

To: T10 Technical Committee
From: Rob Elliott (elliott@hp.com)
Date: 6 June 2007
Subject: 07-263r0 SAM-4 SCSI Initiator Port and Target Port capabilities attributes

Revision history

Revision 0 (6 June 2007) First revision

Related documents

T10/sam4r10 - SCSI Architecture Model - 4 (SAM-4) revision 10
T10/sas2r10 - Serial Attached SCSI - 2 (SAS-2) revision 10
T10/adt2r03 - Automation Device Transport - 2 (ADT-2) revision 3
T10/fcp4r00 - Fibre Channel Protocol - 4 (FCP-4) revision 0
T10/srp-r16a - SCSI RDMA Protocol (SRP) revision 16a
T10/06-341 - SAM-4 Request and Response Fence for protocol services (Rob Elliott, HP)
T10/07-072 - FCP-4 QUERY TASK task management function (Rob Elliott, HP)
T10/07-143 - FCP-4 QUERY TASK SET task management function (Rob Elliott, HP)
T10/07-144 - FCP-4 QUERY UNIT ATTENTION task management function (Rob Elliott, HP)

IETF/RFC 3270 - iSCSI

IETF/draft-ietf-ips-iscsi-impl-guide-08.txt - iSCSI Implementor's Guide revision 8 (by Mallikarjun Chadalapaka, HP). Available as <http://www.ietf.org/internet-drafts/draft-ietf-ips-iscsi-impl-guide-08.txt> through November 2007.

Overview

Each SCSI transport protocol supports a different subset of the features in SAM-4. For example, ADT only supports 16 byte CDBs, while SAS, FCP, ADT, SRP, and iSCSI all support at least 260 bytes (but their maximum sizes are not all identical). The Retry Delay Timer feature is only supported in SAS-2 and FCP-4; it has not yet been added to SRP, iSCSI or ADT.

A SCSI port may not always implement all the features defined by its protocol, particularly as the transport protocols are updated (e.g., FCP-3 to FCP-4). Sometimes the protocol explicitly defines a feature as optional. For example, if Sequence Level Recovery is being used, FCP-4 prohibits using bidirectional commands. The hardware, device driver or operating system may impose constraints as well. For example, most operating systems do not support bidirectional commands even though all the SCSI transport protocols nominally support them.

Conceptually, an application client needs to understand the limits of the SCSI initiator port, and a device server needs to understand the limits of the SCSI target port - both those limits imposed by the SCSI transport protocol it uses and any extra limits imposed by the implementation (e.g., just because the protocol supports bidirectional commands doesn't mean the SCSI port always supports them). These are proposed to be reported in new UML attributes for initiator port and target port objects (defined in the SCSI Initiator Port and SCSI Target Port classes). For the Execute Command () and task management procedure call input and output arguments, attributes are defined to indicate whether certain arguments are supported (e.g., Retry Delay Timer) and indicate the maximum values for other arguments (e.g., CDB size). Attributes indicating other SCSI initiator port information like the SCSI transport protocol are also defined (the port name and the port identifier were already defined as attributes).

This does not mean an HBA (or the controller on a target side) is required to literally implement a data structure containing these values; it just means the HBA and its device driver must understand the limits and reject any attempts by an application client to exceed those limits. For example, if an HBA doesn't support a full 260 byte CDB, it should report an error to the application client rather than silently truncating a CDB, or have a interface that simply does not allow such a CDB to be specified. The T11.5 SM-HBA-2 standard could be enhanced to report some or all of these attributes.

A new Service Response is proposed to represent returning an error to the application client. SAM-4 doesn't really cover target ports returning errors to device servers; that is not changed by this proposal.

An informative annex is included listing the values supported by each of the SCSI transport protocols.

Suggested changes to SAM-4

4.3.3 Request/Response ordering

Request or response transactions are said to be in order if, relative to a given pair of sending and receiving SCSI ports, transactions are delivered in the order they were sent.

A sender may require control over the order in which its requests or responses are presented to the receiver (e.g., the sequence in which requests are received is often important whenever a SCSI initiator device issues a series of commands with the ORDERED [task](#) attribute to a logical unit as described in clause 8). In this case, the order in which these commands are completed, and hence the final state of the logical unit, may depend on the order in which these commands are received. The SCSI initiator device may develop knowledge about the state of pending commands and task management functions and may take action based on the nature and sequence of SCSI target device responses (e.g., a SCSI initiator device should be aware that further responses are possible from an aborted command because the command completion response may be delivered out of order with respect to the abort response).

The manner in which ordering constraints are established is vendor specific. An implementation may delegate this responsibility to the application client (e.g., the device driver). In-order delivery may be an intrinsic property of a service delivery subsystem or a requirement established by the SCSI transport protocol standard.

The order in which task management requests are processed is not specified by the SCSI architecture model. The SCSI architecture model does not require in-order delivery of such requests or processing by the task manager in the order received. To guarantee the processing order of task management requests referencing a specific logical unit, an application client should not have more than one such request pending to that logical unit.

To simplify the description of behavior, the SCSI architecture model assumes in-order delivery of requests or responses to be a property of a service delivery subsystem. This assumption does not constitute a requirement. The SCSI architecture model makes no assumption about and places no requirement on the ordering of requests or responses for different I_T nexuses.

[The SCSI Initiator Port Request Ordering attribute \(see 4.5.7.xx\) indicates to an application client the level of ordering, if any, provided by the SCSI initiator port and its SCSI transport protocol for requests.](#)

[The SCSI Target Port class Response Ordering attribute \(see 4.5.6.xx\) indicates to a device server the level of ordering, if any, provided by the SCSI target port and its SCSI transport protocol for responses.](#)

4.5.5 SCSI Port class

4.5.5.1 SCSI Port class overview

A SCSI Port class (see figure 18) contains the:

- a) SCSI Target Port class (see 4.5.6) that contains the:
 - A) Task Router class (see 4.5.8);
- b) SCSI Initiator Port class (see 4.5.7.1); or
- c) both.

