ENDL
TEXAS

Date: 20 June 2008
To: T10 Technical Committee
From: Ralph O. Weber
Subject: OSD-2 Snapshots and related enhancements

Introduction

The SNIA OSD TWG *Snapshots Proposal* describes the ability to replicate entire partitions in the form of snapshots, clones, or both. Discussions subsequent to development of the *Snapshots Proposal* have identified several additional features that are useful or necessary to effect the snapshots concept in SCSI.

The following features are proposed for addition to OSD-2:

- Snapshots/Clones as described in the *Snapshots Proposal* (see change 1), with the following exceptions:
  - All features related to the REPAIR bit, and
  - Multi-capability security with usage of the CDB continuation described in 08-185 employed instead of what is described in the *Snapshots Proposal*
- Definition of the well known collections model and a well known collection whose membership is the equivalent of the output from a LIST command (see change 2)
- Definition of names (LINKED and TRACKING) for existing collection types, plus a LIST for the well known collection whose membership is the equivalent of the output from a LIST command (see change 2)
- Changes in details of how the various collection types are specified that are not intended to substantially change the way the collection types work (see change 2)
- Definition of a Command Tracking attributes page and is usage for tracking commands that affect multiple objects with sufficient generalization to cover both the existing multi-object commands and snapshot commands (see change 2)
- Definition of an IMMED_TR bit that provides speedy completion of long-running commands similar, but not identical to the immed bit defined in other SCSI standards (see change 3)
- The failure of the REMOVE command definition to mention collection membership is corrected (see change 3)

Note: All CDB formats in this proposal assume that 08-185 is approved for incorporation prior to or concurrently with this proposal.

Revision History

r0 Initial revision
r1 Complete proposal as originally intended & update to match 08-158r1

Since r1 is effectively an initial draft (or at least the completion of the initial draft), there are no change bars in r1.

Unless otherwise indicated additions are shown in blue, deletions in ~~red strikethrough~~, and comments in green.

In some instances text is moved from its current subclause to another, sometimes new, subclause. When this occurs, the text is shown in ~~purple strikethrough~~ where it is removed and purple (no strikethrough) where it is inserted.

18484 Preston Road, Suite 102 PMB 178, Dallas, TX 75252 214-912-1373 972-596-2775 Fx ENDL_TX@computer.org

Change 1 – Snapshots and Clones

Description

The snapshot system described in the SNIA OSD TWG *Snapshots Proposal* are specified here.

The permissions changes that support the commands defined in change 1 appear in change 10 so that all of the permissions changes for all of the commands defined in this proposal can be reviewed as a group.

Proposed changes in OSD-2 r03

4.d Object duplication

{{All of 4.d is new. The use of change markups is suspended for the remainder of 4.d. It is suggested that 4.d be placed between 4.11 (Policy/Storage management) and 4.12 (Security).}}

4.d.1 Overview

The following mechanisms are defined for duplicating the data and attributes contained in one or more user objects and collections in new user objects and collections:

a) The CREATE SNAPSHOT command (see 4.d.2);
b) The CREATE CLONE command (see 4.d.2);
c) The REFRESH SNAPSHOT command (see 4.d.2);
d) The RESTORE PARTITION FROM SNAPSHOT command (see 4.d.2); and
e) The COPY USER OBJECTS command (see 6.h) {{see 08-185.}}.

A model for the partition snapshot and clone mechanisms appears in 4.d.2.

The COPY USER OBJECTS command: … {{see 08-185.}}

4.d.2 Snapshot partitions and clone partitions

4.d.2.1 Overview

The following commands create, update, manage, and remove copies of all the user objects, collections, and attributes between two or more partitions:

a) The CREATE SNAPSHOT command (see 4.d.2.2, 4.d.2.4, and 6.e);
b) The CREATE CLONE command (see 4.d.2.3 and 6.d);
c) The REFRESH SNAPSHOT command (see 6.r);
d) The RESTORE PARTITION FROM SNAPSHOT command (see 6.s);
e) The DETACH CLONE command (see 4.d.2.5 and 6.f); and
f) The REMOVE PARTITION command (see 6.27).

In the context of snapshots and clones, the following types of partitions are identified:

a) Primary (i.e., not a snapshot or a clone);
b) Snapshot; and
c) Clone.

Snapshots and clones are partitions that are full copies of a source partition. Other similarities and differences between snapshots and clones are shown in table x200.

Table x200 — Comparison of snapshots and clones

Feature		Partition type		
		Primary	**Snapshot**	**Clone**
Allowed to be the source partition for a CREATE SNAPSHOT command (see 6.e)		Yes	No	Yes
Allowed to be the source partition for a CREATE CLONE command (see 6.f)		No	Yes	No
Allowed use in a REFRESH SNAPSHOT command (see 6.r)	Source	Yes	No	Yes
	Destination	No	Yes	No
Allowed use in a RESTORE PARTITION FROM SNAPSHOT command (see 6.s)	Source	No	Yes	No
	Destination	Yes	No	Yes
Allowed to be the partition specified in a REMOVE PARTITION command (see 6.27)		Yes	Yes	Yes
Allowed to be the partition specified in a DETACH CLONE command (see 6.f)		No	No	Yes
Time ordered history (i.e., chain) of partition duplicates maintained		No	Yes	No
The source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) indicates the primary partition from which this partition is descended		n/a	Yes	Yes
Writable in normal (i.e., non error recovery) usage		Yes	No	Yes

4.d.2.2 Snapshot history chains

The device server maintains snapshot forward attribute and the snapshot backward attribute in the Snapshots Information attributes page (see 7.1.2.e) to form a double linked chain of the snapshot partitions descended from a primary or clone partition. The attribute values are:

a) The Partition_ID (see 4.6.2) of the partition to which the history chain attribute points; or
b) Zero or an undefined attribute (see 3.1.50) to indicate no history chain linkage exists.

An application client may trace backward or forward in time using these attributes.

Figure x1 shows the history chain for a primary partition with a single snapshot. The conditions show in figure x1 would be present after a CREATE SNAPSHOT command (see 6.e) with partition 1 as the source partition and partition 2 as the destination. Since the contents of snapshot is fixed in time while the primary partition continues to evolve, time might be view as flowing in the direction of the arrow.

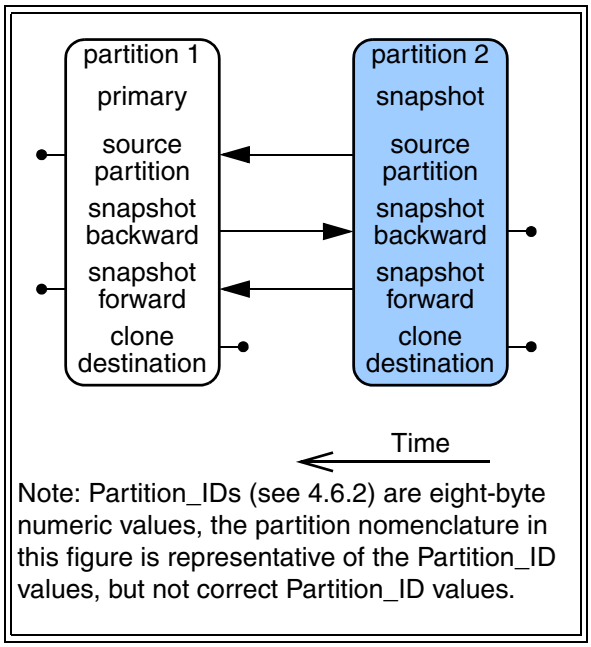


Figure x1 — Snapshot history chain after first CREATE SNAPSHOT command

The attributes in the Snapshots Information attributes page (see 7.1.2.e) that maintain the history chain for figure x1 are summarized in table x201.

Table x201 — Snapshot history chain attributes for one CREATE SNAPSHOT command

Attribute	Partition 1 primary	Partition 2 snapshot
source partition	undefined [a] or zero	partition 1
snapshot backward	partition 2	undefined [a] or zero
snapshot forward	undefined [a] or zero	partition 1
clone destination	undefined [a] or zero	undefined [a] or zero
Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this table is representative of the Partition_ID values, but not correct Partition_ID values.		
[a] See 3.1.50.		

If a second snapshot is taken at a later time, the history chain would become as shown in figure x2.

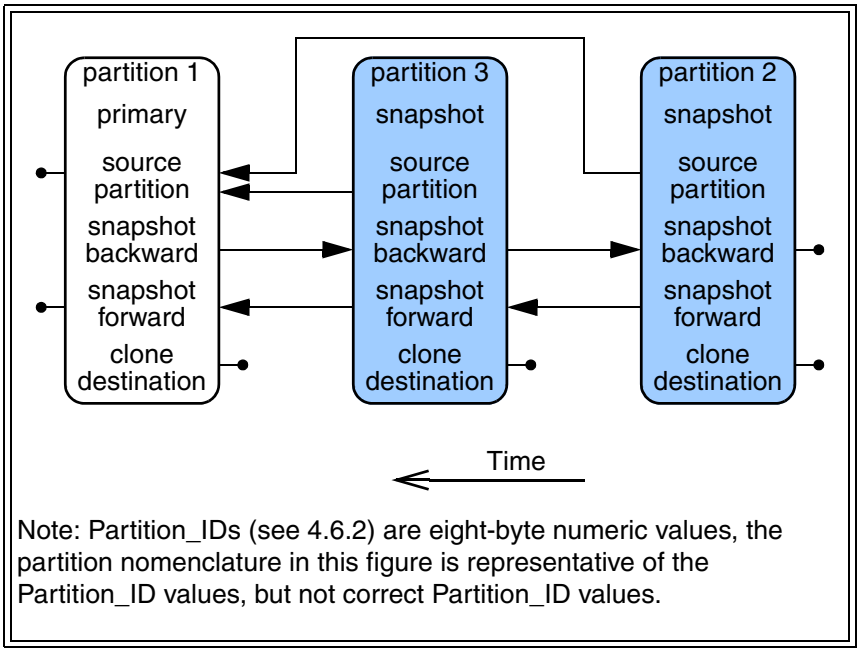


Figure x2 — Snapshot history chain after second CREATE SNAPSHOT command

The attributes in the Snapshots Information attributes page (see 7.1.2.e) that maintain the history chain for figure x2 are summarized in table x202.

Table x202 — Snapshot history chain attributes for two CREATE SNAPSHOT commands

Attribute	Partition 1 primary	Partition 3 snapshot	Partition 2 snapshot
source partition	undefined [a] or zero	partition 1	partition 1
snapshot backward	partition 3	partition 2	undefined [a] or zero
snapshot forward	undefined [a] or zero	partition 1	partition 3
clone destination	undefined [a] or zero	undefined [a] or zero	undefined [a] or zero
Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this table is representative of the Partition_ID values, but not correct Partition_ID values.			
[a] See 3.1.50.			

4.d.2.3 Clone chains

Because clone partitions are writable, they evolve over time in the same way that the partition from which the snapshot was taken evolves. Therefore, no certain trampler relationship is possible with clone partitions.

The clone destination attribute and source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) provide the only linkage between clone partitions and the snapshot partitions from which they are derived.

The presence of multiple clone destination attributes in the Snapshots Information attributes page allows more than one clone partition to be created from a single snapshot partition.

Figure x3 builds on figure x2 (see 4.d.2.2) to show the effects of a CREATE CLONE command (see 6.d) with partition 3 as the source partition and partition 4 as the destination.

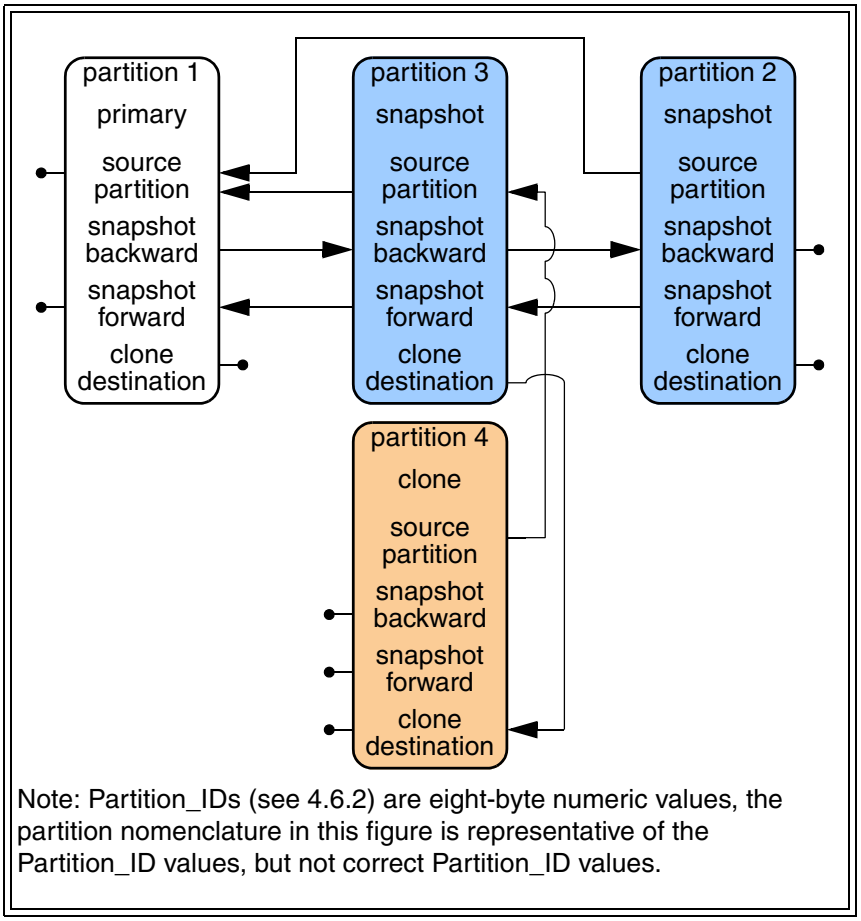


Figure x3 — Snapshot/clone chains after a first CREATE CLONE command

The attributes in the Snapshots Information attributes page (see 7.1.2.e) that maintain the clone chain for figure x3 are summarized in table x203.

