To: T10 Technical Committee
From: Greg Pellegrino (Greg.Pellegrino@compaq.com)
and Rob Elliott, Compaq Computer Corporation (Robert.Elliott@compaq.com)
Date: 5 March 2001
Subject: SRP InfiniBand™ annex

**Revision History**

Revision 0: 5 Jan 2001 first revision
Revision 1: 11 Jan 2001 with updates from Houston SRP meeting.
Revision 2: 18 Feb 2001 with updates from Orlando SRP meeting and San Francisco IBTA AWG FTF meeting. Change bars removed due to amount of changes.
Revision 3: 3 Mar 2001 with updates from Denver SRP and IBTA AWG joint meeting. Still no change bars. Removed most support material.

**Related Documents**

T10/srp-r03 – SCSI over RDMA protocol revision 3 (by Ed Gardner)
T10/fcp2r06 – SCSI over Fibre Channel protocol revision 6 (by Bob Snively)

Access controls:
T10/99-245r9 Access Controls (by Jim Hafner)
T10/00-261r0 Discussion of editorial changes to Access Controls (by Jim Hafner)
T10/00-287r1 TransportIDs for Access Controls (by Jim Hafner)
T10/00-381r0 Three minor modifications to Access Controls (by Jim Hafner)
T10/01-026r0 SPC-3 Access Control conflicts due to TransportIDs (by Rob Elliott)

T10/00-268r6 Defining Targets/Initiators as Ports (by George Penokie)
T10/00-425r1 Long Identifiers in SPC-3, SAM-2, SBC-2 and other XOR issues (by Jim Hafner)

T11/FC-GS-3 Revision 6.42 (Section 6.1.2.3 Platform Object)
T11/99-697v0 Management Server Platform Extension (by Duane Baldwin) (source of FC-GS-3)

InfiniBand Architecture Volume 1 – General Specifications, Release 1.0
InfiniBand Architecture Volume 2 – Physical Specifications, Release 1.0
InfiniBand Architecture Volume 3 – Application of InfiniBand, Release 0.9 (draft)

IETF/draft-ietf-ips-iscsi-disc-reqts-01.txt – iSCSI naming and discovery requirements (IPS working group)
IETF/draft-ietf-ips-iscsi-name-disc-00.txt - iSCSI naming and discovery (IPS working group)
IETF/draft-bakke-iscsi-wwui-urn-00 - A URN Namespace for iSCSI World-Wide Unique Identifiers (IPS working group)

**Overview**

This proposes topics and text for an InfiniBand annex for the SCSI over RDMA (SRP) standard.

The goal is to identify all optional InfiniBand features that must be implemented to ensure useful, interoperable SRP devices. An annex in InfiniBand Volume 1, taken from Volume 3, will describe how boot devices, a subset of SRP devices, are specifically identified and the minimum command sets that may be depended upon (the “Storage Boot Wire Protocol”).

All text outside [brackets] is part of the suggested text.

**Major open issues as of 5 March 2001**

1. Definition of the initiator and target port identifier. Can we use the iSCSI WWUI (255 bytes) or should we define our own (shorter) software-generated worldwide unique identifier?

2. Will SRP_LOGIN_REQ fit in the 92-byte PrivateData limit for InfiniBand connection requests? If the iSCSI WWUI is used, it won't fit; SRP_LOGIN_REQ would need to be split into multiple IUs.
3. Third-party target descriptor formats

**Suggested text**.

Annex A (normative)
SRP for InfiniBand™ Architecture

[footnote] InfiniBand is a trademark and service mark of the InfiniBand Trade Association.

### A.1 Related documents

InfiniBand™ Architecture Volume 1 – General Specifications. Release 1.0. InfiniBand℠ Trade Association.

IETF RFC 2373 - IP Version 6 Addressing Architecture. R. Hinden and S. Deering. Internet Engineering Task Force.

### A.2 Glossary

[Items may be added to the main SRP glossary or defined in the annex. Since they are only used in the annex, I suggest defining them in the annex.]