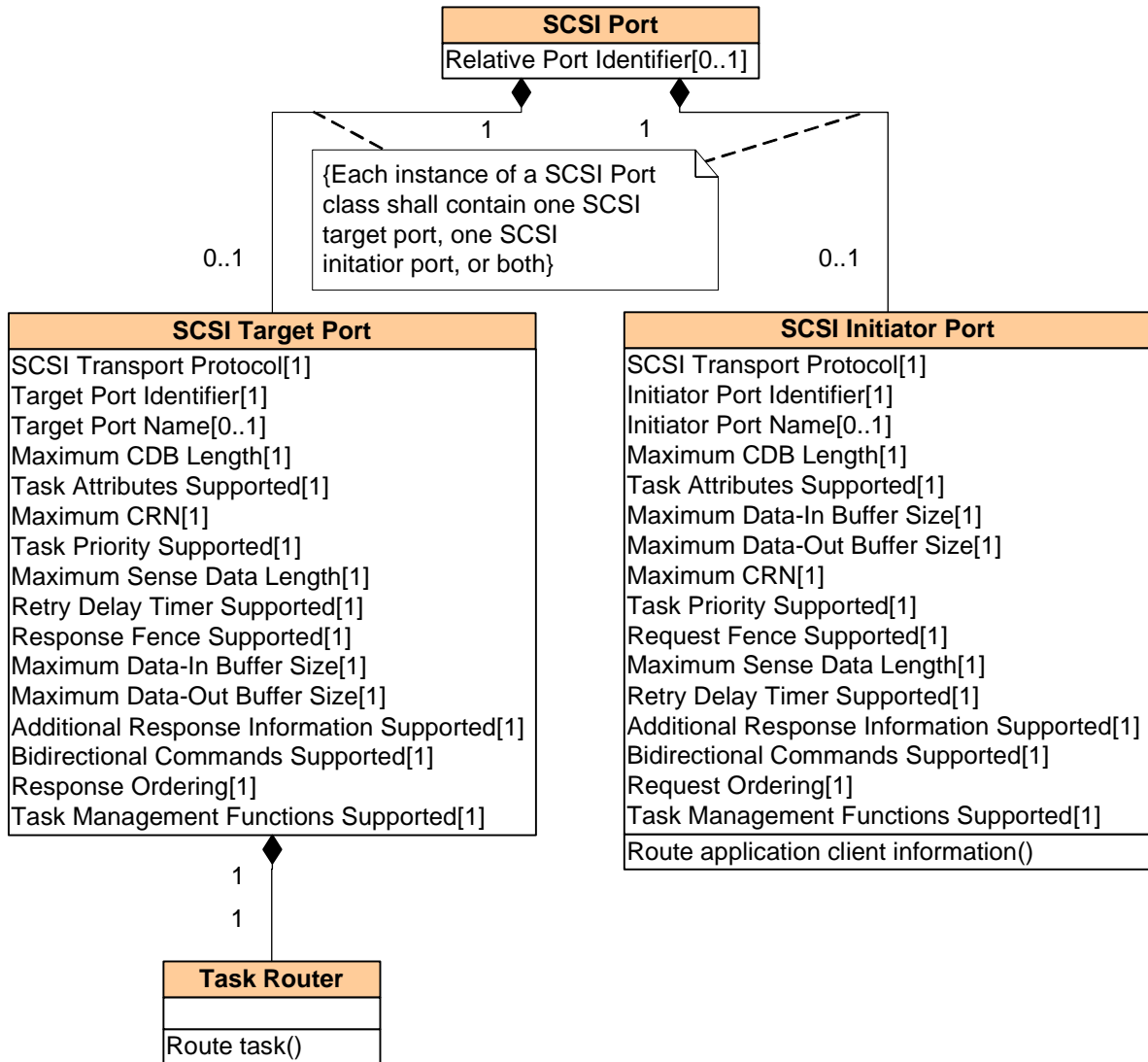


Figure 1 — SCSI Port class diagram [\[modified to add tons of attributes\]](#)

Each instance of a SCSI Port class shall contain:

- a) one SCSI target port that shall contain:
 - A) one task router;
- b) one SCSI initiator port; or
- c) both.

4.5.5.2 Relative Port Identifier attribute

The Relative Port Identifier attribute identifies a SCSI target port or a SCSI initiator port relative to other SCSI ports in a SCSI target device and any SCSI initiator devices contained within that SCSI target device. A SCSI target device may assign relative port identifiers to its SCSI target ports and any SCSI initiator ports. If relative port identifiers are assigned, the SCSI target device shall assign each of its SCSI target ports and any SCSI initiator ports a unique relative port identifier from 1 to 65 535. SCSI target ports and SCSI initiator ports share the same number space.

Relative port identifiers may be retrieved through the Device Identification VPD page (see SPC-3) and the SCSI Ports VPD page (see SPC-3).

The relative port identifiers are not required to be contiguous. The relative port identifier for a SCSI port shall not change once assigned unless physical reconfiguration of the SCSI target device occurs.

4.5.6 SCSI Target Port class

4.5.6.1 SCSI Target Port class overview

A SCSI Target Port class (see figure 18) contains the:

- a) Task Router class (see 4.5.8);

The SCSI Target Port class connects SCSI target devices to a service delivery subsystem.

4.5.6.2 SCSI Transport Protocol attribute

The SCSI Transport Protocol attribute contains the SCSI transport protocol of the SCSI target port. This value may be encoded like the protocol identifier in SPC-4.

4.5.6.3 Target Port Identifier attribute

The Target Port Identifier attribute contains a target port identifier (see 3.1.117) for a SCSI target port. The target port identifier is a value by which a SCSI target port is referenced within a domain.

4.5.6.4 Target Port Name attribute

A Target Port Name attribute contains an optional name (see 3.1.68) of a SCSI target port that is world wide unique within the SCSI transport protocol of the SCSI domain of that SCSI target port. A SCSI target port may have at most one name. A SCSI target port name shall never change and may be used to persistently identify the SCSI target port.

A SCSI transport protocol standard may require that a SCSI target port include a SCSI target port name if the SCSI target port is in a SCSI domain of that SCSI transport protocol. The SCSI target port name may be made available to other SCSI devices or SCSI ports in the given SCSI domain in SCSI transport protocol specific ways.

4.5.6.5 Maximum CDB Length attribute

The Maximum CDB Length attribute contains the maximum length of the Execute Command () and Send SCSI Command Received () CDB argument that is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.6 Task Attributes Supported attribute

The Maximum CDB Length attribute contains the Execute Command () and SCSI Command Received () Task Attribute argument values (see table 38 in 8.6.1) that are supported by the SCSI target port and its SCSI transport protocol.

4.5.6.7 Maximum CRN attribute

The Maximum CRN attribute contains the maximum value of the Execute Command () and SCSI Command Received () CRN argument supported by the SCSI target port and its SCSI transport protocol. A value of zero indicates CRN is not supported.