Table x203 — Snapshot/clone chain attributes for one CREATE CLONE command

Attribute	Partition 1 primary	Partition 2 snapshot	Partition 3 snapshot	Partition 4 clone
source partition	see table x202 in 4.d.2.2		see table x202 in 4.d.2.2	partition 3
snapshot backward				undefined [a] or zero
snapshot forward				undefined [a] or zero
clone destination			partition 4	undefined [a] or zero
Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this table is representative of the Partition_ID values, but not correct Partition_ID values.				
[a] See 3.1.50.				

The effects of a second CREATE CLONE command (see 6.d) with partition 3 as the source are shown in figure x4.

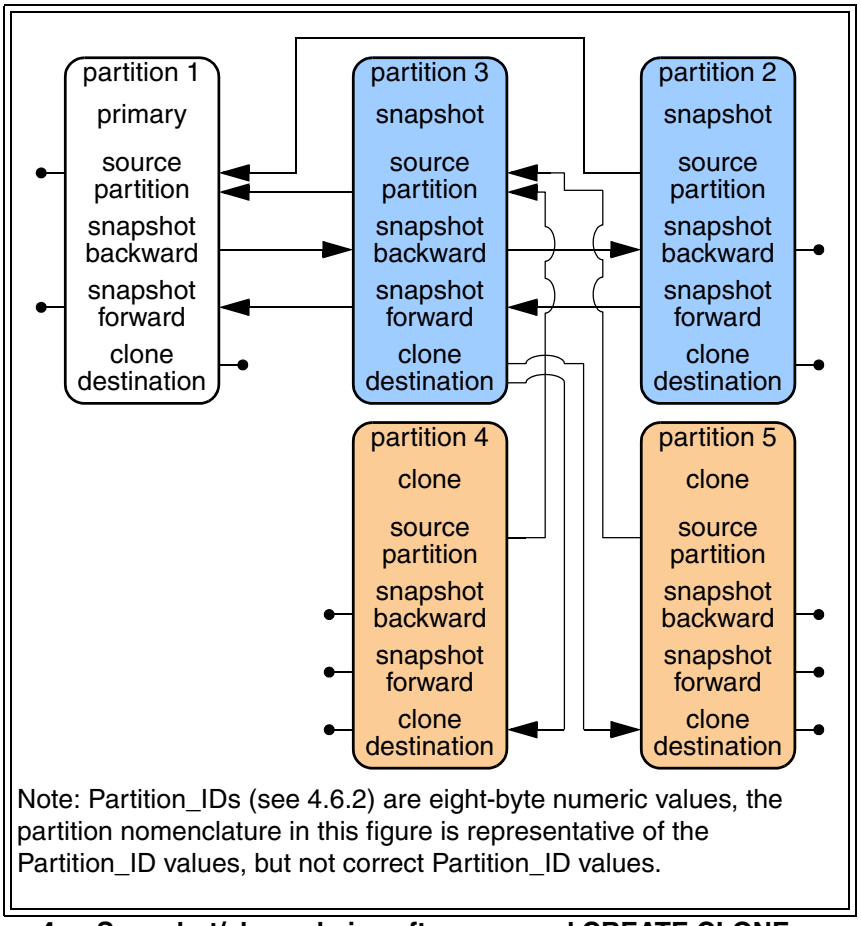


Figure x4 — Snapshot/clone chains after a second CREATE CLONE command

The attributes in the Snapshots Information attributes page (see 7.1.2.e) that maintain the clone chain for figure x4 are summarized in table x204.

Table x204 — Snapshot/clone chain attributes for two CREATE CLONE commands

Attribute	Partition 1 primary	Partition 2 snapshot	Partition 3 snapshot	Partition 4 clone	Partition 5 clone
source partition	see table x202 in 4.d.2.2		see table x202 in 4.d.2.2	partition 3	partition 3
snapshot backward				undefined [a] or zero	
snapshot forward				undefined [a] or zero	
clone destination [b]			partition 4	undefined [a] or zero	
clone destination [b]			partition 5	undefined [a] or zero	

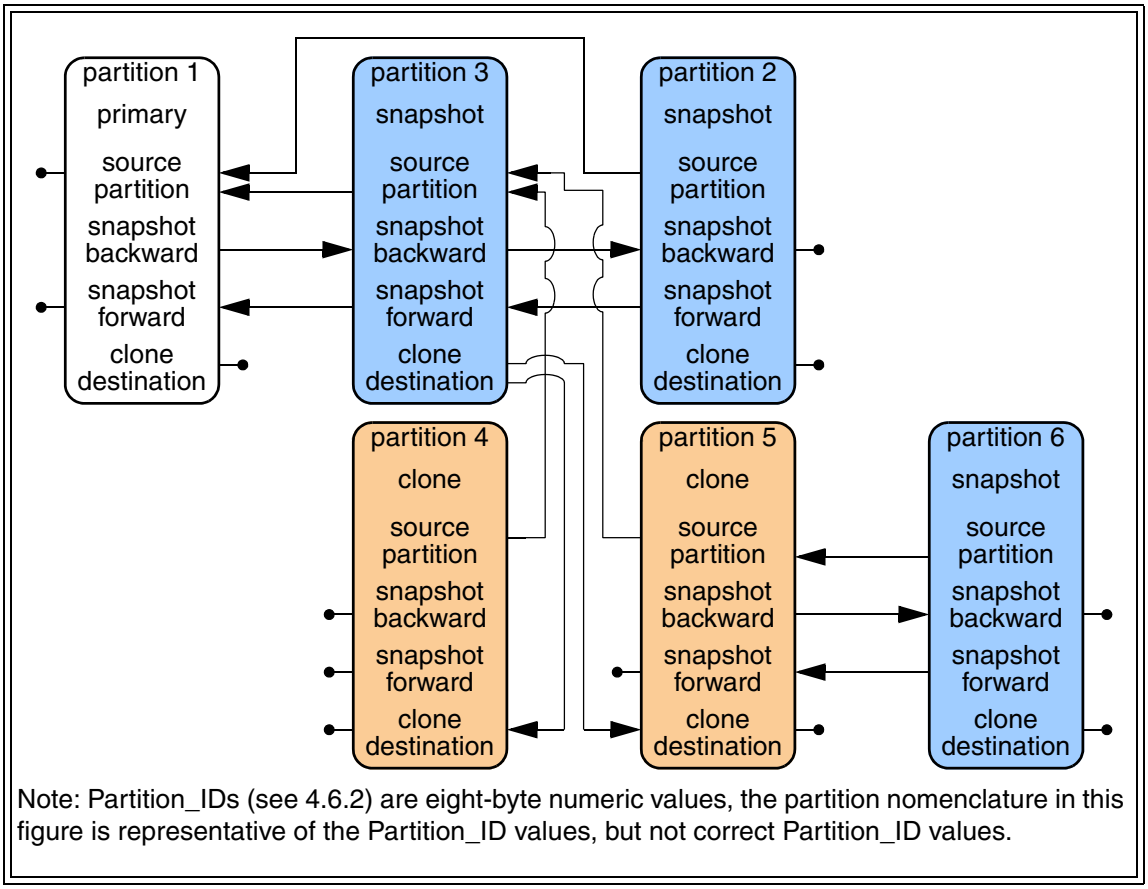
Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this table is representative of the Partition_ID values, but not correct Partition_ID values.

[a] See 3.1.50.

[b] The Snapshots Information attributes page (see 7.1.2.e) defines several attribute numbers with the name clone destination. Each clone destination attribute points to a different clone partition.

4.d.2.4 Snapshots of clones

A clone partition may be specified as the source partition for a CREATE SNAPSHOT command (see 6.e). Figure x5 builds on figure x4 (see 4.d.2.3) to show the effects of a CREATE SHAPSHOT command with clone partition 5 as the source partition and partition 6 as the destination.



Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this figure is representative of the Partition_ID values, but not correct Partition_ID values.

Figure x5 — Snapshot/clone chains after a CREATE SNAPSHOT command on a clone partition

The attributes in the Snapshots Information attributes page (see 7.1.2.e) that maintain the new history chain in figure x5 are summarized in table x205.

Table x205 — Snapshot/clone chain attributes for a CREATE SNAPSHOT command on a clone partition

Attribute	Partition 1 primary	Partitions 2 and 3 snapshots	Partition 4 clone	Partition 5 clone	Partition 6 snapshot
source partition	see table x202 in 4.d.2.2	see table x204 in 4.d.2.3		partition 3	partition 5
snapshot backward				partition 6	undefined [a] or zero
snapshot forward				undefined [a] or zero	partition 5
clone destination				undefined [a] or zero	undefined [a] or zero

Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this table is representative of the Partition_ID values, but not correct Partition_ID values.

[a] See 3.1.50.

4.d.2.5 Detaching a clone partition

A clone partition may be detached from its source partition using a DETACH CLONE command (see 6.f). Figure x6 builds on figure x5 (see 4.d.2.4) to show the effects of a DETACH CLONE command for partition 5.

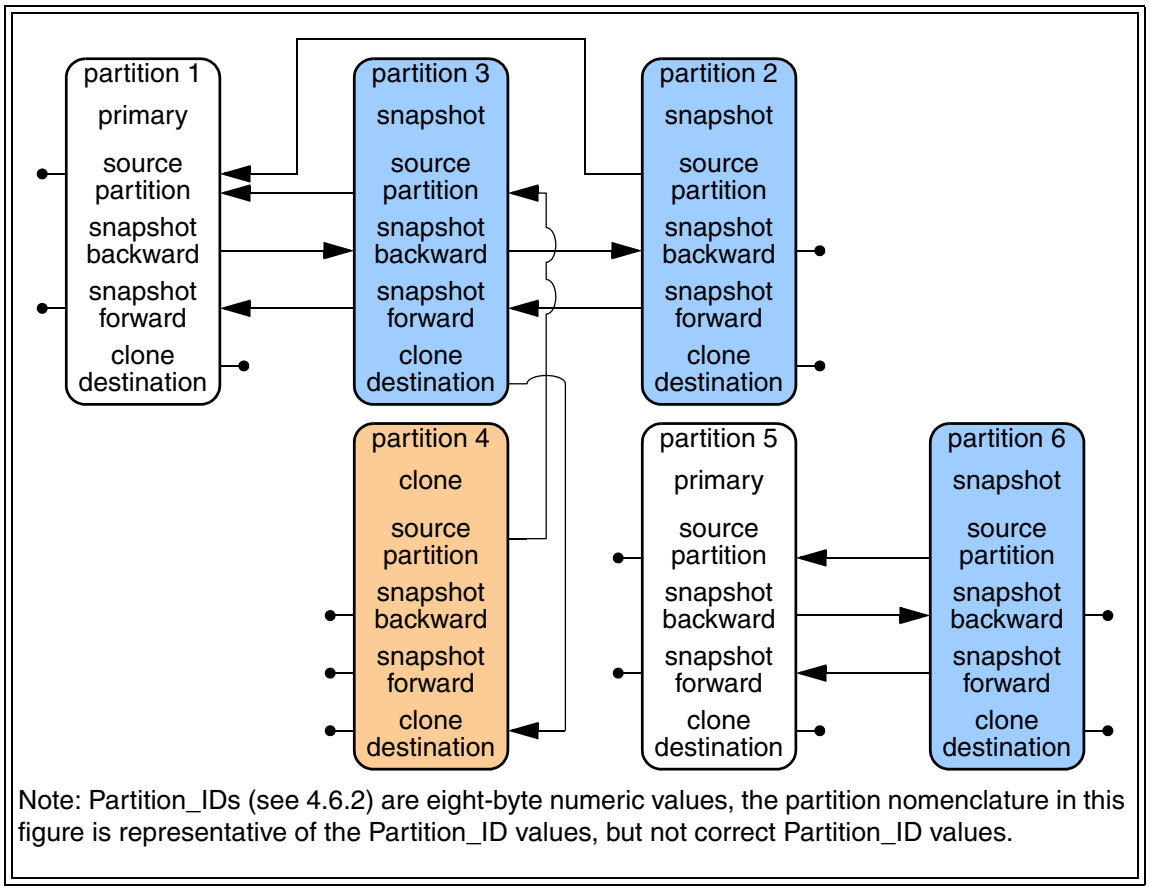


Figure x6 — Snapshot/clone chains after a DETACH CLONE command

The attributes in the Snapshots Information attributes page (see 7.1.2.e) that maintain the two history chains shown in figure x6 are summarized in table x206.