#### A.2.1 Overview

See the InfiniBand Architecture Volume 1 glossary for full definitions of InfiniBand terms.

### A.2.2 Definitions

A.2.2.1 **Channel adapter:** Device that terminates a link and executes transport-level functions. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.2 **Communication manager:** The software, hardware, or combination of the two that supports the communication management mechanisms and protocols used to establish and release InfiniBand connections. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.3 **Consumer:** The direct user of verbs. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.4 **Global ID (GID):** A port address used for directing packets between subnets. A GID is a valid 128-bit IPv6 address. Source and destination GIDs are carried in an optional global routing header. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.5 **General service interface:** An interface providing management services other than subnet manager. Uses the well-known Queue Pair 1. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.6 **IO unit:** One or more IO controllers attached to the fabric through a single channel adapter. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.7 **IO controller:** The part of an IO unit that provides IO services. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.8 **IPv6 address:** A 128-bit address constructed in accordance with IETF RFC 2373 for Internet Protocol version 6. See IETF RFC 2373.

A.2.2.9 **Local ID (LID):** A port address used for directing packets within a subnet. Source and Destination LIDs are carried in every packet header. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.10 **Management datagram (MAD):** A packet used for communication to manage an InfiniBand network. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.11 **Packet:** The indivisible unit of InfiniBand Architecture data transfer and routing, consisting of one or more headers, a packet payload, and one or two CRCs. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.12 **Port:** Location on a channel adapter to which a link connects. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.13 **Queue pair (QP):** An interface used for communication. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.14 **Queue pair number (QPN):** A value that identifies a queue pair within a channel adapter. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.15 **Service ID:** A value that allows a Communication Manager to associate an incoming connection request with the entity providing the service. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.16 **Subnet:** A set of InfiniBand ports connected via switches that have a common Subnet ID and are managed by a common Subnet Manager. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.17 **Subnet manager:** Entity that configures and controls a subnet. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

A.2.2.18 **Verbs:** An abstract description of the functionality of a channel adapter. An operating system may expose some or all of the verb functionality through its programming interface. See InfiniBand™ Architecture Volume 1 – General Specifications, release 1.0.

### A.2.3 Acronyms

APM: Automatic path migration
CRC: Cyclic redundancy check
GID: Global ID
IBA: InfiniBand™ architecture
IPv6: Internet Protocol version 6
LID: Local ID
MAD: Management datagram
QP: Queue pair
QPN: Queue pair number
REP: Communication Management Reply MAD
REQ: Communication Management Request MAD
RTU: Communication Management Ready to Use MAD
ROM: Read only memory
SRP: SCSI over RDMA protocol
WWUI: Worldwide unique identifier

## A.3 InfiniBand Architecture overview

This annex specifies requirements for mapping SCSI over RDMA protocol (SRP) onto the InfiniBand Architecture (IBA), a transport that provides the necessary RDMA semantics.

An IBA processor unit contains consumers, one or more channel adapters, each containing one or more ports and queue pairs (see Figure 1).

[Editor’s note: IBA Volume 1 calls this a “processor node.” The SRP group disliked the term “node.” Dare we stray from Volume 1 terminology?]
[Editor’s note: label ports as such in the figure]

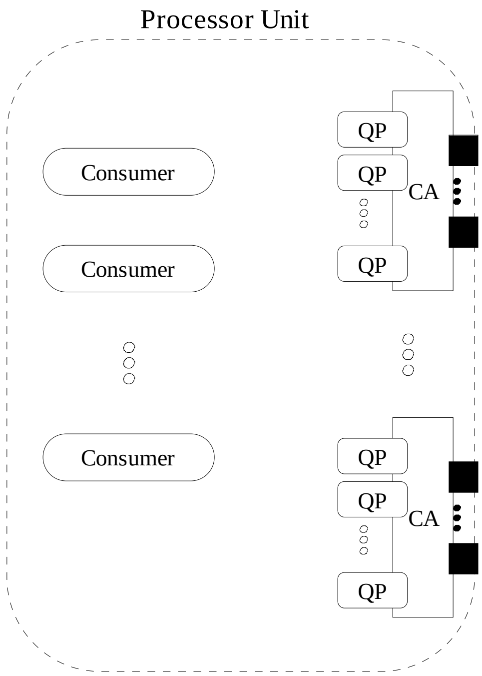


**Figure 1. Processor Unit (derived from InfiniBand Architecture specification).**

An IBA IO unit contains a channel adapter with one or more ports, one or more queue pairs, and one or more IO controllers (see Figure 2). Each IO controller may advertise multiple service IDs. [Editor's note: label ports as such in the figure]