4.5.6.8 Task Priority Supported attribute

The Task Priority Supported attribute indicates if the Execute Command () and Send SCSI Command Received () Task Priority argument is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.9 Maximum Sense Data Length attribute

The Maximum Sense Data Length attribute contains the maximum value of the Execute Command () and Send Command Complete () Sense Data Length argument that is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.10 Retry Delay Timer Supported attribute

The Retry Delay Timer Supported attribute indicates if the Execute Command () and Send Command Complete () Retry Delay Timer argument is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.11 Response Fence Supported attribute

The Response Fence Supported attribute indicates if the Execute Command () and Send Command Complete () Response Fence argument is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.12 Maximum Data-In Buffer Size attribute

The Maximum Data-In Buffer Size attribute contains:

- a) the maximum value of the Execute Command () Data-In Buffer Size argument; and
- b) the maximum value of the Send Data-In () Application Client Buffer Offset argument plus one (e.g., if the maximum data-in buffer size is 100000000h, then the maximum buffer offset is FFFFFFFFh),

that is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.13 Maximum Data-Out Buffer Size attribute

The Maximum Data-Out Buffer Size attribute contains:

- a) the maximum value of the Execute Command () Data-Out Buffer Size argument; and
- b) the maximum value of the Receive Data-Out () Application Client Buffer Offset argument plus one (e.g., if the maximum data-in buffer size is 100000000h, then the maximum buffer offset is FFFFFFFFh),

that is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.14 Additional Response Information Supported attribute

The Additional Response Information Supported attribute indicates if the task management procedure call and Task Management Function Executed () Additional Response Information argument is supported by the SCSI target port and its SCSI transport protocol.

4.5.6.15 Bidirectional Commands Supported attribute

The Bidirectional Commands Supported attribute indicates if bidirectional commands are supported by the SCSI target port and its SCSI transport protocol.

4.5.6.16 Response Ordering attribute

The Response Ordering attribute indicates the level of response ordering that the SCSI target port and its SCSI transport protocol ensure. The attribute contains one of the values defined in table 1.

Table 1 — SCSI Target Port class Response Ordering attribute

| <u>Value</u> | <u>Description</u> |
|-----------------------|---|
| <u>None</u> | <u>Command and task management function responses may or may not be delivered to the SCSI initiator port in the order in which the device server invoked the Send Command Complete () and Task Management Function Executed () transport protocol service responses.</u> |
| <u>Command</u> | <u>Command responses are delivered to the SCSI initiator port in the order in which the device server invoked the Send SCSI Command () transport protocol service responses.</u> <u>Task management function responses may or may not be delivered to the SCSI initiator port in the order in which the device server invoked the Task Management Function Executed () transport protocol service responses.</u> <u>There is no ordering between commands and task management function responses.</u> |
| <u>Full</u> | <u>Commands and task management function responses are delivered to the SCSI initiator port in the order in which the device server invoked the Send Command Complete () and Task Management Function Executed () transport protocol service responses.</u> |

4.5.6.17 Task Management Functions Supported attribute

The Task Management Functions Supported attribute indicates the task management procedure calls (see table 34 in 7.1) that are supported by the SCSI target port and its SCSI transport protocol.

4.5.7 SCSI Initiator Port class

4.5.7.1 SCSI Initiator Port class overview

The SCSI Initiator Port class:

- a) routes information (e.g., commands and task management functions) between an application client and the services delivery subsystem using the route application client information operation; and
- b) connects SCSI initiator devices to a service delivery subsystem.

4.5.7.2 SCSI Transport Protocol attribute

The SCSI Transport Protocol attribute contains the SCSI transport protocol of the SCSI initiator port. This value may be encoded like the protocol identifier in SPC-4.

4.5.7.3 Initiator Port Identifier attribute

The Initiator Port Identifier attribute contains the initiator port identifier for a SCSI initiator port. The initiator port identifier is a value by which a SCSI initiator port is referenced within a domain.

4.5.7.4 Initiator Port Name attribute

An Initiator Port Name attribute contains an optional name (see 3.1.68) of a SCSI initiator port that is world wideunique within the SCSI transport protocol of the SCSI domain of that SCSI initiator port. A SCSI initiator port may have at most one name. A SCSI initiator port name shall never change and may be used to persistently identify the SCSI initiator port.

A SCSI transport protocol standard may require that a SCSI initiator port include a SCSI initiator port name if the SCSI initiator port is in a SCSI domain of that SCSI transport protocol. The SCSI initiator port name may

be made available to other SCSI devices or SCSI ports in the given SCSI domain in SCSI transport protocol specific ways.

4.5.7.5 Maximum CDB Length attribute

The Maximum CDB Length attribute contains the maximum length of the Execute Command () and Send SCSI Command () CDB argument that is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.6 Task Attributes Supported attribute

The Maximum CDB Length attribute contains the Execute Command () and Send SCSI Command () Task Attribute argument values (see table 38 in 8.6.1) that are supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.7 Maximum Data-In Buffer Size attribute

The Maximum Data-In Buffer Size attribute contains the maximum value of the Execute Command () and Send SCSI Command () Data-In Buffer Size argument that is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.8 Maximum Data-Out Buffer Size attribute

The Maximum Data-Out Buffer Size attribute contains the maximum value of the Execute Command () and Send SCSI Command () Data-Out Buffer Size argument that is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.9 Maximum CRN attribute

The Maximum CRN attribute contains the maximum value of the Execute Command () and Send SCSI Command () CRN argument supported by the SCSI initiator port and its SCSI transport protocol. A value of zero indicates CRN is not supported .

4.5.7.10 Task Priority Supported attribute

The Task Priority Supported attribute indicates if the Execute Command () and Send SCSI Command () Task Priority argument is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.11 Request Fence Supported attribute

The Request Fence Supported attribute indicates if the Execute Command () and Send SCSI Command () Request Fence argument is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.12 Maximum Sense Data Length attribute

The Maximum Sense Data Length attribute contains the maximum value of the Execute Command () and Command Complete Received () Sense Data Length argument that is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.13 Retry Delay Timer Supported attribute

The Retry Delay Timer Supported attribute indicates if the Execute Command () and Command Complete Received () Retry Delay Timer argument is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.14 Additional Response Information Supported attribute

The Additional Response Information Supported attribute indicates if the task management procedure call and Received Task Management Function Executed () Additional Response Information argument is supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.15 Bidirectional Commands Supported attribute

The Bidirectional Commands Supported attribute indicates if bidirectional commands are supported by the SCSI initiator port and its SCSI transport protocol.

4.5.7.16 Request Ordering attribute

The Request Ordering attribute indicates the level of request ordering that the SCSI initiator port and its SCSI transport protocol ensure. The attribute contains one of the values defined in table 2.

