WORKING DRAFT

NCITS T10 Project 999D

Revision 10a
March 4, 1998

Information Technology— SCSI-3 Medium Changer Commands (SMC)

This is a draft proposed American National Standard of the National Committee for Information Technology Standards. As such this is not a completed standard. The T10 Technical Committee may modify this document as a result of comments received during the public review process and the approval of this document as a standard. Use of the information contained herein is at your own risk.

Permission is granted to members of NCITS and ISO, their technical committees, and their associated task groups to reproduce this document for the purposes of NCITS and ISO standardization activities without further permission, provided this notice is included. All other rights are reserved. Any commercial or for-profit use is strictly prohibited.

SMC Technical Editor:

Erich Oetting
Storage Technology Corporation
MS 4223
2270 South 88th St.
Louisville, CO, 80028-4223

Telephone: (303) 673-2178
Facsimile: (303) 673-2568
Email: erich_oetting@stortek.com

Reference number
ISO/IEC **** : 199x
ANSI NCITS 314 – 199x
Printed March 4, 1998

Points of Contact:

T10 Chair
John B. Lohmeyer
Symbios Logic, Inc.
4420 Arrows West Drive
Colorado Springs, CO 80907-3444

Tel: (719) 533-7560
Fax: (719) 533-7036
Email: john.lohmeyer@symbios.com

T10 Vice–Chair
Lawrence J. Lamers
Adaptec
691 South Milpitas Blvd.
Milpitas, CA 95035

Tel: (408) 957-7817
Fax: (408) 957-7193
Email: ljlamers@aol.com

NCITS Secretariat
NCITS Secretariat
1250 Eye Street, NW Suite 200
Washington, DC 20005

Telephone: (202) 737-8888
Facsimile: (202) 638-4922
Email: ncits@itic.nw.dc.us
Web: http://www.ncits.org

T10 Electronic Information

Internet address for subscription to the T10 reflector:	majordomo@symbios.com
Internet address for distribution via T10 reflector:	t10@symbios.com
T10 Web page:	http://www.symbios.com/t10
SCSI BBS:	(719) 533-7950

Document Distribution

Global Engineering
15 Inverness Way East
Englewood, CO 80112-5704

Telephone: (800) 854-7179 or (303) 792-2181
Facsimile: (303) 792-2192

ANSI
NCITS 314–199x

American National Standard
for Information Technology —

SCSI-3 Medium Changer Commands (SMC)

Secretariat
Information Technology Industry Council

Approved mm dd yy
American National Standards Institute, Inc.

ABSTRACT

This standard defines the SCSI commands and model for independent medium changer devices and attached medium changer functions integrated into other SCSI devices.

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer. Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that effort be made towards their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give interpretation on any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

CAUTION NOTICE: This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken periodically to reaffirm, revise, or withdraw this standard. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

CAUTION: The developers of this standard have requested that holders of patents that may be required for the implementation of the standard, disclose such patents to the publisher. However, neither the developers nor the publisher have undertaken a patent search in order to identify which, if any, patents may apply to this standard.

As of the date of publication of this standard and following calls for the identification of patents that may be required for the implementation of the standard, no such claims have been made. No further patent search is conducted by the developer or the publisher in respect to any standard it processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this standard.

Foreword

The SCSI-3 Medium Changer Commands (SMC) standard specifies the commands and external behavioral characteristics of a device server that declares itself a medium changer in the device type field of the INQUIRY command response data. This standard also specifies the behavior of the attached medium changer commands available when the MCHNGR bit is set to one in INQUIRY command response data.

SMC is specified independent of any service delivery subsystem used to carry commands, command parameter data, command response data and status. The SMC standard conforms to the requirements specified in the SCSI-3 Architecture Model (SAM) standard.

With any technical document there may arise questions of interpretation as new products are implemented. The NCITS committee has established procedures to issue technical opinions concerning the standards developed by the NCITS organization. These procedures may result in SCSI Technical Information Bulletins being published by NCITS.

These bulletins, while reflecting the opinion of the Technical Committee which developed the standard, are intended solely as supplementary information to users of the standard. This standard, ANSI NCITS.314-199x, as approved through the publication and voting procedures of the American National Standards Institute, is not altered by these bulletins. Any subsequent revision to this standard may or may not reflect the contents of these Technical Information Bulletins.

Current NCITS practice is to make Technical Information Bulletins available through:

Global Engineering
15 Inverness Way East
Englewood, CO 80112-5704

Telephone: (303) 792-2181 or
(800) 854-7179

Facsimile: (303) 792-2192

Introduction

The SCSI-3 Medium Changer Command Set (SMC) standard is divided into seven clauses:

- Clause 1 is the scope.
- Clause 2 enumerates the normative and informative references that apply to this standard
- Clause 3 describes definitions, symbols, abbreviations and conventions used in this standard.
- Clause 4 is an overview of this standard.
- Clause 5 describes the model for this device class.
- Clause 6 describes the commands and responses.
- Clause 7 describes the parameters.
- Annex A is the bibliography.

Contents

Figures

Tables

Annexes

American National Standard for Information Technology - SCSI-3 Medium Changer Commands (SMC)

1 Scope

This standard defines the command set extensions for operation of SCSI medium changer devices, and command set extensions that allow medium changer functions in other types of SCSI devices.

The objectives of the SCSI-3 Medium Changer Commands standard are:

- To permit an application client to communicate with a logical unit that declares itself to be a medium changer device in the DEVICE TYPE field of the INQUIRY command response data over a SCSI service delivery subsystem.
- To permit an application client to access the medium changer functions in a logical unit that sets the MCHNGR bit in INQUIRY command response data.
- To define commands to manage the operation of SCSI medium changer devices.

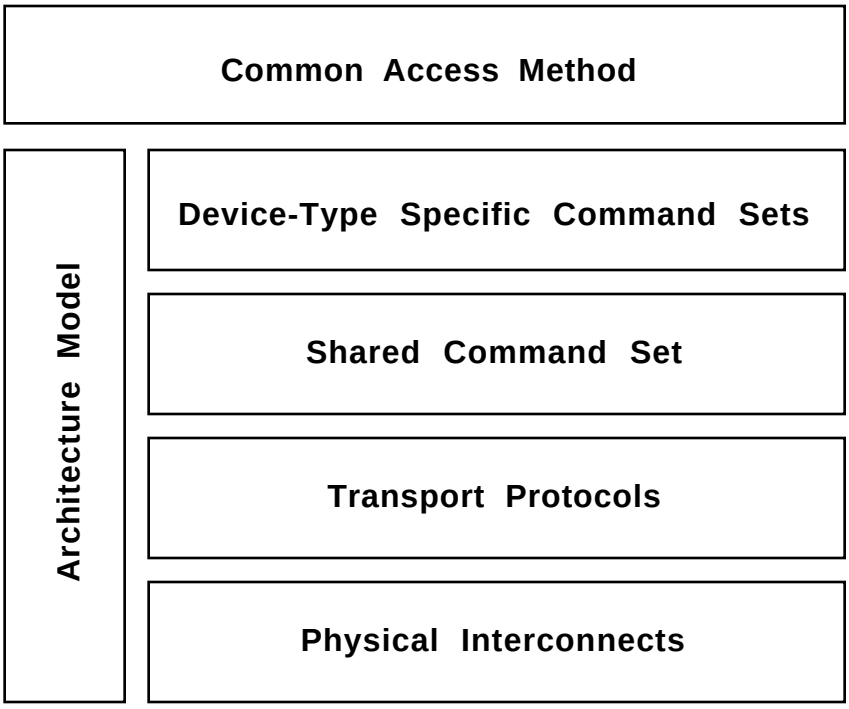

Figure 1 — General structure of SCSI standards

Figure 1 is intended to show the general structure of SCSI-3 standards. The figure is not intended to imply a relationship such as a hierarchy, protocol stack, or system architecture.

At the time this standard was generated examples of the SCSI-3 general structure included:

Physical Interconnects:

Fibre Channel Arbitrated Loop [NCITS T11/960D]
Fibre Channel – Physical and Signaling Interface [X3.230-1994/AM1-1996]
High Performance Serial Bus [ANSI/IEEE 1394–1995]
SCSI-3 Parallel Interface [X3.253-1995]
SCSI-3 Fast-20 Parallel Interface [X3.277-1996]
SCSI Parallel Interface – 2 [NCITS T10/1142D]
Serial Storage Architecture Physical Layer 1 [X3.293-1996]
Serial Storage Architecture Physical Layer 2 [NCITS.307]

Transport Protocols:

SCSI-3 Interlocked Protocol [X3.292-1997]
Serial Storage Architecture Transport Layer 1 [X3.295-1996]
SCSI-3 Fibre Channel Protocol [X3.269-1996]
SCSI-3 Fibre Channel Protocol - 2 [NCITS T10/1144D]

SCSI Serial Bus Protocol – 2 [NCITS T10/1155D]
Serial Storage Architecture SCSI-3 Protocol [NCITS.309]
Serial Storage Architecture Transport Layer 2 [NCITS.308]

Shared Command Set:

SCSI-3 Primary Commands [X3.301-1997]

Device-Type Specific Command Sets:

SCSI-3 Block Commands [NCITS.306]
SCSI-3 Enclosure Services [NCITS.305]
SCSI-3 Stream Commands [NCITS T10/997D]
SCSI-3 Medium Changer Commands (this standard)
SCSI-3 Controller Commands [X3.276-1997]
SCSI-3 Controller Commands - 2 [NCITS T10/1255D]
SCSI-3 Multimedia Command Set [X3.304]
SCSI-3 Multimedia Command Set – 2 [NCITS T10/1228D]

Architecture Model:

SCSI-3 Architecture Model [X3.270-1996]
SCSI-3 Architecture Model - 2 [NCITS T10/1157D]

Common Access Method:

SCSI Common Access Method [X3.232-1996]
SCSI Common Access Method - 3 [NCITS T10/990D]

The term SCSI is used wherever it is not necessary to distinguish between the versions of SCSI. The Small Computer System Interface–2 [X3.131-1994] is referred to herein as SCSI–2. The term SCSI-3 in this standard refers to versions of SCSI defined since SCSI–2.

2 Normative References

The following standards contain provisions that, through reference in the text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Copies of the following documents can be obtained from ANSI: Approved ANSI standards, approved and draft international and regional standards (ISO, IEC, CEN/CENELEC, ITUT), and approved standards of other countries (including BSI, JIS, and DIN). For further information, contact ANSI Customer Service Department at 212–642–4900 (telephone), 212–302–1286 (fax) or via the World Wide Web at http://www.ansi.org.

ANSI X3.270-1996, *Information Technology – SCSI-3 Architecture Model*
ANSI X3.301-1997, *Information Technology – SCSI-3 Primary Commands*

3 Definitions, symbols and abbreviations

3.1 Definitions

This clause contains a glossary of special terms used in this standard. These terms apply to SMC and do not constitute a comprehensive glossary for SCSI-3.

3.1.1 additional sense code: The value in the ADDITIONAL SENSE CODE and ADDITIONAL SENSE CODE QUALIFIER fields of REQUEST SENSE command response data (see SPC).

3.1.2 application client: An object that is the source of SCSI commands. Further definition of an application client may be found in the SCSI-3 Architecture Model (SAM).

3.1.3 attached medium changer: A medium changer that accepts commands issued to the same logical unit as a primary SCSI device that is not a medium changer.

3.1.4 autosense data: The sense data that is automatically delivered to the application client by the device server in a protocol-specific manner when a command completes with a CHECK CONDITION or COMMAND TERMINATED status (see SAM).

3.1.5 byte: Indicates an 8-bit construct.

3.1.6 command: A request describing a unit of work to be performed by a device server. A detailed definition of a command may be found in SAM.

3.1.7 command descriptor block: The structure up to 16 bytes in length used to communicate commands from an application client to a device server.

3.1.8 data transfer element: A component of a medium changer used to access the data stored on a volume. The address in medium changer element space of a primary device.

3.1.9 device service request: A request, submitted by an application client, conveying a SCSI command to a device server. A detailed definition of a device service request may be found in SAM.