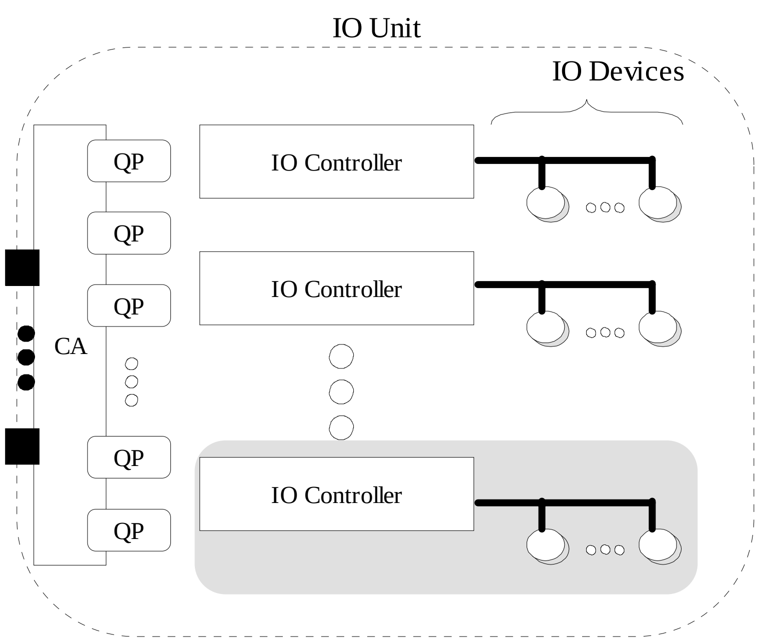


**Figure 2. IO Unit (derived from InfiniBand Architecture specification).**

Each port has a globally unique identifier (GUID) called a port GUID. Each channel adapter has a channel adapter GUID (which applies to all ports on the channel adapter). Each IO controller has an IO controller GUID.

Each port is assigned a local ID (LID) or a range of LIDs by the subnet manager.  Each port has one or more global IDs (GID).  Each GID is globally unique, formed in part from the port GUID.  The subnet manager provides GUID to GID/LID resolution.

Table 1 summarizes the IBA names and addresses relevant to SRP.

**Table 1.  InfiniBand Architecture names and addresses**

| Name | Scope of uniqueness | Description |
|---|---|---|
| Port GUID | worldwide | Identifies a port within a channel adapter |
| Channel adapter GUID (Node GUID) | worldwide | Identifies a channel adapter.  Not used for SRP mapping. |
| IO controller GUID | worldwide | Identifies an IO controller in an IO unit.  Not used for SRP mapping.  [Editor's note: probably not used…] |
| LID | subnet | address assigned by subnet manager to each port |
| GID (IPv6) | worldwide | address assigned by subnet manager; subnet prefix plus the port GUID |

### A.4 SCSI architecture mapping

Figure 3 illustrates how SCSI initiator ports, initiator devices, target ports, and target devices map to InfiniBand Architecture objects.  The figure also illustrates the mapping of the I-T-L nexus onto InfiniBand Architecture objects.  The figure shows an initiator in a processor unit and a target in an IO unit.