Table 2 — SCSI Initiator Port class Request Ordering attribute

| <u>Value</u> | <u>Description</u> |
|----------------|---|
| <u>None</u> | Commands and task management functions may or may not be delivered to the SCSI target port in the order in which the application client invoked the <u>Execute Command ()</u> procedure call and the task management procedure calls (i.e., the <u>Send SCSI Command ()</u> and <u>Send Task Management Request ()</u> transport protocol service requests). |
| <u>Command</u> | <p>Commands are delivered to the SCSI target port in the order in which the application client invoked the <u>Execute Command ()</u> procedure calls (i.e., the <u>Send SCSI Command ()</u> transport protocol service requests).</p> <p>Task management functions may or may not be delivered to the SCSI target port in the order in which the application client invoked the task management procedure calls (i.e., <u>Send Task Management Request ()</u> transport protocol service requests).</p> <p>There is no ordering between commands and task management functions.</p> |
| <u>Full</u> | Commands and task management functions are delivered to the SCSI target port in the order in which the application client invoked the <u>Execute Command ()</u> procedure call and the task management procedure calls (i.e., the <u>Send SCSI Command ()</u> and <u>Send Task Management Request ()</u> transport protocol service requests). |

4.5.7.17 Task Management Functions Supported attribute

The Task Management Functions Supported attribute indicates the task management procedure calls (see table 34 in 7.1) that are supported by the SCSI initiator port and its SCSI transport protocol.

5 SCSI command model

5.1 The Execute Command procedure call

An application client requests the processing of a command by invoking the SCSI transport protocol services described in 5.4, the collective operation of which is modeled in the following procedure call:

**Service Response = Execute Command (IN (I_T_L_Q Nexus, CDB, Task Attribute, [Data-In Buffer Size], [Data-Out Buffer], [Data-Out Buffer Size], [Command Reference Number], [Task Priority]),
OUT ([Data-In Buffer], [Sense Data], [Sense Data Length], Status, [Retry Delay Timer]))**

The SCSI Initiator Port class (see 4.5.7) indicates to an application client the values for the input and output arguments supported by the SCSI initiator port and its SCSI transport protocol.

The SCSI Target Port class (see 4.5.6) indicates to a device server the values for the input and output arguments supported by the SCSI target port and its SCSI transport protocol.

Input arguments:

I_T_L_Q Nexus: The I_T_L_Q nexus identifying the task (see 4.12).

CDB: Command descriptor block (see 5.2).

Task Attribute: A value specifying one of the task attributes defined in 8.6. SCSI transport protocols may or may not provide the ability to specify a different task attribute for each task (see 8.6.1). For a task that processes linked commands, the Task Attribute shall be the value specified for the first command in a series of linked commands. The Task Attribute specified for the second and subsequent commands shall be ignored.

Data-In Buffer Size: The number of bytes available for data transfers to the Data-In Buffer (see 5.4.3). SCSI transport protocols may interpret this argument to include both the size and the location of the Data-In Buffer.

Data-Out Buffer: A buffer containing command specific information to be sent to the logical unit (e.g., data or parameter lists needed to process the command). The buffer size is indicated by the Data-Out Buffer Size argument. The content of the Data-Out Buffer shall not change during the lifetime of the command (see 5.5) as viewed by the application client.

Data-Out Buffer Size: The number of bytes available for data transfers from the Data-Out Buffer (see 5.4.3).

Command Reference Number (CRN): When this argument is used, all sequential commands of an I_T_L nexus shall include a CRN argument that is incremented by one. The CRN shall be set to one for each I_T_L nexus involving the SCSI port after the SCSI port receives a hard reset or detects I_T_L nexus loss. The CRN shall be set to one after it reaches the maximum CRN value supported by the protocol. The CRN value zero shall be reserved for use as defined by the SCSI transport protocol. It is not an error for the application client to provide this argument when CRN is not supported by the SCSI transport protocol or logical unit.

Task Priority: The priority assigned to the task (see 8.7).

Output arguments:

Data-In Buffer: A buffer to contain command specific information returned by the logical unit by the time of command completion. The Execute Command procedure call shall not return a status of GOOD, CONDITION MET, INTERMEDIATE, or INTERMEDIATE-CONDITION MET unless the buffer contents are valid. The application client shall treat the buffer contents as invalid unless the command completes with a status of GOOD, CONDITION MET, INTERMEDIATE, or INTERMEDIATE-CONDITION MET. While some valid data may be present for other values of status, the application client should rely on additional information from the logical unit (e.g., sense data) to determine the state of the buffer contents. If the command ends with a service response of SERVICE DELIVERY OR TARGET FAILURE, the application client shall consider this argument to be undefined.

Sense Data: A buffer containing sense data returned in the same I_T_L_Q nexus transaction (see 3.1.47) as a CHECK CONDITION status (see 5.8.6). The buffer length is indicated by the Sense Data Length argument. If the command ends with a service response of SERVICE DELIVERY OR TARGET FAILURE, the application client shall consider this argument to be undefined.

Sense Data Length: The length in bytes of the Sense Data.

Status: A one-byte field containing command completion status (see 5.3). If the command ends with a service response of SERVICE DELIVERY OR TARGET FAILURE, the application client shall consider this argument to be undefined.

Retry Delay Timer: Additional information about the indicated status code (see 5.3.2).

Service Response assumes one of the following values:

TASK COMPLETE: A logical unit response indicating that the task has ended. The Status argument shall have one of the values specified in 5.3.

ILLEGAL ARGUMENT: [The command has been ended due to an unsupported input argument \(e.g., CDB too long\). All output arguments are invalid.](#)

SERVICE DELIVERY OR TARGET FAILURE: The command has been ended due to a service delivery failure (see 3.1.120) or SCSI target device malfunction. All output ~~parameters~~arguments are invalid.

The SCSI transport protocol events corresponding to a response of TASK COMPLETE or SERVICE DELIVERY OR TARGET FAILURE shall be specified in each SCSI transport protocol standard.

...

5.2 Command descriptor block

The CDB defines the operation to be performed by the device server. [CDB formats are defined in SPC-4.](#)

[The SCSI Initiator Port class Maximum CDB Length attribute \(see 4.5.7.xx\) indicates to an application client the maximum CDB length supported by a SCSI initiator port and its SCSI transport protocol.](#)

[The SCSI Target Port class Maximum CDB Length attribute \(see 4.5.6.xx\) indicates to a device server the maximum CDB length supported by a SCSI target port and its SCSI transport protocol.](#)

For all commands, if the logical unit detects an invalid parameter in the CDB, then the logical unit shall not process the command.

All CDBs shall have an OPERATION CODE [field](#) as the first byte.

Some operation codes provide for modification of their operation based on a service action. In such cases, the combination of operation code value and service action code value may be modeled as a single, unique command determinate. The location of the SERVICE ACTION field in the CDB varies depending on the operation code value.

All CDBs shall contain a CONTROL byte (see table 24). The location of the CONTROL byte within a CDB depends on the CDB format (see SPC-3).

5.3.2 Retry delay timer codes

The retry delay timer codes are specified in table 26 and provide additional information about the reason for the status code.