3.1.10 device type: The type of device (or device model) implemented by the device server.

3.1.11 element: An addressable physical component of a medium changer device that can serve as the location of a removable unit of data storage medium.

3.1.12 extent: An extent is a specified number of logical blocks, typically identified by a starting logical block address and a count of the number of blocks in the extent.

3.1.13 field: A group of one or more contiguous bits.

3.1.14 hard reset: A target response to a reset event or TARGET RESET task management function. A detailed definition of hard reset may be found in SAM.

3.1.15 host: A device with the characteristics of a primary computing device, typically a personal computer, workstation, minicomputer, mainframe computer, or auxiliary computing device or server. A host includes application clients, one or more initiators, and may in some cases take on a target role.

3.1.16 import/export element: A location within a medium changer device that can be accessed by both the medium transport elements and by the operator or an external device.

3.1.17 independent medium changer: A medium changer addressed as a separate SCSI device or logical unit.

3.1.18 initiator: A SCSI device containing application clients that originate device service requests to be processed in a device server. A detailed definition of an initiator may be found in SAM.

3.1.19 linked command: One in a series of SCSI commands executed by a single task, which collectively make up a discrete I/O operation. A detailed definition of a linked command may be found in SAM.

3.1.20 logical unit: An externally addressable entity within a target that implements a SCSI device model and contains a device server. A detailed definition of a logical unit may be found in SAM.

3.1.21 logical unit number: An encoded 64-bit identifier for a logical unit. A detailed definition of a logical unit number may be found in SAM.

3.1.22 medium: One unit of media, equivalent to volume as defined in this standard.

3.1.23 medium changer: A medium changer mechanizes the movement of media to and from the device that records on or reads from the media.

3.1.24 medium transport element: A component of a medium changer device that is used to move volumes.

3.1.25 one: Value of 1, the logical true condition of a variable.

3.1.26 page: Several commands use regular parameter structures that are referred to as pages. These pages are identified with a value known as a page code.

3.1.27 port: A port is a portion of the service delivery interface of a SCSI-3 device. A SCSI-3 device may have more than one port. Each port may attach to the same or a different physical interface.

3.1.28 primary device: A device for reading or writing data on medium. Examples are magnetic disk drives, cartridge tape drives, optical disk drives and CD-ROM drives. Use in a medium changer environment implies that the device supports removable volumes.

3.1.29 protocol-specific: Requirements for the referenced item are defined by a SCSI-3 protocol standard. A detailed definition of protocol-specific may be found in SAM.

3.1.20 SCSI device: A device that is connected to a service delivery subsystem and supports a SCSI application protocol. A detailed definition of a SCSI device may be found in SAM.

3.1.31 SCSI domain: The interconnection of two or more SCSI devices and a service delivery subsystem forms a SCSI domain. A detailed definition of a SCSI domain may be found in SAM.

3.1.32 sense data: Data describing an error or device exception condition that a device server delivers to an application client (see SPC). Sense data may be delivered in response to a REQUEST SENSE command or as autosense data.

3.1.33 sense key: Contents of the SENSE KEY field of REQUEST SENSE command response data.

3.1.34 service delivery subsystem: That part of a SCSI I/O system that transmits service requests to a logical unit and returns logical unit responses to an initiator. A detailed definition of a service delivery subsystem may be found in SAM.

3.1.35 status: One byte of response information sent from a device server to an application client upon completion of each command. A detailed definition of status may be found in SAM.

3.1.36 storage element: A component of a medium changer device used only for physical storage of a volume.

3.1.37 system: A system is one or more SCSI domains operating as a single configuration.

3.1.38 target: A SCSI device containing one or more logical units that receive and execute commands from an initiator. A detailed definition of a target may be found in SAM.

3.1.39 task: An object within a logical unit that represents the work associated with a command or a group of linked commands. A detailed definition of a task may be found in SAM.

3.1.40 task set: A group of tasks within a logical unit, whose interaction is dependent on the queuing and autocontingent allegiance rules defined in SAM.

3.1.41 third-party: When used in reference to RESERVE, or RELEASE commands, third-party means a reservation made on behalf of another device (e.g., a processor device requests that a direct-access device reserve itself for use by a sequential-access device).

3.1.42 unit attention condition: A state that a logical unit maintains while it has asynchronous status information to report to one or more initiators. A detailed definition of the unit attention condition may be found in SAM.

3.1.43 vendor-specific: Something (e.g., a bit, field, code value, etc.) that is not defined by this standard and may be vendor defined.

3.1.44 volume: The recording media and its carrier that is removable from a primary device and may be moved from one element to another by a medium changer.

3.1.45 volume rotation: The process of changing the orientation of a volume. In particular this refers to inverting a two-sided volume cartridge so that a data transport element that accesses only one side at a time may access data on the other side.

3.1.46 zero: Value of 0, the logical false condition of a variable.

3.2 Symbols and abbreviations

- **CDB** command descriptor block
- **I/O** input/output
- **ID** identifier
- **LSB** least significant bit
- **LUN** logical unit number
- **MSB** most significant bit
- **MMC** SCSI-3 Multi–Media Commands standard
- **RSVD** reserved field or bit
- **SAM** SCSI-3 Architecture Model
- **SBC** SCSI-3 Block Commands standard
- **SCSI-3** Only standards identified in the foreword as being part of the SCSI-3 standard document set
- **SMC** SCSI-3 Medium Changer Commands standard
- **SPC** SCSI-3 Primary Commands standard
- **SSC** SCSI-3 Stream Commands standard

3.3 Keywords

3.3.1 expected: A keyword used to describe the behavior of the hardware or software in the design models assumed by this standard. Other hardware and software design models may also be implemented.

3.3.2 invalid: A keyword used to describe an illegal or unsupported bit, byte, word, field or code value. Receipt of an invalid bit, byte, word, field or code value shall be reported as error.

3.3.3 mandatory: A keyword indicating an item that is required to be implemented as defined in this standard.

3.3.4 may: A keyword that indicated flexibility of choice with no implied preference.

3.3.5 obsolete: A keyword indicating that an item was defined in prior SCSI standards but has been removed from this standard.

3.3.6 optional: A keyword that describes features that are not required to be implemented by this standard. However, if any optional feature defined by this standards is implemented, then it shall be implemented as defined in this standard.

3.3.7 reserved: A keyword referring to bits, bytes, words, fields and code values that are set aside for future standardization. A reserved bit, byte, word or field shall be set to zero, or in accordance with a future extension to this standard. Recipients may check reserved bits, bytes, words or fields for zero values and report errors if non-zero values are received. Receipt of reserved code values in defined fields shall be reported as error.

3.3.8 shall: A keyword indicating a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other products that conform to this standard.

3.3.9 should: A keyword indicating flexibility of choice with a strongly preferred alternative; equivalent to the phrase "it is strongly recommended".

3.4 Conventions

Certain words and terms used in this standard have a specific meaning beyond the normal English meaning. These words and terms are defined either in 3.1 or in the text where they first appear. Names of commands, statuses, sense keys, additional sense codes, and additional sense code qualifiers are in all uppercase (e.g., REQUEST SENSE). Field names and bit names are in SMALL CAPS. Lower case is used for words having the normal English meaning.

Fields containing only one bit are usually referred to as the name bit instead of the name field.

Numbers that are not immediately followed by lowercase b or h are decimal values.

Numbers immediately followed by lowercase b (xxb) are binary values.

Numbers or upper case letters immediately followed by lowercase h (xxh) are hexadecimal values.

Lists sequenced by letters (e.g., a-red, b-blue, c-green) show no priority relationship between the listed items. Numbered lists (e.g., 1-red, 2-blue, 3-green) show a priority ordering between the listed items. If a conflict arises between text, tables, or figures, the order of precedence to resolve the conflicts is text; then tables; and finally figures. Not all tables or figures are fully described in the text. Tables show data format and values. NOTES do not constitute any requirements for implementors.

4 Overview

The SCSI-3 medium changer device class specifies a logical unit that is involved, primarily, with the movement of removable volumes in a controlled environment without human intervention. The SCSI-3 device classes that provide for removable volumes are block, multi–media and sequential. (See the SBC, MMC and SSC standards.)

A medium changer logical unit receives commands to move volumes between various element types in the element address space of the medium changer. The element types are storage, data transfer, medium transport, and import/export. A volume handling robotic subsystem, addressed as a medium transport element, moves volumes within a medium changer.

A medium changer logical unit maintains an inventory of volumes and the element address they can be found. The medium changer logical unit reports this inventory when requested as well as identifying the element addresses assigned to different types of elements.

Different levels of sophistication may be implemented in how this inventory is managed, reported, detected and maintained. The elements in a medium changer may be reserved to different initiators. For example, one data transport element may be reserved for exclusive use by one initiator. The primary device, located at that data transfer element, may then be attached to various systems for their use. In some cases, the primary device associated with a data transfer element may not be a SCSI device.

The split between load and unload control of the medium and read and write control by a primary device is a key feature of this device class. The mechanism for coordinating this kind of sophisticated activity is not specified in this standard. The medium changer device class provides the means for mount/dismount management only.

Figure 2 shows an example of an independent medium changer. The data transfer elements (primary devices) shown may be any type of removable media device such as a tape drive, disk drive or optical drive. Supporting multiple types of removable media within the same medium changer is also permitted. Also, the ports on each primary device may or may not attach to the same service delivery subsystem and the interfaces to the primary devices may or may not be SCSI. The number and arrangement of elements is arbitrary.

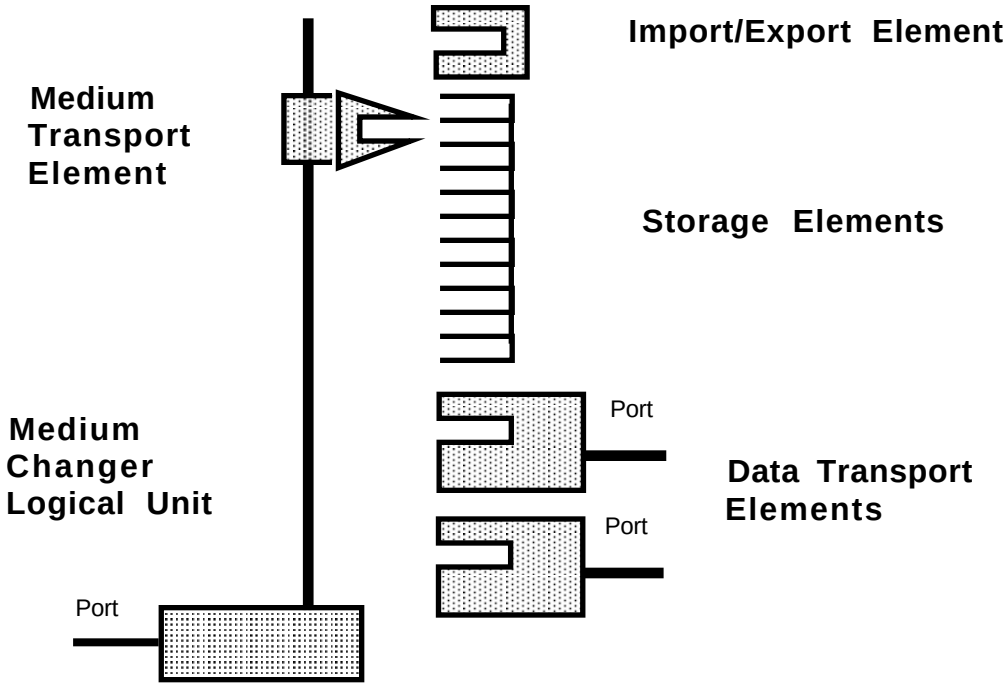

Figure 2 — Example independent medium changer device.

The independent changer model applies to implementations where the medium changer is addressed as a separate logical unit. The logical unit for the medium changer may be accessed via the same SCSI ID as the primary device, or via a different SCSI ID. This type of medium changer may support more than one primary device.

