

IP over InfiniBand
Internet Draft
[draft-ietf-ipoib-ibif-mib-09.txt](http://www.ietf.org/drafts/ietf-ipoib-ibif-mib-09.txt)
Expires: December 2006

Hal Rosenstock
HNR Consulting
June 2006

Definitions of Managed Objects for InfiniBand Interface Types

<[draft-ietf-ipoib-ibif-mib-09.txt](http://www.ietf.org/drafts/ietf-ipoib-ibif-mib-09.txt)>

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at

<http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at

<http://www.ietf.org/shadow.html>.

Abstract

InfiniBand Architecture (IBA) specifies a high speed, channel based, switched fabric architecture that delivers scalable performance in data centers.

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in Internet community. In particular, it defines objects for managing IBA defined InfiniBand interfaces.

Copyright

Copyright (C) The Internet Society (2006).

Table of Contents

| | | |
|-----------------------|--|--------------------|
| 1. | Introduction..... | 3 |
| 2. | The Internet-Standard Management Framework..... | 3 |
| 3. | Introduction to InfiniBand..... | 3 |
| 3.1 | Relation to MIB-2..... | 4 |
| 3.2 | Relation to the Interfaces MIB..... | 5 |
| 3.2.1 | Layering Model..... | 5 |
| 3.2.2 | Virtual Circuits..... | 5 |
| 3.2.3 | IfTestTable..... | 5 |
| 3.2.4 | IfRcvAddressTable..... | 5 |
| 3.2.5 | IfPhysAddress..... | 6 |
| 3.2.6 | ifType..... | 6 |
| 3.2.7 | Specific Interface MIB Objects..... | 6 |
| 3.3 | Structure of the MIB..... | 10 |
| 3.4 | Mapping of IBA managed attribute fields..... | 10 |
| 3.5 | Case Diagram..... | 13 |
| 4. | InfiniBand Interface MIB Definitions..... | 15 |
| 5. | Security Considerations..... | 26 |
| 6. | IANA Considerations..... | 26 |
| 7. | Revision History..... | 27 |
| 7.1 | Changes from <draft-ietf-ipoib-ibif-mib-08.txt>..... | 27 |
| 8. | References..... | 27 |
| 8.1 | Normative References..... | 27 |
| 8.2 | Informative References..... | 28 |
| 9. | Acknowledgements..... | 28 |
| 10. | Author's Address..... | 28 |
| 11. | Intellectual Property Notice..... | 28 |
| 12. | Full Copyright Statement..... | 29 |

1. Introduction

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it defines objects for managing IBA defined InfiniBand interfaces.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [\[RFC2119\]](#).

For an introduction to the concepts of InfiniBand, see [\[IBTAARCH\]](#).

2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to [section 7 of RFC 3410](#) [\[RFC3410\]](#).

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIV2, which is described in STD 58, [RFC 2578](#) [\[RFC2578\]](#), STD 58, [RFC 2579](#) [\[RFC2579\]](#) and STD 58, [RFC 2580](#) [\[RFC2580\]](#).

3. Introduction to InfiniBand

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it defines objects for managing InfiniBand interfaces. The managed objects provide extensions to the objects defined in the Interface Group MIB [\[RFC2863\]](#). This memo provides managed objects based on Sections [14](#) - [16](#) of the InfiniBand Architecture specification [\[IBTAARCH\]](#). The intent of this memo is to define value added manageability of IBA devices by defining SNMP managed objects and at the same time avoid making SNMP management a requirement for device operation.

IBA, unlike most physical layer network protocols, specifies a non-SNMP external management protocol that controls a wide range of functions necessary for IBA devices to operate. The functions

performed by the InfiniBand managers and agents, including the IBA Subnet Manager(SM) and Subnet Management Agents(SMA) in the managed devices include, but are not limited to, device configuration, fault detection, performance monitoring, and device configuration persistence.

Even given this native management capability, SNMP manageability of any networking components in today's networks is highly desirable or even a requirement especially in the area of fault and performance management. Obviously, having two management paradigms simultaneously controlling a device or portions thereof would require a high degree of cooperation between paradigms to avoid conflicts. Therefore, to give maximum visibility but avoid these conflicts, the managed objects defined in this memo offer read-only access to the attributes defined by InfiniBand native management wherever a conflict with IBA defined management exists.

The counters defined in this memo are based on IBA defined counters. But in order to support the semantic behavior defined by SNMP SMI [[RFC2578](#)] conversions are required in several aspects of numerous InfiniBand specified counters. These changes deal primarily with counter width differences, latch-on-wrap (also known as "saturate at all 1's"), clear-on-read (reset to zero), and writeability of the InfiniBand Architecture native counters as defined in [[IBTAARCH](#)]. Mechanisms for doing the required conversion are not defined by this memo.

Instances of the managed object types defined herein represent attributes of a port on an InfiniBand communications medium device (IBA specification prefers the term "port" to "interface"). IBA defined ports can be paralleled/striped over multiple physical links called "physical lanes." Aggregations of 1x, 4x, 8x, and 12x physical lanes are defined by the IBA. The IBA treats these aggregations as one port. The objects defined herein do the same.