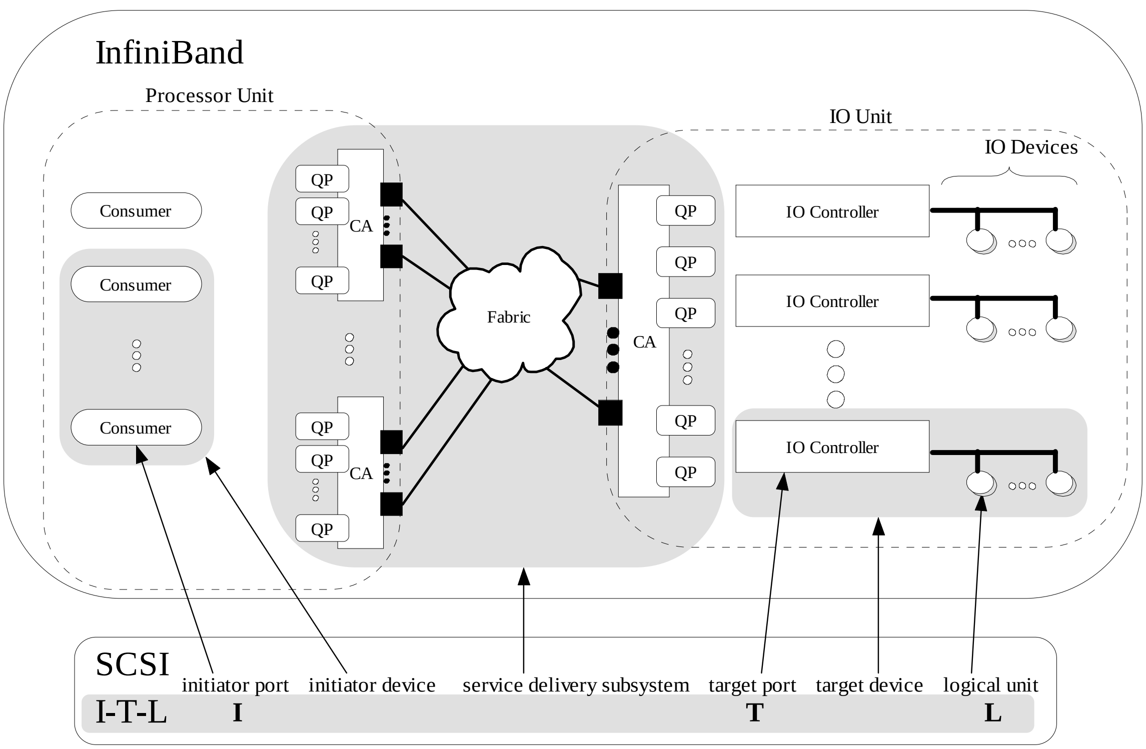


**Figure 3.  SCSI architecture to InfiniBand Architecture mapping example**

[Editor's note: In the table below, Fibre Channel FCP-2 and iSCSI protocol mappings are also shown for comparison purposes.  These do not belong in SRP itself.]
[Editor's note:  The table itself is for discussion purposes; the paragraphs that follow contain the resulting definitions appropriate for this section.]

[Editor's note: 00-268r6 clarifies that initiator/target identifier mean initiator/target port identifier. Jim Hafner has suggested further breaking the port identifier into port address (dynamic) and port name (fixed).]