The attached medium changer model permits a subset of the functions of an independent changer to be incorporated directly into a primary device. Only one data transport element is permitted. In this case, only one logical unit is used to access all functions.

A medium changer moves volumes among the several element types accessible to it on command from an initiator. Medium changers shall be capable of reporting the full or empty status of any element address in its domain.

5 Medium changer models

5.1 Independent medium changer

An independent medium changer is a device server that returns 8h in the PERIPHERAL DEVICE TYPE field (see SPC) of INQUIRY command response data.

Independent medium changers for the SMC standard respond to a LUN different from those used by a primary device. Communication with a primary device may use the same service delivery subsystem as the medium changer device, or a different SCSI service delivery subsystem. Primary devices that are not SCSI devices are also permitted. Multiple primary devices may be attached to an independent changer.

If a primary device served by the medium changer is a SCSI device, the primary device may be addressed on a SCSI-3 service delivery subsystem though the same port as the medium changer but with a different LUN or the primary device may be addressed through independent ports and any LUN on the same or a different service delivery subsystem.

The READ ELEMENT STATUS command response data page for each data transfer element may provide the identity of the primary device serviced by a medium changer device. This support is optional since a primary device is not required to be a SCSI-3 device.

5.2 Attached medium changer

An attached medium changer is part of a device server that sets the MCHNGR bit to one in its standard INQUIRY data (see SPC). Attached medium changers respond to the same LUN as a primary device that is not a medium changer. In an attached medium changer, the PERIPHERAL DEVICE TYPE field of INQUIRY command response data returns the type of the primary device.

Two medium changer commands, READ ELEMENT STATUS ATTACHED and MOVE MEDIUM ATTACHED are added to the command set of the primary device. The other commands available depend on the model for the primary device.

5.3 Medium changer elements

A medium changer has an address space separate and distinct from the physical address space of a SCSI-3 service delivery subsystem. The term element is used throughout this standard to refer to one member of the medium changer address space. Within a medium changer, the element addresses are a set of physical locations and mechanisms within the scope of a medium changer device.

Each element is an instance of one of four element types:

- Medium Transport Element
- Storage Element
- Import/Export Element
- Data Transfer Element

Each element is a discrete physical entity with a unique element address that may provide storage for zero or one volume. A volume is in exactly one element at a time. The point in time during movement

when a volume is considered in a different element is implementation dependent. When requested to report element status, a volume shall be reported as being at exactly one element address. The element address shall not be arbitrarily changed by the medium changer. However, multiple initiator environments may cause volumes to be moved without notifying other initiators.

Volumes are addressed indirectly by this model. Volumes can be moved to or from any of the elements of the medium changer device using element addresses. The method of detecting the presence of a volume at any element in a medium changer is vendor specific.

In order to ensure exclusive access to a volume, the element where the volume is located (the element address) may be reserved by an initiator. Exclusive access will be lost if the volume is moved to an unreserved element. Exclusive access will be retained if a volume is moved between two reserved elements. Reservation of the medium transport element used is not required to preserve exclusive access. In an independent medium changer, elements may be reserved by the RESERVE ELEMENT (6), RESERVE ELEMENT (10), or PERSISTENT RESERVE OUT (see SPC) commands. In an attached medium changer, element reservations are only available using the PERSISTENT RESERVE OUT command.

Providing independent storage for medium is optional for medium transport, import export, and data transport elements. The capabilities of a particular medium changer can be determined from mode parameters in the device capabilities page (see 7.3.1).

Note 1—An example of an element not providing independent storage for a volume is a carousel style storage for volumes. The import/export function could be provided by a port which allows operator access to one of the storage elements. In such a medium changer a MOVE MEDIUM command to move a volume from a storage element to the import/export element rotates the carousel to align the addressed storage element to the import/export position. In this case, the import/export element does not provide independent storage but rather access to one of the storage elements.

Each element type shall be assigned a contiguous number range. The number ranges assigned to element types shall not overlap. The number ranges are not required to form one contiguous number range over all element types. Element address zero is reserved for use as the default medium transport element address.

5.4 Medium transport element

A medium transport element contains the functions of the medium changer device that moves a volume from one element address to another. When a medium transport element can serve (even temporarily) as a storage location for a volume, each location where a volume may be held shall have a separate medium transport element address. Support for a medium transport element address being the source and/or destination address in a MOVE MEDIUM and EXCHANGE MEDIUM command is optional. The maximum number of medium transport elements is 127.

In larger medium changer devices, the medium movement functions may be performed by multiple independent robotics subsystems. Each of these subsystems may have a number of medium transport element addresses. Sets of medium transport elements for the same common robotics system shall have their medium transport element addresses assigned contiguously.

Any of the medium transport element addresses within a medium changer may be used in the medium transport element address field of any MOVE MEDIUM or EXCHANGE MEDIUM command. An

initiator may determine the capabilities of the medium transport elements of a medium changer in the transport geometry mode parameters.

Element address zero is reserved for use in the medium transport element address field of MOVE MEDIUM and EXCHANGE MEDIUM commands to direct the medium changer to use a default, or any available medium transport element. Support for element address zero is mandatory.

Attached medium changer devices shall have only one medium transport element. In an attached changer, element address zero is reserved for the medium transport element.

5.5 Storage element

Storage elements are locations of volumes while a volume is not in some other element type. A volume in a storage element is available for access by medium transport elements. If the SMC device has no storage elements, then it shall have at least one import/export element with independent storage capability.

A storage element may be a source or destination address in a MOVE MEDIUM command or the optional EXCHANGE MEDIUM command.

5.6 Import/export element

Import/export elements are locations of volumes thatare being inserted into or withdrawn from the medium changer. A volume in one of these elements is accessible by at least one medium transport element in the medium changer, by the operator, or by another medium changer device (i.e., cascaded medium changer devices). Support for an import/export element is optional.

Any import/export element may be capable of import actions only, export actions only, or both when present in a medium changer logical unit.

An import/export element address may be a source or destination address in a MOVE MEDIUM command or the optional EXCHANGE MEDIUM command. Import/export elements may or may not provide independent storage of a volume.

5.7 Data transfer element

A data transfer element represents the interface between the medium changer and a primary device (e.g., a removable media optical disk drive or tape drive). A data transfer element is considered part of the medium changer and is not part of a primary device.

NOTE 2— It should be possible to place a primary device of a compatible device class in proximity to a medium changer, specify the interface as a data transfer element to the medium changer, and begin operation without any change to the primary device. Closer coordination between the medium changer and a primary device may be required in some implementations. Such coordination is vendor-specific.

Primary devices are capable of reading or writing the medium in a volume. Data transfer elements may also be viewed as medium changer element addresses of volumes loaded in or available for loading in or removal from primary devices (e.g., disk or tape drives). Any data transfer element shall be accessible to at least one medium transport element.

A data transfer element address may be a source or destination address in a MOVE MEDIUM command or the optional EXCHANGE MEDIUM command. Data transfer elements may or may not provide independent storage of a unit of media, see the device capabilities mode parameters. Attached medium changers shall have only one data transfer element.

5.8 Element status maintenance requirements

When a medium changer receives a valid READ ELEMENT STATUS command with a CURDATA bit of zero, the medium changer shall be capable of reporting as command response data various data required by each page type (i.e., full, error, etc.). The medium changer may maintain this information at all times or it may regenerate it after receiving a valid READ ELEMENT STATUS command. The optional INITIALIZE ELEMENT STATUS command may be used to force regeneration of this information.

5.9 Volume tag information

The READ ELEMENT STATUS command response data descriptor format for all element types includes fields that contain volume tag information. These optional fields are used to report volume identification information that the medium changer has acquired either by reading an external label (e.g., bar code labels), by a SEND VOLUME TAG command, or by other means which may be vendor-specific. The same volume tag information shall be available to all initiators whether the volume tag information was assigned by that initiator, by some other initiator, or by the medium changer.

The volume tag information field values shall be independent of any volume identification information recorded on the medium or a volume.

The medium changer command set definition does not impose any requirement that volume tag information be unique over the volumes within the domain of the medium changer. If volume tag information is implemented, the medium changer shall retain the association between volume tag information and a volume as the volume is moved from element address to element address.

Volume tag information provides a means to confirm the identity of a volume that is stored at a medium changer element address. When volume tag information is implemented, this standard does not specify any direct addressing of volumes based on the values in these fields. Optional commands are defined that provide translation between volume tag information and the element addresses of zero or more volumes with matching volume tags information.

The following commands support the optional volume tag functionality:

- SEND VOLUME TAG – used either as a translation request or to associate a volume tag with the volume currently residing at an element address. This is an optional command for independent medium changers;

- REQUEST VOLUME ELEMENT ADDRESS – returns the element address currently associated with the volume tag information transferred with the last SEND VOLUME TAG command. This is an optional command for independent medium changers;

- READ ELEMENT STATUS – optionally reports volume tag information for all element types. Volume tag information is an optional function of a medium changer.

5.10 Primary and alternate volume tag information

Element status descriptors, as optionally reported by the READ ELEMENT STATUS command, permit defining a primary volume tag and an alternate volume tag. Alternate volume tag information provides a means for a system to use different volume identification information for each partition of a volume. Primary volume tag information refers to the logical medium accessible via a MOVE MEDIUM command with the INVERT bit set to zero. Alternate volume tag information refers to the other side of the media (i.e., the side that would be accessed via a MOVE MEDIUM command with the INVERT bit set to one). Some volumes may be recorded on both sides. The INVERT bit setting permits an initiator to select the side to use when a volume is mounted.

All volumes have a primary volume tag information attribute.

5.11 Volume tag information format

Volume tag information consists of a VOLUME IDENTIFICATION field plus a VOLUME SEQUENCE NUMBER field.

Table 1 defines the fields within the primary and alternate volume tag information fields that may be present in READ ELEMENT STATUS descriptors and in the data format for the SEND VOLUME TAG command.

Table 1 — Volume tag information format

Byte \ Bit	7	6	5	4	3	2	1	0
0 31	VOLUME IDENTIFICATION							
32 33	Reserved							
34 35	VOLUME SEQUENCE NUMBER							

The VOLUME IDENTIFICATION field shall consist of a left justified sequence of characters. Unused positions shall be blank (20h) filled. In order for the SEND VOLUME TAG translate with template to work, the characters '*' and '?' (2Ah and 3Fh) shall not appear in the VOLUME IDENTIFICATION field and there shall be no blanks (20h) in the significant part (non blank filled) of the VOLUME IDENTIFICATION field. If volume tag information for a particular element is undefined, the VOLUME IDENTIFICATION field shall be zero filled. The VOLUME SEQUENCE NUMBER is a 2-byte integer field. If the VOLUME SEQUENCE NUMBER is not used, this field shall be zero.

NOTE 3— For compatibility with existing volume label conventions, it is recommended that the characters in the significant non-blank portion of the VOLUME IDENTIFICATION field be restricted to the set: '0'…'9', 'A'…'Z" and '_'.

6 Commands for medium changer logical units

Table 2 — Commands for independent medium changers

Command	Operation Code	Type	Subclause
CHANGE DEFINITION	40h	O	SPC
EXCHANGE MEDIUM	A6h	O	6.1
INITIALIZE ELEMENT STATUS	07h	O	6.2
INQUIRY	12h	M	SPC
LOG SELECT	4Ch	O	SPC
MODE SELECT (6)	15h	O	SPC
MODE SELECT (10)	55h	O	SPC
MODE SENSE (6)	1Ah	O	SPC
MODE SENSE (10)	5Ah	O	SPC
MOVE MEDIUM	A5h	M	6.3
PERSISTENT RESERVE IN	5Eh	O	SPC
PERSISTENT RESERVE OUT	5Fh	O	SPC
POSITION TO ELEMENT	2Bh	O	6.4
PREVENT ALLOW MEDIUM REMOVAL	1Eh	O	SPC
READ BUFFER	3Ch	O	SPC
READ ELEMENT STATUS	B8h	M	6.5
RECEIVE DIAGNOSTIC RESULTS	1Ch	O	SPC
RELEASE ELEMENT (6)	16h	O	6.6
RELEASE ELEMENT (10)	56h	O	6.7
REQUEST VOLUME ELEMENT ADDRESS	B5h	O	6.8
REQUEST SENSE	03h	M	SPC
RESERVE ELEMENT (6)	16h	O	6.9
RESERVE ELEMENT (10)	56h	O	6.10
REZERO UNIT	01h	O	SPC
SEND DIAGNOSTIC	1Dh	M	SPC
SEND VOLUME TAG	B6h	O	6.11
TEST UNIT READY	00h	M	SPC
WRITE BUFFER	3Bh	O	SPC

Key: M = command implementation is mandatory.
O = command implementation is optional.
* = optional operation codes for use by sequential devices only.

