to accomplish this such as:
1) Set the AckReq bit on the last packet of every message thus guaranteeing that the responder will generate the needed explicit response,
2) Set the AckReq bit on the last packet of the message for which an explicit response is desired,
3) If the AckReq bit was not set for the request for which an explicit response was desired, the requester can retry the request (with AckReq set) thus requiring the responder to return a response,
4) If the AckReq bit was not set for the request for which an explicit response was desired, the requester can send a NOP command (e.g. RDMA WRITE request with a length of zero) and set the AckReq bit. This strategy only works if the responder supports RDMA WRITES.
Add New Text

Add the following text below page 292 line 26:

"PSN of the most recent request", as used here and throughout Chapter 9, specifically means the PSN of the most recently received NEW request packet. (This distinguishes it from the PSN of a recently received duplicate request). Thus, the PSN of the most recently received request marks the point of greatest forward progress, as perceived by the responder, while ignoring duplicate requests.

Replace Original Text

C9-192: For an HCA responder using Unreliable Connection service, if the inbound request is for a RDMA WRITE and the requested DMA length in the RETH is non-zero, then the following conditions shall be checked:

• The R_Key field in the RETH is valid.
• The virtual address and length specified in the RETH are within the locally defined limits associated with the R_Key,
• The type of access specified (Write) is within the locally defined limits associated with the R_Key.

With New Text

C9-192: For an HCA responder using Unreliable Connection service, if the inbound request is for a RDMA WRITE and the requested DMA length in the RETH is non-zero, then the following conditions shall be checked:

• The R_Key field in the RETH is valid.
• The virtual address and length is within the locally defined limits associated with the R_Key.

For an RDMA WRITE request, the length check is conducted on a per packet basis and is based on the LRH:PktLen field.

• The type of access specified (Write) is within the locally defined limits associated with the R_Key.

Replace Original Text

If the requester’s RNR NAK retry counter decrements to zero, an RNR NAK retry error occurs.

With New Text

If the requester’s RNR NAK retry counter is zero, and an RNR NAK packet is received, an RNR NAK retry error occurs.

Replace Original Text

o9-157: If a TCA requester implements Reliable Connection service, or if a CA requester implements Reliable Datagram service, in response to a Requester Class E error, the requester shall drop the acknowledge message.

With New Text

o9-157: If a TCA requester implements Reliable Connection service, or if a CA requester implements Reliable Datagram service, in response to a Requester Class E error, the requester shall drop the acknowledge message. There is,
however, an exception to this rule. For reliable connected service, a duplicate
acknowledge message may be used by the responder to carry end-to-end flow
control credits to the requester (an “unsolicited acknowledge”). Thus, if the PSN
of the acknowledge message is one less than the requester’s expected PSN,
the requester must recover the end-to-end credits and discard the remainder of
the message. This behavior is detailed in section 9.7.7.2 End-to-End(Message
Level) Flow Control on page 332.

Page: 411 Line: 7 Section: 9.10.3 Ref: 3628

Replace Original Text
BTH P_Key description
Partition Key; checked against the port partition table and an index in the:
<footnote c>

With New Text
"Partition Key; checked against the port partition table <footnote c>.
CHAPTER 10: SOFTWARE TRANSPORT INTERFACE ERRATA

Page: 418 Line: 34 Section: 10.2.2.2 Ref: 3528
Add New Text

<Add the following text and compliance statement into 10.2.2.2 Destination Addressing before C10-6:>

The verbs consumer should avoid modifying or destroying an Address Handle while there are outstanding WRs that reference that Address Handle. In any case the CI must process any WR, which references a destroyed or modified Address Handle, and complete it either successfully or in error (according to the normal CQE generation rules). The CA may emit the packet only when the completion is successful, however, the destination address may be indeterminate. When the completion is in error, no packet is emitted at all.

C10-X1: The CI shall process any WR that references an Address Handle which was modified or destroyed while the WR has been outstanding. The WR shall be completed either successfully or in error (normal CQE generation rules apply). A packet shall only emitted when the operation completes successfully, though the destination address may be indeterminate. If the completion is in error the CI shall not emit the packet and the reported completion return status shall be Local QP Operation Error.

Page: 419 Line: 8 Section: 10.2.2.2 Ref: 3644
Replace Original Text

If parameters other than those obtained from exactly one applicable PathRecord are used as part of an Address Vector, this can have undesirable consequences including, but not limited to, packet discard and connection teardown. In particular, Address Vector information from Multicast MCMemberRecords and Unicast PathRecords should not be mixed.

With New Text

If parameters other than those obtained from exactly one applicable PathRecord are used as part of an Address Vector, this can have undesirable consequences including, but not limited to, packet discard and connection teardown.

<Delete original text>

Page: 421 Line: 13 Section: 10.2.4.5 Ref: 3682
Replace Original Text

C10-13: The CI shall only allow direct access to the SMI and GSI QPs by privileged mode Consumers.

With New Text

C10-13: The CI shall allow direct access to the SMI and GSI QPs only by privileged mode Consumers.

Page: 421 Line: 42 Section: 10.2.4.4 Ref: 3637
Replace Original Text

When a QP is destroyed, any outstanding Work Requests are no longer considered to be in the scope of the Channel Interface. It is the responsibility of
the Consumer to be able to clean up any associated resources. Destruction of a QP releases any resources allocated below the Channel Interface on behalf of the QP. Outstanding Work Requests will not complete after this Verb returns.

**With New Text**

When a QP is destroyed, any outstanding Work Requests are no longer considered to be in the scope of the Channel Interface. It is the responsibility of the Consumer to be able to clean up any associated resources. Destruction of a QP releases any resources allocated below the Channel Interface on behalf of the QP. Outstanding Work Requests will not complete after this Verb returns.

It is good programming practice to modify the QP into the Error state and retrieve the relevant CQEs prior to destroying the QP. Destroying a QP does not guarantee that CQEs of that QP are deallocated from the CQ upon destruction. Even if the CQEs are already on the CQ, it might not be possible to retrieve them. It is good programming practice not to make any assumption on the number of CQEs in the CQ when destroying a QP. In order to avoid CQ overflow, it is recommended that all CQEs of the destroyed QP are retrieved from the CQ associated with it before resizing the CQ, attaching a new QP to the CQ or reopening the QP, if the CQ capacity is limited.

---

**Add New Text**

<Add to 10.3.1, p 431 figure 123 the SQD2SQD state transition (green arch from SQ Drain to itself).>

---

**Add New Text**

SQD to SQD state transition can be performed when the SQ draining is completed, i.e. the CI has completed processing any outstanding message on a message boundary and processed any incoming acknowledgements, or while the SQ draining is in progress.

However, changing some of the optional attributes on a SQD to SQD transition if the SQ has not been fully drained is not allowed. The CI shall report an immediate error if the verbs consumer performs a SQD to SQD state transition and attempts to change such an optional attribute while the SQ has not been drained yet. Refer to 11.2.3.2 and 11.2.6.2 for the list of attributes that can not be changed while the SQ is still draining. To determine if the SQ has been drained, Software can use the Affiliated Asynchronous Event or use the Query QP or Query EE verb.

It is not allowed to perform SQD to RTS state transition if the SQ has not been drained yet. The CI shall report an immediate error if the consumer attempts to move the QP/EE from SQD to RTS if the SQ has not been drained.