At present, InfiniBand media are identified by the following value of the ifType object in the Interfaces Group MIB [[RFC2863](#)] as defined by the IANA:

infiniband(199) -- assigned by IANA

3.1 Relation to MIB-2

This section applies only when this MIB is used in conjunction with the "old" ([RFC 1213](#)) interface group.

The relationship between an InfiniBand interface and an interface in the context of MIB-2 is one-to-one. As such, the value of an

ifIndex object instance can be directly used to identify corresponding instances of the objects defined herein.

For agents which implement the (now deprecated) ifSpecific object, an instance of that object that is associated with an InfiniBand interface has the OBJECT IDENTIFIER value:

```
infiniband OBJECT IDENTIFIER
 ::= { transmission 199 } --To be assigned by IANA
```

3.2 Relation to the Interfaces MIB

The Interfaces MIB [[RFC2863](#)] requires that any MIB which is an adjunct of the Interface MIB clarify specific areas within the Interfaces MIB. These areas were intentionally left vague in the Interfaces MIB to avoid over-constraining the MIB, thereby precluding management of certain media-types.

[Section 4.0 of \[RFC2863\]](#) enumerates several areas that a media-specific MIB must clarify. Each of these areas is addressed in a following subsection. The implementer is referred to [[RFC2863](#)] in order to understand the general intent of these areas.

3.2.1 Layering Model

This MIB does not provide for layering. There are no sublayers defined by this memo. Higher Layer protocol stacking on the interfaces is not defined by this memo.

3.2.2 Virtual Circuits

This medium does not support virtual circuits and this area is not applicable to this MIB.

3.2.3 IfTestTable

There are no media tests which are instrumented with this MIB.

3.2.4 IfRcvAddressTable

For IBA Routers, Channel Adapters (CAs), and Enhanced Switch Port 0s: Routers, CAs, and Enhanced Switch Port 0s which have been programmed by the IBA Subnet Manager to have more than one address use this table to display the set of additional addresses beyond the "base" Local Identifier (LID) for which the interface will receive packets. If desired to avoid conflicts with the IBA Subnet Manager configuration of this table, this table MAY be implemented as read-only. See 5.2.5 for encoding of the Physical Address (ifPhysAddress).

For IBA Switches:

Additionally for switches, since the ifRcvAddressTable table is explicitly not intended to provide a bridge address filtering mechanism, there is no need to expose the multicast and unicast forwarding table addresses here.

3.2.5 IfPhysAddress

This object contains the 16 bit "base" Local Identifier (LID) currently in effect on this port. Whether the address was assigned by the IBA Subnet Manager or through some other mechanism does not matter. In the event that no LID is currently in effect an Octet String of zero length is returned. For switches the "base" LID of the (conceptual) management port (port 0) shall be used. The address is stored in binary in this object. The address is stored in IBA defined bit order, that is, the high-order bit of the Local Identifier byte 0 is positioned as the high-order bit of the first byte of the ifPhysAddress.

3.2.6 ifType

This MIB applies to interfaces which has the following ifType value:

infiniband(199) -- Assigned by IANA

Information regarding the number of physical lanes that comprise the port may be included in the ifDescr object (i.e., 1x, 4x, 8x, or 12x).

3.2.7 Specific Interface MIB Objects

The following table provides specific implementation guidelines for applying the interface group objects to InfiniBand media.

| Object | Guidelines |
|---------|--|
| ----- | ----- |
| ifIndex | Each InfiniBand interface is represented by an ifEntry. |
| ifDescr | Refer to [RFC2863] . |
| ifType | Refer to section 3.2.6 . |
| ifMtu | The size of the largest packet that is capable of being either sent/received over this interface specified in octets. Actual Maximum Transmission Unit (MTU) in use is |

determined by examining device MTU capability or NeighborMTU value (if valid) in the portInfo table.

ifSpeed

The current effective operational speed of the interface in bits per second. The speed reported compensates for the 8B10B encoding used on the links. For Single Data Rate (SDR) InfiniBand interfaces, this will be equal to 2,000,000,000 (2 billion), 8,000,000,000 (8 billion), 16, 000,000,000 (16 billion), or 24,000,000,000 (24 billion) for the 1x, 4x, 8x, and 12x ports, respectively. Since 4x, 8x, and 12x values are greater than the maximum reportable by this object, this object MUST report its maximum value (4,294,967,295) and ifHighSpeed MUST be used to report the interface's speed. Also, the link speed needs to be factored in as well as there are Single Data Rate (SDR or 2.5 Gbps), Dual Data Rate (DDR or 5.0 Gbps), and Quad Data Rate (QDR or 10.0 Gbps) links. Actual Speed in use is determined by examining device linkWidthSupported or linkWidthActive value (if valid) and linkSpeedSupported or linkSpeedActive in the portInfo table.

ifPhysAddress

Refer to [section 3.2.5](#).

ifAdminStatus

Default value is "up". If write access is implemented the default value for this attribute SHOULD be "up" to avoid requiring the use of a SNMP manager to enable the interface. Support for "testing" is not required. Providing read-write access to this object can potentially cause conflicts with the IBA Subnet Manager which also controls Port State through the SMA.

ifOperStatus

The operational state of the interface. Support for "testing" is not required.