The commands for independent medium changers shall be as shown in table 2.

Operation codes 0Ch and C0h through FFh are vendor-specific. All other operation codes are reserved.

Attached medium changers shall support the READ ELEMENT STATUS ATTACHED and MOVE MEDIUM ATTACHED commands (see table 3) in addition to the commands defined by the primary device type. Attached medium changers shall not support other medium changer commands (e.g. EXCHANGE MEDIUM).

Table 3 — Commands for attached medium changers

Command name	Operation Code	Type	Subclause
MOVE MEDIUM ATTACHED	A7h	M	6.3
MOVE MEDIUM ATTACHED *	A5h	O	6.3
READ ELEMENT STATUS ATTACHED	B4h	M	6.5
READ ELEMENT STATUS ATTACHED *	B8h	O	6.5

Key: M = command implementation is mandatory.
O = command implementation is optional.
* = optional operation codes for use by sequential devices only.

Sequential devices, (primary device type 1) may also use operation codes A5h for MOVE MEDIUM ATTACHED and B8h for READ ELEMENT STATUS ATTACHED.

6.1 EXCHANGE MEDIUM command

The EXCHANGE MEDIUM command (see table 4) provides a means to exchange the volume in the source element address, with the volume located at a destination element address. Support of this command requires that the logical unit have the capability of handling two volumes at the same time or that it emulate this capability.

Support for this command is optional for an independent medium changer. This command has no command parameter data. No command response data is returned.

A reservation conflict shall occur if an EXCHANGE MEDIUM command is received from an initiator other than the one holding a logical unit or element reservation.

Table 4 — EXCHANGE MEDIUM command

Byte \ Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (A6h)							
1	Reserved							
2	MEDIUM TRANSPORT ADDRESS							
3								
4	SOURCE ADDRESS							
5								
6	FIRST DESTINATION ADDRESS							
7								
8	SECOND DESTINATION ADDRESS							
9								
10	Reserved						INV1	INV2
11	CONTROL							

The volume in the SOURCE ADDRESS element is moved to the FIRST DESTINATION ADDRESS element and the volume that previously occupied the FIRST DESTINATION ADDRESS element is moved to the SECOND DESTINATION ADDRESS element. The SECOND DESTINATION ADDRESS element may or may not be the same as the SOURCE ADDRESS element. In the case of a simple exchange, SOURCE ADDRESS and SECOND DESTINATION ADDRESS are the same. The device capabilities page (see 7.3.1) of the MODE SENSE command provides a matrix which defines the supported source element type and first destination element type combinations for EXCHANGE MEDIUM commands when the source element type is the same as second destination element type.

If the SOURCE ADDRESS element is empty, or the FIRST DESTINATION ADDRESS element is empty, the command shall be terminated with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code MEDIUM SOURCE ELEMENT EMPTY. If the SECOND DESTINATION ADDRESS element is full, and not the same as the SOURCE ADDRESS element, the command shall be terminated with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code MEDIUM DESTINATION ELEMENT FULL

The MEDIUM TRANSPORT ADDRESS field specifies the medium transport element that is to be used in executing this command. The default transport element address of zero may be used. If the MEDIUM TRANSPORT ADDRESS element has not been assigned or that element address has been assigned to a different element type, the logical unit shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS.

The SOURCE ADDRESS, the FIRST DESTINATION ADDRESS, and the SECOND DESTINATION ADDRESS fields may represent a storage element, an import/export element, a data transfer element, or a medium transport element. If the element address specified has not been assigned to a specific element of the medium changer, the logical unit shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS.

An INV1 bit of one specifies that the volume shall be inverted prior to depositing the volume into the FIRST DESTINATION ADDRESS element. Support for this bit set to one is optional.

An INV2 bit of one specifies that the volume shall be inverted prior to depositing the volume into the SECOND DESTINATION ADDRESS element. Support for this bit set to one is optional.

If the medium changer does not support volume rotation for handling double sided volumes, the INV1 and INV2 bits should be set to zero. If either of these bits is one, a logical unit which is not capable of volume rotation shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID FIELD IN CDB.

6.2 INITIALIZE ELEMENT STATUS command

The INITIALIZE ELEMENT STATUS command (see table 5) shall cause the medium changer to check all assigned element addresses for volume and any other status relevant to that element address. The intent of this command is to enable the Initiator to get a quick response from a subsequent READ ELEMENT STATUS command. It may be useful to issue this command after a power failure, or if a volume has been changed by an operator, or if configurations have been changed.

Table 5 — INITIALIZE ELEMENT STATUS command

Byte \ Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (07h)							
1	Reserved							
2								
3								
4								
5	CONTROL							

Support for this command is optional for an independent medium changer. This command has no command parameter data. No command response data is returned.

If an implementation does not support this command, the same function is provided in the READ ELEMENT STATUS command.

A reservation conflict shall occur if an INITIALIZE ELEMENT STATUS command is received from an initiator other than the one holding a logical unit or element reservation.

6.3 MOVE MEDIUM command

The MOVE MEDIUM command (see table 6) requests that the target move a volume from a source element to a destination element. Support for this command is mandatory for both independent medium changers and attached medium changers.

A reservation conflict shall occur if a MOVE MEDIUM command is received from an initiator other than the one holding a logical unit or element reservation.

Table 6 — MOVE MEDIUM command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	Reserved							
2	MEDIUM TRANSPORT ADDRESS							
3								
4	SOURCE ADDRESS							
5								
6	DESTINATION ADDRESS							
7								
8	Reserved							
9								
10	Reserved							INV
11	CONTROL							

The MOVE MEDIUM OPERATION CODE for an independent medium changer shall be A5h. An attached medium changer shall use OPERATION CODE A7h for the MOVE MEDIUM ATTACHED command. Attached changers connected to a sequential primary device are also permitted to implement OPERATION CODE A5h as the MOVE MEDIUM ATTACHED command.

This command moves the volume from the element specified by SOURCE ADDRESS to the element specified by DESTINATION ADDRESS.

If the SOURCE ADDRESS element is empty, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code MEDIUM SOURCE ELEMENT EMPTY. If the DESTINATION ADDRESS element is full, and different from the SOURCE ADDRESS element, the target

shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code MEDIUM DESTINATION ELEMENT FULL.

The MEDIUM TRANSPORT ADDRESS field specifies the medium transport element that is to be used in executing this command. Attached medium changers shall set this field to zero. Independent changers may set this field to zero to specify the default medium transport element. If the address specified has not been assigned or has been assigned to an element other than a medium transport element, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS.

The SOURCE ADDRESS and the DESTINATION ADDRESS fields may represent a storage element, an import/export element, a data transfer element, or a medium transport element. If the address specified has not been assigned to a specific element of the medium changer, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS.

The device capabilities MODE SENSE page (see 7.3.1), provides a matrix with the supported source element or destination element combinations for the MOVE MEDIUM command

An INVERT bit of one specifies that the volume shall be inverted or rotated prior to depositing the medium into the DESTINATION ADDRESS element. If the medium changer does not support volume rotation for handling double sided media, the INVERT bit should be set to zero. If this bit is one, a target that is not capable of volume rotation shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID FIELD IN CDB.

6.4 POSITION TO ELEMENT command

The POSITION TO ELEMENT command (see table 7) shall position the MEDIUM TRANSPORT ADDRESS element such that further motion of the MEDIUM TRANSPORT ADDRESS element is unnecessary to execute an appropriate MOVE MEDIUM command between the MEDIUM TRANSPORT ADDRESS element and the DESTINATION ADDRESS element.

Support for this command is optional for independent medium changers. This command has no command parameter data. No command response data is returned.

A reservation conflict shall occur if a POSITION TO ELEMENT command is received from an initiator other than the one holding a logical unit or element reservation.

Table 7 — POSITION TO ELEMENT command

Byte \ Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (2Bh)							
1	Reserved							
2	MEDIUM TRANSPORT ADDRESS							
3								
4	DESTINATION ADDRESS							
5								
6	Reserved							
7								
8	Reserved							INVERT
9	CONTROL							

An INVERT bit value of one requests that the MEDIUM TRANSPORT ADDRESS element be inverted or rotated before positioning in front of the destination element address. A value of zero requests that the orientation of the MEDIUM TRANSPORT ADDRESS element be unchanged before positioning in front of the DESTINATION ADDRESS element. Support for this bit set to one is optional.

6.5 READ ELEMENT STATUS command

The READ ELEMENT STATUS command (see table 8) requests that the target report the status of its internal elements to the initiator. Support for this command is mandatory for both attached medium changers and independent medium changers.

Table 8 — READ ELEMENT STATUS command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE							
1	Reserved			VOLTAG	ELEMENT TYPE CODE			
2	STARTING ELEMENT ADDRESS							
3								
4	NUMBER OF ELEMENTS							
5								
6	Reserved						CURDATA	DVCID
7	ALLOCATION LENGTH							
8								
9								
10	Reserved							
11	CONTROL							

The READ ELEMENT STATUS OPERATION CODE for an independent medium changer shall be B8h. An attached medium changer shall use OPERATION CODE B4h for the READ ELEMENT STATUS ATTACHED command. Attached changers connected to a sequential primary device may also implement OPERATION CODE B8h as the READ ELEMENT STATUS ATTACHED command.

A volume tag (VOLTAG) bit of one indicates that the target shall report volume tag information if this feature is supported. A value of zero indicates that volume tag information shall not be reported. If the volume tag feature is not supported this bit shall be treated as reserved.

If the current data (CURDATA) bit is one, the target shall return element status data without causing device motion. If the CURDATA bit is zero , the target may cause device motion to confirm element status data. Support for the CURDATA bit set to one is mandatory.

A reservation conflict shall occur if a READ ELEMENT STATUS or READ ELEMENT STATUS ATTACHED command with the CURDATA bit set to zero is received from an initiator other than the one

holding a logical unit or element reservation. If the CURDATA bit is set to one, a reservation conflict shall not occur.

The ELEMENT TYPE CODE field specifies the particular element type(s) selected for reporting by this command. A value of zero specifies that status for all element types shall be reported. The element type codes are defined in table 9.

Table 9 —Element type code

Code	Description
0h	All element types reported, (valid in CDB only)
1h	Medium transport element
2h	Storage element
3h	Import/export element
4h	Data transfer element
5h - Fh	Reserved

The STARTING ELEMENT ADDRESS field specifies the minimum element address to report. Only elements with an element type code permitted by the ELEMENT TYPE CODE field, and an element address greater than or equal to STARTING ELEMENT ADDRESS shall be reported. Element descriptor blocks are not generated for undefined element addresses.

The NUMBER OF ELEMENTS field specifies the maximum number of element descriptors to be created by the target for this command. The value specified by this field is not the range of element addresses to be considered for reporting but rather the number of defined elements to report. If the ALLOCATION LENGTH field is not sufficient to transfer all the element descriptors, the target shall transfer all those descriptors that can be completely transferred and this shall not be considered an error.

A device ID (DVCID) bit of one specifies that the target shall return device identifiers, if available, for the specified range. A DVCID bit of zero specifies that the target shall not return device identifiers. If the device ID feature is not supported, the DVCID bit shall be treated as reserved.

6.5.1 Element status data

The data returned by the READ ELEMENT STATUS command is defined in table 10 and 6.5.2 through 6.5.6. Element status data consists of an eight-byte header (see table 10), followed by zero or more element status pages.