---

**Replace Original Text**

C10-38: Receive Work Requests which were submitted to a Receive Queue prior to that queue’s transition into the SQEr state shall continue to be processed
normally. New Receives must be able to be posted to such a Receive Queue.

**With New Text**

C10-38: Receive Work Requests which were submitted to a Receive Queue prior to that queue’s transition into the SQEr state shall continue to be processed normally. New Receives must be able to be posted to such a Receive Queue. Incoming messages which target a QP in the SQEr state must be processed normally.

Page: 438  Line: 34  Section: 10.3.1.7  Ref: 3683

**Replace Original Text**

C10-42: Work Requests subsequent to that which caused the Completion Error leading to the transition into the Error state, including those submitted after the transition, must return the Flush Error completion status through the Completion Queue.

**With New Text**

C10-X2: Incoming messages which target a QP in the Error state must be silently dropped.

C10-42: Work Requests subsequent to that which caused the Completion Error leading to the transition into the Error state, including those submitted after the transition, must return the Flush Error completion status through the Completion Queue.

Page: 442  Line: 3  Section: 10.4.1.1  Ref: 3768

**Replace Original Text**

The Verbs Consumer or the CI may set the path migration state to Migrated when the current path migration state is Armed and the QP/EE state is RTS.

**With New Text**

The Verbs Consumer or the CI may set the path migration state to Migrated when the current path migration state is Armed and the QP/EE state is SQD or RTS.

Page: 442  Line: 26  Section: 10.4.1.1  Ref: 3768

**Replace Original Text**

The Verbs Consumer should only set the path migration state to Migrated when the current path migration state is Armed and the QP/EE state is RTS. The Modify Queue Pair or Modify EE Context Attributes Verbs shall generate an immediate error when the Verbs Consumer attempts to set the path migration state to Migrated under any other condition.

**With New Text**

The Verbs Consumer should set the path migration state to Migrated only when the current path migration state is Armed and the QP/EE state is SQD or RTS. The Modify Queue Pair or Modify EE Context Attributes Verbs shall generate an immediate error when the Verbs Consumer attempts to set the path migration state to Migrated under any other condition.

o10-30: If Automatic Path Migration is supported, the CI shall change the local path migration state to Migrated only when the current state is Armed and the...
VOLUME 1 - GENERAL SPECIFICATIONS  ERRATA

InfiniBand® Trade Association

Page: 453  Line: 11  Section: 10.6.4.4.1  Ref: 3723
Replace Original Text
C10-54.1.2: The CI shall not perform any local protection checking for any Data Segments specified in the Work Request on the receipt of a zero-length message.

With New Text
C10-54.1.2: For UD messages that do not contain a GRH, UC messages, RC messages, and RD messages, the CI shall not perform any local protection checking for any Data Segments specified in the Work Request on the receipt of a zero-length message. For UD messages that contain a GRH, the CI shall perform the Data Segment checks and, if the check pass, surface the GRH. If the checks don't pass, a completion error shall be surfaced.

Page: 473  Line: 16  Section: 10.8.7  Ref: 3751
Replace Original Text
C10-110: The CI shall replace any previous handler with the handler specified in a new Request Completion Event Verb invocation. The Request Completion Event Verb is set on a CQ basis. This is a one-shot notification; at most, one notification will be generated per call to this Verb. Once CQ notifications have been enabled, additional Request Completion Event calls have no effect. The handler will be called once when the next entry is added to the CQ specified as a modifier to this Verb. The presence of Solicited Events may impact this behavior. See 11.4.2.2 Request Completion Notification on page 557 & 9.2.3 Solicited Event (SE) - 1 bit on page 227 for details.

With New Text
C10-110: The CI shall replace any previous handler with the handler specified in a new Set Completion Event Handler Verb invocation. The Request Completion Notification Verb is set on a per-CQ basis. This is a one-shot notification; at most, one notification will be generated per call to this Verb. Once CQ notifications have been enabled, additional Request Completion Notification calls have no effect. The handler will be called once when the next entry is added to the CQ specified as a modifier to this Verb. The presence of Solicited Events may impact this behavior. See 11.4.2.2 Request Completion Notification on page 557 & 9.2.3 Solicited Event (SE) - 1 bit on page 227 for details.

Page: 477  Line: 21  Section: 10.9.4  Ref: 3762
Replace Original Text
o10-55: If the CA ports and the GSI port for switches and routers support the trap SMP for P_Key Violations, then if a packet’s P_Key does not match, the destination node shall send a trap SMP to the SM, specifying the partitioning class and the Bad P_Key Notification method. The body of the trap SMP must contain the header(s) of the offending packet. Like all traps, this one shall not be sent at a frequency faster than the Subnet Timeout.

With New Text
o10-55: If the CA ports and the GSI port for switches and routers support the trap SMP for P_Key Violations, then if a packet’s P_Key does not match, the destination node shall send a trap SMP to the SM, specifying the partitioning class and the Bad P_Key Notification method. The body of the trap SMP must
contain the offending packet's headers fields as specified in Table 118 Traps on page 724 and Table 125 Notice Data Details For Traps 257 and 258 on page 727. Like all traps, this one shall not be sent at a frequency faster than the Subnet Timeout.

Page: 482 Line: 16 Section: 10.10.2.3 Ref: 3782

Replace Original Text
C10-140: Unaffiliated Asynchronous Errors are handled according to type: local catastrophic errors shall place all QP/EEs in the Error State; local port errors shall have no effect on QP/EE State.

With New Text
<Delete original sentence. See additional text associated with this errata in the errata for Chapter 11>

Page: 482 Line: 29 Section: 10.10.3 Ref: 3673

Replace Original Text
No attempt is made to send a message below the Verbs to tear down a connection just because a QP has encountered an error. However, NAK codes which are generated as the result of a QP being in the error state will have an effect on the QP receiving those NAK codes.

With New Text
No attempt is made to send a message below the Verbs to tear down a connection just because a QP has encountered an error.

Page: 483 Line: 11 Section: 10.10.3.1 Ref: 3674

Replace Original Text
C10-144: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall be transitioned to the Error State. Any request in progress on the corresponding Work Queue shall be halted and shall not be completed successfully.

With New Text
C10-144: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall be transitioned to the Error State. Any request in progress on the corresponding Work Queue shall be halted and returned with a completion error, unless the CQ has overflowed or become inaccessible. If the CQ has overflowed or become inaccessible, then request in progress shall not be returned through the CQ.

Page: 485 Line: 7 Section: 10.10.3.2 Ref: 3674

Replace Original Text
o10-64: If the CI supports RD Service, Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and shall not be completed successfully.

With New Text
o10-64: If the CI supports RD Service, Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be
halted and returned with a completion error, unless the CQ has overflowed or become inaccessible. If the CQ has overflowed or become inaccessible, then request in progress shall not be returned through the CQ.

Page: 489 Line: 7 Section: 10.10.3.3 Ref: 3674

Replace Original Text
C10-150: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and shall not be completed successfully.

With New Text
C10-150: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and returned with a completion error, unless the CQ has overflowed or become inaccessible. If the CQ has overflowed or become inaccessible, then request in progress shall not be returned through the CQ.

Page: 490 Line: 18 Section: 10.10.3.4 Ref: 3674

Replace Original Text
C10-155: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and shall not be completed successfully.