| | |
|-------------------|--|
| ifLastChange | Refer to [RFC2863]. |
| ifInOctets | The number of octets in valid packets received on this interface, including the START and END delimiters and the VCRC. This count MUST also include the number of octets in valid Link packets received on this interface. |
| ifInUcastPkts | Refer to [RFC2863]. Note that this does not include link packets, since link control packets are consumed by the interface layer and are not passed to any higher layer protocol. |
| ifInDiscards | Total number of packets received that were discarded. |
| ifInErrors | Refer to [RFC2863]. |
| ifInUnknownProtos | Refer to [RFC2863]. |
| ifOutOctets | The number of octets transmitted in valid packets on this interface, including the START and END delimiters and the VCRC. This count MUST also include the number of octets in valid link packets sent on this interface. |
| ifOutUcastPkts | Refer to [RFC2863]. Note that this does not include link packets, since link packets are generated by the interface layer, and are not passed down from any higher layer protocol. |
| ifOutDiscards | Refer to [RFC2863]. |
| ifOutErrors | Refer to [RFC2863]. |
| ifName | Locally-significant textual name for the interface (e.g., lan0). |
| ifInMulticastPkts | Refer to [RFC2863]. Note that this does not include link packets, since link control packets are consumed by the interface layer and are not passed to any higher layer protocol. Never |

| | |
|--|--|
| | change because these events cannot occur on this medium. |
| ifOutMulticastPkts | Refer to [RFC2863]. Note that this does not include Link packets, since Link packets are created by the interface layer and are not passed in to any higher layer protocol. Never change because these events cannot occur on this medium. |
| ifInBroadcastPkts | Never change because these events cannot occur on this medium. Refer to [RFC2863]. |
| ifOutBroadcastPkts | Never change because these events cannot occur on this medium. Refer to [RFC2863]. |
| ifHCInBroadcastPkts | Never change because these events cannot occur on this medium. Refer to [RFC2863]. |
| ifHCOutBroadcastPkts | Never change because these events cannot occur on this medium. Refer to [RFC2863]. |
| ifHCInOctets ifHCOutOctets | 64-bit versions of counters. Required for interfaces that are capable of operating at 20Mbit/sec or faster, even if the interface is currently operating at less than 20Mbit/sec. |
| ifHCInUcastPkts ifHCInMulticastPkts ifHCOutUcastPkts ifHCOutMulticastPkts | 64-bit versions of packet counters. Required for interfaces that are capable of operating at 640Mbit/sec or faster, even if the interface is currently operating at less than 640Mbit/sec. Never change because these events cannot occur on this medium. |
| ifLinkUpDownTrapEnable | Refer to [RFC2863]. Default is "enabled". |
| ifHighSpeed | The current effective operational speed of the interface in millions of bits per second. For SDR InfiniBand interfaces, this will be |

| | |
|----------------------------|---|
| | equal to 2,000, 8,000, 16,000, or 24,000. For DDR interfaces, this will be equal to 4,000, 16,000, 32,000, or 48,000. For QDR interfaces, this will be equal to 8,000, 32,000, 64,000, or 96,000. |
| ifPromiscuousMode | Usually "false" but one could create a promiscuous device conceivably. Refer to [RFC2863]. |
| ifConnectorPresent | This will be "true". |
| ifAlias | Refer to [RFC2863]. |
| ifCounterDiscontinuityTime | Refer to [RFC2863]. Note that a discontinuity in the Interface MIB counters may also indicate a discontinuity in some or all of the counters in this MIB that are associated with that interface. |
| ifStackHigherLayer | Refer to section 3.2.1 . |
| ifStackLowerLayer | |
| ifStackStatus | |
| ifRcvAddressAddress | Refer to section 3.2.4 . |
| ifRcvAddressStatus | |
| ifRcvAddressType | |

[3.3](#) Structure of the MIB

This MIB is structured as three groups:

- o Detail error statistics on a per interface basis (ibIfPortStats). This table provides visibility into error counters defined by the IBA.
- o Detail traffic statistics on a per interface and virtual lane pair basis (ibIfVLTraffic). This table provides visibility into data flow counters kept on a per virtual lane (VL) basis defined by the IBA.
- o Information that associates the IP over IB link layer addresses with InfiniBand interfaces (ports).

[3.4](#) Mapping of IBA managed attribute fields

This section lists the correlation and conversion between the SNMP

managed objects defined by this memo and the IBA defined attributes they are based on. Note that the relationship expressed below does not address issues relating to counter widths, latching and reset ability differences between the SNMP SMI and IBA object definitions. Note that since multicast counters have not been defined by the IBA specification, IF-MIB multicast counters should never change because these events cannot occur with the InfiniBand medium. Also, based on the IBA counter definitions, multicast (packets and bytes) will be part of the IF-MIB unicast counters.

The following table is organized by the IBA Attributes that fields belong in. In the case where fields from multiple attributes are required, the supplemental attributes are then fully qualified in the formula below. A second section covering Port Sample counters is also included.