Table x206 — Snapshot/clone chain attributes after a DETACH CLONE command

Attribute	Partition 1 primary	Partitions 2 and 3 snapshots	Partition 4 clone	Partition 5 primary	Partition 6 snapshot
source partition	see table x202 in 4.d.2.2	see table x204 in 4.d.2.3		undefined [a] or zero	partition 5
snapshot backward				partition 6	undefined [a] or zero
snapshot forward				undefined [a] or zero	partition 5
clone destination				undefined [a] or zero	undefined [a] or zero
Note: Partition_IDs (see 4.6.2) are eight-byte numeric values, the partition nomenclature in this table is representative of the Partition_ID values, but not correct Partition_ID values.					
[a] See 3.1.50.					

…

4.d.3 Object duplication methods {{see 08-185.}}

4.d.4 Object duplication state management {{see 08-185.}}

4.d.4.1 Time of duplication source object management {{see 08-185.}}

4.d.4.2 Source object freeze duplication management {{see 08-185.}}

4.d.5 Object duplication space accounting {{see 08-185.}}

…

6.d CREATE CLONE {{All of 6.d is new. Change markups suspended.}}

6.d.1 Overview

The CREATE CLONE command (see table x207) causes the OSD device server to allocate and initialize a destination partition as a clone partition (see 4.d.2) and then copy all user objects, collections, and attributes from a source partition to the newly created clone partition.

Table x207 — CREATE CLONE command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)			SERVICE ACTION (88A8h)				
9								(LSB)
10	Reserved			DPO	FUA	ISOLATION		
11	IMMED_TR	Reserved	GET/SET CDBFMT		Reserved			
12	TIMESTAMPS CONTROL							
13	FREEZE	Reserved			TIME OF DUPLICATION			
14	DUPLICATION METHOD							
15	Reserved							
16	(MSB)			SOURCE PARTITION_ID				
23								(LSB)
24	(MSB)			REQUESTED DESTINATION PARTITION_ID				
31								(LSB)
32				Reserved				
47								
48	(MSB)			CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				
51								(LSB)
52				Get and set attributes parameters (see 5.2.4)				
79								
80				Capability (see 5.2.c) {{in 08-185}}				
183								
184				Security parameters (see 5.2.8)				
235								

The contents of the DPO bit and the FUA bit are defined in 5.2.3.

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

If the FREEZE bit is set to zero, the CREATE CLONE command shall not modify the contents of the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition. If the FREEZE bit is set to one and source object freeze duplication management is supported (see 4.d.4.2), then the device server shall modify the contents of the object accessibility attribute in the Partition Information attributes page of the source partition as described in 6.d.2 and 6.d.4.

The TIME OF DUPLICATION field specifies which time of duplication source object management method (see 4.d.4.1) applies to the CREATE CLONE command. If the TIME OF DUPLICATION field is set to DEFAULT (see table x9 in 4.d.4.1 {{see 08-185}}), then the default clone time of duplication method attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition specifies which time of duplication management method applies to the CREATE CLONE command.

The DUPLICATION METHOD field specifies which duplication method (see 4.d.3) applies to the CREATE CLONE command. If the DUPLICATION METHOD field is set to DEFAULT (see table x8 in 4.d.3 {{see 08-185}}), then the default clone duplication method attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition specifies which duplication method applies to the CREATE CLONE command.

The SOURCE PARTITION_ID field contains the Partition_ID (see 4.6.4) of the source partition for the CREATE CLONE command.

The contents of the REQUESTED DESTINATION PARTITION_ID field specify the Partition_ID to be assigned to the created clone partition. If the REQUESTED DESTINATION PARTITION_ID field contains zero any Partition_ID may be assigned.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{in 08-158}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the CDB continuation segment (see 5.x) {{in 08-158}}:

a) Does not contain one extension capabilities CDB continuation descriptor (see 5.y.z) {{in 08-158}}; or
b) Contains any CDB continuation descriptors other than the extension capabilities CDB continuation descriptor.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14. The destination Partition_ID assigned by the CREATE SNAPSHOT command may be obtained from the Current Command attributes page (see 7.1.2.29).

The capability is defined in 4.11.2.2. The CREATE CLONE command accesses two partitions. One capability is necessary for each partition accessed. The capability with the highest value (see table 27 in 4.12.4.1) in the SECURITY METHOD field appears in the CDB. The other capability appears in the CDB continuation segment (see 5.x) {{see 08-185}}.

The security parameters are defined in 5.2.8.

The CREATE SNAPSHOT command does not initialize the partition key or the working keys (see 4.12.9.1) for the destination partition. Proper operation of any security method other than NOSEC (see 4.12.4) requires that the following commands be processed without errors before other commands are addressed to the destination partition:

a) A SET KEY command (see 6.29) that establishes the partition key; and
b) One or more SET KEY commands that establish one or more working keys for the partition.

A CREATE CLONE command whose capability (see 4.11.2.2) has the SET_ATTR bit set to one and POL/SEC bit set to one is allowed to avoid the need for SET KEY commands by setting the default security method attribute to NOSEC in the Partition Policy/Security attributes page (see 7.1.2.22) for the created partition.

If the REQUESTED DESTINATION PARTITION_ID field is not set to zero, SET KEY command are not needed to track the progress of a CREATE CLONE command with the IMMED_TR bit set to one in the following cases:

a) If the READ permission bit is set to one in the capability that allowed creation of the destination partition, that capability may be used in LIST COLLECTION commands (see 6.16) that list the contents of the snapshot/clone tracking well known collection (see 4.6.6.5.3); and
b) If the GET_ATTR permission bit is set to one in the capability that allowed creation of the destination partition, that capability may be used in GET ATTRIBUTES commands (see 6.13) or equivalents that retrieve attributes from the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection.

6.d.2 Processing before the IMMED_TR bit takes effect

A CREATE CLONE command shall not be completed with GOOD status until at least all the operations described in this subclause have been performed. These operations shall before completing the command with GOOD status even if the IMMED_TR bit is set to one.

If the FREEZE bit is set to one and source object freeze duplication management (see 4.d.4.2) is not supported, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD CDB.

If the requested time of duplication source object management method (see 4.d.4.1) is not supported or the requested duplication method (see 4.d.3) is not supported, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the SOURCE PARTITION_ID field contains zero or the Partition_ID (see 4.6.4) of a partition that does not exist, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB, if any of the follow conditions exist in the attribute values in the Snapshots Information attributes page (see 7.1.2.e) of the source partition:

a) The partition type attribute contains 00h (i.e., primary partition);
b) The partition type attribute contains 02h (i.e., clone partition);
c) The clones count attribute contains a value that is equal to the value in the maximum clones count attribute in the Root Information attributes page (see 7.1.2.8); or
d) The branch depth attribute contains a value that is equal to the value in the maximum branch depth attribute in the Root Information attributes page.

If the REQUESTED DESTINATION PARTITION_ID field contains any value other than zero and the device server is unable to assign the requested Partition_ID to the created partition, the partition shall not be created and the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The device server shall not allow the same Partition_ID to be associated with more than one partition at any point in time.

If a CREATE CLONE command causes the value in the number of partitions attribute in the Root Information attributes page (see 7.1.2.8) to exceed the value in the partition count attribute in the Root Quotas attributes page (see 7.1.2.12), then a quota error shall be generated (see 4.10.2). The quota testing principles described in 4.10.3 apply to the testing of the partition count quota.

The device server shall create the requested destination partition and initialize it as if a CREATE PARTITION command (see 6.7) were being processed.

The assigned Partition_ID shall be placed in the Partition_ID attribute in the Current Command attributes page (see 7.1.2.29). The Collection_Object_ID or User_Object_ID attribute in the Current Command attributes page shall be set to zero.

The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination partition shall be set to 8000 0001h (i.e., deny all write accesses except those of CREATE CLONE commands, but allow all read accesses).

The snapshot/clone tracking well known collection (see 4.6.6.5.3) shall be created in the destination partition, and initialized, including at least the following:

a) Every user object and collection in the source partition shall have their User_Object_ID (see 4.6.5) or Collection_Object_ID (see 4.6.6) inserted as a member of the TRACKING collection (see 4.6.6.3),
b) The Command Tracking attributes page (see 7.1.2.c) shall be initialized to include at least the following:
   A) The percent complete attribute shall be set to zero; and
   B) The command status attribute shall be set to 0001 88A8h (i.e., CREATE CLONE command in progress).

The following attributes in the Snapshots Information attributes page (see 7.1.2.e) of the source partition shall be set as follows:

a) One of the clone destination attributes that is undefined (see 3.1.50) shall be defined and set to the Partition_ID (see 4.6.4) of the destination partition; and
b) If it is defined (see 3.1.15), the clones count attribute shall have its valued incremented by one. If the clones count attribute undefined (see 3.1.50), then it shall be defined and set to a value of one.

The following attributes in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition shall be set as follows:

a) The partition type attribute shall be set to 02h (i.e., clone partition);
b) The source partition attribute shall be set to the Partition_ID (see 4.6.4) of the source partition; and
c) The branch depth attribute shall be set as follows:
   A) If the branch depth attribute is defined (see 3.1.15) in the Snapshots Information attributes page of the source partition, then the branch depth attribute value for the destination partition shall be set to one plus the value in the branch depth attribute for the source partition; or
   B) If the branch depth attribute is undefined (see 3.1.50) in the Snapshots Information attributes page of the source partition, then the branch depth attribute value for the destination partition shall be set to one.

If the FREEZE bit is set to one, the device server shall:

a) Note the value of the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition for use in 6.d.4; and
b) Set the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition to 0000 0001h (i.e., allow all read accesses, but deny all write accesses).

6.d.3 Processing after the IMMED_TR bit takes effect, if any

Every user object and collection in the source partition shall be duplicated in the destination clone partition using the:

a) Duplication method (see 4.d.3) specified by the CDB; and
b) Time of duplication method (see 4.d.4.1) specified by the CDB.

The membership and attributes of the snapshot/clone tracking well known collection for the destination partition should be maintained to restarting of an interrupted CREATE CLONE command with the minimum of repeated work (e.g., user objects or collections that have been fully duplicated should be removed from the snapshot/clone tracking well known collection). Other factors (e.g., meeting the requirements of the END time of duplication method (see 4.d.4.1)) may cause user objects and collections to be added to the snapshot/clone tracking well known collection.

6.d.4 Command completion

When and error is encountered or when all user objects and collections in the source partition have been duplicated in the destination clone partition as described in 6.d.3, the CREATE CLONE command processing shall be completed as described in this subclause.

If the FREEZE bit is set to one, the device server shall restore the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition to the value noted in 6.d.2.

At least the following changes shall be made in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition:

a) The command status attribute shall be set to indicate the condition (e.g., success or error) of CREATE CLONE command processing; and
b) If sense data is available, it shall be placed in the sense data attribute.

If the CREATE CLONE command processing complete (i.e., if the percent complete attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition is set to 100) and the command status attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition has been set to 0000 0000h (i.e., GOOD status command completion), then:

a) The create completion time attribute in the Snapshots Information attributes page (see 7.1.2.e) in the destination clone partition shall be set to the value of the clock attribute in the Root Information attributes page (see 7.1.2.8); and
b) The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination clone partition shall be set to 0000 0000h (i.e., allow all accesses).

If the IMMED_TR bit is set to zero, status shall be returned for the CREATE CLONE command.

6.e CREATE SNAPSHOT {{All of 6.e is new. Change markups suspended.}}

6.e.1 Overview

The CREATE SNAPSHOT command (see table x208) causes the OSD device server to allocate and initialize a destination partition as a snapshot partition (see 4.d.2) and then copy all user objects, collections, and attributes from a source partition to the newly created snapshot partition.

Table x208 — CREATE SNAPSHOT command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)			SERVICE ACTION (88A9h)				
9								(LSB)
10	Reserved			DPO	FUA	ISOLATION		
11	IMMED_TR	Reserved	GET/SET CDBFMT		Reserved			
12	TIMESTAMPS CONTROL							
13	FREEZE	Reserved			TIME OF DUPLICATION			
14	DUPLICATION METHOD							
15	Reserved							
16	(MSB)			SOURCE PARTITION_ID				
23								(LSB)
24	(MSB)			REQUESTED DESTINATION PARTITION_ID				
31								(LSB)
32				Reserved				
47								
48	(MSB)			CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				
51								(LSB)
52				Get and set attributes parameters (see 5.2.4)				
79								
80				Capability (see 5.2.c) {{in 08-185}}				
183								
184				Security parameters (see 5.2.8)				
235								

The contents of the DPO bit and the FUA bit are defined in 5.2.3.

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

If the FREEZE bit is set to zero, the CREATE SNAPSHOT command shall not modify the contents of the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition. If the FREEZE bit is set to one and source object freeze duplication management is supported (see 4.d.4.2), then the device server shall modify the contents of the object accessibility attribute in the Partition Information attributes page of the source partition as described in 6.e.2 and 6.e.4.

The TIME OF DUPLICATION field specifies which time of duplication source object management method (see 4.d.4.1) applies to the CREATE SNAPSHOT command. If the TIME OF DUPLICATION field is set to DEFAULT (see table x9 in 4.d.4.1 {{see 08-185}}), then the default snapshot time of duplication method attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition specifies which time of duplication management method applies to the CREATE SNAPSHOT command.

The DUPLICATION METHOD field specifies which duplication method (see 4.d.3) applies to the CREATE SNAPSHOT command. If the DUPLICATION METHOD field is set to DEFAULT (see table x8 in 4.d.3 {{see 08-185}}), then the default snapshot duplication method attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition specifies which duplication method applies to the CREATE SNAPSHOT command.