With New Text
C10-155: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and returned with a completion error, unless the CQ has overflowed or become inaccessible. If the CQ has overflowed or become inaccessible, then request in progress shall not be returned through the CQ.

Page: 491 Line: 28 Section: 10.10.3.5 Ref: 3674

Replace Original Text
C10-160: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and shall not be completed successfully.

With New Text
C10-160: Affiliated Asynchronous Errors shall result in the QP processing being halted such that outstanding Work Requests are not completed successfully by the Channel Interface. The QP shall transition to the Error State. Any request in progress on the corresponding Work Queue shall be halted and returned with a completion error, unless the CQ has overflowed or become inaccessible. If the CQ has overflowed or become inaccessible, then request in progress shall not be returned through the CQ.
Page: 500 Line: 8 Section: 11.2.1.2 Ref: 3623

Replace Original Text  
Ability of this HCA to change the primary physical port for aQP or EEC when transitioning from SQD to RTS state.

With New Text  
Ability of this HCA to change the primary port for a QP or EEC when transitioning from SQD to SQD state.

Page: 500 Line: 29 Section: 11.2.1.3 Ref: 3645

Replace Original Text  
Input Modifiers:
- HCA handle.
- Port Attribute list (one list for each port on this HCA):
  - Optional System Image GUID.

With New Text  
Input Modifiers:
- HCA handle.
- Optional System Image GUID.
- Port Attribute list (one list for each port on this HCA):

Page: 504 Line: 41 Section: 11.2.2.2 Ref: 3603

Replace Original Text  
Handle

With New Text  
Handle

Page: 510 Line: 25 Section: 11.2.3.2 Ref: 3630

Replace Original Text  
Enable/Disable RDMA Operations

With New Text  
<In section 11.2.3.2 Modify QP, p. 510, line 25 (RTS to RTS row), third column (optional attributes), modify the following:>
Enable/Disable RDMA and Atomic Operations

Page: 510 Line: 40 Section: 11.2.3.2 Ref: 3629

Replace Original Text  
Minimum RNR NAK Timer Field (RC and RD QPs only).

With New Text  
<In section 11.2.3.2 Modify QP, p. 510, line 40 (SQErr to RTS row), third column (optional attributes), modify the following:>
Minimum RNR NAK Timer Field (RD QPs only).
**Add New Text**

<Add SQD2SQD state transition to Modify QP verb 11.2.3.2:>

Required attributes: none

Optional attributes:
- Enable/Disable RDMA & Atomic Operations (footnote a)
- Remote Node Address Vector (Connected QPs only) (footnote b)
- Alternate path address information (RC/UC QPs only)
- Path migration state
- Number of Outstanding RDMA Read/atomic ops at destination (footnote b)
- Number of local RDMA Read/atomic responder resources (footnote b)
- Q_Key
- P_Key index (footnote b)
- Local ACK Timeout (RC QP only) (footnote b)
- Retry count (RC QP only) (footnote b)
- RNR retry count (RC QP only) (footnote b)
- Number of WQEs
- Primary physical port associated with QP if HCA supports the capability to change the primary physical port for a QP when transitioning from SQD to SQD state (RC QPs only). (footnote b)
- Minimum RNR NAK Timer Field (RC and RD QPs only)

<Footnote b also needs to be added, stating:>

Footnote b - It is allowed to change this attribute only when the SQ is drained.

**Replace Original Text**

<text in optional attributes column of SQD to RTS Transition>

Remote Node Address Vector (Connected QPs only).
Alternate path address information (RC/UC QPs only).
Path migration state.
Number of Outstanding RDMA Read/atomic ops at destination.
Number of local RDMA Read/atomic responder resources.
Q_Key.
P_Key index.
Local ACK Timeout (RC QP only)
Retry count (RC QP only)
RNR retry count (RC QP only).
Number of WQEs.
Primary physical port associated with QP if HCA supports this capability (RC QPs only).
Current QP State.
Minimum RNR NAK Timer Field (RC and RD QPs only).
Activate send processing.

With New Text

- Enable/Disable RDMA and Atomic Operations (footnote a)
- Q_Key.
- Alternate path address information (RC/UC QPs only).
- Path migration state.
- Number of WQEs.
- Current QP State.
- Minimum RNR NAK Timer Field (RC and RD QPs only).

Page: 518 Line: 25 Section: 11.2.3.4 Ref: 3531

Replace Original Text

C11-12: Outstanding Work Requests on this QP shall not be completed after this Verb returns. Incoming operations destined for a QP that has been destroyed are discarded.

With New Text

C11-12: Incoming operations destined for a QP that has been destroyed are discarded.

The CI does not guarantee that CQEs generated for a QP prior to its destruction can be retrieved from the CQ after that QP has been destroyed.

Page: 525 Line: 15 Section: 11.2.6.2 Ref: 3623

Add New Text

- Remote Node Address Vector (footnote a)
- Alternate path address information
- Path migration state
- Local ACK Timeout (footnote a)
- Retry count (footnote a)
- RNR retry count (footnote a)
- Number of Outstanding RDMA Read/atomic ops at destination (footnote a)
- Number of local RDMA Read/atomic responder resources (footnote a)
- P_Key index (footnote a)
- Primary physical port associated with QP if HCA supports this capability (footnote a)

Footnote a - It is allowed to change this attribute only when the SQ is drained.

Page: 525 Line: 16 Section: 11.2.6.2 Ref: 3623

Replace Original Text

<text in optional attributes column of SQD to RTS Transition>
Remote Node Address Vector.
Alternate path address information.
Path migration state.
Local ACK Timeout.
Retry count.
RNR retry count.
Number of Outstanding RDMA Read/atomic ops at destination.
Number of local RDMA Read/atomic responder resources.
Q_Key.
P_Key index.
Primary physical port associated with EE if HCA supports this capability.
Activate send processing.

With New Text

Completion notification indicator. Must be specified and is only valid if the Send Queue was created with a Signaling Type of Selectable.

Replace Original Text
Invalid Completion Notification Type

With New Text

One or more QPs are allowed to be attached to a multicast group on the HCA. If the maximum number of multicast group attachments has already been reached
for the HCA when a QP attempts to attach to the multicast group, an error is returned.

**With New Text**

One or more QPs are allowed to be attached to a multicast group on the HCA. If the maximum number of multicast group attachments has already been reached for the HCA when a QP attempts to attach to the multicast group, an error is returned.

If an attempt is made to attach a particular QP to the same multicast group that it is already attached to, the operation will apparently succeed (i.e. return operation completed successfully). However, only one copy of each multicast message will be delivered to the attached QP.

---

**Replace Original Text**

Verb Results:
- Operation completed successfully.
- Invalid HCA handle.
- Invalid HCA port number.

**With New Text**

Verb Results:
- Operation completed successfully.
- Invalid HCA handle.

---

**Replace Original Text**

Completion notification indicator. Must be specified and is only valid if the Send Queue was created with a Signaling Type of Selectable.

**With New Text**

Completion Notification Indicator. Must be specified if the Send Queue was created with a Signaling type of Selectable. Ignored if the Send Queue was created with a Signaling type of Non-Selectable.

---

**Replace Original Text**

Invalid Completion Notification Type

**With New Text**

<Delete original text>

---

**Replace Original Text**

<row for Local Length Error Type>

Local Length | Processing | NAK - Remote Operational Error | NAK - Remote Operational Error

**With New Text**

<row for Local Length Error Type>

Local Length | Processing | NAK - Invalid Request | NAK - Invalid Request
Replace Original Text
- Invalid Physical Buffer List entry.

