To: T10 Technical Committee
From: Greg Pellegrino (Greg.Pellegrino@compaq.com)
and Rob Elliott, Compaq Computer Corporation (Robert.Elliott@compaq.com)
Date: 18 February 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.

Related Documents
T10/srp-r02 – SCSI over RDMA revision 2 (by Ed Gardner)
T10/fcp2r05 – FCP-2 revision 5 (by Bob Snively)

T10/99-245r9 Access Controls (by Jim Hafner)
T10/00-261r0 Discussion of editorial changes to Access Controls (by Jim Hafner)
T10/00-268r4 Defining Targets/Initiators as Ports (by George Penokie)
T10/00-287r1 TransportIDs for Access Controls (by Jim Hafner)
T10/00-381r0 Three minor modifications to Access Controls (by Jim Hafner)
T10/00-425r0 Long Identifiers in SPC-3, SAM-2, SBC-2 and other XOR issues (by Jim Hafner)
T10/01-026r0 SPC-3 Access Control conflicts due to TransportIDs (by Rob Elliott)

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 (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. InfiniBand Volume 3 will describe how boot devices, a subset of SRP devices, are specifically identified and handled (“Storage Boot Wire Protocol”).

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

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^{SM} Trade Association.

InfiniBand™ Architecture Volume 2 – Physical Specifications. Release 1.0. InfiniBand℠ Trade Association. [needed for Chassis and Module discussion]
InfiniBand™ Architecture Volume 3 – Application of InfiniBand. Release 0.9 (draft). InfiniBand℠ Trade Association.

IETF RFC 2460 – Internet Protocol, Version 6. S. Deering and R. Hinden. Internet Engineering Task Force. [needed if GID format is referenced]

[rules: do not cross-reference section numbers in other documents. Can refer to section headings.]

A.2 Glossary

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

A.2.1 Overview

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

A.2.2 Acronyms

APM: Automatic path migration
CA: Channel adapter
GID: Global ID
GSA: General service agent
HCA: Host channel adapter
IBA: InfiniBand™ architecture
IOC: IO controller
IPv6: Internet Protocol version 6
LID: Local ID
MAD: Management datagram
QP: Queue pair
QPN: Queue pair number
ROM: Read only memory
SMA: Subnet management agent
SRP: SCSI over RDMA protocol
TCA: Target channel adapter

A.2.3 Definitions

A.2.3.1 Automatic path migration:

A.2.3.2 Channel adapter (CA): Device that terminates a link and executes transport-level functions. One of a Host channel adapter or Target channel adapter.

A.2.3.3 Communications manager: The software, hardware, or combination of the two that supports the communication management mechanisms and protocols used to establish and release InfiniBand connections.

A.2.3.4 Consumer: The direct user of verbs.

A.2.3.5 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.

A.2.3.6 General Service Interface: An interface providing management services other than subnet manager. Uses the well-known Queue Pair 1.

A.2.3.7 Host channel adapter (HCA): A channel adapter used to support processor nodes.

A.2.3.8 IO unit: One or more IO controllers attached to the fabric through a single TCA.

A.2.3.9 IO controller: The part of an IO unit that provides IO services.

A.2.3.10 IPv6 Address: A 128-bit address constructed in accordance with IETF RFC 2460 for IPv6.

A.2.3.11 Local ID (LID): A port address used for directing packets within a subnet. Source and Destination LIDs are carried in every packet header.

A.2.3.12 Management datagram (MAD): A packet used for communication to manage an InfiniBand network.

A.2.3.13 Port: Location on a channel adapter to which a link connects.

A.2.3.14 Queue pair (QP): An interface used for communication.

A.2.3.15 Queue pair number (QPN): A value that identifies a queue pair within a channel adapter.

A.2.3.16 Service ID: A value that allows a Communication Manager to associate an incoming connection request with the entity providing the service.

A.2.3.17 Subnet: A set of InfiniBand ports connected via switches that have a common Subnet ID and are managed by a common Subnet Manager.

A.2.3.18 Subnet manager: Entity that configures and controls a subnet.

A.2.3.19 Subnet management agent (SMA): An entity present in every channel adapter that processes subnet management packets from a subnet manager.

A.2.3.20 Target channel adapter (TCA): A channel adapter used to support I/O units.

A.2.3.21 Verbs: An abstract description of the functionality of a Host Channel Adapter. An operating system may expose some or all of the verb functionality through its programming interface.

A.3 Overview

A.3.1 InfiniBand Architecture

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 one or more host channel adapters (HCAs), each containing one or more ports (see Figure 1).

[Editor's note: InfiniBand Volume 1 uses "processor node". We used "processor unit" to parallel "IO unit."]