The SOURCE PARTITION_ID field contains the Partition_ID (see 4.6.4) of the source partition for the CREATE SNAPSHOT command.

The contents of the REQUESTED DESTINATION PARTITION_ID field specify the Partition_ID to be assigned to the created snapshot partition. If the REQUESTED DESTINATION PARTITION_ID field contains zero any Partition_ID may be assigned.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{in 08-158}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the CDB continuation segment (see 5.x) {{in 08-158}}:

a) Does not contain one extension capabilities CDB continuation descriptor (see 5.y.z) {{in 08-158}}; or
b) Contains any CDB continuation descriptors other than the extension capabilities CDB continuation descriptor.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14. The destination Partition_ID assigned by the CREATE SNAPSHOT command may be obtained from the Current Command attributes page (see 7.1.2.29).

The capability is defined in 4.11.2.2. The CREATE SNAPSHOT command accesses two partitions. One capability is necessary for each partition accessed. The capability with the highest value (see table 27 in 4.12.4.1) in the SECURITY METHOD field appears in the CDB. The other capability appears in the CDB continuation segment (see 5.x) {{see 08-185}}.

The security parameters are defined in 5.2.8.

The CREATE SNAPSHOT command does not initialize the partition key or the working keys (see 4.12.9.1) for the destination partition. Proper operation of any security method other than NOSEC (see 4.12.4) requires that the following commands be processed without errors before other commands are addressed to the destination partition:

a) A SET KEY command (see 6.29) that establishes the partition key; and
b) One or more SET KEY commands that establish one or more working keys for the partition.

A CREATE SNAPSHOT command whose capability (see 4.11.2.2) has the SET_ATTR bit set to one and POL/SEC bit set to one is allowed to avoid the need for SET KEY commands by setting the default security method attribute to NOSEC in the Partition Policy/Security attributes page (see 7.1.2.22) for the created partition.

If the REQUESTED DESTINATION PARTITION_ID field is not set to zero, SET KEY command are not needed to track the progress of a CREATE SNAPSHOT command with the IMMED_TR bit set to one in the following cases:

a) If the READ permission bit is set to one in the capability that allowed creation of the destination partition, that capability may be used in LIST COLLECTION commands (see 6.16) that list the contents of the snapshot/clone tracking well known collection (see 4.6.6.5.3); and
b) If the GET_ATTR permission bit is set to one in the capability that allowed creation of the destination partition, that capability may be used in GET ATTRIBUTES commands (see 6.13) or equivalents that retrieve attributes from the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection.

6.e.2 Processing before the IMMED_TR bit takes effect

A CREATE SNAPSHOT command shall not be completed with GOOD status until at least all the operations described in this subclause have been performed. These operations shall before completing the command with GOOD status even if the IMMED_TR bit is set to one.

If the FREEZE bit is set to one and source object freeze duplication management (see 4.d.4.2) is not supported, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the requested time of duplication source object management method (see 4.d.4.1) is not supported or the requested duplication method (see 4.d.3) is not supported, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the SOURCE PARTITION_ID field contains zero or the Partition_ID (see 4.6.4) of a partition that does not exist, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB, if any of the follow conditions exist in the attribute values in the Snapshots Information attributes page (see 7.1.2.e) of the source partition:

a) The partition type attribute contains 01h (i.e., snapshot partition); or
b) The snapshots count attribute contains a value that is equal to the value in the maximum snapshots count attribute in the Root Information attributes page (see 7.1.2.8).

If the REQUESTED DESTINATION PARTITION_ID field contains any value other than zero and the device server is unable to assign the requested Partition_ID to the created partition, the partition shall not be created and the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The device server shall not allow the same Partition_ID to be associated with more than one partition at any point in time.

If a CREATE SNAPSHOT command causes the value in the number of partitions attribute in the Root Information attributes page (see 7.1.2.8) to exceed the value in the partition count attribute in the Root Quotas attributes page (see 7.1.2.12), then a quota error shall be generated (see 4.10.2). The quota testing principles described in 4.10.3 apply to the testing of the partition count quota.

The device server shall create the requested destination partition and initialize it as if a CREATE PARTITION command (see 6.7) were being processed.

The assigned Partition_ID shall be placed in the Partition_ID attribute in the Current Command attributes page (see 7.1.2.29). The Collection_Object_ID or User_Object_ID attribute in the Current Command attributes page shall be set to zero.

The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination partition shall be set to 8000 0001h (i.e., deny all write accesses except those of CREATE SNAPSHOT commands, but allow all read accesses).

If it is defined (see 3.1.15), the snapshots count attribute in the Snapshots Information attributes page (see 7.1.2.e) of the source partition shall have its valued incremented by one. If the snapshots count attribute in the Snapshots Information attributes page of the source partition is undefined (see 3.1.50), then it shall be defined and set to a value of one.

The following attributes in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition shall be set as follows:

a) The partition type attribute shall be set to 01h (i.e., snapshot partition);
b) The source partition attribute shall be set to the Partition_ID (see 4.6.4) of the source partition; and
c) The branch depth attribute shall be set as follows:
   A) If the branch depth attribute is defined (see 3.1.15) in the Snapshots Information attributes page of the source partition, then the branch depth attribute value for the destination partition shall be set to the same value as the branch depth attribute for the source partition; or
   B) If the branch depth attribute is undefined (see 3.1.50) in the Snapshots Information attributes page of the source partition, then the branch depth attribute value for the destination partition shall be set to zero.

The destination snapshot partition shall be added as the newest entry in the history change as described in 6.r.5.

If the FREEZE bit is set to one, the device server shall:

a) Note the value of the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition for use in 6.e.4; and
b) Set the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition to 0000 0001h (i.e., allow all read accesses, but deny all write accesses).

The snapshot/clone tracking well known collection (see 4.6.6.5.3) shall be created in the destination partition, and initialized, including at least the following:

a) Every user object and collection in the source partition shall have their User_Object_ID (see 4.6.5) or Collection_Object_ID (see 4.6.6) inserted as a member of the TRACKING collection (see 4.6.6.3),
b) The Command Tracking attributes page (see 7.1.2.c) shall be initialized to include at least the following:
   A) The percent complete attribute shall be set to zero; and
   B) The command status attribute shall be set to 0001 88A9h (i.e., CREATE SNAPSHOT command in progress).

6.e.3 Processing after the IMMED_TR bit takes effect, if any

Every user object and collection in the source partition shall be duplicated in the destination snapshot partition using the:

a) Duplication method (see 4.d.3) specified by the CDB; and

b) Time of duplication method (see 4.d.4.1) specified by the CDB.

The membership and attributes of the snapshot/clone tracking well known collection for the destination partition should be maintained to restarting of an interrupted CREATE SNAPSHOT command with the minimum of repeated work (e.g., user objects or collections that have been fully duplicated should be removed from the snapshot/clone tracking well known collection). Other factors (e.g., meeting the requirements of the END time of duplication method (see 4.d.4.1)) may cause user objects and collections to be added to the snapshot/clone tracking well known collection.

6.e.4 Command completion

When and error is encountered or when all user objects and collections in the source partition have been duplicated in the destination snapshot partition as described in 6.e.3, the CREATE SNAPSHOT command processing shall be completed as described in this subclause.

If the FREEZE bit is set to one, the device server shall restore the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition to the value noted in 6.e.2.

At least the following changes shall be made in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition:

a) The command status attribute shall be set to indicate the condition (e.g., success or error) of CREATE SNAPSHOT command processing; and
b) If sense data is available, it shall be placed in the sense data attribute.

If the CREATE SNAPSHOT command processing complete (i.e., if the percent complete attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition is set to 100) and the command status attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition has been set to 0000 0000h (i.e., GOOD status command completion), then:

a) The create completion time attribute in the Snapshots Information attributes page (see 7.1.2.e) in the destination snapshot partition shall be set to the value of the clock attribute in the Root Information attributes page (see 7.1.2.8); and
b) The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination snapshot partition shall be set to 0000 0001h (i.e., allow all read accesses, but deny all write accesses).

If the IMMED_TR bit is set to zero, status shall be returned for the CREATE SNAPSHOT command.

6.f DETACH CLONE {{All of 6.f is new. Change markups suspended.}}

The DETACH CLONE command (see table x209) causes the OSD device server to change a clone partition into a primary partition (see 4.d.2).

Table x209 — DETACH CLONE command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)			SERVICE ACTION (88AAh)				
9								(LSB)
10	Reserved			DPO	FUA	ISOLATION		
11	Reserved		GET/SET CDBFMT		Reserved			
12	TIMESTAMPS CONTROL							
13				Reserved				
15								
16	(MSB)			CLONE PARTITION_ID				
23								(LSB)
24				Reserved				
47								
48	(MSB)			CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				
51								(LSB)
52				Get and set attributes parameters (see 5.2.4)				
79								
80				Capability (see 5.2.c) {{in 08-185}}				
183								
184				Security parameters (see 5.2.8)				
235								

The DETACH CLONE command accesses the following partitions:

a) The clone partition that is specified by the CLONE PARTITION_ID field; and
b) The source partition whose Partition_ID (see 4.6.4) is the value in the source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) of the clone partition.

The contents of the DPO bit and the FUA bit are defined in 5.2.3.

The contents of the ISOLATION field are defined in 5.2.5.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

The CLONE PARTITION_ID field contains the Partition_ID (see 4.6.4) of the clone partition that the DETACH CLONE command is being requested to detach.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB, if attributes in the Snapshots Information attributes page (see 7.1.2.e) of the clone partition have any of the following properties:

a) The partition type attribute contains a value other than 02h (i.e., clone partition);
b) The source partition attribute is undefined (see 3.1.50); or
c) The create completion time attribute is undefined (see 3.1.50) and the refresh completion time attribute is undefined.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{in 08-158}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the CDB continuation segment (see 5.x) {{in 08-158}}:

a) Does not contain one extension capabilities CDB continuation descriptor (see 5.y.z) {{in 08-158}}; or
b) Contains any CDB continuation descriptors other than the extension capabilities CDB continuation descriptor.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14.

The capability is defined in 4.11.2.2. The DETACH CLONE command accesses two partitions. One capability is necessary for each partition (i.e., clone and source) accessed. The capability with the highest value (see table 27 in 4.12.4.1) in the SECURITY METHOD field appears in the CDB. The other capability appears in the CDB continuation segment (see 5.x) {{see 08-185}}.

The security parameters are defined in 5.2.8.

In the Snapshots Information attributes page (see 7.1.2.e) of the source partition, the following changes shall be made in attribute values:

a) One shall be subtracted from the clones count attribute value; and
b) The clone destination attribute whose value matches the Partition_ID (see 4.6.4) of the clone partition shall be made undefined (see 3.1.50).

In the Snapshots Information attributes page (see 7.1.2.e) of the clone partition, the following changes shall be made in attribute values:

a) The partition type attribute shall be set to 00h (i.e., primary partition);
b) The source partition attribute shall be made undefined (see 3.1.50);
c) The branch depth attribute shall be set to zero;
d) The create completion time attribute shall be made undefined;
e) The refresh completion time attribute shall be made undefined.

The branch depth attributes in the Snapshots Information attributes page (see 7.1.2.e) of all partitions chained to the former clone partition shall be updated as follows:

a) The branch depth attribute in all snapshot partitions that have the former clone partition as a source shall be set to zero;
b) The branch depth attribute in each clone partition that has a source snapshot partition whose branch depth attribute is zero shall be set to one;

c) The branch depth attribute in each snapshot partition that has a source clone partition whose branch depth is one shall be set to one;
d) The branch depth attribute in each clone partition that has a source snapshot partition whose branch depth attribute is one shall be set to two;
e) The branch depth attribute in each snapshot partition that has a source clone partition whose branch depth is two shall be set to two;
f) The branch depth attribute in each clone partition that has a source snapshot partition whose branch depth attribute is n shall be set to n plus one; and
g) The branch depth attribute in each snapshot partition that has a source clone partition whose branch depth is n plus one shall be set to n plus one.

6.r REFRESH SNAPSHOT OR CLONE {{All of 6.r is new. Change markups suspended.}}

6.r.1 Overview

The REFRESH SNAPSHOT OR CLONE command (see table x210) causes the OSD device server to:

a) Restart the processing started by a CREATE SNAPSHOT command (see 6.e) or CREATE CLONE command (see 6.d) that was interrupted before completion; or
b) Update the contents of a snapshot partition (see 4.d.2) to match the current contents of its source partition.

Table x210 — REFERESH SNAPSHOT OR CLONE command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)			SERVICE ACTION (88ABh)				
9								(LSB)
10	Reserved			DPO	FUA	ISOLATION		
11	IMMED_TR	Reserved		GET/SET CDBFMT	Reserved			
12	TIMESTAMPS CONTROL							
13	FREEZE	Reserved			TIME OF DUPLICATION			
14	DUPLICATION METHOD							
15	Reserved							
16	(MSB)			PARTITION_ID				
23								(LSB)
24				Reserved				
47								
48	(MSB)			CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				
51								(LSB)
52				Get and set attributes parameters (see 5.2.4)				
79								
80				Capability (see 5.2.c) {{in 08-185}}				
183								
184				Security parameters (see 5.2.8)				
235								

The REFRESH SNAPSHOT OR CLONE command accesses the following partitions:

a) The destination partition that is specified by the PARTITION_ID field; and
b) The source partition whose Partition_ID (see 4.6.4) is the value in the source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition.