[table 26]

[The SCSI Initiator Port class Retry Delay Timer Supported attribute \(see 4.5.7.xx\) indicates to an application client if retry delay timer codes are supported by a SCSI initiator port and its SCSI transport protocol.](#)

[The SCSI Target Port class Retry Delay Timer Supported attribute \(see 4.5.6.xx\) indicates to a device server if retry delay timer codes are supported by a SCSI target port and its SCSI transport protocol.](#)

5.4 SCSI transport protocol services in support of Execute Command

5.4.1 Overview

The SCSI transport protocol services that support the Execute Command procedure call are described in 5.4. Two groups of SCSI transport protocol services are described. The SCSI transport protocol services that support the delivery of the command and status are described in 5.4.2. The SCSI transport protocol services that support the data transfers associated with processing a command are described in 5.4.3.

5.4.2 Command and Status SCSI transport protocol services

5.4.2.1 Command and Status SCSI transport protocol services overview

All SCSI transport protocol standards shall define the SCSI transport protocol specific requirements for implementing the **Send SCSI Command** request (see 5.4.2.2), the **SCSI Command Received** indication (see 5.4.2.3), the **Send Command Complete** response (see 5.4.2.4), and the **Command Complete Received** confirmation (see 5.4.2.5) SCSI transport protocol services.

All SCSI initiator devices shall implement the **Send SCSI Command** request and the **Command Complete Received** confirmation SCSI transport protocol services as defined in the applicable SCSI transport protocol standards.

All SCSI target devices shall implement the **SCSI Command Received** indication and the **Send Command Complete** response SCSI transport protocol services as defined in the applicable SCSI transport protocol standards.

5.4.2.2 Send SCSI Command transport protocol service request

An application client uses the Send SCSI Command transport protocol service request to request that a SCSI initiator port send a SCSI command.

Send SCSI Command transport protocol service request:

Send SCSI Command (IN (I_T_L_Q Nexus, CDB, Task Attribute, [Data-In Buffer Size], [Data-Out Buffer], [Data-Out Buffer Size], [CRN], [Task Priority], [First Burst Enabled]))

Input arguments:

I_T_L_Q Nexus: The I_T_L_Q nexus identifying the task (see 4.12).

CDB: Command descriptor block (see 5.2).

Task Attribute: A value specifying one of the task attributes defined in 8.6. For specific requirements on the Task Attribute argument see 5.1.

Data-In Buffer Size: The number of bytes available for data transfers to the Data-In Buffer (see 5.4.3). SCSI transport protocols may interpret the Data-In Buffer Size to include both the size and the location of the Data-In Buffer.

Data-Out Buffer: A buffer containing command specific information to be sent to the logical unit (e.g., data or parameter lists needed to process the command (see 5.1)). The content of the Data-Out Buffer shall not change during the lifetime of the command (see 5.5) as viewed by the application client.

Data-Out Buffer Size: The number of bytes available for data transfers from the Data-Out Buffer (see 5.4.3).

CRN: When CRN is used, all sequential commands of an I_T_L nexus shall include a CRN argument that is incremented by one (see 5.1).

Task Priority: The priority assigned to the task (see 8.7).

First Burst Enabled: An argument specifying that a SCSI transport protocol specific number of bytes from the Data-Out Buffer shall be delivered to the logical unit without waiting for the device server to invoke the Receive Data-Out SCSI transport protocol service.

5.4.2.3 SCSI Command Received transport protocol service indication

A SCSI target port uses the SCSI Command Received transport protocol service indication to notify a device server that it has received a SCSI command.

SCSI Command Received transport protocol service indication:

SCSI Command Received (IN (I_T_L_Q Nexus, CDB, Task Attribute, [CRN], [Task Priority], [First Burst Enabled]))

Input arguments:

I_T_L_Q Nexus: The I_T_L_Q nexus identifying the task (see 4.12).

CDB: Command descriptor block (see 5.2).

Task Attribute: A value specifying one of the task attributes defined in 8.6. For specific requirements on the Task Attribute argument see 5.1.

CRN: When a CRN argument is used, all sequential commands of an I_T_L nexus shall include a CRN argument that is incremented by one (see 5.1).

Task Priority: The priority assigned to the task (see 8.7).

First Burst Enabled: An argument specifying that a SCSI transport protocol specific number of bytes from the Data-Out Buffer are being delivered to the logical unit without waiting for the device server to invoke the Receive Data-Out SCSI transport protocol service.

5.4.2.4 Send Command Complete transport protocol service response

A device server uses the Send Command Complete transport protocol service response to request that a SCSI target port transmit command complete information.

Send Command Complete transport protocol service response:

Send Command Complete (IN (I_T_L_Q Nexus, [Sense Data], [Sense Data Length], Status, Service

Response, [Retry Delay Timer])

Input arguments:

I_T_L_Q Nexus: The I_T_L_Q nexus identifying the task (see 4.12).

Sense Data: If present, a Sense Data argument instructs the SCSI target port to return sense data to the SCSI initiator port (see 5.8.6).

Sense Data Length: The length in bytes of the sense data to be returned to the SCSI initiator port.

Status: Command completion status (see 5.1).

Service Response: Possible service response information for the command (see 5.1).

Retry Delay Timer: The Retry Delay Timer code for the command (see 5.3.2).

5.4.2.5 Command Complete Received transport protocol service confirmation

A SCSI initiator port uses the Command Complete Received transport protocol service confirmation to notify an application client that it has received command complete information.

Command Complete Received transport protocol service confirmation:

Command Complete Received (IN (I_T_L_Q Nexus, [Data-In Buffer], [Sense Data], [Sense Data Length], Status, Service Response, [Retry Delay Timer]))

Input arguments:

I_T_L_Q Nexus: The I_T_L_Q nexus identifying the task (see 4.12).

Data-In Buffer: A buffer containing command specific information returned by the logical unit on command completion (see 5.1).

Sense Data: Sense data returned in the same I_T_L_Q nexus transaction (see 3.1.47) as a CHECK CONDITION status (see 5.8.6).

Sense Data Length: The length in bytes of the received sense data.

Status: Command completion status (see 5.1).

Service Response: Service response for the command (see 5.1).

Retry Delay Timer: The Retry Delay Timer code for the command (see 5.3.2).

5.4.3 Data transfer SCSI transport protocol services**5.4.3.1 Introduction**

The data transfer services described in 5.4.3 provide mechanisms for moving data to and from the SCSI initiator port in response to commands transmitted using the Execute Command procedure call. All SCSI transport protocol standards shall define the protocols required to implement these services.

The application client's Data-In Buffer and/or Data-Out Buffer each appears to the device server as a single, logically contiguous block of memory large enough to hold all the data required by the command (see figure 38). This standard allows either unidirectional or bidirectional data transfer. The processing of a command may require the transfer of data from the application client using the Data-Out Buffer, or to the application client using the Data-In Buffer, or both to and from the application client using both the Data-In Buffer and the Data-Out Buffer.