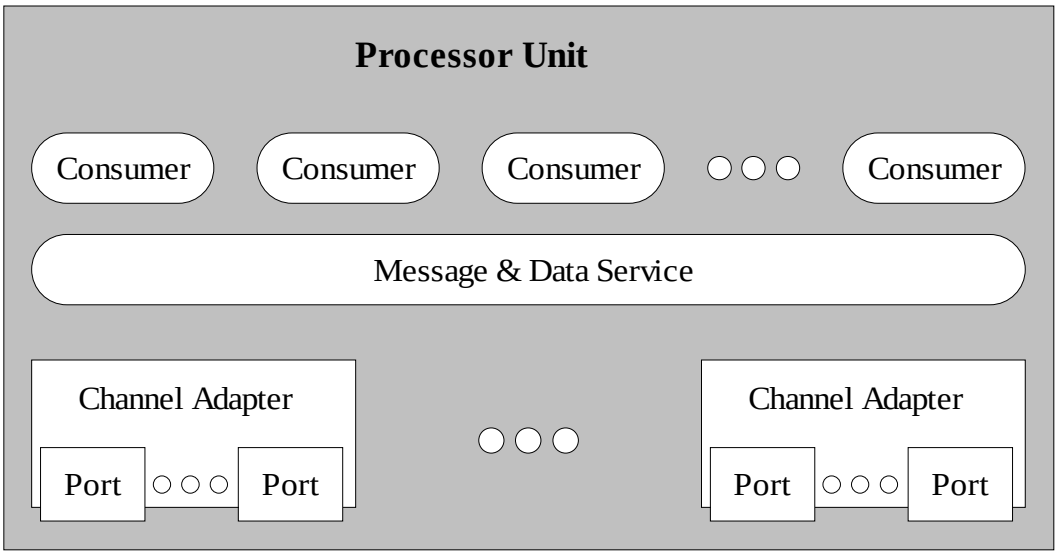

Figure 1. Processor Unit (derived from InfiniBand Architecture specification).

An IBA IO unit contains a target channel adapter (TCA) with one or more ports, a subnet management agent (SMA), general service agents (GSAs), and one or more IO controllers (IOCs) (see Figure 2). Each IO controller may advertise multiple service IDs.

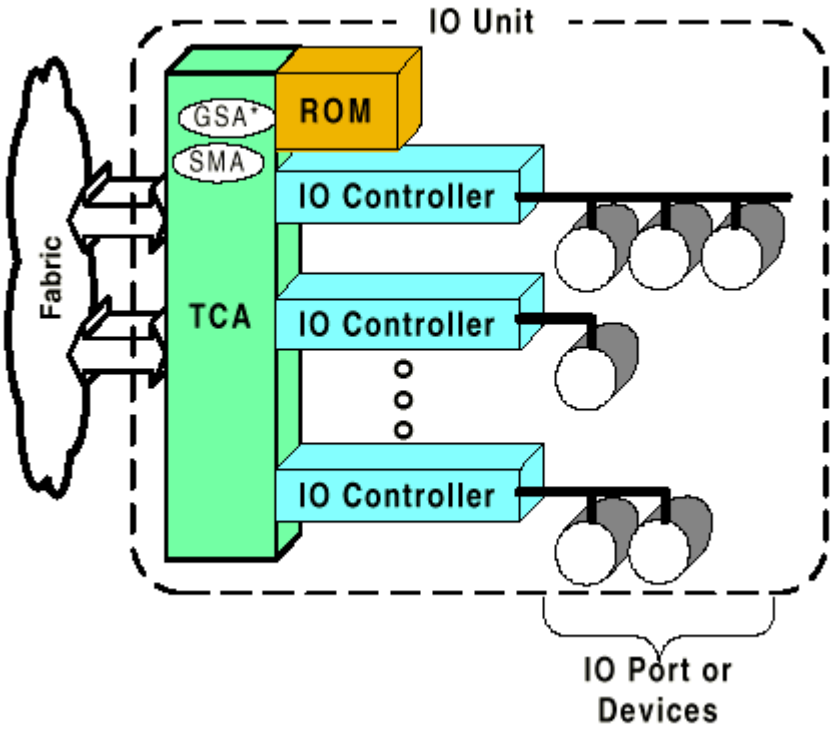

Figure 2. IO Unit (from InfiniBand Architecture specification).

Figure 3 shows the InfiniBand architecture objects used in the SRP mapping.