With New Text
<Delete original text>

Replace Original Text
Invalid EE Context State - Operation is not legal for the specified EE Context state.

With New Text
Invalid EE Context State - Operation is not legal for the specified EE Context state.
All the above completion errors are applicable for WRs submitted through the Post Send Request and Post Receive Request verbs, except for the Memory Window Bind Error status. WR submitted through the Bind Memory Window verb must complete with one of the following error codes: Success, Memory Window Bind Error, Local QP Operation Error or Work Request Flushed Error.

Replace Original Text
Send Queue Drained - Indicates that the Send Queue of the specified Queue Pair has completed the outstanding messages in progress when the state change was requested and, if applicable, has received all acknowledgements for those messages.

With New Text
Send Queue Drained - Indicates that the Send Queue of the specified Queue Pair or EE has completed the outstanding Messages in progress when the state change was requested and, if applicable, has received all acknowledgements for those messages; The event is also generated if the transition to SQD has been aborted by a transition into SQError, Error or Reset state.

Replace Original Text
C11-40.1.1: The CI shall generate an Invalid Request Local Work Queue Error when the transport layer detects a transport OpCode violation at the Responder. The affiliated QP shall be placed in the error state.

With New Text
C11-40.1.1: For RC Service, the CI shall generate an Invalid Request Local Work Queue Error when the transport layer detects a transport OpCode violation at the Responder. The Responder's affiliated QP shall be placed in the error state.

Replace Original Text
C11-40.1.2: The CI shall generate a Local Access Violation Work Queue Error when the transport layer detects a Request access violation at the Responder. The affiliated QP shall be placed in the error state.

With New Text
C11-40.1.2: For RC Service, the CI shall generate a Local Access Violation Work Queue Error when the transport layer detects a Request access violation at the Responder. The Responder's affiliated QP shall be placed in the error state.
Replace Original Text  
C11-42: The CI shall generate a Port Error when the link is declared unavailable.

With New Text  
C11-42: The CI shall generate a Port Error when the link is declared unavailable. Port Errors shall have no effect on QP/EE State.
CHAPTER 12: COMMUNICATION MANAGEMENT ERRATA

Page: 581 Line: 23 Section: 12.6.7 Ref: 3666

Replace Original Text
If non-zero, the length in bytes of valid Additional Reject Information

With New Text
If non-zero, the length in bytes of valid Additional Reject Information. The sender is not required to provide Additional Reject Information even if the Reason code it places in the REJ message allows for it. If the sender decides not to provide Additional Reject Information, it shall set this field to a value of zero.

Page: 582 Line: 19 Section: 12.6.7.2 Ref: 3666

Replace Original Text
Meaning of Additional Reject Information Field

With New Text
Meaning of Additional Reject Information Field (when present)

Page: 584 Line: 23 Section: 12.6.7.2 Ref: 3707

Replace Original Text
The consumer decided to reject the communication or EE context setup establishment attempt for reasons other than those listed in the other REJ codes. (Typically this happens based upon information being conveyed in the PrivateData field of a message.)

With New Text
The consumer decided to reject the communication or EE context setup establishment attempt for reasons other than those listed in the other REJ codes. Typically this happens based upon information being conveyed in the PrivateData field of a message. It can also happen because the Consumer decided for reasons unrelated to any CM message it received to terminate the communication or EE context setup establishment attempt. This would therefore be the appropriate Reason code to use if the Consumer decided to destroy the EEC in the midst of the communication or EE context setup establishment attempt.

Page: 587 Line: 18 Section: 12.6.11 Ref: 3663

Replace Original Text
DREP is sent in response to DREQ, and signifies that the sender has received the DREP.

With New Text
DREP is sent in response to DREQ, and signifies that the sender has received the DREQ.
Replace Original Text: comparing the PortInfo:SubnetPrefix fields
With New Text: comparing the PortInfo:GidPrefix fields

Replace Original Text: this field must the the LID
With New Text: this field must be the LID

Replace Original Text: If non-zero, the length in bytes of valid Additional Information
With New Text: If non-zero, the length in bytes of valid Additional Information. The sender is not required to provide Additional Information even if the AP status code it places in the APR message allows for it. If the sender decides not to provide Additional Information, it shall set this field to a value of zero.

Replace Original Text: "Completion" for the communication release protocol means that the communication is released; this protocol never fails to run to completion.
With New Text: "Completion" for the communication release protocol means that the communication is released; this protocol never fails to run to completion. The diagrams in this section assume that the Consumer runs the IBA communication establishment and release protocols to completion (successful or otherwise). It is legal, however, for the Consumer to destroy the EEC involved in the communication establishment/release attempt before that attempt has run to completion. If this occurs, the Consumer whose EEC has been destroyed has one of two options: they can ignore all incoming messages that pertain to the communication establishment/release attempt involving the EEC (and let the communications establishment attempt timeout), or they can send a REJ message with a "Consumer Reject" Reason code if they are in a CM state (i.e. REP Rcvd, MRA(REP) Sent, REQ Rcvd, MRA Sent) that allows a REJ to be sent.

Replace Original Text: In many of the states of the InfiniBand™ communication establishment and release protocols, there is a defined set of input messages that can legally be received while in that state. The general rule for handling input messages that cannot be legally received and acted upon while in that state is to ignore them. A CM shall not retry the REQ, REP, or DREQ messages more than the
With New Text

Several of the states in the CM protocols are ephemeral; the transition out of these states depends only upon a decision being made locally, not on any new input being received from the other party in the communication establishment or release attempt. The ephemeral states are:

- Peer Compare
- Timeout
- REJ Retry
- REP Rcvd
- DREQ Rcvd
- DREP Timeout
- REQ Rcvd
- REJ Sent
- RTU Timeout

The general rule for handling any input message that is received while in an ephemeral state is to hold that message pending until the CM protocol enters a non-ephemeral state.

In many of the non-ephemeral states of the InfiniBand™ communication establishment and release protocols, there is a defined set of input messages that can legally be received while in that state. Once the CM protocol has entered a non-ephemeral state, the general rule for handling an input message that cannot be legally received and acted upon while in that state is to ignore it. A CM shall not retry the REQ, REP, or DREQ messages more than the number of times specified by \textit{REQ:Max CM Retries}. 

Replace Original Text

The ServiceID implicitly defines whether the service is client/server or peer to peer, but the server application must inform its CM so that the CM will handle the inbound REQ correctly.

With New Text

The ServiceID implicitly defines whether the service is client/server or peer to peer, but the server application must inform its CM so that the CM will handle the inbound REQ correctly.