Some Abbreviations used:

POH = Packet Overhead = 4 = 1 START + 1 END + 2 VCRC Octets;

SLP = Size Link flow control Packet = 8 = 1 START + 1 END + 6 flow control octets;

MHB = Max Header Bytes = 126 octets; includes all packet fields except framing characters. See [[IBTAARCH](#)] sec 7.7.8.

| IBA Attribute and Field | Corresponding SNMP Object |
|-------------------------|---------------------------|
|-------------------------|---------------------------|

PortCounters see [[IBTAARCH](#)] Table:225

| | |
|--|-------------------|
| (.PortRcvData * 4) + (.PortRcvPkts * POH) + (PortFlowCtlCounters. PortRcvFlowPkts * SLP) = | IF-MIB:ifInOctets |
|--|-------------------|

| | |
|--|---------------------|
| .PortRcvConstraintErrors + .VL15Dropped = | IF-MIB:ifInDiscards |
|--|---------------------|

| | |
|--|-------------------|
| .PortRcvRemotePhysicalErrors + PortCounters.PortRcvErrors = | IF-MIB:ifInErrors |
|--|-------------------|

| | |
|----------------|----------------------|
| .PortRcvPkts = | IF-MIB:ifInUcastPkts |
|----------------|----------------------|

| | |
|--|--------------------|
| .PortXmitData * 4 + (PortFlowCtlCounters.PortXmitFlowPkts * SLP) + (.PortXmitPkts * POH) = | IF-MIB:ifOutOctets |
|--|--------------------|

| | |
|--|----------------------|
| .PortXmitDiscards + .PortXmitConstraintErrors = | IF-MIB:ifOutDiscards |
|--|----------------------|


```

.PortXmitPkts +
.PortXmitDiscards +
.PortXmitConstraintErrors =          IF-MIB:ifOutUcastPkts

.SymbolErrorCounter =                ibIfPortSymbolErrs

.LinkErrorRecoveryCounter =          ibIfPortLinkErrRecovery

.PortRcvRemotePhysicalErrors =       ibIfPortStatRcvRemPhyErrs

.LinkDownedCounter =                 ibIfPortLinkDowned

.LocalLinkIntegrityErrors =           ibIfPortStatLinkIntegrityErrs

.ExcessiveBufferOverrunErrors =       ibIfPortStatExcBufOverrunErrs

.VL15Dropped =                       ibIfPortStatVL15Dropped

```

Note: If PortCountersExtended see [[IBTAARCH](#)] Table:238 supported use these rather than PortCounters for the above where applicable and additionally:

```

.PortUnicastRcvPkts =                IF-MIB:ifHCInUcastPkts

.PortUnicastXmitPkts +
.PortXmitDiscards +
.PortXmitConstraintErrors =          IF-MIB:ifHCOutUcastPkts

.PortMulticastRcvPkts =              IF-MIB:ifHCInMulticastPkts

.PortMulticastXmitPkts =             IF-MIB:ifHCOutMulticastPkts

```

PortRcvErrorDetails see [[IBTAARCH](#)] Table:228

```

.PortLocalPhysicalErrors =            ibIfPortStatLocalPhyErrs
.PortMalformedPacketErrors =          ibIfPortStatMalPktErrs

```

PortXmitDiscardDetails see [[IBTAARCH](#)] table:229

```

.PortInactiveDiscards =               ibIfPortStatInactDiscards
.PortNeighborMTUDiscards =            ibIfPortStatNeighMTUDiscards
.PortSwLifetimeLimitDiscards =        ibIfPortStatSwLifetimeDiscards
.PortSwHOQLimitDiscards =             ibIfPortStatHOQLifetimeDiscards

```

PortInfo see [[IBTAARCH](#)] Table:145


```

-----
    .NeighborMTU =                                IF-MIB:ifMtu

    .linkWidthActive *
    .linkSpeedActive =                            IF-MIB:ifSpeed

```

| IBA Port Sample Counter | Corresponding SNMP Object |
|-------------------------|---------------------------|
|-------------------------|---------------------------|

CounterSelect Values see [[IBTAARCH](#)] table:189

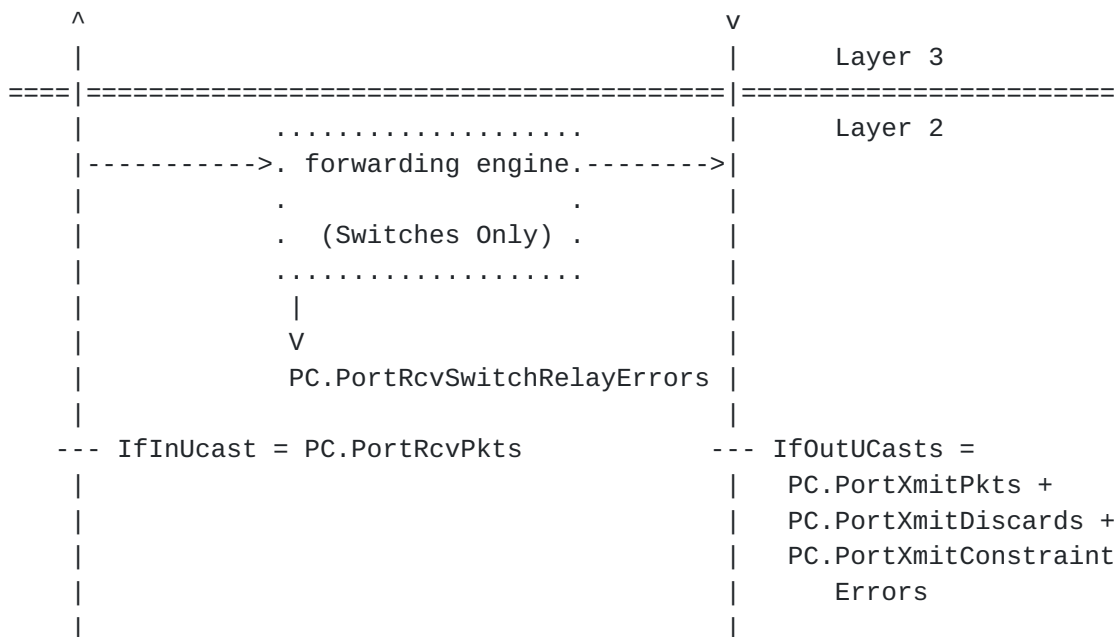
```

-----
PortRcvDataVL * 4 +
  ( PortXmitPktsVL * POH) =                ibIfVLIInOctets
PortRcvPktsVL =                            ibIfVLIInPkts
PortXmitDataVL * 4 +
  ( PortXmitPktsVL * POH) =                ibIfVLOutOctets
PortXmitPktsVL =                            ibIfVLOutPkts