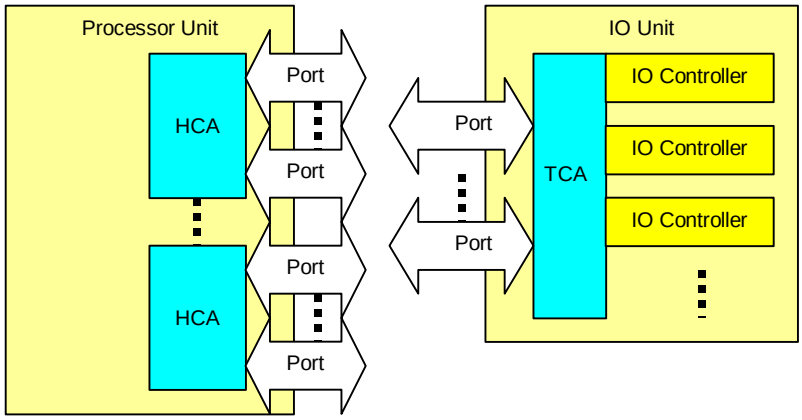

Figure 3. InfiniBand Architecture objects.

Each port has a globally unique identifier called a port GUID. Each channel adapter has a CA GUID (which is shared between all of the channel adapters ports). Each IO controller has an IOC 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 addressable entities relevant to SRP.

Table 1. IBA names and addressable entities

Name	Scope of uniqueness	Description
Port GUID	worldwide	EUI-64 identifying a port within a channel adapter
Channel adapter GUID/Node GUID	worldwide	EUI-64 identifying a channel adapter. Not used for SRP mapping.
IO Controller GUID	worldwide	EUI-64 identifying an IO controller in an IO unit
LID	subnet	assigned by subnet manager to each port
GID (IPv6)	worldwide	assigned by subnet manager; subnet prefix plus the port GUID

A.3.2 SCSI Mapping

Figure 4 illustrates how SCSI initiator ports, initiator devices, target ports, and target devices map to InfiniBand Architecture objects.

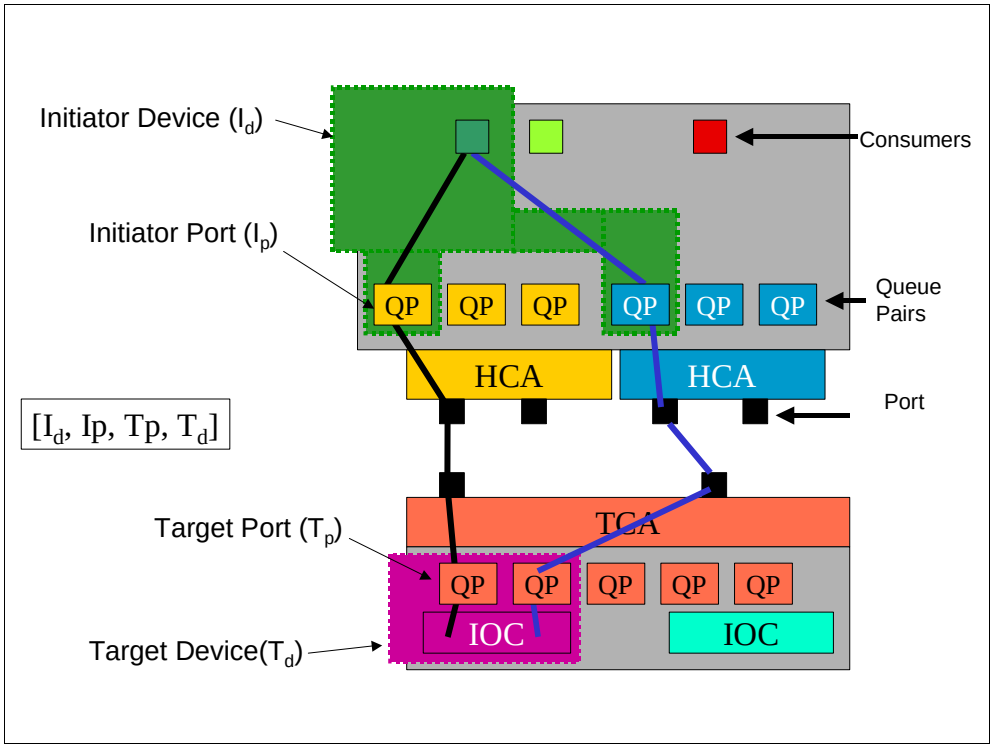

Figure 4. SCSI architecture mapping to InfiniBand architecture example.

[Editor's note: Fibre Channel FCP-2 and iSCSI protocol mappings are also shown for comparison purposes. These do not belong in SRP itself.]

[Editor's note: 00-268r4 splits initiator/target identifier into initiator/target device identifier and initiator/device port identifier. Jim Hafner has suggested further breaking the port identifier into port address and port name.]