When the CM stat is IDLE, LISTEN, or TimeWait, the CEP is allowed to be in any of the Error, Reset, or Initialized states. In these CM states, which particular state the CEP is in is outside the scope of this specification. The CM should avoid transitioning a QP from the Error state to the Reset state before it has notified the Consumer that the QP has entered the Error state, and given the Consumer an opportunity to dequeue any Work Completions that may be associated with that QP. If the CM doesn't follow this suggestion the Consumer may be unable to dequeue the Work Completions associated with a QP after the QP is transitioned to the Reset state.
Replace Original Text  
Entry - CEP to Reset
Send REQ - Send REQ / CM to REQ Sent / CEP to Initialized

With New Text  
Send REQ - Send REQ / CM to REQ Sent / CEP to Initialized

Replace Original Text  
Entry - CM: Start Timer / CEP to Reset

With New Text  
Entry - CM: Start Timer

Replace Original Text  
Entry - CEP to Reset
Receive REQ - CM to REQ Rcvd / CEP to Initialized

With New Text  
Receive REQ - CM to REQ Rcvd / CEP to Initialized

Replace Original Text  
Entry - CM: Start Timer / CEP to Reset

With New Text  
Entry - CM: Start Timer

Replace Original Text  
If non-zero, the length in bytes of valid Additional Information

With New Text  
If non-zero, the length in bytes of valid Additional Information. The sender is not required to provide Additional Information even if the Status code it places in the SIDR_REP message allows for it. If the sender decides not to provide Additional Information, it shall set this field to a value of zero.

Replace Original Text  
The Status field tells whether the QPN and Q_Key fields are valid, and if not valid, the reason a valid QPN and Q_Key were not provided. Depending upon the value of the Status field, the Additional Information field may or may not have valid contents.

With New Text  
The Status field tells whether the QPN and Q_Key fields are valid, and if not valid, the reason a valid QPN and Q_Key were not provided. Depending upon the value of the Additional Information Length field, the Additional Information field may or may not have valid contents.
Page: 622  Line: 30  Section: 12.11.2.1  Ref: 3666

Replace Original Text

Meaning of Additional Information Field

With New Text

Meaning of Additional Information Field (when present)
Add New Text
Implementors must be aware that it is possible for retries of a Set() request to result in different GetResp() responses. For example, assume a Set() reached the responder and was successfully processed by the responder, but the corresponding GetResp() was dropped in the fabric. The sender then has no indication that the Set() was processed, and as a result may retry the Set(). Such a retry may result in a GetResp() with the Status field set to an error code because the Set() request was received twice by the responder. This may be the case for a Set() that causes some entity to be deleted from a table; the retried Set() will attempt to delete an entity that no longer exists and as a result returns an error.

Replace Original Text
The value shall not be zero.

With New Text
<Delete Original Text>

Replace Original Text
C13-30.1.1: If a management class defines any Notice attributes, then the class manager shall support both the reception of Trap()s and the Notice Queue operations of polling and deleting entries.
C13-30.1.2: An agent shall either send a Trap() or store a notice in its Notice Queue or both when the condition or event corresponding to the notice occurs.

With New Text
C13-30.1.1: If a management class defines any Notice attributes, then the class manager shall support both the reception of Trap()s and the Notice Queue operations of polling and deleting entries, unless otherwise specified by a compliance statement for that class.
C13-30.1.2: An agent shall either send a Trap() or store a notice in its Notice Queue or both when the condition or event corresponding to the notice occurs, unless otherwise specified by a compliance statement for that class.

Replace Original Text
C13-34: For each endport, unless redirected (see 13.5.2 GSI Redirection on page 665), SA or GS MADs to be processed at that port shall be destined to Queue Pair 1.

With New Text
C13-34: For each endport, unless redirected (see 13.5.2 GSI Redirection on page 665), SA or GS MADs to be processed by agents at that port shall be destined to Queue Pair 1.
Page 672 Line: 35 Section: 13.5.3.3 Ref: 3662

**Replace Original Text**

```
and ((LRH:LNH=0b10) & (LRH:PktLen =72)) | ((LRH:LNH=0b01) & (LRH:PktLen =82)) /* C7-11:, C13-3: */
```

**With New Text**

```
and ((LRH:LNH=0b10) & (LRH:PktLen =72)) | ((LRH:LNH=0b11) & (LRH:PktLen =82)) /* C7-11:, C13-3: */
```

Page 674 Line: 23 Section: 13.5.3.3 Ref: 3668

**Add New Text**

```
<Add the following comment block near first "diamond" of Figure 159>
Note: If PortInfo:IsSMDisabled=1, then PortInfo:IsSM must be 0 <cross ref to C14-70>. 
```

Page 674 Line: 29 Section: 13.5.3.3 Ref: 3669

**Replace Original Text**

```
<Arrow in Figure 159 labeled "GetResp, Trap or otherwise" from "Method" diamond to “Drop Packet” oval>
```

**With New Text**

```
<Get rid of arrow specified in Current Spec Text.  
Change “Get (SMInfo), Set(SMInfo)” arrow label from “Method” diamond to “Optional:....” rounded rectangle to the following:>
```

```
Get(SMInfo),  
Set(SMInfo),  
GetResp, Trap or otherwise
```

Page 677 Line: 3 Section: 13.5.3.3 Ref: 3670

**Replace Original Text**

```
<reference to C14-19: 1>
```

**With New Text**

```
<reference to C14-13: 1>
```

Page 684 Line: 18 Section: 13.6.2.1 Ref: 3691

**Replace Original Text**

```
RMPPFlags.Active
```

**With New Text**

```
RMPPFlags.Active
```

Page 685 Line: 38 Section: 13.6.2.2 Ref: 3692

**Replace Original Text**

```
Also note that in that same table, “sender” refers to the entity that sends the MAD containing the status code, not to the “Sender” role in RMPP; i.e., the sender can be either a Sender or a Receiver.
```

**With New Text**

```
Also note that in that same table, “transmitter of the packet” refers to the entity that sends the MAD containing the status code, i.e., the transmitter of the packet can be either a Sender or a Receiver.
```
Replace Original Text: The sender must terminate
With New Text: The transmitter of the packet must terminate

Replace Original Text: shall cause the receiver to emit
With New Text: shall cause the packet's recipient to emit

Replace Original Text: An ACK or RETRY packet was received for a segment that has not been sent.
With New Text: An ACK packet was received for a segment that has not been sent.

Replace Original Text: was received by the sender
With New Text: was received by the transmitter of the packet

Replace Original Text: The sender does not
With New Text: The transmitter of the packet does not

Replace Original Text: ; if this time period expires without receiving the expected ACK, the Sender retries the last DATA packet in the window.
With New Text: <replace text starting at ';' to end of paragraph with:>
. See <ref to Figure 178> for the Sender behavior when this time period expires.

Replace Original Text: sender
With New Text: Sender

Replace Original Text: specified by the receiver have been sent
With New Text: specified by the Receiver have been sent
Page: 693  Line: 30  Section: 13.6.4  Ref: 3718

Replace Original Text  receiver
With New Text  Receiver

Page: 693  Line: 33  Section: 13.6.4  Ref: 3694

Replace Original Text  does not result in the sender incorrectly
With New Text  does not result in the Sender incorrectly

Page: 695  Line: 17  Section: 13.6.5.1  Ref: 3695

Replace Original Text  ES > NS
With New Text  ES >= NS

Page: 696  Line: 25  Section: 13.6.5.2  Ref: 3696

Replace Original Text  TID Check?
With New Text  <remove question mark>
  TID Check

Page: 696  Line: 29  Section: 13.6.5.2  Ref: 3714

Replace Original Text  Version Check?
With New Text  <remove question mark>
  Version Check

Page: 697  Line: 14  Section: 13.6.5.3  Ref: 3719

Replace Original Text  receiver
With New Text  Receiver

Page: 698  Line: 6  Section: 13.6.5.4  Ref: 3702

Replace Original Text  ABORT(T2L), terminate
With New Text  ABORT(T2L), purge context