The contents of the DPO bit and the FUA bit are defined in 5.2.3.

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

If the FREEZE bit is set to zero, the REFRESH SNAPSHOT OR CLONE command shall not modify the contents of the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) of the source partition. If the FREEZE bit is set to one and source object freeze duplication management is supported (see 4.d.4.2), then the device server shall modify the contents of the object accessibility attribute in the Partition Information attributes page of the source partition as described in 6.r.2 and 6.r.4.

The TIME OF DUPLICATION field specifies which time of duplication source object management method (see 4.d.4.1) applies to the REFRESH SNAPSHOT OR CLONE command. If the TIME OF DUPLICATION field is set to DEFAULT (see table x9 in 4.d.4.1 {{see 08-185}}), then which time of duplication source object management method is used is specified as follows:

a) If the partition type attribute in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition is set to 01h (i.e., snapshot partition), then the default snapshot time of duplication method attribute in the Partition Information attributes page of the source partition specifies which time of duplication management method applies to the REFRESH SNAPSHOT OR CLONE command; or
b) If the partition type attribute in the Snapshots Information attributes page of the destination partition is set to 02h (i.e., clone partition), then the default clone time of duplication method attribute in the Partition Information attributes page of the source partition specifies which time of duplication management method applies to the REFRESH SNAPSHOT OR CLONE command.

The DUPLICATION METHOD field specifies which duplication method (see 4.d.3) applies to the REFRESH SNAPSHOT OR CLONE command. If the DUPLICATION METHOD field is set to DEFAULT (see table x8 in 4.d.3 {{see 08-185}}), then which duplication method is used is specified as follows:

a) If the partition type attribute in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition is set to 01h (i.e., snapshot partition), then the default snapshot duplication method attribute in the Partition Information attributes page of the source partition specifies which duplication method applies to the REFRESH SNAPSHOT OR CLONE command; or
b) If the partition type attribute in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition is set to 02h (i.e., clone partition), then the default snapshot duplication method attribute in the Partition Information attributes page of the source partition specifies which duplication method applies to the REFRESH SNAPSHOT OR CLONE command.

The contents of PARTITION_ID field (i.e., the Partition_ID of the destination partition) are defined in 5.2.7.

If the source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition is undefined (see 3.1.50), the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{in 08-158}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the CDB continuation segment (see 5.x) {{in 08-158}}:

a) Does not contain one extension capabilities CDB continuation descriptor (see 5.y.z) {{in 08-158}}; or
b) Contains any CDB continuation descriptors other than the extension capabilities CDB continuation descriptor.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14.

The capability is defined in 4.11.2.2. The REFRESH SNAPSHOT OR CLONE command accesses two partitions. One capability is necessary for each partition (i.e., source and destination) accessed. The capability with the highest value (see table 27 in 4.12.4.1) in the SECURITY METHOD field appears in the CDB. The other capability appears in the CDB continuation segment (see 5.x) {{see 08-185}}.

The security parameters are defined in 5.2.8.

6.r.2 Processing before the IMMED_TR bit takes effect

A REFRESH SNAPSHOT OR CLONE command shall not be completed with GOOD status until at least all the operations described in this subclause have been performed. These operations shall before completing the command with GOOD status even if the IMMED_TR bit is set to one.

If the snapshot forward attribute value in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition is not equal to the source partition attribute value in the Snapshots Information attributes page of the destination partition and the support for refreshing attribute in the Root Information attributes page (see 7.1.2.8) contains MOST RECENT ONLY (i.e., 01h), then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the FREEZE bit is set to one and source object freeze duplication management (see 4.d.4.2) is not supported, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the requested time of duplication source object management method (see 4.d.4.1) is not supported or the requested duplication method (see 4.d.3) is not supported, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the partition type attribute in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition contains 00h (i.e., primary partition), then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the of the destination partition contains 0000 0000h (i.e., allow all accesses), then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The refresh completion time in the Snapshots Information attributes page (see 7.1.2.e) of the destination partition shall be made undefined (see 3.1.50).

If the Snapshots Information attributes page (see 7.1.2.e) of the destination partition contains the following attribute values:

a) The partition type attribute contains 01h (i.e., snapshot partition); and
b) The snapshot forward attribute value is not equal to the source partition attribute value:

then:

1) The destination partition shall be unlinked from the history chain as described in 6.r.6; and
2) The destination partition shall be added as the newest entry in the history change as described in 6.r.5.

If the FREEZE bit is set to one, the device server shall:

a) Note the value of the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition for use in 6.r.4; and
b) Set the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition to 0000 0001h (i.e., allow all read accesses, but deny all write accesses).

The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination partition shall be set to 8000 0001h (i.e., deny all write accesses except those of REFRESH SNAPSHOT OR CLONE commands, but allow all read accesses).

The snapshot/clone tracking well known collection (see 4.6.6.5.3) shall be update in the destination partition to include at least the following:

a) Every user object and collection in the source partition shall have their User_Object_ID (see 4.6.5) or Collection_Object_ID (see 4.6.6) inserted as a member of the TRACKING collection (see 4.6.6.3),
b) The Command Tracking attributes page (see 7.1.2.c) shall be initialized to include at least the following:
   A) The percent complete attribute shall be set to zero; and
   B) The command status attribute shall be set to 0001 88ABh (i.e., REFRESH SNAPSHOT OR CLONE command in progress).

6.r.3 Processing after the IMMED_TR bit takes effect, if any

Every user object and collection in the source partition shall be duplicated in the destination snapshot partition using the:

a) Duplication method (see 4.d.3) specified by the CDB; and
b) Time of duplication method (see 4.d.4.1) specified by the CDB.

The membership and attributes of the snapshot/clone tracking well known collection for the destination partition should be maintained to restarting of an interrupted REFRESH SNAPSHOT OR CLONE command with the minimum of repeated work (e.g., user objects or collections that have been fully duplicated should be removed from the snapshot/clone tracking well known collection). Other factors (e.g., meeting the requirements of the END time of duplication method (see 4.d.4.1)) may cause user objects and collections to be added to the snapshot/clone tracking well known collection.

6.r.4 Command completion

When and error is encountered or when all user objects and collections in the source partition have been duplicated in the destination snapshot partition as described in 6.r.3, the REFRESH SNAPSHOT OR CLONE command processing shall be completed as described in this subclause.

If the FREEZE bit is set to one, the device server shall restore the object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the source partition to the value noted in 6.r.2.

At least the following changes shall be made in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition:

a) The command status attribute shall be set to indicate the condition (e.g., success or error) of REFRESH SNAPSHOT OR CLONE command processing; and
b) If sense data is available, it shall be placed in the sense data attribute.

If the REFRESH SNAPSHOT OR CLONE command processing complete (i.e., if the percent complete attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition is set to 100) and the command status attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the destination partition has been set to 0000 0000h (i.e., GOOD status command completion), then:

a) The refresh completion time attribute in the Snapshots Information attributes page (see 7.1.2.e) in the destination partition shall be set to the value of the clock attribute in the Root Information attributes page (see 7.1.2.8); and
b) The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination snapshot partition shall be set to 0000 0001h (i.e., allow all read accesses, but deny all write accesses).

If the IMMED_TR bit is set to zero, status shall be returned for the REFRESH SNAPSHOT OR CLONE command.

6.r.5 Linking a snapshot as the most recent entry in the history chain

To add a snapshot partition in the history chain as the most recent snapshot, the snapshot backward and snapshot forward attributes in the Snapshots Information attributes page (see 7.1.2.e) of the following partitions shall be set as shown in table x211:

a) Source partition (i.e., the partition whose Partition_ID (see 4.6.4) is the value in the source partition attribute of the snapshot partition to be added as the most recent snapshot);
b) Destination partition (i.e., the snapshot partition to be added as the most recent snapshot); and

c) Previous newest partition, if any (i.e., the partition whose Partition_ID in the snapshot backward attribute of the source partition before any changes are made).

Table x211 — Snapshot backward and forward attribute values set to add the most recent entry

Partition	Snapshots Information attributes page attribute	
	Snapshot backward	**Snapshot forward**
Source partition	The Partition_ID of the destination partition	not modified
Destination partition	The Partition_ID of the previous newest partition [a]	The Partition_ID of the source partition
Previous newest partition [b]	not modified	The Partition_ID of the destination partition

[a] If the snapshot backward attribute of the source partition is undefined, this attribute shall also be undefined.

[b] If the snapshot backward attribute of the source partition is undefined, the changes shown in this row are not made.

6.r.6 Unlinking a snapshot from the history chain

To unlink a snapshot partition from the history chain, the snapshot backward and snapshot forward attributes in the Snapshots Information attributes page (see 7.1.2.e) of the following partitions shall be set as shown in table x211:

a) Destination partition (i.e., the snapshot partition to be unlinked);
b) Newer partition (i.e., the partition whose Partition_ID (see 4.6.4) is the value in the snapshot forward attribute of the snapshot partition to be unlinked before any changes are made); and
c) Older partition, if any (i.e., the partition whose Partition_ID is the value in the snapshot backward attribute of the snapshot partition to be unlinked before any changes are made).

Table x212 — Snapshot backward and forward attribute values set to unlink an entry

Partition	Snapshots Information attributes page attribute	
	Snapshot backward	**Snapshot forward**
Newer partition	The value of the snapshot backward attribute of the destination partition [a]	not modified
Older partition	not modified	The value of the snapshot forward attribute of the destination partition [a]

[a] If the specified attribute is undefined (see 3.1.50) the attribute to be set shall be made undefined.

6.s RESTORE PARTITION FROM SNAPSHOT {{All of 6.s is new. Change markups suspended.}}

6.s.1 Overview

The RESTORE PARTITION FROM SNAPSHOT command (see table x213) causes the OSD device server to update the contents of a main partition (i.e., primary partition or clone partition) to match the contents of a snapshot partition(see 4.d.2).

Table x213 — RESTORE PARTITION FROM SNAPSHOT command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)			SERVICE ACTION (88ACh)				
9								(LSB)
10	Reserved			DPO	FUA	ISOLATION		
11	IMMED_TR	Reserved	GET/SET CDBFMT		Reserved			
12	TIMESTAMPS CONTROL							
13	Reserved							
14	DUPLICATION METHOD							
15	Reserved							
16	(MSB)			SNAPSHOT PARTITION_ID				
23								(LSB)
24				Reserved				
47								
48	(MSB)			CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				
51								(LSB)
52				Get and set attributes parameters (see 5.2.4)				
79								
80				Capability (see 5.2.c) {{in 08-185}}				
183								
184				Security parameters (see 5.2.8)				
235								

The RESTORE PARTITION FROM SNAPSHOT command accesses the following partitions:

a) The snapshot partition that is specified by the SNAPSHOT PARTITION_ID field; and
b) The main partition whose Partition_ID (see 4.6.4) is the value in the source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) of the snapshot partition.

The contents of the DPO bit and the FUA bit are defined in 5.2.3.

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

The DUPLICATION METHOD field specifies which duplication method (see 4.d.3) applies to the RESTORE PARTITION FROM SNAPSHOT command. If the DUPLICATION METHOD field is set to DEFAULT (see table x8 in 4.d.3 {{see 08-185}}), then the default snapshot duplication method attribute in the Partition Information attributes page (see 7.1.2.9) of the main partition specifies which duplication method applies to the RESTORE PARTITION FROM SNAPSHOT command.

The SNAPSHOT PARTITION_ID field contains the Partition_ID (see 4.6.4) of the snapshot partition for the RESTORE PARTITION FROM SNAPSHOT command.

If the source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) of the snapshot partition is undefined (see 3.1.50), the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{in 08-158}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the CDB continuation segment (see 5.x) {{in 08-158}}:

a) Does not contain one extension capabilities CDB continuation descriptor (see 5.y.z) {{in 08-158}}; or
b) Contains any CDB continuation descriptors other than the extension capabilities CDB continuation descriptor.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14.

The capability is defined in 4.11.2.2. The RESTORE PARTITION FROM SNAPSHOT command accesses two partitions. One capability is necessary for each partition (i.e., snapshot and main) accessed. The capability with the highest value (see table 27 in 4.12.4.1) in the SECURITY METHOD field appears in the CDB. The other capability appears in the CDB continuation segment (see 5.x) {{see 08-185}}.

The security parameters are defined in 5.2.8.

6.s.2 Processing before the IMMED_TR bit takes effect

A RESTORE PARTITION FROM SNAPSHOT command shall not be completed with GOOD status until at least all the operations described in this subclause have been performed. These operations shall before completing the command with GOOD status even if the IMMED_TR bit is set to one.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB, if attributes in the Snapshots Information attributes page (see 7.1.2.e) of the snapshot partition have any of the following properties:

a) The partition type attribute contains a value other than 01h (i.e., snapshot partition);
b) The create completion time attribute is undefined (see 3.1.50) and the refresh completion time attribute is undefined.

If the requested duplication method (see 4.d.3) is not supported, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the main partition shall be set to 8000 0001h (i.e., deny all write accesses except those of RESTORE PARTITION FROM SNAPSHOT commands, but allow all read accesses).

The snapshot/clone tracking well known collection (see 4.6.6.5.3) shall be update in the snapshot partition to include at least the following:

a) Every user object and collection in the snapshot partition shall have their User_Object_ID (see 4.6.5) or Collection_Object_ID (see 4.6.6) inserted as a member of the TRACKING collection (see 4.6.6.3),
b) The Command Tracking attributes page (see 7.1.2.c) shall be initialized to include at least the following:
   A) The percent complete attribute shall be set to zero; and
   B) The command status attribute shall be set to 0001 88ACh (i.e., RESTORE PARTITION FROM SNAPSHOT command in progress).