Table 2. SCSI mapping to InfiniBand Architecture objects

SCSI Architecture	SRP for	*[Fibre Channel*	*[iSCSI]*	[Comment]

	InfiniBand	***FCP-2]***		
Initiator device	Consumer + Initiator QP(s)	*Set of FC ports*	*ISCSI session*	
Initiator port	Initiator QP	*FC port*	*iSCSISession*	
Target device	IOC + Initiator QP(s)	*Set of FC ports*	*Session*	
Target port	Initiator QP	*FC port*	*Session*	
Initiator identifier (SAM-2 existing term)	iSCSI WWUI (254 byte name) (communicated in SRP login)	*Address identifier of initiator port, i.e. 24 bit FC address*	*iSCSI WWUI (254 byte name)*	
Initiator device identifier (00-268r4)	iSCSI WWUI (254 byte name) (communicated in SRP login)	*N/A today Could use the FC-GS-3 platform name (255 bytes) FC-GS-4 could define it as an iSCSI WWUI format*	*iSCSI WWUI (254 byte name)*	Potentially compatible across protocols Put the definition of the format of initiator device identifier into SAM-2 so it can be shared across protocols. Use iSCSI's WWUI ASCII format.
Initiator port identifier (00-268r4)	Target's QPN	*Address identifier of initiator port, i.e. 24 bit FC address*	*Set of initiator IP addresses and IP port numbers*	Jim Hafner's proposal replaces initiator port identifier with initiator port address and initiator port name
Initiator port address (Jim Hafner's proposal)	Target's QPN	*Address identifier of initiator port, i.e. 24 bit FC address*	*Set of initiator IP addresses and IP port numbers*	As viewed by the IOC. Limits one initiator to a QP.
Initiator port name (Jim Hafner's proposal)	Initiator port GUID	*Initiator port worldwide_name*	*Set of initiator DNS names and IP port numbers*	IOC shall keep mapping of Initiator port GUIDs to Initiator Device Identifier to maintain reservations over multiple channels within a connection.
What a target uses to tell initiator traffic apart (e.g. for reservation checking and access control checking)	Initiator port address	*Initiator port address*	*Initiator port address*	
What a target uses to identify the same initiator returning after a logout for persistent reservations.	Initiator device identifier	*Initiator port's world wide identifier*	*Identifier device identifier*	Note that semantics change from FC to iSCSI and SRP. FC used a port-level construct; iSCSI and SRP want to use a higher-level construct.
What is carried in Access Controls TransportID (carried in the data payload of the	Initiator device identifier	*Initiator port's WWN*	*Initiator device identifier*	

ACCESS CONTROL OUT command)				
Target identifier (SAM-2 existing term)		*Address identifier of target port, i.e. 24 bit FC address*		
Target device identifier (00-268r4)		*N/A*		
Target port identifier (00-268r4)	Initiator's QPN	*Address identifier of target port, i.e. 24 bit FC address*		
Target Port Address (Jim Hafner's proposal)	Initiator's QPN	*Address identifier of target port, i.e. 24 bit FC address*	*Set of target IP addresses and IP port numbers*	
Target Port Name (Jim Hafner's proposal)	Target port GUID	*Target port worldwide_name*	*Set of target DNS names and IP port numbers*	
Target identifier for Extended Copy, XOR commands, and Alias commands	a) LID (on copy manager's subnet) and GID (of target device) with target device identifier checking b) target device identifier with suggested LID (on copy manager's subnet) and GID (of target device) suggestion	*c) target port worldwide_name* *d) address identifier of target port* *e) address identifier of target port with worldwide_name checking*	*TBD* *Maybe IP address + IP port with target device identifier checking*	
I-T nexus	Connection?	fully qualified exchange identifier		
I-T-L nexus				
I-T-L-Q nexus		fully qualified exchange identifier + logical unit number		

[Editor's Option 1 for defining an initiator device]
An SRP initiator device in a processor node is a set of consumers that share an initiator device identifier. An SRP initiator device in an IO unit is an IO controller.
[Editor's Option 2 for defining an initiator device
An SRP initiator device is identified by an initiator device identifier.
]

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

Initiator and target devices may communicate over multiple QP pairs (channels) simultaneously. Commands shall be completed over the same channel as submitted. Command ordering shall be maintained within individual channels. No command ordering is implied over concurrent channels.

A.3.3 Communication Management Overview

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

A.3.4 Device Identifiers

As described in the main body of SRP, an initiator device identifier and a target device identifier are exchanged during SRP login in SRP_Login_Req and SRP_Login_Rsp, respectively. The initiator device identifier is used and recorded by a target for persistent reservations. The target device identifier is used for third party commands such as extended copy.