```

3.5 Case Diagram

The following "Case Diagram" visually depicts the mapping described in [section 3.4](#). Again it should be noted that since no multicast counters are defined by the IBA specification they are not covered here.



| | |
|---|--|
| <pre> --> IfInDiscards = PC.PortRcvConstraintErrs + PC.VL15Dropped --- IfInOctets = PC.PortRcvData * 4 + PC.PortRcvPkts * POH + PFCC.PortRcvFloPkts * SLP --> IfInUnknownProtos = 0 --> IfInErrors = PC.PortRcvRemotePhysicalErrors + PC.PortRcvErrors ===== ^ </pre> | <pre> --> IfOutDiscards = PC.PortXmitDiscards + PC.PortXmitConstraint Errors --> IfOutErrors = 0 --> IfOutOctets = (PC.PortXmitData * 4)+ (PFCC.PortXmitFlowPkts * SLP) + (PC.PortXmitPkts * POH) ===== V </pre> |
| | Layer 1 |

Abbreviations:

PRED = PortRcvErrorDetails Table

PC = Port Counters Table

PFCC = PortFlowCtlCounters Table

POH = Packet Overhead = 4bytes

SLP = Size Link FlowCtl Packt = 8

MB = MaxHeader bytes = 126

```

PC.PortXmitDiscards = PortInactiveDiscards +
                      PortNeighborMTUDiscards +
                      PortSwLifeTimeLimitDiscard +
                      PortSwHOQLifetimeLimitDiscards

```

```

PC.PortRcvErrors = PRED.PortLocalPhysicalErrors +
                   PRED.PortMalformedPacketErrors +
                   PRED.PortBufferOverrunErrors

```

```

PC.PortRcvSwitchRelayErrors = PRED.PortDLIDMapErrs +
                              PRED.PortVLMapErrs +
                              PRED.PortLoopingErr

```


4. InfiniBand Interface MIB Definitions

```
IB-IF-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-COMPLIANCE, OBJECT-GROUP          FROM SNMPv2-CONF
    MODULE-IDENTITY, OBJECT-TYPE,
    Unsigned32, Counter32, Counter64          FROM SNMPv2-SMI
    InterfaceIndex                             FROM IF-MIB
    IbVirtualLane, infinibandMIB,
    IbIpoibClientIdentifier                    FROM IB-TC-MIB;
```

```
ibIfMIB MODULE-IDENTITY
```

```
    LAST-UPDATED "200606270000Z" -- 27 June 2006 00:00:00 GMT
```

```
    ORGANIZATION "IETF IP over IB (IPOIB) Working Group"
```

```
    CONTACT-INFO
```

```
        "Hal Rosenstock
        Postal: HNR Consulting
                200 Old Harvard Road
                Boxboro MA 01719-1834
                United States
        Email: hnrose@earthlink.net
```

```
        Email comments to the IPOIB WG Mailing List at
        ipoverib@ietf.org."
```

```
    DESCRIPTION
```

```
        "Copyright (C) The Internet Society (2006). The initial
        version of this MIB module was published in RFC XXXX; for
        full legal notices see the RFC itself. Supplementary
        information may be available on
        http://www.ietf.org/copyrights/ianamib.html.
```

```
        This module contains managed object definitions for
        managing InfiniBand interfaces."
```

```
    REVISION
```

```
        "200606270000Z" -- 27 June 2006 00:00:00 EST
```

```
    DESCRIPTION
```

```
        "Initial version published as part of RFC XXXX."
```

```
 ::= { infinibandMIB 2 }
```

```
--
```

```
-- Top-Level Object Identifier Assignments
```

```
--
```

```
    ibIfObjects OBJECT IDENTIFIER ::= { ibIfMIB 1 }
    ibIfNotifications OBJECT IDENTIFIER ::= { ibIfMIB 2 }
    ibIfConformance OBJECT IDENTIFIER ::= { ibIfMIB 3 }
```



```

_ _*****
-- IpoIB Detailed per interface/port statistics
-- ibIfPortStat Table
_ _*****

```

ibIfPortStatTable OBJECT-TYPE

SYNTAX SEQUENCE OF IbIfPortStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table provides detail statistics for all InfiniBand interfaces attached to a particular system. There will be one row in this table for each InfiniBand interface in the system."