6.s.3 Processing after the IMMED_TR bit takes effect, if any

Every user object and collection in the snapshot partition shall be duplicated in the main snapshot partition using the duplication method (see 4.d.3) specified by the CDB.

The membership and attributes of the snapshot/clone tracking well known collection for the snapshot partition should be maintained to restarting of an interrupted RESTORE PARTITION FROM SNAPSHOT command with the minimum of repeated work (e.g., user objects or collections that have been fully duplicated should be removed from the snapshot/clone tracking well known collection).

6.s.4 Command completion

When and error is encountered or when all user objects and collections in the source partition have been duplicated in the destination snapshot partition as described in 6.s.3, the RESTORE PARTITION FROM SNAPSHOT command processing shall be completed as described in this subclause.

At least the following changes shall be made in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the snapshot partition:

a) The command status attribute shall be set to indicate the condition (e.g., success or error) of RESTORE PARTITION FROM SNAPSHOT command processing; and
b) If sense data is available, it shall be placed in the sense data attribute.

If the RESTORE PARTITION FROM SNAPSHOT command processing complete (i.e., if the percent complete attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the snapshot partition is set to 100) and the command status attribute in the Command Tracking attributes page (see 7.1.2.c) of the snapshot/clone tracking well known collection (see 4.6.6.5.3) in the snapshot partition has been set to 0000 0000h (i.e., GOOD status command completion), then:

a) The restore completion time attribute in the Snapshots Information attributes page (see 7.1.2.e) in the main partition shall be set to the value of the clock attribute in the Root Information attributes page (see 7.1.2.8);
b) The restore Partition_ID attribute in the Snapshots Information attributes page in the main partition shall be set to the Partition_ID of the snapshot partition;
c) The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the main snapshot partition shall be set to 0000 0000h (i.e., allow all accesses).

If the IMMED_TR bit is set to zero, status shall be returned for the RESTORE PARTITION FROM SNAPSHOT command.

…

6.27 REMOVE PARTITION

6.27.1 Overview

The REMOVE PARTITION command (see table 111) deletes a partition from the OSD logical unit. ~~If there are any collections or user objects in the partition, the command shall be terminated with a CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to PARTITION OR COLLECTION CONTAINS USER OBJECTS.~~ If the command status attribute in the Command Tracking attributes page (see 7.1.2.c) associated with the partition is defined (see 3.1.15) and set to a value between 0001 8800h and 0001 8FFFh (e.g., if a CREATE SNAPSHOT command (see 6.e) is processing objects in the partition), then the command shall be terminated with CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to COMMAND SEQUENCE ERROR.

Table 111 — REMOVE PARTITION command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)							
9	SERVICE ACTION (888Ch)							(LSB)
10	Reserved			DPO	FUA	ISOLATION		
~~11~~	IMMED_TR	Reserved	~~GET/SET CDBFMT~~		~~Reserved~~			
11	Reserved		GET/SET CDBFMT		Reserved	REMOVE SCOPE		
12	TIMESTAMPS CONTROL							
13	{{No other changes in the body of table 111.}}							
…								

…

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The REMOVE SCOPE field (see table x214) specifies the scope the partition removal operations requested with respect to:

a) user objects and collections within the partition; and
b) Clone partitions (see 4.d.2).

Table x214 — REMOVE SCOPE field

Snapshots count / Clones count [a]		zero / zero	zero / non-zero	non-zero / zero
REMOVE SCOPE field	**Reference**			
000b	6.27.2	Remove the partition	Terminate the command [b]	
001b	6.27.3	Remove user objects and collections from the partition, and then remove the partition		
010b	6.27.4	Remove the partition	Detach clones, and then perform remove partition functions	Terminate the command [b]
011b	6.27.5	Remove user objects and collections from the partition, and then remove the partition		
100b to 111b	Reserved			

[a] The snapshots count attribute and clones count attribute are found in the Snapshots Information attributes page (see 7.1.2.e) of the specified partition. As described in 4.d.2, snapshot partitions are not allowed to head other snapshot history chains. Also clone partitions and primary partitions are not allowed to have clone partitions. Therefore, the case where the snapshots count attribute is not zero and the clones count attribute is not zero is not allowed by the snapshots model.

[b] The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

{{The SNIA OSD TWG Snapshots proposal v3.14 says: "If the FORCE_REMOVAL bit is set, the REMOVE PARTITION command deletes the named partition, all of its objects and collections, **and all of its underlying clones and snapshots**." I have been unable to devise a mechanism for removing all the 'underlying clones and snapshots' in a recoverable and restartable way. The detaching clones function shown in table x214 is the closest I have been able to come.}

…

6.27.2 Removing an empty partition with no descending snapshots or clones

6.27.2.1 Empty partition CDB validation

If there are any collections or user objects in the partition, the command shall be terminated with CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to PARTITION OR COLLECTION CONTAINS USER OBJECTS.

If the IMMED_TR bit is set to one, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB. If the IMMED_TR bit is set to zero, the command shall be processed as described in this subclause.

6.27.2.2 Empty partition unlinking

If the partition type attribute in the Snapshots Information attributes page (see 7.1.2.e) is set to 01h (i.e., snapshot partition), then

a) The partition shall be unlinked from the history chain as the destination partition described in 6.r.6; and
b) The source partition attribute in the Snapshots Information attributes page shall be used to locate the snapshot's source partition and one shall be subtracted from the snapshots count attribute value in that page.

If the partition type attribute in the Snapshots Information attributes page (see 7.1.2.e) is set to 02h (i.e., clone partition), then:

1) The source partition attribute in the Snapshots Information attributes page shall be used to locate the clone's source partition, and
2) The following changes shall be made in the attributes in the Snapshots Information attributes page of the source partition:
   A) One shall be subtracted from the clones count attribute value in that page; and
   B) The clone destination attribute in that page whose value matches the Partition_ID (see 4.6.4) of the partition being removed shall be made undefined (see 3.1.50).

6.27.2.3 Empty and unlinked partition removal

A partition is deleted from an OSD logical unit by changing its representation so that a LIST command (see 6.15) with the PARTITION_ID field set to zero does not contain the Partition_ID (see 4.6.4) of the deleted partition.

6.27.3 Removing user objects and collections before removing a partition

6.27.3.1 Processing before the IMMED_TR bit takes effect

A REMOVE PARTITION command with the REMOVE SCOPE field set to 001b or 011b shall not be completed with GOOD status until at least all the operations described in this subclause have been performed. These operations shall before completing the command with GOOD status even if the IMMED_TR bit is set to one.

If the snapshot/clone tracking well known collection (see 4.6.6.5.3) exists and the command status attribute is not set to zero in the Command Tracking attributes page (see 7.1.2.c), then the REMOVE PARTITION command shall be terminated with CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to COMMAND SEQUENCE ERROR.

The object accessibility attribute in the Partition Information attributes page (see 7.1.2.9) in the destination partition shall be set to 8000 0001h (i.e., deny all write accesses except those of REMOVE PARTITION commands, but allow all read accesses).

The snapshot/clone tracking well known collection (see 4.6.6.5.3) shall be created in the destination partition, and initialized, including at least the following:

a) Every user object and collection in the source partition shall have their User_Object_ID (see 4.6.5) or Collection_Object_ID (see 4.6.6) inserted as a member of the TRACKING collection (see 4.6.6.3),
b) The Command Tracking attributes page (see 7.1.2.c) shall be initialized to include at least the following:
   A) The percent complete attribute shall be set to zero; and
   B) The command status attribute shall be set to 0001 888Ch (i.e., REMOVE PARTITION command in progress).

6.27.3.2 Processing after the IMMED_TR bit takes effect, if any

Removal of the user objects and collections in the partition shall proceed as follows:

1) The equivalent of a REMOVE command (see 6.24) shall be processed for each user object in the snapshot/clone tracking well known collection; and
2) The equivalent of a REMOVE COLLECTION command (see 6.25) shall be processed for each collection in the snapshot/clone tracking well known collection.

The membership and attributes of the snapshot/clone tracking well known collection for the partition should be maintained to restarting of an interrupted REMOVE PARTITION command with the minimum of repeated work (e.g., user objects or collections that have been removed should be removed from the snapshot/clone tracking well known collection).

6.27.3.3 Command completion

The partition shall be unlinked as described in 6.27.2.2

The empty and unlinked partition shall be removed as described in 6.27.2.3.

{{Requirement to reset the object accessibility attribute and cleanup the snapshot/clone tracking collection could be added here, but this seems like too much of a good thing.}}

6.27.4 Detaching clones from an empty partition

If the REMOVE_SCOPE field is set to 010b, the following operations shall be performed:

1) The REMOVE PARTITION CDB shall be validated for use in processing an empty partition as described in 6.27.2.1;
2) The equivalent of a DETACH CLONE command (see 6.f) shall be performed for every Partition_ID (see 4.6.4) identified by a clone destination attribute in the Snapshots Information attributes page (see 7.1.2.e);
3) The partition shall be unlinked as described in 6.27.2.2; and
4) The empty partition shall be removed as described in 6.27.2.3.

6.27.5 Detaching clones from a populated partition

A REMOVE PARTITION command with the REMOVE SCOPE field set to 011b shall not be completed with GOOD status until the equivalent of a DETACH CLONE command (see 6.f) has been performed for every Partition_ID (see 4.6.4) identified by a clone destination attribute in the Snapshots Information attributes page (see 7.1.2.e). The clone partitions shall be detached before completing the command even if the IMMED_TR bit is set to one.

After the clones have been detached, the REMOVE PARTITION command shall be processed as described in 6.27.3.

…

7.1.2.e Snapshots Information attributes page

{{All of 7.1.2.e is new. Change markups suspended. This subclause should be inserted immediately before the Current Command attributes page (see 7.1.2.29.}}

The Snapshots Information attributes page (P+7h) shall contain the attributes listed in table x215.

Table x215 — Snapshots Information attributes page contents

Attribute Number	Length (bytes)	Attribute	Application Client Settable	OSD Logical Unit Provided
0h	0 or 40	Page identification	No	Yes
1h	0 or 1	Partition type	No	Yes
2h to 7Fh		Reserved	No	
80h	0 or 8	Source partition	No	Yes
81h	0 or 8	Snapshot backward	No	Yes
82h	0 or 8	Snapshot forward	No	Yes
83h to FFFFh	0 or 8	Clone destination	No	Yes
1 0000h to 2 0000h		Reserved	No	
2 0001h	0 or 4	Snapshots count	No	Yes
2 0002h	0 or 4	Clones count	No	Yes
2 0003h to 2 000Bh		Reserved	No	
2 000Ch	0 or 4	Branch depth	No	Yes
2 000Dh to 2 0010h		Reserved	No	
2 0011h	0 or 6	Create completion time	No	Yes
2 0012h	0 or 6	Refresh completion time	No	Yes
2 0013h	0 or 6	Restore completion time	No	Yes
2 0014h	0 or 8	Restore Partition_ID	No	Yes
2 0015 to FFFF FFFEh		Reserved	No	

If it is defined (see 3.1.15), the page identification attribute (number 0h) shall have the format described in 7.1.2.2 with the VENDOR IDENTIFICATION field containing the ASCII characters "INCITS" and the ATTRIBUTES PAGE IDENTIFICATION field containing the ASCII characters "T10 Snapshots Information".

If it is defined (see 3.1.15), the partition type attribute (number 1h) (see table x216) indicates the characteristics of the partition with respect to the snapshots model (see 4.d.2). If the partition type attribute is undefined (see 3.1.50), the partition is a primary partition.

Table x216 — Partition type attribute values

Partition type attribute value	Description
00h	Primary partition (i.e., not a snapshot partition or a clone partition)
01h	Snapshot partition
02h	Clone partition
03h to FFh	Reserved

If it is defined (see 3.1.15), the source partition attribute (number 80h) contains the contents of the SOURCE PARTITION_ID field in the CREATE SNAPSHOT command (see 6.e) or CREATE CLONE command (see 6.d) that created the partition. If the source partition attribute is undefined (see 3.1.50), then one of the following is true:

a) The partition was not created by a CREATE SNAPSHOT command or a CREATE CLONE command, or
b) The partition was created by a CREATE CLONE command and later detached by a DETACH CLONE command (see 6.f).

If it is defined (see 3.1.15), the snapshot backward attribute (number 81h) contains the Partition_ID (see 4.6.2) of the next older snapshot partition in the history chain (see 4.d.2.2). If the snapshot backward attribute is undefined (see 3.1.50), then the partition has never been a source partition or a destination partition in a CREATE SNAPSHOT command (see 6.e).

If it is defined (see 3.1.15), the snapshot forward attribute (number 82h) contains the Partition_ID (see 4.6.2) of the next newer snapshot partition in the history chain (see 4.d.2.2). If the snapshot forward attribute is undefined (see 3.1.50), then the partition has never been a source partition or a destination partition in a CREATE SNAPSHOT command (see 6.e).