**Table 2. SCSI mapping to InfiniBand Architecture objects**

| **SCSI Architecture** | **SRP for InfiniBand** | ***[Fibre Channel FCP-2]*** | ***[iSCSI]*** | ***[Comment]*** |
|---|---|---|---|---|
| Initiator or target device in a processor unit | One or more consumers | *Set of FC ports* | *iSCSI session* | |
| Initiator or target port in a processor unit | Set of consumers that share a port identifier | *FC port* | *iSCSI session* | |
| Initiator or target device in an IO unit | IO controller + one or more IO devices | | | |
| Initiator or target port in an IO unit | IO controller | | | |
| | | | | |
| Initiator identifier (aka initiator port identifier in 00-268r6) | iSCSI WWUI (255 byte name) (communicated in SRP login) | *Address identifier of initiator port, i.e. 24 bit FC address* | *iSCSI WWUI (255 byte name)* | |
| What a target uses to tell initiator traffic apart (e.g. for reservation checking and access control checking) | Initiator port identifier | *Initiator port identifier* | *Initiator port identifier* | |
| What a target uses to identify the same initiator returning after a logout for persistent reservations. | Initiator port identifier | *Initiator port's world wide identifier* | *Identifier port identifier* | *Note that semantics change from FC to iSCSI and SRP. FC used a hardware port-level construct; iSCSI and SRP want to use a higher-level software construct.* |
| What is carried in Access Controls TransportID (carried in the data payload of the ACCESS CONTROL OUT command) | Initiator port identifier | *Initiator port's WWN* | *Initiator port identifier* | |
| | | | | |
| Target identifier (aka target port identifier in 00-268r6) | iSCSI WWUI (255 byte name) (communicated in SRP login) | *Address identifier of target port, i.e. 24 bit FC address* | *iSCSI WWUI (255 byte name)* | |
| Target identifier for Extended Copy, XOR commands, and Alias commands | target port identifier + suggested LID (on copy manager's subnet) + suggested GID (of target port) | *One of:*<br>*a) target port worldwide name*<br><br>*b) address identifier of target port*<br><br>*c) address* | *TBD; maybe:*<br><br>*IP address + IP port with target port identifier checking*<br><br>*DNS name + IP port with target* | |

| | | | | |
|---|---|---|---|---|
| | | *identifier of target port with worldwide name checking* | *port identifier checking* | |

An SRP initiator device in a processor node is a set of consumers. An SRP initiator device in an IO unit is an IBA IO controller.

An SRP initiator port in a processor node is a set of consumers that share an initiator port identifier. An SRP initiator port in an IO unit is an IBA IO controller.

An SRP target device in a processor node is a set of consumers. An SRP target device in an IO unit is an IBA IO controller plus one or more IBA IO devices.

An SRP target port in a processor node is a set of consumers that share a target port identifier. An SRP target port in an IO unit is an IBA IO controller.

The service delivery subsystem contains queue pairs, channel adapters, and the InfiniBand fabric.

### A.5 Port names and identifiers

Initiator and target ports in a processor unit shall use WWUIs as port names, as described in Table 3.

[Editor's note: the iSCSI WWUI format could be defined in SAM-2 so all protocols can use it.]

**Table 3. SCSI port name in a processor unit**

| Name | Size | Description |
|---|---|---|
| World Wide Unique Identifier (WWUI) | Up to 255 bytes | A UTF-8 string.<br><br>The first bytes indicate the type. The types are:<br>"iscsi." – reverse DNS name<br>"eui." – EUI-64 name<br>"oui." – naming authority is the 24-bit OUI assigned by IEEE<br>"dns." – Domain Name Server authority<br><br>The remainder is worldwide unique, guaranteed by the naming authority indicated by the type.<br><br>(see draft-ietf-ips-iscsi-name-disc-00.txt for details) |

Initiator and target ports in an IO controller shall use either an IO controller GUID or a WWUI as port names, as described in Table 4:

**Table 4. SCSI port name in an IO unit**

| Name | Size | Description |
|---|---|---|
| WWUI | Up to 255 bytes | See Table 3 |
| IO controller GUID | 8 bytes | Identifies an IO controller in an IO unit |

The initiator port identifier and a target port identifier are exchanged during SRP login in SRP_Login_Req and SRP_Login_Rsp.

## A.6 Communication management

### A.6.1 Communication management overview

Communications Managers on each InfiniBand device manage InfiniBand connections using MADs over the General Service Interface on each system. SRP ports shall use the active/passive (client/server) connection establishment protocol. The processor node or IO controller containing the SRP target port shall act as the server and the processor node or IO controller containing the SRP initiator port shall act as the client.

### A.6.2 Discovering SRP target ports

To discover the service ID of an SRP target port in an IO Unit, an SRP initiator port may use this sequence:
1) Retrieve the IOUnitInfo attribute from an IO unit using a DevMgtGet MAD to determine the presence and slot number of each IO controller attached to the IO unit;
2) retrieve the IOControllerProfile attributes from each IO controller, each of which includes a ServiceEntries table; and,
3) search the ServiceEntries table for entries with service names of “SRP.T10.NCITS”.