[The SCSI Initiator Port class Bidirectional Commands Supported attribute \(see 4.5.7.xx\) indicates if bidirectional commands are supported by a SCSI initiator port and its SCSI transport protocol to an application client.](#)

[The SCSI Target Port class Bidirectional Commands Supported attribute \(see 4.5.6.xx\) indicates if bidirectional commands are supported by a SCSI target port and its SCSI transport protocol to a device server.](#)

...

7 Task management functions

7.1 ~~Introduction~~ Task management function procedure calls

An application client requests the processing of a task management function by invoking the SCSI transport protocol services described in 7.12, the collective operation of which is modeled in the following procedure call format:

~~Service Response = Function name (IN (nexus), OUT ([additional response information])~~
Service Response = Function Name (IN (Nexus), OUT ([Additional Response Information])

where:

| | |
|--|---|
| <u>Function Name</u> | <u>is one of the task management function names listed in table 34</u> |
| | <u>is either:</u> |
| <u>Nexus</u> | <u>a) an I T Nexus argument;</u> |
| | <u>b) an I T L Nexus Argument; or</u> |
| | <u>c) an I T L Q Nexus argument</u> |
| <u>Fence</u> | <u>is the Fence input argument described below</u> |
| <u>Additional Response Information</u> | <u>is the Additional Response Information output argument described below</u> |

The task management function names are summarized in table 34.

[table 34]

The SCSI Initiator Port class (see 4.5.7) indicates to an application client the values for the input and output arguments supported by the SCSI initiator port and its SCSI transport protocol.

The SCSI Target Port class (see 4.5.6) indicates to a device server the values for the input and output arguments supported by the SCSI target port and its SCSI transport protocol.

Input arguments:

~~**Nexus:** An I_T nexus, I_T_L nexus, or I_T_L_Q nexus (see 4.7) identifying the task or tasks affected by the task management function.~~

~~**I_T Nexus:** A SCSI initiator port and SCSI target port nexus (see 4.7).~~

~~**I_T_L Nexus:** A SCSI initiator port, SCSI target port, and logical unit nexus (see 4.7).~~

~~**I_T_L_Q Nexus:** A SCSI initiator port, SCSI target port, logical unit, and task tag nexus (see 4.7).~~

I T Nexus: The I T nexus (see 4.7) affected by the task management function.

I T L Nexus: The I T L nexus (see 4.7) affected by the task management function.

I T L Q Nexus: The I T L Q nexus (see 4.7) affected by the task management function.

Output arguments:

Additional Response Information: If supported by the transport protocol and the logical unit, then three bytes that are returned along with the service response for certain task management functions (e.g., QUERY UNIT ATTENTION). Transport protocols may or may not support the Additional Response Information argument. A transport protocol supporting the Additional Response Information argument may or may not require that logical units accessible through a target port using that transport protocol support the Additional Response Information argument..

~~One of the following SCSI transport protocol specific responses shall be returned:~~

Service Response assumes one of the following values:

FUNCTION COMPLETE: A task manager response indicating that the requested function is complete. Unless another response is required, the task manager shall return this response upon completion of a task management request supported by the logical unit or SCSI target device to which the request was directed.

FUNCTION SUCCEEDED: An optional task manager response indicating that the requested function is supported and completed successfully. This task manager response shall only be used by functions

that require notification of success (e.g., QUERY TASK).

FUNCTION REJECTED: An task manager response indicating that the requested function is not supported by the logical unit or SCSI target device to which the function was directed.

INCORRECT LOGICAL UNIT NUMBER: An optional task router response indicating that the function requested processing for an incorrect logical unit number.

ILLEGAL ARGUMENT: The command has been ended due to an unsupported input argument. All output arguments are invalid.

SERVICE DELIVERY OR TARGET FAILURE: The request was terminated due to a service delivery failure (see 3.1.120) or SCSI target device malfunction. The task manager may or may not have successfully performed the specified function. All output arguments are invalid.

Each SCSI transport protocol standard shall define the events comprising each of these service responses.

The task manager response to task management requests is subject to the presence of access restrictions, as managed by ACCESS CONTROL OUT and ACCESS CONTROL IN commands (see SPC-3), as follows:

- a) A task management request of ABORT TASK, ABORT TASK SET, CLEAR ACA, I_T NEXUS RESET, or QUERY TASK shall not be affected by the presence of access restrictions;
- b) A task management request of CLEAR TASK SET or LOGICAL UNIT RESET received from a SCSI initiator port that is denied access to the logical unit, either because it has no access rights or because it is in the pending-enrolled state, shall not cause any changes to the logical unit; and c) The task management function service response shall not be affected by the presence of access restrictions.

7.2 ABORT TASK

RequestProcedure call:

Service Response = ABORT TASK (IN (I_T_L_Q Nexus))

Description:

This function shall be supported by all logical units.

The task manager shall abort the specified task, if any, as described in 5.6.2. Previously established conditions, including MODE SELECT parameters, reservations, and ACA shall not be changed by the ABORT TASK function.

A response of FUNCTION COMPLETE shall indicate that the task was aborted or was not in the task set. In either case, the SCSI target device shall guarantee that no further requests or responses are sent from the task. All SCSI transport protocol standards shall support the ABORT TASK task management function.

7.3 ABORT TASK SET

RequestProcedure call:

Service Response = ABORT TASK SET (IN (I_T_L Nexus))

Description:

This function shall be supported by all logical units.

The task manager shall abort all tasks in the task set that were received on the specified I_T nexus as described in 5.6. Tasks received on other I_T nexuses or in other task sets shall not be aborted. This task management function performed is equivalent to a series of ABORT TASK requests.

Other previously established conditions, including MODE SELECT parameters, reservations, and ACA shall not be changed by the ABORT TASK SET function.

All SCSI transport protocol standards shall support the ABORT TASK SET task management function.

7.4 CLEAR ACA

RequestProcedure call:

Service Response = CLEAR ACA (IN (I_T_L Nexus))

Description:

This function shall be supported by a logical unit if it supports ACA (see 5.2).

For the CLEAR ACA task management function, the task set shall be the one defined by the TST field in the Control mode page (see SPC-3).

An application client requests a CLEAR ACA using the faulted I_T nexus (see 3.1.38) to clear an ACA condition from the task set serviced by the logical unit. The state of all tasks in the task set shall be modified as described in 8.8. For a task with the ACA task attribute (see 8.6.5) receipt of a CLEAR ACA function shall have the same effect as receipt of an ABORT TASK function (see 7.2) specifying that task. If successful, this function shall be terminated with a service response of FUNCTION COMPLETE.

If the task manager clears the ACA condition, any task within that task set may be completed subject to the requirements for task set management specified in clause 8.

The service response for a CLEAR ACA request received from an I_T nexus other than the faulted I_T nexus shall be FUNCTION REJECTED.

All SCSI transport protocol standards shall support the CLEAR ACA task management function.