[Editor's note: Table 3 presents the possible fields that may be assigned to the initiator device identifier. If a generic format like that used for iSCSI WWUI is chosen, then move this information from the annex to main document. If an InfiniBand specific format is chosen, then the annex needs to define it.]

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

Table 3. Initiator device identifier

Name	Size	Description
World Wide Unique Identifier (WWUI)	Up to 254 bytes	The first byte indicates the length, the second byte indicates the type, and the remaining bytes define the type. The types are: 00 – No Authority (not guaranteed to be unique) 01 – ASCII (using reversed DNS name as Naming Authority) 02 – IEEE EUI-64 03 – Unicode (DNS naming authority) 04 – Generic Binary WWUI (to be considered)

Table 4 contains options for target device identifiers:

Table 4 Target device identifier

Name	Size	Description
WWUI	Up to 254 bytes	See initiator device identifier.
IOC GUID	8 bytes	EUI-64 identifying an IO controller in an IO unit
Channel adapter GUID	8 bytes	EUI-64 identifying a channel adapter.
Port GUID	8 bytes	EUI-64 identifying a port within a

		channel adapter.

]

Devices that use Automatic Path Migration shall define Device Identifiers with a name scoped above the Channel Adapter level.

A.3.5 Establishing a connection

<Editor's Note: IBA defines methods for identifying nodes. We should not attempt to redefine these methods. How much background do we need to state here to specify that SRP_LOGIN_IUs are included in the PrivateData field of the REQ?>

To discover how to establish an InfiniBand connection to an SRP target device, the client may retrieve the IOUnit and ServiceEntries attributes from an IO unit using a DevMgtGet MAD. This provides the list of IOCs, their supported protocols, and their associated Service Entries. The client filters the list to find a Service Name for "SRP.T10.NCITS". The Service Id associated may then be used to in the communication management process to resolve QP(s) for the IOC.

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 device. If it accepts the connection request, the server returns a queue pair number (QPN) in a Response (REP) message. The client replies with a Ready To Use (RTU) message.

The PrivateData field of the REQ message shall include an SRP_LOGIN_REQ IU. The SRP target device may choose to refuse the connection with Communication Manager Reject Code 28 (consumer reject), based on parsing this login message. If it accepts the login, the SRP target device shall return an SRP_LOGIN_RSP IU in the PrivateData field of the REP MAD. This creates an SRP connection at the same time the InfiniBand connection is created.

[Editor's note: iSCSI 254-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.3.6 Releasing a connection

LOGOUT IU is sent as a SEND operation. Initiator devices and target devices may send LOGOUT IU. The sender shall disconnect upon receipt of ACK to the LOGOUT IU. The sender may disconnect at any time after sending the LOGOUT IU. The receiver of a LOGOUT IU shall respond with ACK and disconnect.

A.4 InfiniBand protocol requirements

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

SRP target devices shall not use RDMA transfer lengths that exceed the maximum transfer size of 2^31.

SRP target devices shall ensure that the sum of the packet payloads adds up to the requested transfer size.

SRP target devices shall implement the DevMgt class of general management services.

Support for various transport functions is described in Table 5.

Table 5. Transport function support requirements.

Transport functions	Initiator device	Target device
Send to	Required	Required
Send from	Required	Required
RDMA write to	Required	Prohibited
RDMA write from	Prohibited	Required
RDMA read to	Required for data-out commands	Prohibited
RDMA read from	Prohibited	Required for data-out commands
RDMA Write with immediate data (to or from)	Prohibited	Prohibited
ATOMIC (to or from)	Prohibited	Prohibited

SRP target devices in an IO unit shall include device management IOUnit attributes as described in Table 6.

Table 6. IOUnit attributes for SRP devices

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

SRP target devices in an IO unit shall include the device management IOControllerProfile attributes as described in Table 7.

Table 7. IOControllerProfile attributes for SRP devices

Field	Description
[IO controller] GUID	The GUID shall be the same as that reported for LUN 0 in INQUIRY VPD Page 83h Type 2h (EUI-64)
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 devices shall use 0x0000. 0x0001 is reserved for future SRP-2 devices.
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 <significantly larger than one preferred>
Controller Operations Capability Mask	These bits shall be set to one: 0: ST (Send Messages to IOCs) 1: SF (Send Messages from IOCs) 5: WF (RDMA Write Requests from IOCs) This bit shall be set to 1 by SRP target devices supporting data-out commands: 3: RF (RDMA Read Requests from IOCs) These bits may (shall?) be set to zero: 2: RT (RDMA Read Requests to IOCs) 4: WT (RDMA Write Requests to IOCs) 6: AT (Atomic Operations to IOCs) 7: AF (Atomic Operations from IOCs)
Controller Services Capability Mask	This bit is set for SRP devices with boot support 1: SBWP Storage Boot Wire Protocol
Service Entries	At least one
ID String	

The SRP target device's IO controller 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.