Page: 699  Line: 8  Section: 13.6.5.4  Ref: 3701

Replace Original Text  terminate
With New Text  purge context
Replace Original Text: setting WF from packet
With New Text: setting NS = WF = 1 and WL = NewWindowLast (NWL) from ACK packet

Replace Original Text: <Type? decision block>
With New Text: <Type? outputs should be the following:
DATA - as is
ACK - as is
other - as is, but change block at end of "other" arrow to be "discard pkt, ABORT(BadT), terminate"
ABORT, STOP - new output flow that goes to block "discard pkt, terminate"
Note that the "False" flow from IsDS? still goes to the same block that "other" goes to.>

Replace Original Text: receiver
With New Text: Receiver

Replace Original Text: an STOP
With New Text: a STOP

Replace Original Text: ABORT(TMR), terminate
With New Text: ABORT(TMR), purge context

Replace Original Text: <Type? decision block>
With New Text: <Type? outputs should be the following:
DATA - as is
ACK - as is
other - as is, but change block at end of "other" arrow to be "discard pkt, ABORT(BadT), terminate"
ABORT, STOP - as is, but make arrow go to a block that
has "discard pkt, terminate">

Page: 701  Line: 28  Section: 13.6.5.5  Ref: 3700

Replace Original Text  ABORT(N2S)
With New Text  ABORT(W2S)

Page: 703  Line: 11  Section: 13.6.6.1  Ref: 3704

Replace Original Text  ABORT(TMR), terminate
With New Text  ABORT(TMR), purge context

Page: 703  Line: 18  Section: 13.6.6.1  Ref: 3737

Replace Original Text  <Type? decision block>
With New Text  <Type? outputs should be the following:
DATA - as is
ACK - as is
other - as is, but change block at end of "other" arrow to be "discard pkt, ABORT(BadT), terminate"
ABORT, STOP - new output flow that goes to block "discard pkt, terminate">

Page: 704  Line: 4  Section: 13.6.6.2  Ref: 3721

Replace Original Text  its receiver to begin sending
With New Text  its recipient (the Sender) to begin sending

Page: 704  Line: 13  Section: 13.6.6.2  Ref: 3705

Replace Original Text  ABORT(TMR), terminate
With New Text  ABORT(TMR), purge context
CHAPTER 14: SUBNET MANAGEMENT ERRATA

Page: 723  Line: 37  Section: 14.2.5  Ref: 3769

Add New Text

Several of the SM attributes described in the sections that follow (LinearForwardingTable, RandomForwardingTable, MulticastForwardingTable, VLArbitrationTable, GUIDInfo, and P_KeyTable) are used to load and read contents of tables in switches and CAs. Each of those attributes uses a block of table entries and an offset into the table specified using the MadHeader:AttributeModifier component. Successive block elements are loaded or read starting at the specified offset.

The number of table entries in a block is fixed, and the offset (AttributeModifier) is specified in units of block size. Also, there is no requirement that the actual table lengths, as indicated by related Cap or Top components, be an integer multiple of the block size. <Footnote: All of the table attributes except LinearForwardingTable use a Cap component only. For example, SwitchInfo:RandomFDBCap is used for RandomForwardingTable. The LinearForwardingTable uses both a Cap and a Top. See <ref to Section 14.2.5.10 LinearForwardingTable>.> This implies that the last, i.e., largest offset, block that is loaded or read from a table may overflow past the end of a table, implicitly addressing table entries that are invalid. (When the table size happens to be less than the block size, such a "last" block will also be the first, and only valid, block.)

As long as some of a block's entries address valid table entries, this is not an error. The block elements corresponding to invalid table entries are ignored on write and read back as all zeros.

However, if none of the block entries address valid table entries, it must mean that the AttributeModifier value specifies an offset that is past the end (Cap or Top) of the table. In this case a MADHeader:Status.Code of 7 is returned, since that is the status code for an invalid attribute modifier value <ref to Section 13.4.7 Status Field>. The attribute contents accompanying that Status Code are implementation-specific.

Page: 732  Line: 21  Section: 14.2.5.5  Ref: 3770

Replace Original Text

Valid values are from 0 to 31 and are further limited by the size of the GUIDCap of the port. Any entries in the block beyond the end of the GUID table are ignored on write and read back as zero.

With New Text

Valid values are from 0 to 31 and are further limited by the size of the GUIDCap of the port; see <ref to Sec 14.2.5 Attributes>. 
Page: 744  Line: 25  Section: 14.2.5.7  Ref: 3771

**Replace Original Text**

Valid values are 0 - 2047, and are further limited by the size of the P_Key table (specified by the PartitionCap for CAs, routers, and switch management ports or PartitionEnforcementCap for external ports on switches) for that node.

**With New Text**

Valid values are 0 - 2047, and are further limited by the size of the P_Key table for that node (specified by the PartitionCap for CAs, routers, and switch management ports or PartitionEnforcementCap for external ports on switches). See <ref to Sec 14.2.5 Attributes>.

Page: 744  Line: 34  Section: 14.2.5.7  Ref: 3779

**Replace Original Text**

Any entries in the P_KeyTable Block beyond the end of the P_KeyTable are ignored on write and read back as zero.

**With New Text**

<Delete Original Text>

Page: 746  Line: 33  Section: 14.2.5.9  Ref: 3784

**Add New Text**

<Add the following beyond line 33 as a new paragraph:>

The size of the VLArbitrationTable, based on PortInfo:VLArbitrationHighCap and PortInfo:VLArbitrationLowCap, further limits the valid values. See <ref to Sec 14.2.5 Attributes>.

Page: 747  Line: 5  Section: 14.2.5.9  Ref: 3612

**Replace Original Text**

The interpretation is as follows:

1 - values 0 -31 of low priority
2 - values 32 -63 of low priority
3 - values 0 - 31 of high priority
4 - values 32 -63 of high priority

**With New Text**

<Remove Original Text from the description column.>

Page: 747  Line: 24  Section: 14.2.5.10  Ref: 3772

**Replace Original Text**

and are further limited by the size of the Linear Forwarding Table of the switch. Any entries in the block beyond the end of the table are ignored on write and read back as zero.

**With New Text**

and are further limited by the size of the LinearForwardingTable of the switch. An implementation may choose either SwitchInfo:LinearFDBCap or SwitchInfo:LinearFDBTop to denote the size of the LinearForwardingTable. See <ref to Sec 14.2.5 Attributes>.

Page: 748  Line: 5  Section: 14.2.5.11  Ref: 3773

**Replace Original Text**

Valid values are from 0 to 3071, and are further limited by the size of the
Random Forwarding Table of the switch.
Any entries in the RandomForwardingTable Block beyond the end of the
RandomForwardingTable are ignored on write and read back as zero.

**With New Text**
Valid values are from 0 to 3071, and are further limited by the size of the
RandomForwardingTable of the switch based on SwitchInfo:RandomFDBCap.
See <ref to Sec 14.2.5 Attributes>.

Page: 748  Line: 9  Section: 14.2.5.11  Ref: 3621

**Add New Text**
<Append the following text to the paragraph on pg 748, line 8>
If an invalid port number is written into an entry that has the Valid bit set to 1,
packets sent to the LIDs specified in the entry will be discarded and that entry’s
Port shall be read back as 0xFF to indicate that an invalid port number was
used. If two or more entries specify LID ranges that overlap based on LMCs, and
the Valid bit for these entries is set to 1, switch forwarding behavior for packets
sent to these LIDs is implementation-dependent.