7.5 CLEAR TASK SET

Request Procedure call:

Service Response = CLEAR TASK SET (IN (I_T_L Nexus))

Description:

This function shall be supported by logical units.

For the CLEAR TASK SET task management function, the task set shall be the one defined by the TST field in the Control mode page (see SPC-3).

All tasks in the task set shall be aborted as described in 5.6.

All pending status and sense data for the task set shall be cleared. Other previously established conditions, including MODE SELECT parameters, reservations, and ACA shall not be changed by the CLEAR TASK SET function.

All SCSI transport protocol standards shall support the CLEAR TASK SET task management function.

7.6 I_T NEXUS RESET

Request Procedure call:

Service Response = I_T NEXUS RESET (IN (I_T Nexus))

Description:

SCSI transport protocols may or may not support I_T NEXUS RESET and may or may not require logical units accessible through SCSI target ports using such transport protocols to support I_T NEXUS RESET.

Each logical unit accessible through the SCSI target port shall perform the I_T nexus loss functions specified in 6.3.4 for the I_T nexus on which the function request was received, then the SCSI target device shall return a FUNCTION COMPLETE response. After returning a FUNCTION COMPLETE response, the logical unit(s) and the SCSI target port shall perform any additional functions specified by the SCSI transport protocol.

7.7 LOGICAL UNIT RESET

Request Procedure call:

Service Response = LOGICAL UNIT RESET (IN (I_T_L Nexus))

Description:

This function shall be supported by all logical units.

Before returning a FUNCTION COMPLETE response, the logical unit shall perform the logical unit reset functions specified in 6.3.3.

NOTE 11 - Previous versions of this standard only required LOGICAL UNIT RESET support in logical units that supported hierarchical logical units.

All SCSI transport protocol standards shall support the LOGICAL UNIT RESET task management function.

7.8 QUERY TASK

Request Procedure call:

Service Response = QUERY TASK (IN (I_T_L_Q Nexus))

Description:

SCSI transport protocols may or may not support QUERY TASK and may or may not require logical units accessible through SCSI target ports using such transport protocols to support QUERY TASK.

The task manager in the specified logical unit shall:

- a) if the specified task is present in the task set, then return a service response set to FUNCTION SUCCEEDED; and
- b) if the specified task is not present in the task set, then return a service response set to FUNCTION COMPLETE.

7.9 QUERY TASK SET

Request Procedure call:

Service Response = QUERY TASK SET (IN (I_T_L Nexus))

Description:

SCSI transport protocols may or may not support QUERY TASK SET and may or may not require logical units accessible through SCSI target ports using such transport protocols to support QUERY TASK SET.

The task manager in the specified logical unit shall:

- a) if there is any task present in the task set from the specified I_T nexus, then return a service response set to FUNCTION SUCCEEDED; and
- b) if there is no task present in the task set from the specified I_T nexus, then return a service response set to FUNCTION COMPLETE.

7.10 QUERY UNIT ATTENTION

Request Procedure call:

Service Response = QUERY UNIT ATTENTION (IN (I_T_L Nexus), OUT ([Additional Response Information]))

Description:

A SCSI transport protocol may or may not support QUERY UNIT ATTENTION. A SCSI transport protocol supporting QUERY UNIT ATTENTION may or may not require logical units accessible through SCSI target ports using that transport protocol to support QUERY UNIT ATTENTION.

The task manager in the specified logical unit shall:

- a) if there is a unit attention condition (see 5.8.7) or a deferred error (see SPC-3) pending for the specified I_T nexus, then return a service response set to FUNCTION SUCCEEDED; and
- b) if there is no unit attention condition or deferred error pending for the specified I_T nexus, then return a service response set to FUNCTION COMPLETE.

If the service response is not FUNCTION SUCCEEDED, then the task manager shall set the Additional Response Information argument to 000000h.

If the service response is FUNCTION SUCCEEDED, the task manager shall set the Additional Response Information argument as defined in table 35.

7.11 Task management function lifetime

The task manager shall create a task management function upon receiving a Task Management Request Received indication (see 7.12). The task management function shall exist until:

...

7.12 Task management SCSI transport protocol services

7.12.1 Task management SCSI transport protocol services overview

The SCSI transport protocol services described in this subclause are used by a SCSI initiator device and SCSI target device to process a task management procedure call. The following arguments are passed:

Nexus: An I_T nexus, I_T_L nexus, or I_T_L_Q nexus (see 4.12).

Function Identifier: Argument encoding the task management function to be performed.

All SCSI transport protocol standards shall define the SCSI transport protocol specific requirements for implementing the **Send Task Management Request** request (see 7.10.2), the **Task Management Request Received** indication (see 7.10.3), the **Task Management Function Executed** response (see 7.10.4), and the **Received Task Management Function Executed** (see 7.10.5) confirmation (see 7.10.5) SCSI transport protocol services.

A SCSI transport protocol standard may specify different implementation requirements for the **Send Task Management Request** request SCSI transport protocol service for different values of the Function Identifier argument.

All SCSI initiator devices shall implement the **Send Task Management Request** request and the **Received Task Management Function Executed** confirmation SCSI transport protocol services as defined in the applicable SCSI transport protocol standards.

All SCSI target devices shall implement the **Task Management Request Received** indication and the **Task Management Function Executed** response SCSI transport protocol services as defined in the applicable SCSI transport protocol standards.

7.12.2 Send Task Management Request transport protocol service request

An application client uses the Send Task Management Request transport protocol service request to request that a SCSI initiator port send a task management request.

Send Task Management Request transport protocol service request:

Send Task Management Request (IN (Nexus, Function Identifier))

Input arguments:

Nexus: An I_T nexus, I_T_L nexus, or I_T_L_Q nexus (see 4.12).

Function Identifier: Argument encoding the task management function to be performed.

7.12.3 Task Management Request Received transport protocol service indication

A SCSI target port uses the Task Management Request Received transport protocol service indication to notify a task manager that it has received a task management request.

Task Management Request Received transport protocol service indication:

Task Management Request Received (IN (Nexus, Function Identifier))

Input arguments:

Nexus: An I_T nexus, I_T_L nexus, or I_T_L_Q nexus (see 4.12).

Function Identifier: Argument encoding the task management function to be performed.

7.12.4 Task Management Function Executed transport protocol service response

A task manager uses the [Send Task Management Request](#) [Task Management Function Executed](#) transport protocol service response to request that a SCSI target port transmit task management function executed information.

Task Management Function Executed transport protocol service response:

Task Management Function Executed (IN (Nexus, Service Response, [\[Additional Response Information\]](#)))

Input arguments:

Nexus: An I_T nexus, I_T_L nexus, or I_T_L_Q nexus (see 4.12).

Service Response: An encoded value representing one of the following:

FUNCTION COMPLETE: The requested function has been completed.

FUNCTION SUCCEEDED: The requested function is supported and completed successfully.