SRP target devices in an IO unit shall include the Service Name/Service ID pair in the device management ServiceEntries attribute pair as described in Table 8.

Table 8. ServiceEntries attribute pair for SRP devices

Field	Length	Description
ServiceName_n	320	"SRP.T10.NCITS"
ServiceID_n	64	Assigned by Communications Manager within Service Id range assigned by InfiniBand volume 1. The format is: 0xFEnn:nnnn:ttpp:ppii where: nn:nnnn is the OUI currently assigned to NCITS (0x006093). [Editor's note: check that canonical order is handled correctly] tt is the value currently assigned, by NCITS, to T10 (0x01). pp:pp is the value assigned by T10 to indicate SRP (0x00:00). ii is an index to allow multiple SRP target devices at a single IO unit or CA (0x00 through 0xFF).

A.5 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 wants to use higher level identifiers so don't document a port GUID format. Thus, there is NO non-alias Extended Copy target descriptor proposed for SRP InfiniBand.]

A.6 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 T10/00-425r0, which is new and subject to change.]

The identifier/address formats in **Error! Reference source not found.**Table 9 are used by the REPORT ALIASES and CHANGE ALIASES commands to identify SRP target devices.

Table 9. Identifier/Address formats for SRP (for InfiniBand)

Protocol Identifier	Protocol Description	Type	Type Description	Format/Length	[Comments]
04h	SRP	10h	LID/GID with target device identifier checking	2-byte LID + 16 byte GID + 254 bytes	Works without a name server
		20h	Target device identifier	254 bytes	Requires a name server that can provide a LID/GID when given a target device identifier. May be usable on all transports.
		21h	Target device identifier plus suggested LID/GID	254 bytes	Requires a name server that can provide a LID/GID when given a target device identifier. The LID/GID specified is just an optimization.

A.7 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.]**Error! Reference source not found.**

Table 10. TransportID for SRP (for InfiniBand)

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 DEVICE IDENTIFIER							
…								
261								

A.8 Notes for Annex Development (will be discarded)

The following text is editorial and review notes. This is not part of the proposed annex.

[Channel Adapter is a preferable term to Node and is an errata to volume 1 currently]
[Editor's note: Figure 1 taken from IBA Figure 17. Figure 171 might be a better source.]
[Need permission to reuse diagrams from IBTA, Bill may have suitable source]

[Editor's note: Figure 2 taken from IBA Figure 35. Figure 171 might be a better source – it doesn't show a ROM.]

[NOTE: remove all this volume 2 stuff]
[Editor's note: Figure 3is a picture adding in module and chassis concepts from Volume 2 that may also be useful:]
[removed:
An IBA module contains one or more CAs, switches, or routers. An IBA chassis contains one or more IBA modules. Each IBA module may contain a module GUID. Each IBA chassis may contain a chassis GUID.
]

[Editor's note: if an Srp initiator device is implemented in an IO controller, the IO controller cannot share [T]CAs and still follow the IO unit model. It can act as a single-CA processor node, though.]
<shouldn't be any difference than a single initiator behind multiple HCAs?>

[Editor's note: if an SRP target device is implemented by a host using an HCA, it has to support inbound DevMgt MADs and look like an IO controller inside an IO unit. Unlike the initiator, defining SRP target for processor nodes does not describe how to find them.]
<This seems to be describing the host as a processor target, an idea that predates the AER>

[if Figure 4. SCSI architecture mapping to InfiniBand architecture example.

[Editor's note: Fibre Channel FCP-2 and iSCSI protocol mappings are also shown for comparison purposes. These do not belong in SRP itself.]

[Editor's note: 00-268r4 splits initiator/target identifier into initiator/target device identifier and initiator/device port identifier. Jim Hafner has suggested further breaking the port identifier into port address and port name.]

Table 2 is kept, show APM too]

[Editor's note: it is beyond the scope of SRP to define a Configuration Manager/name server like device to help an initiator find IO unit LID/GIDs worth querying for SRP services. Assuming that such devices have been found, a connection is established as described below.]

[Bob: The initiator port identifier is the target's QPN. By talking to its own QP communication just happens to the initiator.
The initiator's GID shall serve as the world wide identification used to track initiator port identifier changes. This is the initiator device ID.
]

[NOTE: page 532 of volume 1 Remote Port GID says GidInfo where it should say GuidInfo. 3 instances of this total.]