Page: 748  Line: 32  Section: 14.2.5.12  Ref: 3615

**Replace Original Text**
The ten low-order bits of the AttributeModifier are a pointer to a block of 32
PortMask entries to which this attribute applies.

**With New Text**
The nine low-order bits of the AttributeModifier are a pointer to a block of 32
PortMask entries to which this attribute applies.

Page: 748  Line: 33  Section: 14.2.5.12  Ref: 3774

**Replace Original Text**
Valid values are limited by the size of the Multicast Forwarding Table of the
switch. See Figure 190 MulticastForwardingTable Bit Layout on page 749 for
how blocks of PortMasks are mapped into MulticastForwardingTable entries.
Any entries in the MulticastForwardingTable Block beyond the end of the
MulticastForwardingTable are ignored on write and read back as zero.

**With New Text**
Valid values are limited by the size of the MulticastForwardingTable of the switch
based on SwitchInfo:MulticastFDBCap; see <ref to Sec 14.2.5 Attributes>. See
<ref to Figure 190 MulticastForwardingTable Bit Layout on page 749> for how
blocks of Port-Masks are mapped into MulticastForwardingTable entries.

Page: 749  Line: 19  Section: 14.2.5.12  Ref: 3604

**Replace Original Text**
<Figure 190 on pg 749>

**With New Text**
<Note, the figure number shown below will be 190 when inserted in the actual
source text. It is otherwise in this document due to autonumbering.>
### Figure 1 MulticastForwardingTable Bit Layout

<table>
<thead>
<tr>
<th>AM 31:28 (p bits)</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>...</th>
<th>0</th>
<th>0</th>
<th>MIF for Multicast Forwarding Table Entry</th>
<th>AM 8:0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Port #</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>16</td>
<td>15</td>
<td>14</td>
<td>01</td>
</tr>
<tr>
<td>PortMask 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>PortMask 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>PortMask 31</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>MulticastForwardingTable Blocks (shaded)</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**MLID for Multicast Forwarding Table Entry**
- 0xC000: 0
- 0xC001: 0
- 0xC01E: 0
- 0xC01F: 0
- 0xC020: 1
- 0xC021: 1

**PortMask 0**: 31
**PortMask 1**: 1
**PortMask 31**: 0

**Figure 1 MulticastForwardingTable Bit Layout**
CHAPTER 15: SUBNET ADMINISTRATION ERRATA

Page: 793 Line: 4 Section: 15.2.5.12 Ref: 3625
Replace Original Text <Table 167>
InformInfo | 432 | 192
With New Text InformInfo | 288 | 192

Page: 801 Line: 20 Section: 15.2.5.16 Ref: 3710
Replace Original Text For example, consider a SubnAdmGetTable(PathRecord) query by template with:
With New Text For example, consider a SubnAdmGetTable(PathRecord) query with:

Page: 807 Line: 18 Section: 15.2.5.17 Ref: 3622
Replace Original Text Reserved 24 393 Reserved
With New Text Reserved 23 393 Reserved

Page: 822 Line: 38 Section: 15.4.6 Ref: 3727
Replace Original Text <Compliance marker text. Changed here in the source files, but visible only in Chapter 20, e.g., p. 994, line 12, C15-0.1.30.>
SubnAdmGet Query by Template
With New Text SubnAdmGet Query
CHAPTER 16: GENERAL SERVICES ERRATA

Page: 837 Line: 37 Section: 16.1.3.3 Ref: 3616

Replace Original Text <Sample Select Value column:>
0x0007-0x3FFF

With New Text 0x0006-0x3FFF

Page: 848 Line: 19 Section: 16.1.4.1 Ref: 3690

Replace Original Text Total number of packets received on the port that were discarded because they could not be forwarded by the switch relay due to VL mapping errors.

With New Text Packet discards due to VL mapping behavior are not considered errors, so the behavior of this counter is implementation-dependent. However, it is recommended that this counter be used to count the total number of packets received on the port that were discarded because they could not be forwarded by the switch relay due to VL mapping behavior <ref to 7.6.6 VL Mapping Within a Subnet>.

Page: 860 Line: 4 Section: 16.2.1 Ref: 3658

Replace Original Text <Figure 197>

Infiniband Subnet

With New Text InfiniBand Subnet

Page: 901 Line: 17 Section: 16.7.3.1 Ref: 3643

Replace Original Text <Row for Bit 11 of Table>
11 | IsRawDatagramCapable | The CA is capable of receiving Raw Datagramtraffic over the port to which the Get(ClassPortInfo) as sent.

With New Text 11 | Undefined | Undefined; any use of this field is vendor-dependent.
no errata
CHAPTER 18: SWITCHES ERRATA

Page: 926 Line: 8 Section: 18.1.1 Ref: 3760

Replace Original Text
Port 0 is not required to implement IB link level flow control.

With New Text
Port 0 is not required to implement IB link level flow control.

Port 0 is either in enhanced or base mode, an enhanced port can't be reduced to a base port dynamically.

Page: 926 Line: 12 Section: 18.1.1 Ref: 3761

Replace Original Text
Port 0 is not required to implement the IB physical layer electrical, optical, or mechanical requirements.

With New Text
Port 0 is not required to implement the IB physical layer electrical, optical, or mechanical requirements.

Enhanced port 0 link speed will be set to the nearest InfiniBand equivalent, e.g. A PCI link may have the Linkwidth set to 1x.

Enhanced port 0 PortPhysicalState shall support Linkup only.

Page: 928 Line: 33 Section: 18 Ref: 3763

Replace Original Text
C18-6.1.1 With the exception of packets arriving on the management virtual lane, switches shall be capable of forwarding packets of size from the minimum valid packet (including Raw packets) up to the supported MTU plus 126 bytes.

With New Text
C18-6.1.1 With the exception of packets arriving on the management virtual lane, switches shall be capable of forwarding packets of size from the minimum valid packet (Raw packets may be excluded) up to the supported MTU plus 126 bytes.

Page: 937 Line: 30 Section: 18.2.4.3.2 Ref: 3620

Add New Text
Note: Switches that implement a Random Forwarding Table shall discard all unicast packets for which the port number in the forwarding table entry corresponding to the packet's DLID is set to a port identifier that does not exist.
CHAPTER 19: ROUTERS ERRATA

No errata
CHAPTER 20:      VOLUME 1 COMPLIANCE SUMMARY ERRATA

Page: 994 Line: 12 Section: 20.8 Ref: 3709, 3727
Replace Original Text  C15-0.1.30: SubnAdmGet Query by Template
With New Text          C15-0.1.30: SubnAdmGet Query
ANNEX A1: I/O INFRASTRUCTURE ERRATA

Replace Original Text
Using information from the DevMgtGet(IOControllerProfile), ...
With New Text
Using controller and protocol information from the DevMgtGet(), ...