::= { ibIfObjects 1 }

ibIfPortStatEntry OBJECT-TYPE

SYNTAX IbIfPortStatEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Statistics for a particular interface to an InfiniBand medium."

INDEX { ibIfPortStatIfIndex }

::= { ibIfPortStatTable 1 }

IbIfPortStatEntry ::= SEQUENCE {

```

    ibIfPortStatIfIndex InterfaceIndex,
    ibIfPortSymbolErrs Counter32,
    ibIfPortLinkErrRecovery Counter32,
    ibIfPortLinkDowned Counter32,
    ibIfPortStatLocalPhyErrs Counter32,
    ibIfPortStatMalPktErrs Counter32,
    ibIfPortStatRcvRemPhyErrs Counter32,
    ibIfPortStatRcvConstrErrs Counter32,
    ibIfPortStatInactDiscards Counter32,
    ibIfPortStatNeighMTUDiscards Counter32,
    ibIfPortStatSwLifetimeDiscards Counter32,
    ibIfPortStatHOQLifetimeDiscards Counter32,
    ibIfPortStatLinkIntergrityErrs Counter32,
    ibIfPortStatExcBufOverrunErrs Counter32,
    ibIfPortStatVL15Dropped Counter32

```

}

ibIfPortStatIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

| | | |
|------------|-----------------------|-----------|
| Rosenstock | Expires December 2006 | [Page 16] |
|------------|-----------------------|-----------|

"The ifIndex value of the InfiniBand interface to which
these port statistics apply."
::= { ibIfPortStatEntry 1 }

ibIfPortSymbolErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of minor link errors detected on one or more
physical lanes."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section
16.1.3.5](#); Table 225 PortCounters::SymbolErrorCounter."

::= { ibIfPortStatEntry 2 }

ibIfPortLinkErrRecovery OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of times the Port Training state machine has
successfully completed the link error recovery process."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section
16.1.3.5](#); Table 225 PortCounters::LinkErrorRecoveryCounter."

::= { ibIfPortStatEntry 3 }

ibIfPortLinkDowned OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of times the Port Training state machine
has failed the link error recovery process and downed
the link."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section
16.1.3.5](#); Table 225 PortCounters::LinkDownedCounter."

::= { ibIfPortStatEntry 4 }

ibIfPortStatLocalPhyErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of packets received on the port that contain
local physical errors (ICRC, VCRC, FCCRC, and

all physical errors that cause entry into the BAD

Rosenstock

Expires December 2006

[Page 17]

PACKET or BAD PACKET DISCARD states of the packet receiver state machine)."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::PortRcvErrors."

::= { ibIfPortStatEntry 5 }

ibIfPortStatMalPktErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of packets received on the port that contain malformed packet errors
- data packets: LVer, length, VL
- link packets: operand, length, VL"

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::PortRcvErrors."

::= { ibIfPortStatEntry 6 }

ibIfPortStatRcvRemPhyErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of packets marked with the EBP delimiter received on the port."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::PortRcvRemotePhysicalErrors."

::= { ibIfPortStatEntry 7 }

ibIfPortStatRcvConstrErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of packets received on the port that are discarded for the following reasons:
- FilterRawInbound is true and packet is raw
- PartitionEnforcementInbound is true and packet fails partition key check or IP version check."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::PortRcvConstraintErrors."

::= { ibIfPortStatEntry 8 }

ibIfPortStatInactDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of outbound packets discarded by the port because it is not in the active state."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.4.2](#);
Table 229 PortXmitDiscardDetails::PortInactiveDiscards."

::= { ibIfPortStatEntry 9 }

ibIfPortStatNeighMTUDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of outbound packets discarded by the port because packet length exceeded the neighbor MTU."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.4.2](#);
Table 229 PortXmitDiscardDetails::PortNeighborMTUDiscards."

::= { ibIfPortStatEntry 10 }

ibIfPortStatSwLifetimeDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of outbound packets discarded by the port because the Switch Lifetime Limit was exceeded. Note this object is only incremented for switches."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.4.2](#);
Table 229
PortXmitDiscardDetails::PortSwLifetimeLimitDiscards."

::= { ibIfPortStatEntry 11 }

ibIfPortStatHOQLifetimeDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of outbound packets discarded by the port because the switch HOQ lifetime was exceeded. Note this object is only incremented for switches."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.4.2](#);

Table 229 PortXmitDiscardDetails::PortSwHOQLimitDiscards."

Rosenstock

Expires December 2006

[Page 19]

```
::= { ibIfPortStatEntry 12 }
```

ibIfPortStatLinkIntergrityErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times that the count of local physical errors exceeded the LocalPhyErrors threshold."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::LocalLinkIntegrityErrors.; Also see Table 145 PortInfo."

```
::= { ibIfPortStatEntry 13 }
```

ibIfPortStatExcBufOverrunErrs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times that OverrunErrors consecutive flow control update periods occurred with at least one overrun error in each period."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::ExcessiveBufferOverrunErrors.; Also see Table 145 PortInfo."

```
::= { ibIfPortStatEntry 14 }
```

ibIfPortStatVL15Dropped OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of incoming VL15 packets dropped due to resource limitations on the selected port (e.g., lack of buffers)."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.5](#); Table 225 PortCounters::VL15Dropped."

```
::= { ibIfPortStatEntry 15 }
```

```
_ _*****
-- IPoIB per port VL Traffic Objects
-- ibIfVLTraffic Table
_ _*****
```

ibIfVLTrafficTable OBJECT-TYPE

SYNTAX SEQUENCE OF IbIfVLTrafficEntry

Rosenstock

Expires December 2006

[Page 20]

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table provides traffic statistics for all virtual lanes that are configured on an InfiniBand interface. There will always be at least one data virtual lane in the range VL0-VL14 configured on an interface, as well as control channel VL15. Configuration of what VLS are available on a particular interface is done by IBA native management."