Table 10 — Element status data

Byte \ Bit	7	6	5	4	3	2	1	0
0 1	FIRST ELEMENT ADDRESS REPORTED							
2 3	NUMBER OF ELEMENTS AVAILABLE							
4	Reserved							
5 6 7	BYTE COUNT OF REPORT AVAILABLE (all pages, n-7)							
8 n	Element status page(s)							

The FIRST ELEMENT ADDRESS REPORTED field indicates the element address of the element with the smallest element address found to meet the CDB request.

The NUMBER OF ELEMENTS AVAILABLE field indicates the number of elements meeting the request in the command descriptor block. The status for these elements is returned if sufficient ALLOCATION LENGTH was specified.

The BYTE COUNT OF REPORT AVAILABLE field indicates the number of bytes of element status page data available for all elements meeting the request in the command descriptor block. This value shall not be adjusted to match the ALLOCATION LENGTH available.

NOTE 4 — The READ ELEMENT STATUS command can be issued with an ALLOCATION LENGTH of eight bytes in order to determine the ALLOCATION LENGTH required to transfer all the element status data specified by the command.

6.5.2 Element status page

The element status page is defined in table 11. Each element status page includes an eight-byte header followed by zero or more element descriptor blocks. The header includes the element type code, the length of each descriptor block and the number of bytes of element descriptor information that follow the header for this element type.

Table 11 —Element status page

Byte \ Bit	7	6	5	4	3	2	1	0
0	ELEMENT TYPE CODE							
1	PVOLTAG	AVOLTAG	Reserved					
2 3	ELEMENT DESCRIPTOR LENGTH (z)							
4	Reserved							
5 6 7	BYTE COUNT OF DESCRIPTOR DATA AVAILABLE (this page, y - 7)							
8 y	Element descriptor(s)							

The ELEMENT TYPE CODE field indicates the element type (see table 9) reported by this page.

A primary volume tag (PVOLTAG) bit of one indicates that the PRIMARY VOLUME TAG INFORMATION field is present in each of the following element descriptor blocks. A value of zero indicates that these bytes are omitted from the element descriptors that follow.

An alternate volume tag (AVOLTAG) bit of one indicates that the ALTERNATE VOLUME TAG INFORMATION field is present in each of the following element descriptor blocks. A value of zero indicates that these bytes are omitted from the element descriptors that follow.

The ELEMENT DESCRIPTOR LENGTH field indicates the number of bytes in each element descriptor.

The BYTE COUNT OF DESCRIPTOR DATA AVAILABLE field indicates the number of bytes of element descriptor data available for elements of this element type meeting the request in the CDB. This value shall not be adjusted to match the ALLOCATION LENGTH available.

Each element descriptor includes the element address and status flags; it may also contain sense code information as well as other information depending on the element type (see 6.5.3 through 6.5.6).

6.5.3 Medium transport element descriptor

Table 12 defines the medium transport element descriptor.

Table 12 —Medium transport element descriptor

Bit / Byte	7	6	5	4	3	2	1	0
0	ELEMENT ADDRESS							
1								
2	Reserved					EXCEPT	RSVD	FULL
3	Reserved							
4	ADDITIONAL SENSE CODE							
5	ADDITIONAL SENSE CODE QUALIFIER							
6	Reserved							
8								
9	SVALID	INVERT	Reserved					
10	SOURCE STORAGE ELEMENT ADDRESS							
11								
	•••							
(36 bytes)	PRIMARY VOLUME TAG INFORMATION (field omitted if PVOLTAG=0)							
(36 bytes)	ALTERNATE VOLUME TAG INFORMATION (field omitted if AVOLTAG =0)							
	•••							
(1 byte)	Reserved				CODE SET			
(1 byte)	Reserved				IDENTIFIER TYPE			
(1 byte)	Reserved							
(1 byte)	IDENTIFIER LENGTH (x)							
(x bytes)	IDENTIFIER							
	•••							
to z-1	Vendor-specific							

The ELEMENT ADDRESS field gives the address of the medium changer element whose status is reported by this element descriptor block.

An exception (EXCEPT) bit of one indicates the element is in an abnormal state. An exception bit of zero indicates the element is in a normal state. If this bit is one, information on the abnormal state may be available in the ADDITIONAL SENSE CODE and ADDITIONAL SENSE CODE QUALIFIER fields.
A FULL bit value of one indicates that the element contains a unit of media. A value of zero indicates that the element does not contain a unit of media. When the EXCEPT bit is one, the value of the FULL bit is not valid.

The ADDITIONAL SENSE CODE field may provide specific information on an abnormal element state. The values in this field are as defined for the ADDITIONAL SENSE CODE field of REQUEST SENSE command response data (see SPC). This field is valid only if the EXCEPT bit is one.

The ADDITIONAL SENSE CODE QUALIFIER field may provide more detailed information on an abnormal element state. The values in this field are as defined for the ADDITIONAL SENSE CODE QUALIFIER field of REQUEST SENSE command response data (see SPC). This field is valid only if the EXCEPT bit is one.

A source valid (SVALID) bit value of one indicates that the SOURCE STORAGE ELEMENT ADDRESS field and the INVERT bit information are valid. A value of zero indicates that the values in these fields are not valid.

An INVERT bit value of one indicates that the unit of media now in this element was inverted by MOVE MEDIUM or EXCHANGE MEDIUM operations since it was last in the SOURCE STORAGE ELEMENT ADDRESS. A value of zero indicates that no inversion occurred during the operation.

The SOURCE STORAGE ELEMENT ADDRESS field provides the address of the last storage element this unit of media occupied. This field is valid only if the SVALID bit is one.

The PRIMARY VOLUME TAG INFORMATION and ALTERNATE VOLUME TAG INFORMATION fields provide for identifying the unit of media residing in this element (see 5.9). Either or both of these fields may be omitted for all the element descriptor blocks that comprise an element status page as indicated by the PVOLTAG and AVOLTAG bits in the element status page header.

The CODE SET field and IDENTIFIER TYPE field are defined in 6.5.7.

The IDENFIFIER LENGTH field contains the length in bytes of the IDENTIFIER field (see 6.5.7). If no device identifier is available, or the DVCID bit in the CDB is zero, the IDENTIFIER LENGTH field shall be zero and the CODE SET and IDENTIFIER TYPE fields shall also be zero.

The IDENTIFIER field provides a device identifier for this medium transport element as defined in 6.5.7. If no device identifier is available for this element, or the DVCID bit in the CDB is zero, this field shall be omitted.

6.5.4 Storage element descriptor

Table 13 defines the storage element descriptor.

Table 13 — Storage element descriptor

Byte \ Bit	7	6	5	4	3	2	1	0
0	ELEMENT ADDRESS							
1								
2	Reserved				ACCESS	EXCEPT	RSVD	FULL
3	Reserved							
4	ADDITIONAL SENSE CODE							
5	ADDITIONAL SENSE CODE QUALIFIER							
6	Reserved							
8								
9	SVALID	INVERT	Reserved					
10	SOURCE STORAGE ELEMENT ADDRESS							
11								
•••								
(36 bytes)	PRIMARY VOLUME TAG INFORMATION (field omitted if PVOLTAG=0)							
(36 bytes)	ALTERNATE VOLUME TAG INFORMATION (field omitted if AVOLTAG=0)							
•••								
(1 byte)	Reserved				CODE SET			
(1 byte)	Reserved				IDENTIFIER TYPE			
(1 byte)	Reserved							
(1 byte)	IDENTIFIER LENGTH (x)							
(x bytes)	IDENTIFIER							
•••								
to z-1	Vendor-specific							

An ACCESS bit value of one indicates that access to the element by a medium transport element is allowed. An ACCESS bit of zero indicates that access to the element by the medium transport element is denied. When the EXCEPT bit is one, the value of the ACCESS bit is not valid.

The SOURCE STORAGE ELEMENT ADDRESS field provides the address of the last storage element this unit of media occupied. This element address value may or may not be the same as this element. This field is valid only if the SVALID bit is one.

The CODE SET field and IDENTIFIER TYPE field are defined in 6.5.7.

The IDENFIFIER LENGTH field contains the length in bytes of the IDENTIFIER field (see 6.5.7). If no device identifier is available, or the DVCID bit in the CDB is zero, the IDENTIFIER LENGTH field shall be zero and the CODE SET and IDENTIFIER TYPE fields shall also be zero.

The IDENTIFIER field provides a device identifier for this storage element as defined in 6.5.7. If no device identifier is available for this element, or the DVCID bit in the CDB is zero, this field shall be omitted.

For fields not defined in this subclause, see 6.5.3.

6.5.5 Import/export element descriptor

Table 14 defines the import/export element descriptor.

Table 14 — Import/export element descriptor

Byte \ Bit	7	6	5	4	3	2	1	0
0 1	ELEMENT ADDRESS							
2	Reserved		INENAB	EXENAB	ACCESS	EXCEPT	IMPEXP	FULL
3	Reserved							
4	ADDITIONAL SENSE CODE							
5	ADDITIONAL SENSE CODE QUALIFIER							
6 8	Reserved							
9	SVALID	INVERT	Reserved					
10 11	SOURCE STORAGE ELEMENT ADDRESS							
•••								
(36 bytes)	PRIMARY VOLUME TAG INFORMATION (field omitted if PVOLTAG=0)							
(36 bytes)	ALTERNATE VOLUME TAG INFORMATION (field omitted if AVOLTAG=0)							
•••								
(1 byte)	Reserved				CODE SET			
(1 byte)	Reserved				IDENTIFIER TYPE			
(1 byte)	Reserved							
(1 byte)	IDENTIFIER LENGTH (x)							
(x bytes)	IDENTIFIER							
•••								
To z-1	Vendor-specific							

An import enable (INENAB) bit of one indicates that the import/export element supports movement of media into the scope of the medium changer device. An INENAB bit of zero indicates that this element does not support import actions.

An export enable (EXENAB) bit of one indicates that the import/export element supports movement of media out of the scope of the medium changer device. An EXENAB bit of zero indicates that this element does not support export actions.

An ACCESS bit of one indicates that access to the import/export element by a medium transport element is allowed. An ACCESS bit of zero indicates access to the import/export element by medium transport elements is denied. When the EXCEPT bit is one, the value of the ACCESS bit is invalid.

NOTE 5— An example of when access would be denied is when the operator has exclusive access to the import/export element.

An import export (IMPEXP) bit of one indicates the unit of media in the import/export element was placed there by an operator. An IMPEXP bit of zero indicates the unit of media in the import/export element was placed there by the medium transport element. When the EXCEPT bit is one, the value of the IMPEXP bit is invalid.

The CODE SET field and IDENTIFIER TYPE field are defined in 6.5.7.

The IDENFIFIER LENGTH field contains the length in bytes of the IDENTIFIER field (see 6.5.7). If no device identifier is available, or the DVCID bit in the CDB is zero, the IDENTIFIER LENGTH field shall be zero and the CODE SET and IDENTIFIER TYPE fields shall also be zero.

The IDENTIFIER field provides a device identifier for this import/export element as defined in 6.5.7. If no device identifier is available for this element, or the DVCID bit in the CDB is zero, this field shall be omitted.

For fields not defined in this subclause, see 6.5.3.

6.5.6 Data transfer element descriptor

Table 15 defines the data transfer element descriptor.

Table 15 —Data transfer element descriptor

Bit / Byte	7	6	5	4	3	2	1	0
0	ELEMENT ADDRESS							
1								
2	Reserved				ACCESS	EXCEPT	RSVD	FULL
3	Reserved							
4	ADDITIONAL SENSE CODE							
5	ADDITIONAL SENSE CODE QUALIFIER							
6	NOT BUS	RSVD	ID VALID	LU VALID	RSVD	LOGICAL UNIT NUMBER		
7	SCSI BUS ADDRESS							
8	Reserved							
9	SVALID	INVERT	Reserved					
10	SOURCE STORAGE ELEMENT ADDRESS							
11								
	•••							
(36 bytes)	PRIMARY VOLUME TAG INFORMATION (field omitted if PVOLTAG=0)							
(36 bytes)	ALTERNATE VOLUME TAG INFORMATION (field omitted if AVOLTAG=0)							
	•••							
(1 byte)	Reserved				CODE SET			
(1 byte)	Reserved				IDENTIFIER TYPE			
(1 byte)	Reserved							
(1 byte)	IDENTIFIER LENGTH (x)							
(x bytes)	IDENTIFIER							
	•••							
to z-1	Vendor-specific							

An ACCESS bit value of one indicates access to the data transfer element by the medium transport element is allowed. A value of zero indicates access to the data transfer element by a medium transport element is denied.