Page: 1005 Line: 27 Section: Annex A1.2.4.1.1 Ref: 3745
Replace Original Text
In order to determine the compatibility strings for an IOC, a host sends a DevMgtGet(IOControllerProfile) to the managed I/O unit containing the IOC. The IOControllerProfile attribute contains a set of components identifying the IOC and its vendor (VendorID, IocDeviceID, Device Version, Subsystem VendorID, SubsystemID), and a set of components identifying the protocol supported by the IOC (I/O Class, I/O Subclass, Protocol, and Protocol Version).
These components of the IOControllerProfile attribute are in binary format. See the Identifiers Annex for additional details regarding the IOControllerProfile component values corresponding to various protocols.
The binary value of each component of the IOControllerProfile attribute is converted into its ASCII representation with a leading designator. For example, if the IOControllerProfile Device Version component was 0xabcd, which is a 16-bit quantity, then the ASCII string into which it is converted is 'vabcd', which is a 40-bit quantity. Table 264 specifies how the components from the IOControllerProfile attribute are converted into an ASCII string.

With New Text
In order to determine the compatibility strings for an IOC, a host sends DevMgtGet()s to the managed I/O unit containing the IOC requesting IOControllerProfile and other pertinent attributes. The attributes contains a set of components identifying the IOC and its vendor (VendorID, IocDeviceID, Device Version, Subsystem VendorID, SubsystemID), and a set of components identifying the protocol supported by the IOC (I/O Class, I/O Subclass, Protocol, and Protocol Version). These components are in binary format. See the Application Specific Identifiers Annex for additional details regarding the component values corresponding to various protocols.
The binary value of each component is converted into its ASCII representation with a leading designator. For example, if the IOControllerProfile Device Version component was 0xabcd, which is a 16-bit quantity, then the ASCII string into which it is converted is 'vabcd', which is a 40-bit quantity. Table 264 specifies how the components from the DevMgt attributes are converted into an ASCII string.

Page: 1006 Line: 5 Section: Annex A1.2.4.1.1 Ref: 3746
Replace Original Text
IOControllerProfile Component Name
With New Text
Component Name
Replace Original Text

CommMgt

With New Text

ComMgt

Replace Original Text

The solution is to “query by template” and use the Component Mask to indicate that the partition is a wildcard. Since the SA only returns information associated with partitions that are valid for the host (i.e., only information about nodes with which the host can communicate), the result is that the SA responds with information for all of the host’s partitions and each record returned indicates the partition that the host needs to use.

With New Text

The solution is to query the SA using the Component Mask to indicate that the partition is a wildcard. Since the SA only returns information associated with partitions that are valid for the host (i.e., only information about nodes with which the host can communicate), the result is that the SA responds with information for all of the host’s partitions and each record returned indicates the partition that the host needs to use. See Chapter 15 section titled “Administration of Query Subsystem” for details on “querying by Component Mask”. The following are examples of how the host can apply this concept.

Replace Original Text

• IOControllerProfile components:
  :
  :

• ServiceEntries components:

With New Text

{Delete “• IOControllerProfile components:” on p1017 line 41}
{Delete “• ServiceEntries components:” on p1018 line 6}
ANNEX A2: CONSOLE SERVICE PROTOCOL ERRATA

Replace Original Text
• The IOC’s Class, Subclass, Protocol, and Version values for its DevMgt IOControllerProfile attribute are as follows:

With New Text
• The IOC’s Class, Subclass, Protocol, and Version values for its DevMgt attributes are as follows:
ANNEX A3: APPLICATION SPECIFIC IDENTIFIERS ERRATA

Page: 1052 Line: 18 Section: Annex A3.1 Ref: 3741

Replace Original Text  A3.3 I/O Controller Identification - Specifies policies for creating and interpreting vendor and protocol specific values in Device Management class IOControllerProfile attributes.

With New Text  A3.3 I/O Controller Identification - Specifies policies for creating and interpreting vendor and protocol specific values in Device Management class attributes.

Page: 1068 Line: 4 Section: Annex A3.3 Ref: 3750

Replace Original Text  The DevMgt class IOControllerProfile attribute contains components that identify an I/O controller. These components are: VendorID, DeviceID, Device Version, Subsystem VendorID, SubsystemID, IO Class, IO Subclass, Protocol, Protocol Version, ID String. In addition, each I/O controller has at least one associated ServiceEntries attribute.


Page: 1068 Line: 33 Section: Annex A3.3.2 Ref: 3743

Replace Original Text  Generic information fields refer to the IO Class, IO Subclass, Protocol, and Protocol Version components of the IoControllerProfile. An I/O Controller (IOC) uses these fields to indicate that it supports a standard I/O protocol. A host uses these fields to match the IOC with an I/O driver that performs that I/O protocol (see I/O Annex for driver matching).

CA3-6: A managed I/O unit shall implement a Device Management Agent as per Section 16.3.
CA3-7: An IOC shall specify IoControllerProfile: Class, Subclass, and Protocol values in accordance with A3.3 I/O Controller Identification.

With New Text  Generic information fields refer to the IO Class, IO Subclass, Protocol, and Protocol Version components in DevMgt attributes. An I/O Controller (IOC) uses these fields to indicate that it supports a standard I/O protocol. A host uses these fields to match the IOC with an I/O driver that performs that I/O protocol (see I/O Annex for driver matching).

CA3-6: A managed I/O unit shall implement a Device Management Agent as per Section 16.3 (or a supplemental Annex that supercedes Section 16.3).
CA3-7: An IOC shall specify: Class, Subclass, and Protocol values in accordance with A3.3 I/O Controller Identification.
A3.4.2 SERVICEENTRIES ATTRIBUTE

Each IOC provides a list of protocol identifiers in one or more ServiceEntries attributes. The number of ServiceEntries attributes for a particular IOC is indicated in the IOControllerProfile. Each ServiceEntries attribute can indicate up to 4 protocol identifiers. Typically, an I/O driver uses this information for establishing connections with the IOC.

For each protocol the IOC supports, the IOC provides a service entry. Each entry contains a ServiceName and its associated Service ID. Service names need to be unique within the scope of the IOC (i.e., the IOC cannot list the same service name more than once in any of its ServiceEntries attributes). For proprietary drivers, the IOC vendor defines a service name and for standard I/O protocols, the organization defining the I/O protocol defines the service name.

A3.4.2 IOC SERVICE RECORDS

Each IOC provides a list of protocol identifiers in one or more service entry attributes. The number of attribute records for a particular IOC is indicated in the IOControllerProfile. Typically, an I/O driver uses this information for establishing connections with the IOC.

For each protocol the IOC supports, the IOC provides a service entry. Each entry contains a ServiceName and its associated Service ID. Service names need to be unique within the scope of the IOC (i.e., the IOC cannot list the same service name more than once in any of its service entries). For proprietary drivers, the IOC vendor defines a service name and for standard I/O protocols, the organization defining the I/O protocol defines the service name.
Replace Original Text

<In the event column of table 303 Data Sink Mode Transition Events, change S=1 to S=0>
Receive ModeChange message with S=1 and Mode=COMB_MODE

With New Text

Receive ModeChange message with S=0 and Mode=COMB_MODE
Annex A5: Booting Annex Errata

Page: 1147 Line: 20 Section: Annex A5.1.2 Ref: 3659
Replace Original Text
Infiniband
With New Text
InfiniBand

Page: 1150 Line: 28 Section: Annex A5.1.3 Ref: 3660
Replace Original Text
Infiniband
With New Text
InfiniBand
ANNEX A6: BOO  T INFORMATION SERVICE ERRATA

No errata