The service ID associated with each matching service name may be used in the communication management process to open InfiniBand connections to IO controllers containing SRP target ports.

Discovering service IDs of SRP target ports in processor units is beyond the scope of this standard.

### A.6.3 Establishing a connection

To establish an InfiniBand connection, the client places the service ID in a Communication Management Request (REQ) message. The server associates the request with the appropriate SRP target port. The PrivateData field of the REQ message shall include an SRP_LOGIN_REQ IU. The SRP target port may choose to refuse the connection based on the SRP_LOGIN_REQ IU content by using a Communication Manager Reject value of “consumer reject.”

If the server accepts the connection request and SRP login, the server returns a queue pair number (QPN) in a Response (REP) message. The PrivateData field of the REP message shall include an SRP_LOGIN_RSP IU. The SRP initiator port may choose to refuse the connection based on the SRP_LOGIN_REQ IU content by using a Communication Manager Reject value of “consumer reject.”

If the client accepts the connection reply and SRP login response, it replies with a Ready To Use (RTU) message and both an InfiniBand and an SRP connection are opened. It may start using the connection for communication.

[Editor’s note: iSCSI 255-byte WWUIs are too large to fit into CM REQ, which is limited to 92 bytes. The CM REP is limited to 204 bytes. LOGIN information in the CM message allows SRP involvement in connection rejection. Like to have some sort of minimal LOGIN information in CM REQ/REP, then longer LOGIN IUs with device identifiers exchanged with SEND operations. SEND operations will have to be handled as multiple SEND messages since will be subject to MTU default size until message length is negotiated – presumably after LOGIN.]

### A.6.4 Releasing a connection

Either the SRP initiator port or SRP target port may send an SRP_LOGOUT IU with a SEND operation. The sender shall disconnect upon receipt of an InfiniBand transport level acknowledgement to the SRP_LOGOUT IU. The sender may disconnect if its send queue has transitioned to an error state. The receiver of a LOGOUT IU shall respond with an InfiniBand transport acknowledgement and disconnect.

## A.7 InfiniBand protocol requirements

SRP target ports and SRP initiator ports shall support the Reliable Connection transport service type.

SRP target ports implemented in IO units shall implement the DevMgt class of general management services.

SRP initiator ports and SRP target ports shall support the transport functions described in Table 5.

**Table 5. Transport operation support requirements**

| Transport functions | SRP initiator port | SRP target port |
|---|---|---|
| Send to | Mandatory | Mandatory |
| Send from | Mandatory | Mandatory |
| RDMA write to | Mandatory | Not used |
| RDMA write from | Not used | Mandatory |
| RDMA read to | Mandatory for data-out commands | Not used |
| RDMA read from | Not used | Mandatory for data-out commands |
| RDMA Write with immediate data (to or from) | Not used | Not used |
| ATOMIC (to or from) | Not used | Not used |

IO units containing an IO controller acting as an SRP target ports shall report the device management IOUnit attributes as described in Table 6.

**Table 6. IOUnit attributes for SRP target ports**

| Field | Description |
|---|---|
| Change ID | |
| Max Controllers | At least one |
| Option ROM | |
| Controller List | At least one IO controller must be present. |

IO controllers acting as SRP target ports shall report the device management IOControllerProfile attributes as described in Table 7.

**Table 7. IOControllerProfile attributes for SRP target ports**

| Field | Description |
|---|---|
| [IO controller] GUID | |
| Device ID | |
| Vendor ID | |
| Device Version | |
| Subsystem Vendor ID | |
| Subsystem [Device] ID | |
| IO Class | [TBD assigned by AWG] |
| IO Subclass | [TBD assigned by AWG] |