FUNCTION REJECTED: The task manager does not implement the requested function.

INCORRECT LOGICAL UNIT NUMBER: An optional task router response indicating that the function requested processing for an incorrect logical unit number.

SERVICE DELIVERY OR TARGET FAILURE: The request was terminated due to a service delivery failure (see 3.1.112) or SCSI target device malfunction. The task manager may or may not have successfully performed the specified function.

[Additional Response Information:](#) [The Additional Response Information output argument for the task management procedure call \(see 7.1\):](#)

7.12.5 Received Task Management Function Executed transport protocol service confirmation

A SCSI initiator port uses the Received Task Management Function Executed transport protocol service confirmation to notify an application client that it has received task management function executed information.

Received Task Management Function Executed transport protocol service confirmation:

Received Task Management Function Executed (IN (Nexus, Service Response, [\[Additional Response Information\]](#)))

Input arguments:

Nexus: An I_T nexus, I_T_L nexus, or I_T_L_Q nexus (see 4.12).

Service Response: An encoded value representing one of the following:

FUNCTION COMPLETE: The requested function has been completed.

FUNCTION SUCCEEDED: The requested function is supported and completed successfully.

FUNCTION REJECTED: The task manager does not implement the requested function.

INCORRECT LOGICAL UNIT NUMBER: An optional task router response indicating that the function requested processing for an incorrect logical unit number.

SERVICE DELIVERY OR TARGET FAILURE: The request was terminated due to a service delivery failure (see 3.1.112) or SCSI target device malfunction. The task manager may or may not have successfully performed the specified function.

[Additional Response Information:](#) [The Additional Response Information output argument for the task management procedure call \(see 7.1\):](#)

Each SCSI transport protocol shall allow a **Received Task Management Function Executed** confirming completion of the requested task to be associated with the corresponding **Send Task Management Request**.

8.6 Task attributes

The application client shall assign a task attribute (see table 38) to each task.

SCSI transport protocols shall provide the capability to specify a unique task attribute for each task.

[The SCSI Initiator Port class Task Attributes Supported attribute \(see 4.5.7.xx\) indicates to an application client the task attributes supported by a SCSI initiator port and its SCSI transport protocol. The Extended](#)

[INQUIRY Data VPD page \(see SPC-4\) and the standard INQUIRY data NORMACA bit indicate to an application client the task attributes supported by a SCSI target port and its SCSI transport protocol](#)

[The SCSI Target Port class Task Attributes Supported attribute \(see 4.5.6.xx\) indicates to a device server if bidirectional the task attributes supported by a SCSI target port and its SCSI transport protocol.](#)

Annex B

(informational)

SCSI Initiator Port attributes per SCSI transport protocol [all new]

Table B.1 lists the values of the SCSI Initiator Port attributes that a SCSI initiator port using each different SCSI transport protocol is able to return, and the values of the SCSI Target Port attributes that a SCSI target port using that SCSI transport protocol is able to return.

Table B.1 — SCSI Initiator Port and SCSI Target Port attributes by SCSI transport protocol (part 1 of 2)

| Attribute | ADT-2 | FCP-4 | iSCSI | SAS-2 | SRP |
|--|---|-----------|--------|-------------------|-----------|
| Maximum CDB Length (in bytes) ^a | 16 | 268 | 65 550 | 268 | 268 |
| Task Attributes Supported | SIMPLE, HEAD OF QUEUE, ORDERED, and ACA | | | | |
| Maximum Data-In Buffer Size (in bytes) | FFFFFFFFh | | | | |
| Maximum Data-Out Buffer Size (in bytes) | FFFFFFFFh | | | | |
| Maximum CRN ^b | zero | FFh | zero | zero | zero |
| Task Priority Supported | no | yes | no | yes | no |
| Request Fence Supported ^h and Response Fence Supported ⁱ | no | no | yes | yes | no |
| Maximum Sense Data Length (in bytes) ^c | FFFFh | FFFFFFFFh | FFFFh | 3E8h ^e | FFFFFFFFh |
| Retry Delay Timer Supported | no | yes | no | yes | no |
| Additional Response Information Supported | no | yes | no | yes | no |

^a SPC-4 defines the maximum length of a CDB as being 260 bytes.

^b A Maximum CRN of zero indicates that CRN is not supported.

^c SPC-4 defines the maximum length of sense data as being 252 bytes.

^d The task management function name is not an argument to a procedure call; it is the name of the procedure call itself.

^e 3E8h represents 1 000 bytes, which keeps the sense data in one RESPONSE frame.

^f FCP-4 ensures commands are delivered in order if CRN is used.

^g SAS-2 ensures commands and task management functions are delivered in order if there is only one connection at a time for an I_T nexus.

^h SCSI Initiator Port attribute only.

ⁱ SCSI Target Port attribute only.

Table B.1 — SCSI Initiator Port and SCSI Target Port attributes by SCSI transport protocol (part 2 of 2)

| Attribute | ADT-2 | FCP-4 | iSCSI | SAS-2 | SRP |
|--|---|------------------------------|---|---------------------------|---|
| Bidirectional Commands Supported | yes | | | | |
| Request Ordering | None | Command ^f or None | Full | Full ^g or None | None |
| Task Management Functions Supported ^d | ABORT TASK, ABORT TASK SET, CLEAR TASK SET, LOGICAL UNIT RESET, CLEAR ACA, QUERY TASK | All | ABORT TASK, ABORT TASK SET, CLEAR TASK SET, LOGICAL UNIT RESET, CLEAR ACA | All | ABORT TASK, ABORT TASK SET, CLEAR TASK SET, LOGICAL UNIT RESET, CLEAR ACA |
| <p>^a SPC-4 defines the maximum length of a CDB as being 260 bytes.</p> <p>^b A Maximum CRN of zero indicates that CRN is not supported.</p> <p>^c SPC-4 defines the maximum length of sense data as being 252 bytes.</p> <p>^d The task management function name is not an argument to a procedure call; it is the name of the procedure call itself.</p> <p>^e 3E8h represents 1 000 bytes, which keeps the sense data in one RESPONSE frame.</p> <p>^f FCP-4 ensures commands are delivered in order if CRN is used.</p> <p>^g SAS-2 ensures commands and task management functions are delivered in order if there is only one connection at a time for an I_T nexus.</p> <p>^h SCSI Initiator Port attribute only.</p> <p>ⁱ SCSI Target Port attribute only.</p> | | | | | |

Editor's Note 1: ADT-2 is expected to add QUERY TASK SET and QUERY UNIT ATTENTION before it is published.

Editor's Note 2: iSCSI is likely to add Task Priority, Retry Delay Timer, QUERY TASK SET, and QUERY UNIT ATTENTION via a "SAM-4 enhancements" RFC in the future

[\[end of all new section\]](#)