[Editor's note: if Reliable Datagrams were supported, then the initiator's EE Context would need to be retained along with the QPN.]

[Editor's note: Since the Port GUID is a worldwide unique name, the name persists through power loss.]

[Mapping initiators from FC to IB by mapping the FC port WWN to an IB initiator device identifier should work fine. For target device initiators too?]

[Editor's note: It would be beneficial to use an identifier of wider scope than the Port GUID like the Node GUID or Platform GUID. A Node GUID would be unaffected by alternate path migration, LID changes, and queue pair number selection. The Platform GUID would be unaffected by CA swaps as well. However, this issue also applies to Fibre Channel and other transports. Adding this feature to Persistent Reservations in SPC-3 for all transports will be proposed separately.]
If Automatic Path Migration is supported and the initiator identifies a second address in its REQ packet during connection establishment, the target shall also record as part of an initiator identifier the second address. If requested by the initiator, the target shall start using that address in place of the original address.

[using APM, the target shall remember both primary and alternate port GUIDs as the initiator device identifiers. After an APM switch, the same queue pair is used so it is transparent to the target. After a complete logout/disconnect, on a new login, if the new primary or alternate matches either of the old primary or alternate, the initiator shall be considered the same. The target shall record the new pair over the old pair for future logouts.]

The target port identifier used by an initiator shall be the target LID, GID, and QPN.
[The target port identifier used by an initiator shall be the initiator's QPN.
The target device identifier is the target's GID.]

[Editor's note: if Reliable Datagrams were supported, then the target's EE Context would need to be retained along with the QPN.]

[Editor's note: need to define the target port identifier also used during persistent reservations?]

The initiator port identifier used by a target and recorded by a target for persistent reservations shall be the initiator LID, GID, and QPN. The initiator device identifier shall serve as the world wide identification also maintained by persistent reservations, used to track initiator port identifier changes.

[Editor's note: Ed Gardner suggests "SRP.T10.NCITS". IBTA TWG needs to assign this string. Boot-capable devices (with Boot ROMs) will advertise "SBWP.IBTA" as well.]
[Should we claim a ServiceID range also? Propose to AWG]
[Reason for ServiceID range not understood. Why not just query the name each time? A range doesn't hurt anything…]

[no plans to authenticate logins]
[should this be the ONLY way to login? Yes. After a SRP LOGOUT, can an initiator send a new SRP LOGIN again without creating a new IB connection? No. SRP LOGOUT means the recipient must do an IB disconnect within a period of time if the source device's IB disconnect has not yet arrived.]

[Editors Option 2:][make this generic to SRP, as part of the login IU itself for all transports]

The PrivateData field of the REQ message includes a 128-bit initiator device identifier. This uniquely identifies an Srp initiator device and may be shared across multiple channel adapters. The PrivateData payload shall be constructed as shown in **Error! Reference source not found.**.

The REP message includes a 128-bit target device identifier. This uniquely identifies an SRP target and may be shared across multiple channel adapters. The PrivateData payload shall be constructed as shown in **Error! Reference source not found.**.

In an IO controller, an initiator or target device identifier shall be constructed by concatenating the IO controller GUID to a unique 64-bit suffix. In a processor node, an initiator or platform GUID shall be constructed by concatenating a unique 64-bit suffix to either a chassis GUID or a port GUID accessible to the platform.
-
[Coordinate the Initiator/Target Identifier definitions with the SNIA Fibre Channel Management work group's definition of platform ID generation algorithm.]

[define a format for the PrivateData – don't just dump the IU into it]

[Add the following tables to SRP main text]

Table 11. Initiator Device Identifier

Byte Bit	7	6	5	4	3	2	2	0
0	TYPE (02H)							
1	RESERVED							
2	RESERVED							
3	RESERVED							
4	(MSB) INITIATOR DEVICE IDENTIFIER MOST SIGNIFICANT 64 BITS (FROM CHASSIS GUID, PORT GUID, OR IO CONTROLLER GUID)							
11								
12	INITIATOR DEVICE IDENTIFIER LEAST SIGNIFICANT 64 BITS							
19								(LSB)

Table 12. Target Device Identifier

Byte Bit	7	6	5	4	3	2	2	0
0	TYPE (03H)							
1	RESERVED							
2	RESERVED							
3	RESERVED							
4	(MSB) TARGET DEVICE IDENTIFIER MOST SIGNIFICANT 64 BITS (FROM CHASSIS GUID, PORT GUID, OR IO CONTROLLER GUID)							
11								
12	TARGET DEVICE IDENTIFIER LEAST SIGNIFICANT 64 BITS							
19								(LSB)