| | |
|---|---|
| Protocol | [TBD assigned by AWG] |
| Protocol Version | SRP ports shall use 0x0000.<br>[Editor's note: 0x0001 is reserved for SRP-2] |
| Service Connections | At least one |
| Initiators Supported | At least one |
| Send Message Depth | At least one |
| RDMA Read Depth | At least one |
| Send Message Size | [Large enough to hold connection establishment with private data containing SRP login] |
| RDMA Transfer Size | At least one<br>[Editor's note: significantly larger than one preferred] |
| Controller Operations Capability Mask | These bits shall be set to one:<br>0: ST (Send Messages to IO controllers)<br>1: SF (Send Messages from IO controllers)<br>5: WF (RDMA Write Requests from IO controllers)<br><br>This bit shall be set to one by SRP target ports supporting data-out commands:<br>3: RF (RDMA Read Requests from IO controllers)<br><br>These bits may be set to zero:<br>2: RT (RDMA Read Requests to IO controllers)<br>4: WT (RDMA Write Requests to IO controllers)<br>6: AT (Atomic Operations to IO controllers)<br>7: AF (Atomic Operations from IO controllers) |
| Controller Services Capability Mask | Bit 1 may be set for SRP ports with boot support:<br>1: SBWP Storage Boot Wire Protocol |
| Service Entries | At least one |
| ID String | |

An IO controller acting as an SRP target port shall register with its Communications Manager a Service Name string of "SRP.T10.NCITS". This string is assigned an IO SERVICE ID type service ID by the Communications Manager.

IO controllers acting as SRP target ports shall include at least one Service Name/Service ID pair in the device management ServiceEntries attribute pair as described in Table 8.

**Table 8. ServiceEntries attribute pair for SRP target ports**

| Field | Length | Description |
|---|---|---|
| ServiceName_n | 320 | "SRP.T10.NCITS" |
| ServiceID_n | 64 | Assigned by the IO controller |

### A.8 Extended Copy target descriptor

[Editor's note: this goes in SPC-3 rather than in the protocol standard.]

[Editor's note: There are 3 FC target descriptors:

a) port WWN
b) port ID (D_ID, the 24-bit FC address)
c) port ID (D_ID) with port WWN checking.

None of them are node-based or platform-based.

The SRP equivalents (not necessarily what we want to use) are roughly:

a) Port GUID (8 bytes)
b) LID/GID (2 + 16 bytes = 18 bytes)
c) LID/GID with Port GUID checking (2 + 16 + 8 = 26 bytes)

With the 1 byte TYPE field and 8 byte LUN fields added, only the first fits in the 24-bit target descriptor. The REPORT/CHANGE ALIASES method must be used to create longer descriptors – see T10/00-425 and below.

We assume SRP needs to use higher level identifiers so don't document a port GUID format (the only one that would fit). Thus, there is NO non-alias Extended Copy target descriptor proposed for SRP InfiniBand.]

### A.9 REPORT ALIASES and CHANGE ALIASES Identifier/Address format

[Editor's note: this may go in SPC-3 rather than in the protocol standard.]

[Editor's note: This is based on Jim Hafner's T10/00-425r1, which is new and subject to change.]

The identifier/address formats in Table 9 are used by the REPORT ALIASES and CHANGE ALIASES commands to identify SRP target ports.

[Editor's note: should a Type range be reserved for each transport – InfiniBand, FC-VI, VI over IP, etc.?]

**Table 9. Identifier/Address formats for SRP for InfiniBand**

| Protocol Identifier | Protocol Description | Type | Type Description | Format/Length | *[Comments]* |
|---|---|---|---|---|---|
| 04h | SRP | 10h | InfiniBand LID/GID with target port identifier checking | 2-byte LID + 16-byte GID + 255 bytes | *Works without a name server.* |
| | | 20h | Target port identifier | 255 bytes | *Requires a name server that can provide a LID/GID when given a target port identifier.* |
| | | 21h | Target port identifier plus suggested InfiniBand LID/GID | 255 bytes + 2-byte LID + 16-byte GID | *Requires a name server that can provide a LID/GID when given a target port identifier. The LID/GID specified is just a hint.* |

### A.10 Access Controls TransportID

[Editor's note: this goes in SPC-3 rather than in the protocol standard.]

The TransportID used by a target to identify initiators for access controls shall have the format described in Table 10.