NOTE 6— Access to the data transfer element by medium transport elements might be denied if a data transfer operation was under way. Note that a one value in this bit may not be sufficient to ensure a successful operation. This bit can only reflect the best information available to the medium changer device, which may not accurately reflect the state of the primary (data transfer) device.

The NOT BUS bit, ID VALID bit, SCSI BUS ADDRESS field, LU VALID bit and LOGICAL UNIT NUMBER field return information about addressing other devices connected to an independent medium changer. These fields shall be set to zero in an attached medium changer. If the SCSI bus address of the other device is over 255, the ID VALID bit shall be set to zero. If the lun address is over 7, the LU VALID bit shall be set to zero.

A not this bus (NOT BUS) bit value of one indicates that the SCSI BUS ADDRESS and LOGICAL UNIT NUMBER values may not be valid for the SCSI bus used to select the medium changer device. A NOT BUS bit value of zero indicates that the SCSI address and logical unit values, are on the same bus as the medium changer device. If the ID VALID and LU VALID bits are zero, this bit shall be ignored.

An ID VALID bit value of one indicates that the SCSI BUS ADDRESS field contains valid information. An LU VALID bit value of one indicates that the LOGICAL UNIT NUMBER field contains valid information.

The SCSI BUS ADDRESS field, if valid, provides the SCSI address of the primary device served by the medium changer at this element address.

The LOGICAL UNIT NUMBER field, if valid, provides the logical unit number within the SCSI bus device of the primary device served by the medium changer at this element address.

The CODE SET field and IDENTIFIER TYPE field are defined in 6.5.7.

The IDENFIFIER LENGTH field contains the length in bytes of the IDENTIFIER field (see 6.5.7). If no device identifier is available, or the DVCID bit in the CDB is zero, the IDENTIFIER LENGTH field shall be zero and the CODE SET and IDENTIFIER TYPE fields shall also be zero.

The IDENTIFIER field provides a device identifier for the primary device associated with this data transfer element as defined in 6.5.7. If no device identifier is available for this element, or the DVCID bit in the CDB is zero, this field shall be omitted.

For fields not defined in this subclause, see 6.5.3.

6.5.7 Identification descriptor

Table 16 defines the identification descriptor fields returned in element descriptors.

Table 16 — Identification descriptor fields

Byte \ Bit	7	6	5	4	3	2	1	0
(1 byte)	Reserved				CODE SET			
(1 byte)	Reserved				IDENTIFIER TYPE			
(1 byte)	Reserved							
(1 byte)	IDENTIFIER LENGTH (x)							
(x bytes)	IDENTIFIER							

The CODE SET, IDENTIFIER TYPE, IDENTIFIER LENGTH and IDENTIFIER fields in element descriptors are defined by the device identification page in SPC. Device identifiers may be available for some or all elements in a medium changer. If no device identifier is available or the DVCID bit in the CDB is zero, the IDENTIFIER LENGTH shall be zero, the IDENTIFIER field is omitted, and the CODE SET and IDENTIFIER TYPE fields shall be zero.

For a data transfer element, the IDENTIFIER field returns a device identifier from the primary device (disk or tape drive) associated with this element. The same CODE SET, IDENTIFIER TYPE, IDENTIFIER LENGTH and IDENTIFIER fields should be available via an INQUIRY command (see SPC) issued to the primary device.

For an import/export element, the IDENTIFIER field returns a unique identifier for the import/export device. An element used to exchange media between two medium changers should return the same CODE SET, IDENTIFIER TYPE, IDENTIFIER LENGTH and IDENTIFIER fields via either medium changer.

For a storage or medium transport element the CODE SET, IDENTIFIER TYPE, IDENTIFIER LENGTH and IDENTIFIER fields refer to the element, and are not an identifier for a volume stored in this location.

6.6 RELEASE ELEMENT (6) command

The RELEASE ELEMENT (6) command (see table 17) is used to release a previously reserved logical unit, or, if the element release option is implemented, to release previously reserved elements within units.

Table 17 — RELEASE ELEMENT (6) command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (17h)							
1	Reserved			Obsolete				ELEMENT
2	RESERVATION IDENTIFICATION							
3	Reserved							
4								
5	CONTROL							

Attached medium changers shall implement the RESERVE (6) command as defined in the SPC. Element reservations are not allowed in attached medium changer usage.

Note 7—The ELEMENT bit is in the same position as the EXTENT bit in the SPC definition of RESERVE (6). An attached medium changer with a primary device that supports extents can not distinguish element from extent reservations.

The RESERVE ELEMENT and RELEASE ELEMENT commands provide the basic mechanism for contention resolution in multiple initiator systems. A reservation may only be released by a RELEASE command from the initiator that made it. It is not an error for an application client to attempt to release a reservation that is not currently valid, or is held by another initiator. In this case, the device server shall return GOOD status without altering any other reservation.

6.6.1 Logical unit release (Mandatory)

If the ELEMENT bit is zero, this command shall cause the device server to terminate all unit and element reservations that are active from the initiator to the specified logical unit. The RESERVATION IDENTIFICATION field in the command descriptor shall be ignored by the device server.

6.6.2 Element release (Optional)

If the ELEMENT bit is one and the element release option is implemented, this command shall cause any reservation from the requesting initiator with a matching reservation identification to be terminated. Other reservations from the requesting initiator shall remain in effect. If the ELEMENT bit is one and the element release option is not implemented, the RELEASE ELEMENT command shall be terminated with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID FIELD IN CDB. This option shall be implemented if the element reservation option (see 6.9.2) is implemented.

6.7 RELEASE ELEMENT (10) command

The RELEASE ELEMENT (10) command (see table 18) is used to release a previously reserved logical unit, or if the element release option is implemented, to release previously reserved elements within units. This clause describes only those instances where the RELEASE ELEMENT (10) command differs from the RELEASE ELEMENT (6) command. Except for the instances described in this clause, the RELEASE ELEMENT (10) command shall function exactly like the RELEASE ELEMENT (6) command (see 6.6).

Table 18— RELEASE ELEMENT (10) command

Byte \ Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (57h)							
1	Reserved			3RDPTY	Reserved		LONGID	ELEMENT
2	RESERVATION IDENTIFICATION							
3	THIRD PARTY DEVICE ID							
4	Reserved							
5								
6								
7	PARAMETER LIST LENGTH							
8								
9	CONTROL							

6.7.1 Third-party release (Mandatory)

The third-party release option of the RELEASE ELEMENT command allows an initiator to release a unit or elements within a unit that were previously reserved using third-party reservation (see 6.10.1). Third-party release shall be implemented if third-party reserve is implemented.

If the third-party (3RDPTY) bit is zero, the third-party release option is not requested. If the 3RDPTY bit is one and the target implements the third-party release option, the target shall release the specified unit or elements, but only if the reservation was made using the third-party reservation option by the initiator that is requesting the release for the same SCSI device as specified in the THIRD PARTY DEVICE ID field.

The format of THIRD PARTY DEVICE ID is protocol specific. If the THIRD PARTY DEVICE ID value associated with the reservation release is smaller than 255, the LONGID bit may be zero and the THIRD PARTY DEVICE ID value sent in the CDB. If the LONGID bit is zero, the PARAMETER LIST LENGTH field shall be set to zero. If the THIRD PARTY DEVICE ID value is greater than 255, the LONGID bit shall be one.

Device servers that support device IDs greater than 255 shall accept commands with LONGID equal to one. Device servers whose devices IDs are limited to 255 or smaller may reject commands with LONGID equal to one with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID FIELD IN CDB.

If the LONGID bit is one, the PARAMETER LIST LENGTH shall be eight, and the parameter list shall have the format shown in table 19. If the LONGID bit is one, the THIRD PARTY DEVICE ID field in the CDB shall be ignored. If the LONGID bit is one and the PARAMETER LIST LENGTH is not eight, the device server shall return a CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code PARAMETER LIST LENGTH ERROR.

Table 19— RELEASE ELEMENT (10) parameter list

Bit / Byte	7	6	5	4	3	2	1	0
0	THIRD PARTY DEVICE ID							
7								

6.8 REQUEST VOLUME ELEMENT ADDRESS command

The REQUEST VOLUME ELEMENT ADDRESS command (see table 20) is used to transfer the results of a SEND VOLUME TAG command. Multiple REQUEST VOLUME ELEMENT ADDRESS commands may be used to retrieve the results of a single SEND VOLUME TAG command with the translate option.

Support for this command is optional for independent medium changers. This command has no command parameter data. This command returns command response data.

A reservation conflict shall occur if a REQUEST VOLUME ELEMENT ADDRESS command is received from an initiator other than the one holding a logical unit or element reservation.

The command response data returned by this command consists of a header as defined by table 21, plus zero or more element type specific pages in the same format as defined by the READ ELEMENT STATUS command (see 6.5).

For each SEND VOLUME TAG command, the logical unit shall be able to report in response to a REQUEST VOLUME ELEMENT ADDRESS command zero or more elements that match a volume tag template in element address order. Once information for a given element address has been reported, only higher element addresses will be reported by subsequent REQUEST VOLUME ELEMENT ADDRESS commands.

If a REQUEST VOLUME ELEMENT ADDRESS command is received and no prior SEND VOLUME TAG command has been executed or the element list has been completely reported for the most recent successful SEND VOLUME TAG command, the logical unit shall return command response data consisting of only the volume element address header.

NOTE 8— In order to ensure the successful completion of a SEND VOLUME TAG, REQUEST VOLUME ELEMENT ADDRESS command sequence in a multi-initiator environment, it may be necessary to reserve the logical unit to the initiator prior to sending the SEND VOLUME TAG command and release the logical unit after the last REQUEST VOLUME ELEMENT ADDRESS command has completed.

Table 20 — REQUEST VOLUME ELEMENT ADDRESS command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (B5h)							
1	Reserved			VOLTAG	ELEMENT TYPE CODE			
2	ELEMENT ADDRESS							
3								
4	NUMBER OF ELEMENTS TO REPORT							
5								
6	Reserved							
7	ALLOCATION LENGTH							
8								
9								
10	Reserved							
11	CONTROL							

A volume tag (VOLTAG) bit of one indicates that the logical unit shall report volume tag information, if implemented by the logical unit. A value of zero indicates that volume tag information shall not be reported. Support for this bit set to one is optional.

The ELEMENT TYPE CODE field specifies the particular element type(s) selected for reporting by this command. A value of zero specifies that status for all element types shall be reported. The element type codes are defined by table 9. This field value determines the element types to be reported from information prepared by the most recently successful SEND VOLUME TAG command.

The ELEMENT ADDRESS field gives a medium changer element address whose interpretation depends on the SEND ACTION CODE field (see table 27) of the last successful SEND VOLUME TAG command. The SEND ACTION CODE is returned in the volume element address header. When the last SEND ACTION CODE was a translate, the ELEMENT ADDRESS field gives the minimum element address to be reported by this command. When the SEND ACTION CODE is assert, replace, and undefine, the ELEMENT ADDRESS field gives the particular element whose volume tag information was modified.

The NUMBER OF ELEMENTS TO REPORT field specifies the maximum number of elements to be reported for this command. The value specified in this field is the number of elements to report of those that match

the last SEND VOLUME TAG command translate template. If the ALLOCATION LENGTH is not sufficient to transfer all the element descriptors, the logical unit shall transfer all those descriptors that can be completely transferred and this shall not be considered an error.