::= { ibIfObjects 2 }

ibIfVLTrafficEntry OBJECT-TYPE

SYNTAX IbIfVLTrafficEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Information about a particular virtual lane (VL)."

INDEX { ibIfVLTrafficIfIndex, ibIfVLIndex }

::= { ibIfVLTrafficTable 1 }

IbIfVLTrafficEntry ::= SEQUENCE {

ibIfVLTrafficIfIndex InterfaceIndex,

ibIfVLIndex IbVirtualLane,

ibIfVLOutOctets Counter64,

ibIfVLOutPkts Counter64,

ibIfVLInOctets Counter64,

ibIfVLInPkts Counter64

}

ibIfVLTrafficIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The ifIndex value of the InfiniBand interface to which these virtual lane (VL) traffic statistics apply."

::= { ibIfVLTrafficEntry 1 }

ibIfVLIndex OBJECT-TYPE

SYNTAX IbVirtualLane

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Identifies what Virtual Lane (VL) instance is being addressed."

::= { ibIfVLTrafficEntry 2 }

ibIfVLOutOctets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets transmitted in valid data packets on this interface, including the START and END delimiters and the VCRC for this VL."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.3](#);
Table 223 CounterSelect::PortXmitDataVL."

::= { ibIfVLTrafficEntry 3 }

ibIfVLOutPkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets successfully sent on this VL."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.3](#);
Table 223 CounterSelect::PortXmitPktVL."

::= { ibIfVLTrafficEntry 4 }

ibIfVLInOctets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets in valid data packets received on this interface, including the START and END delimiters and the VCRC for this VL."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.3](#);
Table 223 CounterSelect::PortRcvDataVL."

::= { ibIfVLTrafficEntry 5 }

ibIfVLInPkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets successfully received on this VL.
This does not include link packets, since link packets are generated by the interface layer, and are not passed from any higher layer protocol."

REFERENCE

"InfiniBand Architecture Release 1.2 Vol. 1. [Section 16.1.3.3](#);

Table 223 CounterSelect::PortRcvPktVL."

Rosenstock

Expires December 2006

[Page 22]

```
 ::= { ibIfVLTrafficEntry 6 }
```

```

_*****
-- IPoIB Link Layer Address Table
-- ibIpoibLinkLayerAddr Table
_*****

```

ibIpoibLinkLayerAddrTable OBJECT-TYPE

SYNTAX SEQUENCE OF IbIpoibLinkLayerAddrEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table contains information about all IPOIB link layer addresses associated with this InfiniBand interface (port)."

```
 ::= { ibIfObjects 3 }
```

ibIpoibLinkLayerAddrEntry OBJECT-TYPE

SYNTAX IbIpoibLinkLayerAddrEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Information about specific IPOIB link layer addresses associated with this InfiniBand interface (port)."

INDEX { ibIpoibLinkLayerIfIndex, ibIpoibLinkLayerIndex }

```
 ::= { ibIpoibLinkLayerAddrTable 1 }
```

IbIpoibLinkLayerAddrEntry ::= SEQUENCE {

ibIpoibLinkLayerIfIndex InterfaceIndex,

ibIpoibLinkLayerIndex Unsigned32,

ibIpoibLinkLayerAddr IbIpoibClientIdentifier

}

ibIpoibLinkLayerIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The ifIndex of the IPOIB InfiniBand interface (port)."

```
 ::= { ibIpoibLinkLayerAddrEntry 1 }
```

ibIpoibLinkLayerIndex OBJECT-TYPE

SYNTAX Unsigned32(1..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index of the IPOIB Link Layer address in the table."

```
 ::= { ibIpoibLinkLayerAddrEntry 2 }
```


ibIpoibLinkLayerAddr OBJECT-TYPE

SYNTAX IbIpoibClientIdentifier

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A Client Identifier for this interface (port). An implementation may associate multiple IPOIB interfaces on the same port. It is up to the implementation to ensure a unique Client Identifier when multiple IPOIB interfaces are defined over the same port and same GID. A unique, invariant interface-id value must be included in addition to the GID within the Client Identifier definition to achieve this. Note: a port may also be associated with multiple GIDs. Therefore, multiple IPOIB interfaces may exist on the same port while using a different GID from among the GIDs associated with the port. In either case, the IbIpoibLinkLayerAddr MUST be unique."

::= { ibIpoibLinkLayerAddrEntry 3 }

ibIfCompliances OBJECT IDENTIFIER ::= { ibIfConformance 1 }

ibIfMIBGroups OBJECT IDENTIFIER ::= { ibIfConformance 2 }

```

_ _*****
-- Compliance Statements
_ _*****

```

ibIfBasicCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION "The basic implementation requirements for managed network entities that have InfiniBand network interfaces."