Each defined (see 3.1.15) clone destination attribute (numbers 83h to FFFFh) contains the Partition_ID (see 4.6.2) of a clone partition (see 4.d.2.3). If all clone destination attributes are undefined (see 3.1.50), then one of the following is true:

a) The partition has never been a source partition for a CREATE CLONE command (see 6.f), or
b) All clone partitions for which this partition was the source have been:
   A) Detached by DETACH CLONE commands (see 6.f); or
   B) Removed by REMOVE PARTITION commands (see 6.27).

There is no significance to which clone destination attribute numbers are defined and which are undefined.

If the snapshot backward attribute is defined (see 3.1.15) in a primary partition or clone partition, then the snapshots count attribute (number 2 001h) is defined and contains the number of snapshots in the history chain (see 4.d.2.2) that the primary partition or clone partition heads. If the snapshot backward attribute is undefined (see 3.1.50) or the partition is a snapshot partition, the snapshots count attribute is undefined.

If any clone destination attribute is defined (see 3.1.15), then the clones count attribute (number 2 002h) is defined and contains the number of clone destination attributes that are defined in the partition. If all clone destination attributes are undefined (see 3.1.50), the clones count attribute is undefined.

If it is defined (see 3.1.15), the create completion time attribute (number 2 0011h) contains value of the clock attribute in the Root Information attributes page (see 7.1.2.8) at the completion of the CREATE SNAPSHOT

command (see 6.e) or CREATE CLONE command (see 6.d) that created this partition. The create completion time is undefined (see 3.1.50) if any of the following are true:

a) The partition was not created by a CREATE SNAPSHOT command or a CREATE CLONE command; or
b) The CREATE SNAPSHOT command or CREATE CLONE command has not yet completed.

If it is defined (see 3.1.15), the refresh completion time attribute (number 2 0012h) contains value of the clock attribute in the Root Information attributes page (see 7.1.2.8) at the completion of the most recent REFRESH SNAPSHOT command (see 6.r). The refresh completion time is undefined (see 3.1.50) if any of the following are true:

a) The partition has never been the destination of a REFRESH SNAPSHOT command; or
b) The most recent REFRESH SNAPSHOT command has not yet completed.

If it is defined (see 3.1.15), the restore completion time attribute (number 2 0013h) contains value of the clock attribute in the Root Information attributes page (see 7.1.2.8) at the completion of the most recent RESTORE PARTITION FROM SNAPSHOT command (see 6.s). The restore completion time is undefined (see 3.1.50) if any of the following are true:

a) The partition has never been the destination of a RESTORE PARTITION FROM SNAPSHOT command; or
b) The most recent RESTORE PARTITION FROM SNAPSHOT command has not yet completed.

If it is defined (see 3.1.15), the restore Partition_ID attribute (number 2 0014h) contains value in the PARTITION_ID field of the most recent RESTORE PARTITION FROM SNAPSHOT command (see 6.s) that has completed. The restore completion time is undefined (see 3.1.50) if any of the following are true:

a) The partition has never been the destination of a RESTORE PARTITION FROM SNAPSHOT command; or
b) The most recent RESTORE PARTITION FROM SNAPSHOT command has not yet completed.

If it is defined (see 3.1.15), the branch depth attribute (number 0002 000Ch) indicates the nesting depth of a snapshot partition or clone partition. The branch depth of a primary partition is zero, and the branch depth attribute is undefined (see 3.1.50) for primary partitions. Other branch depth values increase from the primary partition value as follows:

a) All snapshot partitions that have the primary partition as their source partition have a branch depth of zero;
b) All clone partitions that have a snapshot partition with a branch depth of zero as their source partition, have a branch depth of one;
c) All snapshot partitions that have a clone partition with a branch depth of one as their source partition, have a branch depth of one;
d) All clone partitions that have a snapshot with a branch depth of one as their source partition, have a branch depth of two;
e) All snapshot partitions that have a clone partition with a branch depth of n as their source partition, have a branch depth of n; and
f) All clone partitions that have a snapshot partition with a branch depth of n as their source partition, have a branch depth of n plus one.

If a command attempts to set an attribute that table x215 states is not application client settable, then the command shall be terminated as described in 7.1.n {{see 08-181}}.

…

7.1.2.8 Root Information attributes page

The Root Information attributes page (R+1h) shall contain the attributes listed in table 127.

Table 127 — Root Information attributes page contents

Attribute Number	Length (bytes)	Attribute	Application Client Settable	OSD Logical Unit Provided
…	…	…	…	…
123h	1	Data/attributes atomicity multiplier	No	Yes
124h to 1FFh		Reserved	No	
1C1h	0 or 4	Maximum snapshots count	No	Yes
1C2h	0 or 4	Maximum clones count	No	Yes
1C3h to 1CBh		Reserved	No	
1CCh	0 or 4	Maximum branch depth	No	Yes
1CDh to 1FFh		Reserved	No	
124h to 1FFh		Reserved	No	
200h to 2FFh	0 or 4	Supported object duplication method	No	Yes
300h to 30Fh	0 or 4	Supported time of duplication method	No	Yes
310h	0 or 4	Support for duplicated object freezing	No	Yes
311h	0 or 1	Support for snapshot refreshing	No	Yes
~~124h~~ 312h to FFFF FFFEh		Reserved	No	
…	…	…	…	…

…

If it is defined (see 3.1.15), the maximum snapshots count attribute (number 1C1h) shall contain the non-zero number that is the largest value allowed in any snapshots count attribute in any Snapshots Information attributes page (see 7.1.2.e). If the maximum snapshots count attribute is defined, the following commands shall be supported:

a) The CREATE SNAPSHOT command (see 6.e);
b) The REFRESH SNAPSHOT command (see 6.r); and
c) The RESTORE PARTITION FROM SNAPSHOT command (see 6.s).

If it is defined (see 3.1.15), the maximum clones count attribute (number 1C2h) shall contain the non-zero number that is the largest value allowed in any clones count attribute in any Snapshots Information attributes page (see 7.1.2.e). If the maximum clones count attribute is defined, the following commands shall be supported:

a) The CREATE CLONE command (see 6.d); and
b) The DETACH CLONE command (see 6.f).

If the CREATE SNAPSHOT command (see 6.e) is supported and the CREATE CLONE command (see 6.d) is supported, then the maximum branch depth attribute (number 1CCh) shall be defined (see 3.1.15) and shall

contain largest value allowed in any branch depth attribute in any Snapshots Information attributes page (see 7.1.2.e).

…

{{The supported object duplication method attributes are defined in 08-185.}}

{{The supported time of duplication method attributes are defined in 08-185.}}

{{The supported duplicated object freezing attributes are defined in 08-185.}}

…

If it is defined (see 3.1.15), the support for snapshot refreshing attribute (number 311h) (see table x217) shall indicate how the REFRESH SNAPSHOT command (see 6.r) is supported. If the support for snapshot refreshing attribute is undefined (see 3.1.50), then the REFRESH SNAPSHOT command is not supported.

Table x217 — Support for snapshot refreshing attribute values

Value	Name	Description
00h	Reserved	
01h	MOST RECENT ONLY	The REFRESH SNAPSHOT command is allowed only if the value in the source partition attribute in the Snapshots Information attributes page (see 7.1.2.e) for the source partition is equal to the value in the snapshot forward attribute in the Snapshots Information attributes page.
02h to FEh	Reserved	
FFh	UNLIMITED	The REFRESH SNAPSHOT command has no limits on the source partition.

…

Change 2 – New collections and collection types

Description

If the functionality normally associated with an IMMED bit is to be supported by an OSD, a way needs to be defined to track the progress of such functions, repaired failed functions, and restart interrupted functions. Preliminary attempts to do this were made with the type 1 collections used by multi-object commands. Snapshots require enhancements and clarifications to this scheme.

Also, the SNIA multi-object commands described a 'mega collection' (to be called the all user objects in partition well known collection in this proposal) whose membership is any user object whose User_Object_ID would be returned by a LIST command.

The processing of multi-object commands (see 4.6.6.6) needs changes to account for user objects that have been removed since the time the TRACKING collection was created.

Proposed changes in OSD-2 r03

3.1.a user tracking collection: A TRACKING collection (see 4.6.6.3) with a Collection_Object_ID that is greater than or equal to 10000h (i.e., a TRACKING collection that is not a well-known collection (see 4.6.6.5)).

…

4.6.2 Identifying OSD objects

The combination of Partition_ID and User_Object_ID uniquely identifies the root object, each partition, each collection, and each user object. Partition_ID and User_Object_ID values are assigned as shown in table 3.

Table 3 — Partition_ID and User_Object_ID value assignments

Partition_ID	User_Object_ID or Collection_Object_ID	Description
0h	0h	Root object
0h	1h - FFFF FFFF FFFF FFFFh	Reserved
1h to FFFFh	0h - FFFF FFFF FFFF FFFFh	Reserved
10000h to FFFF FFFF FFFF FFFFh	0h	Partition [a]
~~10000h to FFFF FFFF FFFF FFFFh~~	~~1h to FFFFh~~	~~Reserved~~
10000h to FFFF FFFF FFFF FFFFh	1h to 0FFFh	Reserved
10000h to FFFF FFFF FFFF FFFFh	1000h to BFFFh	Well known collections [b]
10000h to FFFF FFFF FFFF FFFFh	C000h to FFFFh	Reserved
10000h to FFFF FFFF FFFF FFFFh	10000h to FFFF FFFF FFFF FFFFh	Collection or User object [c]

[a] Partition_ID values assigned by the OSD logical unit in response to application client requests.
[b] Well known collections have constant Collection_Object_ID values and may be members of any partition (see 4.6.6.5).
[c] Collection_Object_ID values and User_Object_ID values assigned by the OSD logical unit in response to application client requests.

4.6.3 Root object

…

4.6.6 Collections

4.6.6.1 Overview

Support for collections is optional. If collections are not supported:

a) The length of attribute number 4h in the User Object Directory attributes page (see 7.1.2.7) shall be zero for every user object (i.e., no Collections attributes pages identified); and
b) Zero shall be returned as the length of attribute number 0h in every Collections attributes page (see 7.1.2.19).

A partition may contain zero or more collections each of which may contain zero or more user objects. One user object may be a member of zero or more collections. ~~If the collection type attribute in the Collection Information attributes page (see 7.1.2.10) contains 00h, user objects are added to or removed from the membership of a collection by setting attribute values in the user object's Collections attributes page (see 7.1.2.19).~~

Collections have the Partition_ID of the partition to which they belong and a Collection_Object_ID (see 4.6.2) that is assigned by the OSD logical unit when the collection is created. A collection is a member of only one partition.

Within a single partition, no collection shall be assigned the same Collection_Object_ID as any User_Object_ID and no user object shall be assigned the same User_Object_ID value as any Collection_Object_ID (i.e., collections and user objects share the same number space for their identifier values).

~~A collection is created using the CREATE COLLECTION command (see 6.6) and deleted using the REMOVE COLLECTION command (see 6.25). The page format of the Collections attributes page (see 7.1.2.19) lists all the collections in which a user object is a member.~~ The LIST COLLECTION command (see 6.16) lists all the collections in a partition or all the user objects that are members of a collection.

{{N.B. A LIST COLLECTION command that lists user object members of the all user objects in partition well known LIST collection (see 4.6.6.5.2) is equivalent to a LIST command.}}

A collection does not contain a read/write data area. The device server shall terminate all READ commands, WRITE commands, and APPEND commands sent to the collection with ~~a~~ CHECK CONDITION status, setting the sense key to ILLEGAL REQUEST and the additional sense code to INVALID FIELD IN CDB.

The collection type attribute in the Collection Information attributes page indicates the type of the collection as shown in table 133 (see 7.1.2.10). Different collection types have different operational characteristics. The following collection types are defined:

a) LINKED (see 4.6.6.2);
b) TRACKING (see 4.6.6.3); and
c) LIST (see 4.6.6.4).

4.6.6.2 LINKED collections

The device server maintains a linkage between the user object entries in a LINKED collection and the actual user objects (e.g., if a user object that is a member of a LINKED collection is removed from the partition, all entries for the user object are removed from all LINKED collections of which the user object is a member).

A LINKED collection is created using the CREATE COLLECTION command (see 6.6) and deleted using the REMOVE COLLECTION command (see 6.25). The page format of the Collections attributes page (see 7.1.2.19) lists all the collections in which a user object is a member.

~~If the collection type attribute in the Collection Information attributes page (see 7.1.2.10) contains 00h, user~~ User objects are added to or removed from the membership of a LINKED collection by setting attribute values in the user object's Collections attributes page (see 7.1.2.19).

4.6.6.3 TRACKING collections

The members of a TRACKING collection have no linkage to the actual user objects (e.g., removal of a user object from the partition or replacement of a user object with another user object having the same User_Object_ID have no effect on the user object's membership in the TRACKING collection).

TRACKING collections are used to track the progress of commands that operate on multiple objects (e.g., multi-object commands (see 4.6.6.6) and the CREATE SNAPSHOT command (see 6.e)).

The Collection_Object_ID of a TRACKING collection affects the collection's operational characteristics as shown in table x218.

Table x218 — TRACKING collection operational characteristics

Collection_Object_ID	Allowed members	Created by	Removed by	Dynamic addition of members allowed
0h	Reserved (see table 3 in 4.6.2)			
1h to FFFFh	user objects and collections	The device server in response to specific commands (e.g., the CREATE SNAPSHOT command (see 6.e))		Yes
10000h to FFFF FFFF FFFF FFFFh	user objects	A CREATE USER TRACKING COLLECTION command (see 6.t)	A REMOVE COLLECTION command (see 6.25)	No

TRACKING collections shall include the Command Tracking attributes page (see 7.1.2.c).