Table 21 — Volume element address header

Byte \ Bit	7	6	5	4	3	2	1	0
0 1	FIRST ELEMENT ADDRESS REPORTED							
2 3	NUMBER OF ELEMENTS REPORTED							
4	Reserved			SEND ACTION CODE				
5 6 7	BYTE COUNT OF REPORT AVAILABLE (all pages, x - 7)							
8 x	Element status page(s)							

The SEND ACTION CODE field (see table 27) reports the function performed by the last SEND VOLUME TAG command.

For fields not defined in this subclause, see the READ ELEMENT STATUS command description.

6.9 RESERVE ELEMENT (6) command

The RESERVE ELEMENT (6) command (see table 22) is used to reserve a logical unit or, if the element reservation option is implemented, elements within a logical unit.

Table 22 — RESERVE ELEMENT (6) command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (16h)							
1	Reserved				Obsolete			ELEMENT
2	RESERVATION IDENTIFICATION							
3	ELEMENT LIST LENGTH							
4								
5	CONTROL							

The RESERVE ELEMENT (6) and RELEASE ELEMENT (6) commands provide the basic mechanism for contention resolution in multiple-initiator systems.

If the RESERVE ELEMENT (6) command is implemented, then the RELEASE ELEMENT (6) also shall be implemented. Attached medium changers shall implement the RESERVE (6) command as defined in SPC. Element reservations are not allowed in attached medium changer usage.

NOTE 9 — The ELEMENT bit is in the same position as the EXTENT bit in the SPC definition of RESERVE (6). An attached medium changer with a primary device that supports extents can not distinguish element from extent reservations.

6.9.1 Logical unit reservation (Mandatory)

If the ELEMENT bit is zero, this command shall request that the entire logical unit be reserved for the exclusive use of the initiator as defined in SPC. If the ELEMENT bit is zero, the RESERVATION IDENTIFICATION and the ELEMENT LIST LENGTH fields shall be ignored.

If the unit, or any element within the unit, is reserved for another initiator, the target shall respond by returning RESERVATION CONFLICT status.

If, after honoring the reservation, any other initiator attempts to perform any command on the reserved unit other than an INQUIRY, REQUEST SENSE, RESERVE, RELEASE command, or a READ ELEMENT STATUS command with a curdata bit of 1, the command shall be rejected with RESERVATION CONFLICT status.

6.9.2 Element reservation (optional)

The RESERVATION IDENTIFICATION field provides a means for an application client to identify each element reservation. This allows an application client in a multiple tasking environment to have multiple reservations outstanding. The RESERVATION IDENTIFICATION field is used in the RELEASE ELEMENT command to specify which reservation is to be released. It is also used in superseding RESERVE ELEMENT commands to specify which reservation is to be superseded.

If the element reservation option is implemented, the element release option (see 6.6.2) shall also be implemented. These options permit multiple groups of elements within the unit to be reserved, each with a separate reservation identification.

If the ELEMENT bit is one, and the element reservation option is implemented, the device server shall process the reservation request as follows:

– The element list shall be checked for valid element addresses. If any element address is invalid for this unit, the command shall be terminated with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS. If any element descriptor in the element list specifies an element already specified in another element descriptor in the list, the command shall be terminated with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS;

– If the requested reservation does not conflict with any active or previously requested reservation, the elements specified shall be reserved until superseded by another valid RESERVE command from the initiator that made the reservation, or until released by a RELEASE command from the same initiator, by a TARGET RESET task management function from any initiator, by a hard reset condition, or by a power on cycle. If either of the last three conditions occur, a unit attention condition shall be generated;

– If the reservation request conflicts with an existing reservation, then the device server shall return RESERVATION CONFLICT status;

– If the ELEMENT bit is one, and the element reservation option is not implemented, the RESERVE command shall be terminated with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID FIELD IN CDB.

The size of the element list shall be defined by the ELEMENT LIST LENGTH parameter. The element list shall consist of zero or more descriptors as shown in table 23. Each element list descriptor defines a series of elements beginning at the specified ELEMENT ADDRESS for the specified NUMBER OF ELEMENTS. If the NUMBER OF ELEMENTS field is zero, the element list shall begin at ELEMENT ADDRESS and continue through the last element address on the unit.

Table 23 — Data format of element list descriptors

Byte \ Bit	7	6	5	4	3	2	1	0
0	Reserved							
1								
2	NUMBER OF ELEMENTS							
3								
4	ELEMENT ADDRESS							
5								

If an initiator issues a command to an element that has been reserved by a different initiator, the command shall not be performed and the command shall be terminated with a RESERVATION CONFLICT status. If a reservation conflict precludes any part of the command, none of the command shall be performed.

6.10 RESERVE ELEMENT (10) command

The RESERVE ELEMENT (10) command (see table 24) is used to reserve a logical unit or, if the element reservation option is implemented, elements within a logical unit. This subclause describes only those instances where the RESERVE ELEMENT (10) command differs from the RESERVE ELEMENT (6) command. Except for the instances described in this clause, the RESERVE ELEMENT (10) command shall function exactly like the RESERVE ELEMENT (6) command (see 6.9).

Table 24 — RESERVE ELEMENT (10) command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (56h)							
1	Reserved			3RDPTY	Reserved		LONGID	ELEMENT
2	RESERVATION IDENTIFICATION							
3	Reserved							
4								
5								
6								
7	PARAMETER LIST LENGTH							
8								
9	CONTROL							

The RESERVE ELEMENT (10) and RELEASE ELEMENT (10) commands provide the basic mechanism for contention resolution in multiple-initiator systems. The third-party reservation allows logical units or elements to be reserved for another specified SCSI device.

If the RESERVE ELEMENT (10) command is implemented, then the RELEASE ELEMENT (10) also shall be implemented. Element reservations are not allowed in attached medium changer usage.

NOTE 11— The ELEMENT bit is in the same position as the EXTENT bit in the SPC definition of RESERVE (10). An attached medium changer with a primary device that supports extents can not distinguish element from extent reservations.

6.10.1 Third-party reservation (Mandatory)

The third-party reservation option of the RESERVE command allows an application client to reserve a logical unit or elements within a logical unit for another SCSI device.

If the third-party (3RDPTY) bit is zero, the third-party reservation option is not requested. If the 3RDPTY bit is one, the RESERVE command shall reserve the specified unit or elements for the SCSI device specified in the THIRD PARTY DEVICE ID field. The target shall preserve the reservation until it is superseded by another valid RESERVE command from the initiator that made the reservation or until it is released by the same initiator, by a TARGET RESET task management function from any initiator, by a hard reset condition, or by a power on cycle. The target shall ignore any attempt to release the reservation made by any other initiator.

After a third–party reservation has been granted, the initiator that sent the RESERVE command shall be treated like any other initiator. Reservation conflicts shall occur in all cases where another initiator is not allowed access due to the reservation.

6.10.2 Superseding reservations

An initiator that holds a current reservation may modify that reservation by issuing another RESERVE command to the same unit and, if the ELEMENT bit is one, using the same reservation identification. The superseding RESERVE command shall release the previous reservation in place when the new reservation request is granted. The current reservation shall not be modified if the new reservation request cannot be granted. If the superseding reservation cannot be granted because of conflicts with a previous active reservation (other than the reservation being superseded), the target shall return RESERVATION CONFLICT status.

NOTE 10— Superseding reservations are principally intended to allow the SCSI device ID to be changed on a reservation using the third-party reservation option.

The format of THIRD PARTY DEVICE ID is protocol specific. If the THIRD PARTY DEVICE ID value associated with the reservation is smaller than 255, the LONGID bit may be zero and the THIRD PARTY DEVICE ID value sent in the CDB. If the LONGID bit is zero, the PARAMETER LIST LENGTH field shall be set to zero. If the THIRD PARTY DEVICE ID value is greater than 255, the LONGID bit shall be one.

If the LONGID bit is one, the THIRD PARTY DEVICE ID field in the CDB shall be ignored. If the LONGID bit is one, the PARAMETER LIST LENGTH field shall be at least eight. If the LONGID bit is one and the PARAMETER LIST LENGTH field is less than eight, the device server shall return a CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code PARAMETER LIST LENGTH ERROR.

Device servers that support device IDs greater than 255 shall accept commands with LONGID equal to one. Device servers whose devices IDs are limited to 255 or smaller may reject commands with LONGID equal to one with CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID FIELD IN CDB.

If both the LONGID and ELEMENT bits are one, then the parameter list shall have the format shown in table 25 and the extent list length shall be the PARAMETER LIST LENGTH minus eight.

Table 25— RESERVE ELEMENT (10) ID & elements parameter list

Bit / Byte	7	6	5	4	3	2	1	0
0	THIRD PARTY DEVICE ID							
7								
8	Element Descriptors (see table 23)							
n								

6.11 SEND VOLUME TAG command

The SEND VOLUME TAG command (see table 26) transfers a volume tag template to be used for a search of existing volume tag information or new volume tag information for one media changer element address. The function of the command is conveyed by the SEND ACTION CODE field value. The REQUEST VOLUME ELEMENT ADDRESS command may be used to transfer the results of a translate search operation.

A reservation conflict shall occur if a SEND VOLUME TAG command is received from an initiator other than the one holding a logical unit or element reservation.

Support for this command is optional for independent medium changers. This command may have command parameter data. No command response data is returned.

Table 26 — SEND VOLUME TAG command

Byte \ Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (B6h)							
1	Reserved				ELEMENT TYPE CODE			
2	ELEMENT ADDRESS							
3								
4	Reserved							
5	Reserved			SEND ACTION CODE				
6	Reserved							
7								
8	PARAMETER LIST LENGTH							
9								
10	Reserved							
11	CONTROL							

The ELEMENT TYPE CODE field specifies an element type specification as defined in the READ ELEMENT STATUS command (see table 9). If the SEND ACTION CODE indicates a translate operation, this field indicates the element types to be searched. If the value is zero, all element types are candidates for a translate operation. If the SEND ACTION CODE does not indicate a translate, this field shall be treated as reserved.

The ELEMENT ADDRESS field gives a medium changer element address whose interpretation depends on the SEND ACTION CODE field. When the SEND ACTION CODE is a translate, the element address field gives the starting element to be examined for satisfaction of the search criteria. When the SEND ACTION CODE is assert, replace, or undefine, the ELEMENT ADDRESS field gives the specific element address where volume tag information for a volume is to be modified.

The SEND ACTION CODE field gives the function to be performed by this command as listed in table 27.

Table 27 — Send action codes

Code	Description
0h	Translate - search all defined volume tags
1h	Translate - search only primary volume tags
2h	Translate - search only alternate volume tags
3h	Reserved
4h	Translate - search all defined tags - ignore sequence numbers
5h	Translate - search primary tags - ignore sequence numbers
6h	Translate - search alternate tags - ignore sequence numbers
7h	Reserved
8h	Assert - as the primary volume tag - if tag now undefined
9h	Assert - as the alternate volume tag - if tag now undefined
Ah	Replace - the primary volume tag - current tag ignored
Bh	Replace - the alternate volume tag - current tag ignored
Ch	Undefine - the primary volume tag - current tag ignored
Dh	Undefine - the alternate volume tag - current tag ignored
Eh – 1Bh	Reserved
1Ch – 1Fh	Vendor-specific

Translate operations request that the logical unit search the volume tag information available for volumes at defined element addresses for volume tag information that matches the template given by the command parameter data. The resulting information may be reported via the REQUEST VOLUME ELEMENT ADDRESS command.

Assert operations define volume tag information for a single volume at an element address that does not currently have defined volume tag information. A CHECK CONDITION status with the sense key set to ILLEGAL REQUEST is reported if the volume at the selected element address already has defined volume tag information. In this case, the original volume tag information shall not be changed. Support for this field set to an assert function value is optional.

Replace operations define or overwrite volume tag information for a single volume at one element address. Any previously defined volume tag information is overwritten. Support for this field set to a replace function value is optional.