MODULE -- this module

-- ibIfStatOptionalPortStatGroup, ibIfVLTrafficGroup, and
-- ibIpoibLinkAddrGroup are not required for basic
-- implementation.

MANDATORY-GROUPS { ibIfStatMandatoryPortStatGroup }

::= { ibIfCompliances 1 }

ibIfFullCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION "The full implementation requirements for managed network entities that have InfiniBand network interfaces."

MODULE -- this module

MANDATORY-GROUPS { ibIfStatMandatoryPortStatGroup,
ibIfStatOptionalPortStatGroup,
ibIfVLTrafficGroup,

ibIpoibLinkAddrGroup }

Rosenstock

Expires December 2006

[Page 24]

```
 ::= { ibIfCompliances 2 }

_*****
-- Units of Conformance
_*****

ibIfStatMandatoryPortStatGroup OBJECT-GROUP
    OBJECTS {
        ibIfPortSymbolErrs,
        ibIfPortLinkErrRecovery,
        ibIfPortLinkDowned,
        ibIfPortStatRcvRemPhyErrs,
        ibIfPortStatRcvConstrErrs,
        ibIfPortStatLinkIntergrityErrs,
        ibIfPortStatExcBufOverrunErrs,
        ibIfPortStatVL15Dropped
    }
    STATUS current
    DESCRIPTION
        "Detailed error statistics group for mandatory
        InfiniBand-based Port Statistics counters."
    ::= { ibIfMIBGroups 1 }

ibIfStatOptionalPortStatGroup OBJECT-GROUP
    OBJECTS {
        ibIfPortStatLocalPhyErrs,
        ibIfPortStatMalPktErrs,
        ibIfPortStatInactDiscards,
        ibIfPortStatNeighMTUDiscards,
        ibIfPortStatSwLifetimeDiscards,
        ibIfPortStatHOQLifetimeDiscards
    }
    STATUS current
    DESCRIPTION
        "Detailed error statistics group for optional
        InfiniBand-based Port Statistics counters."
    ::= { ibIfMIBGroups 2 }

ibIfVLTrafficGroup OBJECT-GROUP
    OBJECTS {
        ibIfVLOutOctets,
        ibIfVLOutPkts,
        ibIfVLInOctets,
        ibIfVLInPkts
    }
    STATUS current
    DESCRIPTION
        "Detailed per VL traffic statistics group."
```


::= { ibIfMIBGroups 3 }

```
ibIpoibLinkAddrGroup OBJECT-GROUP
    OBJECTS {
        ibIpoibLinkLayerAddr
    }
    STATUS current
    DESCRIPTION
        "Detailed per port IPOIB link layer address group."
    ::= { ibIfMIBGroups 4 }

END
```

5. Security Considerations

SNMPv1 by itself is not a secure environment. Even if the network itself is secure (for example by using IPSec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model [RFC 3414](#) [[RFC3414](#)] and the View-based Access Control Model [RFC 3415](#) [[RFC3415](#)] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

6. IANA Considerations

This memo does NOT define any new name spaces which must be maintained by IANA. This memo DOES consume an ifType assignment that has been made by IANA:

```
infiniband(199)  -- ifType assigned by IANA
```

as well as an experimental MIB registration space:

```
infinibandMIB ::= { experimental 117 }  -- Assigned by IANA
```

The infinibandMIB name space will be locally administered by the IPOIB Working group as new MIBs are created by the group.

This document also requests IANA to assign { infinibandMIB 2 } to the IB-IF-MIB specified in this document.

7. Revision History

This section should be removed when this document is published as an RFC.

7.1 Changes from <[draft-ietf-ipoib-ibif-mib-08.txt](#)>

Updated to InfiniBand Architecture Revision 1.2 from 1.1
Support for SDR, DDR, and QDR links
Also added PortCountersExtended

Resolved nits with <[draft-ietf-ipoib-ibif-mib-08.txt](#)>

Also updated some RFC numbers and references

Added revision history

8. References

8.1 Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, [RFC 2578](#), April 1999.
- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, [RFC 2579](#), April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Conformance Statements for SMIv2", STD 58, [RFC 2580](#), April 1999.
- [RFC2863] McCloghrie, K., and Kastenholz F., "The Interfaces Group MIB", [RFC 2863](#), June 2000.
- [RFC3414] U. Blumenthal, B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, [RFC 3414](#), December 2002.
- [RFC3415] B. Wijnen, R. Presuhn, K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management

Protocol (SNMP)", STD 62, [RFC 3415](#), December 2002.

[IBTAARCH] InfiniBand Architecture Specification Volume 1,
Release 1.2, October, 2004, Final Release.

8.2 Informative References

[RFC2665] Flick J., and Johnson J., "Definitions of Managed Objects for the Ethernet-like Interface Types", [RFC 2665](#), August 1999.

[RFC3410] Case, J., Mundy, R., Partain, D. and B. Stewart,
"Introduction and Applicability Statements for
Internet-Standard Management Framework", [RFC 3410](#),
December 2002

9. Acknowledgements

This MIB module was updated based on the original work done by Sean Harnedy and Bill Anderson.

10. Author's Address

Hal Rosenstock
HNR Consulting
200 Old Harvard Road
Boxboro, MA 01719-1834
USA
Email: hnrose@earthlink.net

11. Intellectual Property Notice

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

12. Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