If Automatic Path Migration (APM) is supported, the initiator and/or target may include a second address in its REQ or REP packet that may be used in case of failure of the path to the original address. This address includes an alternate LID and GID.
[note: IB Glossary does not define APM as automatic path migration as it should.]
[make sure Ed includes both initiator device identifier and target device identifier in the login IU exchange]

SRP initiator device's HCA are not required to check inbound RDMA Write transfer lengths (either to verify that the maximum transfer size of 2^31 was not exceeded or to verify that the sum of the packet payloads adds up to the requested transfer size).
<for discussion only, should go away>
[ok, delete it – move to end background material]

[Editor's note: If we want to allow multiple targets within a single IO controller, we need a Service ID range assigned by IBTA. If one target per IO controller suffices, one Service ID suffices. Cris Simpson will push this thru the IBTA TWG.]

[Editor's note: Boot devices are identified this way: An IO controller indicates that it supports the SRP storage protocol for booting by setting the SBWP bit in the IO Controller Profile's Controller Services Capability Mask. See IBA volume 1 release 1.0 Table 222 in section 16.3.3.4. It also sets IOUnitInfo.OptionROM to 1 and provides a ROM image for booting.]

[Editor's note: IBTA AWG may recommend a similar bit for all SRP devices, or leave discovery to the Service ID strings.]

[Editor's note: Ed Gardner would like to define RD too. Need to add a Service ID range for RD, add EE context to the initiator identifier, and make other changes to support it. Ed will write a proposal or defer it until SRP-2.]

End-to-end flow control is not required in either the SRP initiator or target.

[Editor's note: Are Solicited Events needed? Ed thinks maybe... defer until someone asks for it.]

[Editor's note: Should automatic failover support (path migration) be mandatory? We could require the target support responding to automatic path migration requests (Alternate Path Response APR) but not require it to generate them (Load Alternate Path LAP) if it loses contact with the initiator. This way the initiator can force a failover if it wants. This feature would be optional to the initiator. This decision cannot be made until initiator identification is finalized.]

Any Service Level may be negotiated and used. Any Virtual Lanes may be negotiated and used.
[Editor's note: The SL for an SRP connection is chosen by the initiator. The target shall accept any SLs. Should this non-requirement be removed? The issue draws many questions so I suggest mentioning it here.] <remove>

SRP devices shall not set the Fence indicator on any transfers.
[Editor's note: The only case in Volume 1 section 10.8.3 that might break is an RDMA READ followed by a SEND before the RDMA READ completes. This should never occur as the device must wait for data transfer for the RDMA before issuing another SEND (e.g. for an SRP_LOGOUT_REQ IU).]<remove>

[Editor's note: There seems to be little value in listing these non-requirements:

Path MTU size support is device-specific. [all SRP message fit within the minimum size]

SRP devices shall not use multicast operations. [implied by not using unreliable datagrams]

SRP devices are not required to support more than two P_KEYs per port.

SRP devices are not required to support the Bad P_Key Trap and P_Key Violations Counter. [Volume 1 section 10.9.3-4]

SRP devices are not required to check M_Keys for reads or maintain M_Keys through power loss [Volume 1 section 10.9.9]

SRP devices are not required to implement any minimum RNR (resource not ready) NAK Timer field values.

Performance management features are all optional. [Volume 1 Section 16.1.3.2]

]

[Editor's note: be careful of the difference between InfiniBand connection and SRP connection. FCP has a table regarding login/logout effects that might be needed here.]

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.

]

[Editor does not like this table. Allows too many choices. Do we really need to refer to other standards: FC and iSCSI?]
[Table for discussion. If WWUI is chosen then move table from annex to main document]

Table 13 Initiator Device Identifier Options

Name	Size	Description
Process Id	16 bytes	Generated by processor application. Method of generation outside scope of specification.
Channel Adapter GUID	8 bytes	EUI-64 identifying a channel adapter.
Port GUID	8 bytes	EUI-64 identifying a port within a channel adapter.
FC-GS-3 Platform Name	Up to 255 bytes	The first byte indicates the length and the remaining bytes are a text string.
iSCSI World Wide Unique Identifier (WWUI)	Up to 254 bytes	The first byte indicates the length, the second byte indicates the type, and the remaining bytes define the type. The types are: 00 – No Authority (not guaranteed to be unique) 01 – ASCII (using reversed DNS name as Naming Authority) 02 – IEEE EUI-64 03 – Unicode (DNS naming authority) 04 – Generic Binary WWUI (to be considered)
IEEE Registered Extended Format	16 bytes	As used by FC-FS.