[Editor's note: the length of this TransportID is different than that for existing TransportIDs. This existing format supports variable length fields – they just all happen to be 24 bytes today.]

[Editor’s note: this is not currently InfiniBand specific – all SRP transports should be compatible]

**Table 10. TransportID for SRP**

| Byte Bit | 7 | 6 | 5 | 4 | 3 | 2 | 2 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | TYPE (02H) | | | | | | | |
| 1 | RESERVED | | | | | | | |
| 2 | RESERVED | | | | | | | |
| 3 | RESERVED | | | | | | | |
| 4 | RESERVED | | | | | | | |
| .. | RESERVED | | | | | | | |
| 7 | RESERVED | | | | | | | |
| 8 | INITIATOR PORT IDENTIFIER | | | | | | | |
| … | | | | | | | | |
| 262 | | | | | | | | |

**Additional notes**
This text is not part of the proposed annex.

[Additional background on initiator identifier:]
[Excerpt from SPC-2 revision 18c:

> The device server shall preserve the following information for each registration across any reset, and if the APTPL capability is enabled, across any power off period:
> a) Initiator identifier;
> b) reservation key; and
> c) when supported by the protocol, the initiator port's world wide identification.
>
> The device server shall preserve the following reservation information across any reset, and if the APTPL capability is enabled, across any power off period:
> a) Initiator identifier;
> b) reservation key;
> c) scope;
> d) type; and
> e) when supported by the protocol, the initiator port's world wide identification.
>
> For those protocols for which the initiator port's world wide identification is available to the device server the initiator port's world wide identification shall be used to determine if the initiator identifier has changed. This determination shall be made at any time the target detects that the configuration of the system may have changed. If the initiator identifier changed, the device server shall assign the new initiator identifier to the existing registration and reservation of the initiator port having the same world wide identification.

]

[Additional background on FC worldwide names:]
[Excerpt from FCP-2 revision 5:

> **5.3 Use of World Wide Names**
> As specified in FC-FS, each Fibre Channel node and each Fibre Channel port shall have a Worldwide_Name, a unique name using one of the formats defined by FC-FS and its extensions. Each target and its associated logical units has knowledge of the Port_Name of each initiator through the Fibre Channel login process. As a result, the relationship between address identifier of the initiator and a persistent reservation for a logical unit may be adjusted as defined in SPC-2 during those reconfiguration events that may change the address identifier of the initiator. If a target receives a PRLI or a PLOGI from an initiator FCP_Port with a previously known Worldwide_Name but with a changed initiator identifier, the device server shall assign the new initiator identifier to the existing registration and reservation to the initiator port having the same Worldwide_Name.
>
> Each logical unit shall be able to present a Worldwide_Name through the INQUIRY command vital product data device identification page as defined by SPC-2. For devices compliant with this standard and having a LUN 0, the Worldwide_Name of the logical unit having a LUN of 0 may be the same as the Node_Name of the target. The Worldwide_Name for the port shall be different from the Worldwide_Name for the node.

]

[These are the problems we're trying to solve with initiator and target identifiers:

- General
    - returning data to the same initiator that sent the command. Requires knowledge of the volatile fabric address (e.g. 24-bit FC address).
    - maintaining per-initiator mode pages.

- identification is implicit by the transport. Initiator identifier is not carried as a payload in any data structure.

- Reservations
  - recognizing initiators as different to allow or reject commands
  - third party reservations (essentially obsolete). Initiator identifier carried as a payload.
  - persistent reservations – must identifying the initiator is the same through a logout/login. Initiator identifier is implicit, not carried as a payload.

- Access controls
  - need to identify initiators to allow or deny access based on either TransportID or higher-level AccessID
  - TransportID identifies an initiator with a transport level address, so the initiator doesn't need to know about access control itself
  - TransportID carried in data payload
  - recognizing different initiators to allow or reject commands

- Third-party commands
  - need to identify targets to receive another SCSI command
  - target identifier contained in data payload
  - Extended Copy target descriptors
  - XOR block commands like XPWrite Extended and Rebuild
  - proposed Alias commands target descriptors which allow for longer descriptors.

]