Undefine operations cause any previously defined volume tag information for the volume at the specified element address to be cleared. It shall not be considered an error to undefine volume tag information that was not previously defined. Support for this field set to an undefine function value is optional.

If a logical unit implements volume tag information, it may choose to not implement the functions that modify volume tag information. For such an implementation a request for any assert, replace or undefine function shall cause the SEND VOLUME TAG command to be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST.

The PARAMETER LIST LENGTH field shall be zero for undefine functions. The volume tag information sent as command parameter data for translate, assert and replace functions is defined in table 28.

Table 28 — Send volume tag parameters format

Bit / Byte	7	6	5	4	3	2	1	0
0 - 31	VOLUME IDENTIFICATION TEMPLATE							
32 - 33	Reserved							
34 - 35	MINIMUM VOLUME SEQUENCE NUMBER							
36 - 37	Reserved							
38 - 39	MAXIMUM VOLUME SEQUENCE NUMBER							

The VOLUME IDENTIFICATION TEMPLATE field specifies a search template for translate functions and the exact value of the new volume identification information for other SEND VOLUME TAG command functions.

As a search template, this field may contain the wildcard characters '?' and '*' (3Fh and 2Ah).

– '?' will match any single character;

– '*' will match any string of characters. When it appears in a template the remainder of the template at higher offsets in the field is not used.

For an assert, replace, or undefine function, this field shall not contain the '?' or '*' wildcard characters.

The MINIMUM VOLUME SEQUENCE NUMBER field specifies the new sequence number for the assert and replace functions. For a translate, this field specifies the least value in the volume sequence number field of the volume tag information that will meet the search specification.

The MAXIMUM VOLUME SEQUENCE NUMBER field specifies the maximum value in a volume sequence number field of the volume tag information that meets the search specification. This field is ignored for assert, replace and undefine functions.

7 Parameters

This clause defines the parameters used for independent medium changers devices. Attached medium changers shall use the parameters defined for the primary device type.

7.1 Diagnostic parameters

This subclause defines the descriptors and pages for diagnostic parameters used with independent medium changer devices. Attached medium changers shall use the descriptors and pages defined for the primary device type.

The diagnostic page codes for independent medium changer devices are defined in table 29.

Table 29 — Diagnostic page codes

Page code	Description	Subclause
00h	Supported diagnostic page	SPC
01h – 3Fh	Reserved (for all device type pages)	
40h – 7Fh	Reserved	
80H – FFh	Vendor-specific pages	

7.2 Log parameters

This subclause defines the descriptors and pages for log parameters used with independent medium changer devices. Attached medium changers shall use the descriptors and pages defined for the primary device type.

The log page codes for independent medium changer devices are defined in table 30.

Table 30 — Log page codes

Page code	Description	Subclause
00h	Supported log pages	SPC
06h	Non medium error page	SPC
07h	Last n error events page	SPC
06h	Non medium error page	SPC
01H – 05h	Reserved	
08H – 2Fh	Reserved	
30H – 3Eh	Vendor-specific pages	

7.3 Mode parameters

This subclause defines the descriptors and pages for mode parameters used with independent medium changer devices. Attached medium changers shall only return pages and descriptors defined for the primary device type.

The mode parameter list, including the mode parameter header and mode block descriptor, are defined in SPC.

The MEDIUM TYPE code field is contained in the mode parameter header. This field is reserved for independent medium changer devices.

The DEVICE–SPECIFIC PARAMETER field is contained in the mode parameter header. This field is reserved for independent medium changer devices.

The DENSITY CODE field is contained in the mode parameter block descriptor. This field is reserved for independent medium changer devices.

The mode page codes for independent medium changer devices are shown in table 31.

Table 31 — Mode page codes

Page code	Description	Subclause
1Fh	Device capabilities	7.3.1
1Dh	Element address assignment	7.3.2
1Eh	Transport geometry parameters	7.3.3
01H – 1Ch	Reserved	
00h	Vendor-specific (does not require page format)	
20H – 3Eh	Vendor-specific (page format required)	
3Fh	Return all pages (valid only for the MODE SENSE command)	SPC

7.3.1 Device capabilities page

The device capabilities page (see table 32) defines characteristics of the element types of an independent medium changer. Attached medium changers shall not return this page. This information may be employed by the initiator to determine functions permitted by the MOVE MEDIUM and EXCHANGE MEDIUM commands.

Table 32 — Device capabilities page

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	RSVD	PAGE CODE (1Fh)					
1	PARAMETER LENGTH (12h)							
2	Reserved				STORDT	STORI/E	STORST	STORMT
3	Reserved							
4	Reserved				MT->DT	MT->I/E	MT->ST	MT->MT
5	Reserved				ST->DT	ST->I/E	ST->ST	ST->MT
6	Reserved				I/E->DT	I/E->I/E	I/E->ST	I/E->MT
7	Reserved				DT->DT	DT->I/E	DT->ST	DT->MT
8 - 11	Reserved							
12	Reserved				MT<>DT	MT<>I/E	MT<>ST	MT<>MT
13	Reserved				ST<>DT	ST<>I/E	ST<>ST	ST<>MT
14	Reserved				I/E<>DT	I/E<>I/E	I/E<>ST	I/E<>MT
15	Reserved				DT<>DT	DT<>I/E	DT<>ST	DT<>MT
16 - 19	Reserved							

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a nonvolatile, vendor-specific location.

The field names in table 32 use the following element type abbreviations:

- MT — a medium transport element,
- ST — a storage element,
- I/E — an import/export element, and
- DT — a data transfer element.

In the descriptions, XX and YY are any of the element type abbreviations.

A STORXX bit value of one indicates that the defined elements of type XX may provide independent storage for a unit of media. A value of zero indicates that elements of type XX provide virtual sources or destinations, that the location of the unit of media is provided by an element of some other type. The value of STORST is one by the definition of that type (see 5.5).

An XX->YY bit value of one indicates that the medium changer device supports all MOVE MEDIUM commands where the source is element type XX, the destination is element type YY and these element addresses are otherwise valid. An XX->YY bit value of zero indicates that these MOVE MEDIUM commands may or may not be valid depending on the particular elements requested. Those which are not valid will be rejected with ILLEGAL REQUEST.

An XX<>YY bit value of one indicates that the medium changer device supports all EXCHANGE MEDIUM commands where the source is element type XX, FIRST DESTINATION ADDRESS is element type YY, SECOND DESTINATION ADDRESS is the same type as the source element type and these element addresses are otherwise valid. An XX<>YY bit value of zero indicates that these EXCHANGE MEDIUM commands may or may not be valid depending on the particular elements requested. Those which are not valid will be rejected with ILLEGAL REQUEST.

7.3.2 Element address assignment page

The element address assignment page (see table 33) is used to assign addresses to the elements of the independent medium changer (MODE SELECT) and to report those assignments (MODE SENSE). This page also defines the number of each type of element present. An attached medium changer shall not return this page.

Table 33 — Element address assignment page

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	RSVD	PAGE CODE (1Dh)					
1	PARAMETER LIST LENGTH							
2 / 3	FIRST MEDIUM TRANSPORT ELEMENT ADDRESS							
4 / 5	NUMBER OF MEDIUM TRANSPORT ELEMENTS							
6 / 7	FIRST STORAGE ELEMENT ADDRESS							
8 / 9	NUMBER OF STORAGE ELEMENTS							
10 / 11	FIRST IMPORT/EXPORT ELEMENT ADDRESS							
12 / 13	NUMBER OF IMPORT/EXPORT ELEMENTS							
14 / 15	FIRST DATA TRANSFER ELEMENT ADDRESS							
16 / 17	NUMBER OF DATA TRANSFER ELEMENTS							
18 / 19	Reserved							

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a nonvolatile vendor-specific location.

The FIRST MEDIUM TRANSPORT ELEMENT ADDRESS field identifies the first medium transport element contained in the medium changer (other than the default medium transport address of zero). The NUMBER OF MEDIUM TRANSPORT ELEMENTS field defines the total number of medium transport elements contained in the medium changer. If the NUMBER OF MEDIUM TRANSPORT ELEMENTS field in a MODE SELECT command is greater than the default value returned in the MODE SENSE parameter data, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code PARAMETER VALUE INVALID.

The FIRST STORAGE ELEMENT ADDRESS field identifies the first medium storage element contained in the medium changer. The NUMBER OF STORAGE ELEMENTS field defines the total number of medium storage elements contained in the medium changer. If the NUMBER OF MEDIUM STORAGE ELEMENTS field in a MODE SELECT command is greater than the default value returned in the MODE SENSE parameter data, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code PARAMETER VALUE INVALID.

The FIRST IMPORT/EXPORT ELEMENT ADDRESS field identifies the first medium portal that is accessible both by the medium transport devices and also by an operator from outside the medium changer. The NUMBER OF IMPORT/EXPORT ELEMENTS field defines the total number of import/export elements contained in the medium changer and accessible to the medium transport elements. If the NUMBER OF IMPORT/EXPORT ELEMENTS field in a MODE SELECT command is greater than the default value returned in the MODE SENSE parameter data, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code PARAMETER VALUE INVALID.

The FIRST DATA TRANSFER ELEMENT ADDRESS field identifies the first data transfer element contained in the medium changer. The data transfer elements may be either read/write or read-only devices. The NUMBER OF DATA TRANSFER ELEMENTS field defines the total number of data transfer elements contained within the medium changer and accessible to the medium transport elements. If the NUMBER OF DATA TRANSFER ELEMENTS field in a MODE SELECT command is greater than the default value returned in the MODE SENSE parameter data, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code PARAMETER VALUE INVALID.

NOTE 12 — The number of import/export elements or data transfer elements may be zero. The number of storage elements may only be zero when there is at least one import/export element with independent storage capability (see 5.5).

Each element in the medium changer must have a unique address. If the address ranges defined for any of the element types overlap, the target shall return CHECK CONDITION status. The sense key shall be ILLEGAL REQUEST and the additional sense code INVALID ELEMENT ADDRESS.

7.3.3 Transport geometry parameters page

The transport geometry parameters page (see table 34) defines whether each medium transport element of an independent medium changer is a member of a set of elements that share a common robotics subsystem and whether the element is capable of media rotation. One transport geometry descriptor is transferred for each medium transport element, beginning with the first medium transport element. (Other than the default transport element address of zero.) An attached medium changer shall not return this page.

Table 34 — Transport geometry parameters page

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	RSVD	PAGE CODE (1Eh)					
1	PARAMETER LENGTH (n-1)							
2 ... n	Transport geometry descriptor(s)							

The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a nonvolatile vendor-specific location.

The PARAMETER LENGTH specifies the number of bytes of transport geometry descriptors that follow. The geometry of each medium transport element is defined using a two-byte field as defined by table 35.

Table 35 — Transport geometry descriptor

Bit / Byte	7	6	5	4	3	2	1	0
0	Reserved							ROTATE
1	MEMBER NUMBER IN TRANSPORT ELEMENT SET							

A ROTATE bit of one indicates that the medium transport element supports media rotation for handling double-sided media. A ROTATE bit of zero indicates that the medium transport element does not support media rotation.

The MEMBER NUMBER IN TRANSPORT ELEMENT SET field indicates the position of this element in a set of medium transport elements that share a common robotics subsystem. The first element in a set has a MEMBER NUMBER IN TRANSPORT ELEMENT SET of zero.

NOTE 13 — This page reports information about the way transport elements are physically clustered in a system. The model for this is a medium changer device with more than one independent robotics subsystem, where each of these supports multiple transport elements. The elements that are supported by a particular robotics subsystem form a set. This sort of information is helpful for optimization and error recovery in such a large system. (The individual transport element is addressed [not the robotics subsystem]. An element is defined to be a place where a unit of media may be at any point in time. See clause 5.)

Annex A
(informative)
Bibliography

ANSI X3.131-1994, *Information systems – Small computer system interface - 2*
ANSI X3.304-1996, *Information technology – SCSI-3 multi-media commands*
ANSI NCITS 306-199X, *Information technology – SCSI-3 block commands*
ANSI NCITS T10/997D, *Information technology – SCSI-3 stream commands*