4.6.6.4 LIST collections

All LIST collections are well known collections (see 4.6.6.5). The LIST collection's Collection_Object_ID specifies how to determined the collection's membership. A LIST collection's membership is recomputed each time it is retrieved.

4.6.6.5 Well known collections

4.6.6.5.1 Overview

Any partition except partition zero (see 3.1.33) may contain one or more of the well known collections shown in table x219.

Table x219 — Well known collections

Collection_Object_ID [a]	Description	Type		Well Known Collection Reference
		Name	Reference	
1000h to 1081h	Reserved			
1082h	All user objects in partition	LIST	4.6.6.4	4.6.6.5.2
1083h to 8000h	Reserved			
8001h	Snapshot/clone tracking	TRACKING	4.6.6.3	4.6.6.5.3
8002h to BFFFh	Reserved			

[a] Collection_Object_IDs not shown in this table are shown in table 3 (see 4.6.2).

Well known collections are not included in the parameter data returned by a LIST COLLECTION command (see 6.16).

4.6.6.5.2 The all user objects in partition well known collection

The membership of the all user objects in partition well known collection is all user objects in the partition which contains it. The effect of each access to the all user objects in partition well known collection is the equivalent of:

1) Processing a LIST command (see 6.15) with:
   A) The PARTITION_ID field set to the Partition_ID of the partition that contains the all user objects in partition well known collection; and
   B) The LIST_ATTR bit set to zero;
   and
2) defining the collection's membership to match the command's output.

4.6.6.5.3 The snapshot/clone tracking well known collection

The snapshot/clone tracking well known collection is a member of any partition that is:

a) The destination for an incomplete:
   A) CREATE SNAPSHOT command (see 6.e); or
   B) CREATE CLONE command (see 6.d);
   or
b) The partition being processed by a REMOVE PARTITION command (see 6.27) with the REMOVE SCOPE field set to 111b.

The membership and attributes of a snapshot/clone tracking well known collection provide sufficient information to:

a) Track the progress of the creation of the snapshot or clone partition; and
b) Restart an interrupted CREATE SNAPSHOT command, CREATE CLONE command, or REMOVE PARTITION command.

4.6.6.6 ~~4.6.6.2~~ Commands that use collections to affect multiple user objects

Commands such as SET MEMBER ATTRIBUTES (see 6.31) (i.e., multi-object commands) process multiple user objects using the membership of a collection that is not a well known collection (see 4.6.6.5) as a ~~dynamic~~ list of the user objects on which the specified operations are to be performed.

With the exception of the REMOVE MEMBER OBJECTS command (see 6.26), multi-object commands process only TRACKING collections (see 4.6.6.3) ~~whose collection type attribute contains 01h~~. If the COLLECTION_OBJECT_ID field in a multi-object command CDB other than a REMOVE MEMBER OBJECTS command specifies a collection type other than TRACKING (see table 133 in 7.1.2.10) ~~for which the collection type attribute in the Collection Information attributes page (see 7.1.2.10) contains a value other than 01h~~, the command shall be terminated with ~~a~~ CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

If the COLLECTION_OBJECT_ID field in a multi-object command CDB specifies a object that is not a collection or is a well known collection (see 4.6.6.5), the command shall be terminated with ~~a~~ CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

{{The following action is number 3 in the list below.}}

~~After the specified operations have been successfully completed on a user object, that user object shall be removed from the specified collection.~~

Before any user object is processed, the attributes in the Command Tracking attributes page (see 7.1.2.c) shall be updated to reflect active use of the collection for processing the command.

Each user object in the specified collection shall be processed as follows:

1) If the user object has been removed, the specified operation shall not be performed on the non-existent user object. This shall not be considered to be an error;
2) If user object has not been removed and the creation time attribute in the User Object Timestamps attributes page (see 7.1.2.18) is earlier than or equal to (i.e., less than or equal to) the creation time attribute in the Collection Timestamps attributes page (see 7.1.2.17) (i.e., if the user object has not been replaced), then the quotas (see 4.10) that apply to the specified operation shall be evaluated and processing of the operation shall be handled as follows:
   A) If a quota error condition is detected, the multi-object command shall be terminated as described in this subclause; or
   B) If no quota error condition is detected, the specified operation shall be performed on the user object and whether or not an error is detected shall be noted;
3) If the creation time attribute in the User Object Timestamps attributes page is later than (i.e., greater than) the creation time attribute in the Collection Timestamps attributes page, then the specified operation shall not be performed on the user object. This shall not be considered to be an error; ~~and~~
4) If no error has been detected, the user object shall be removed from the specified collection~~.~~; and
5) The attributes in the Command Tracking attributes page shall be updated to reflect completion of processing for the user object.

As a result of these requirements, the following conditions apply:

a) After an error condition that prevented processing of all user objects in the collection is corrected, the same command specifying the same collection may be sent to continue processing; and
b) Application clients may poll to determine the progress of a multi-object command using the LIST COLLECTION command (see 6.16) and/or the contents of the Command Tracking attributes page (see 7.1.2.c).~~; and~~
~~c) Application clients may poll to determine the progress of a multi-object command by retrieving the number of members attribute value in the Collection Information attributes page.~~

NOTE 2 - The LIST command and LIST COLLECTION command are not multi-object commands.

Two multi-object commands shall not concurrently process the same collection. If a multi-object command is received with the COLLECTION_OBJECT_ID field in the CDB specifying the Collection_Object_ID (see 4.6.2) of a collection that is already being processed by a different multi-object command, the command shall be terminated with ~~a~~ CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The device sever may process more than one user object concurrently.

If an error is detected during the processing of a user object:

a) The user object shall not be removed from the collection;
b) Processing that has already been started on any other user object shall be completed to the greatest degree possible and any user objects for which processing is successfully completed shall be removed from the collection;
c) If necessary, the policy access tag attribute in the User Object Policy/Security attributes page (see 7.1.2.24) for any user object for which an error is detected shall be updated as described in 4.11.3.2;
d) Processing shall not be started for any user object that has not already started processing; and
e) When no user objects are being processed~~,~~:
   A) If the IMMED_TR bit (see 5.2.i) is set to zero, the command shall be terminated with the status and sense data corresponding to the first error that was detected~~.~~; and

B) The attributes in the Command Tracking attributes page (see 7.1.2.c) shall be updated to reflect the termination of processing, including the status and additional sense data with which the command, if any would have been or was terminated.

If a multi-object command is terminated as part of processing a command-related condition (see SAM-4), a task management function, or as the result of a SCSI device condition (e.g., logical unit reset) established in response to an event (see SAM-4), then the device server shall:

a) Update the attributes in the Command Tracking attributes page (see 7.1.2.c) reflect the interruption of the command ~~Set the multi-object operation in progress attribute value to zero in the Collection Information attributes page (see 7.1.2.10)~~; and
b) Either:
   A) Establish a consistent, stable state for each user object being processed; or
   B) Set the policy access tag attribute in the User Object Policy/Security attributes page described in 4.11.3.2 for any user object for which it is not possible to establish consistent state.

The device server shall not remove the specified collection upon completion of the multi-object command, even if the collection contains zero user objects.

If the CDB GET/SET CDBFMT field contains 11b (i.e., when list format attributes processing is specified), multi-object commands allow setting and retrieving of both collection attributes and user object attributes. …

… {{No changes are proposed for the processing of attributes by multi-object commands.}}

{{The changes above cover all of the following. Also significant changes are proposed in the attributes to be modified.}}

~~The multi-object operation in progress attribute in the Collection Information attributes page (see 7.1.2.10) shall be set as follows:~~

~~a) To one before an operation is performed as described in this subclause on the first user object in a collection; and~~
~~b) To zero before the processing of a multi-object command is completed or terminated as described in this subclause.~~

…

6.14 GET MEMBER ATTRIBUTES

The GET MEMBER ATTRIBUTES command (see table 74) instructs the device server to return the specified attributes for the specified user tracking collection (see 3.1.a) and the user object members of the user tracking collection before setting the attributes, if any, specified by the command (see 4.8.4). The GET MEMBER ATTRIBUTES command is a multi-object command ~~(see 4.6.6.2)~~ (see 4.6.6.6).

…

6.26 REMOVE MEMBER OBJECTS

The REMOVE MEMBER OBJECTS command (see table 110) instructs the device server to remove all the user objects that are members of the specified user tracking collection (see 3.1.a). The REMOVE MEMBER OBJECTS command is a multi-object command ~~(see 4.6.6.2)~~ (see 4.6.6.6).

…

6.31 SET MEMBER ATTRIBUTES

The SET MEMBER ATTRIBUTES command (see table 119) instructs the device server to set the specified attributes for the specified user tracking collection (see 3.1.a) and user object members of the user tracking collection before retrieving the attributes, if any, specified by the command (see 4.8.4). The SET MEMBER ATTRIBUTES command is a multi-object command ~~(see 4.6.6.2)~~ (see 4.6.6.6).

…

7.1.2.10 Collection Information attributes page

The Collection Information attributes page (C+1h) shall contain the attributes listed in table 132.

Table 132 — Collection Information attributes page contents

Attribute Number	Length (bytes)	Attribute	Application Client Settable	OSD Logical Unit Provided
…	…	…	…	…
Ah	1	Collection type	No	Yes
~~Bh~~	~~4~~	~~Number of members~~	~~No~~	~~Yes~~
~~Ch~~	~~1~~	~~Multi-object operation in progress~~	~~No~~	~~Yes~~
~~Dh~~ Bh to 80h		Reserved	No	
…	…	…	…	…

{{The function served by the multi-object operation in progress attribute is replaced and enhanced by Command Tracking attributes page (see 7.1.2.c).}}

…

The collection type attribute (number Ah) shall identify the characteristics (see table 133) of the collection.

Table 133 — Collection type codes

Code	Name	Description
00h	LINKED	User objects may be added to or removed from the collection using the Collections attributes page (see 7.1.2.19).
01h	TRACKING	~~User objects shall not be added or removed from the collection except as part of processing multi-object commands (see 4.6.6.6).~~ Changes in the Collections attributes page (see 7.1.2.19) shall not affect the membership of TRACKING type collections. Changes in the membership of TRACKING type collections shall not affect the attributes in the Collections attributes page. The membership of TRACKING type collections is maintained by the device server based on processing requested by the application client (e.g., the processing of multi-object commands (see 4.6.6.6)).
02h to EEh		Reserved
EFh	LIST	The membership of a LIST type collection shall be recomputed every time the collection is accessed. All LIST collections are well known collections (see 4.6.6.5) and the collection's Collection_Object_ID specifies how to compute the collection's membership.
F0h to FFh		Reserved
~~02h to FFh~~		~~Reserved~~

~~The number of members attribute (number Bh) shall indicate the number of user objects that are members of the collection.~~

~~The multi-object operation in progress attribute (Ch) shall contain the following:~~

~~a) Zero if no multi-object operations (see 4.6.6.6) are in progress; or~~
~~b) One if a multi-object operation is in progress.~~

~~If a multi-object command is terminated as part of processing a command-related condition (see SAM-4), a task management function, or as the result of a SCSI device condition (e.g., logical unit reset) established in response to an event (see SAM-4), then the device server shall set the multi-object operation in progress attribute value to zero.~~

…

7.1.2.c Command Tracking attributes page

{{This subclause should be inserted immediately before the Collections attributes page (see 7.1.2.19).}}

The Command Tracking attributes page (C+4h) shall contain the attributes listed in table x220.

Table x220 — Command Tracking attributes page contents

Attribute Number	Length (bytes)	Attribute	Application Client Settable	OSD Logical Unit Provided
0h	40	Page identification	No	Yes
1h	1	Percent complete	No	Yes
2h	4	Command status	No	Yes
3h	0 or n	Sense data	No	Yes
4h to Fh		Reserved	No	
~~Bh~~ 10h	~~4~~ 8	Number of members	No	Yes
11h	0 or 8	Objects processed	No	Yes
12h	0 or 8	Newer objects skipped	No	Yes
13h	0 or 8	Missing objects skipped	No	Yes
14h to EFFF FFFFh		Reserved	No	
F000 0000h to FFFF FFFEh	0 or n	Vendor specific [a]	No	Yes

[a] The combination of a TRACKING collection's (see 4.6.6.3) membership and the Command Tracking attributes page attributes shall be sufficient to restart an interrupted command (e.g., an interrupted CREATE SNAPSHOT command (see 6.e)) or a command that was terminated with CHECK CONDITION status. Information in the vendor specific attributes may be needed to fulfill this requirement.

The page identification attribute (number 0h) shall have the format described in 7.1.2.2 with the VENDOR IDENTIFICATION field containing the ASCII characters "INCITS" and the ATTRIBUTES PAGE IDENTIFICATION field containing the ASCII characters "T10 Command Tracking".

The percent complete attribute (number 1h) shall indicate percentage of the processing that has been completed for the command, if any, for which the device sever is using the collection to track processing activities.

The command status attribute (number 2h) shall indicate the processing status (see table x221) of the command, if any, for which the device server is using the collection to track processing activities.

Table x221 — Command status attribute values

Command status attribute value	Description
0000 0000h	Command processing has completed with GOOD status.
0000 0001h to 0000 00FFh	Command processing has completed with the status code (see SAM-4) contained in the least significant byte of the attribute value.
0000 0100h to 0000 7FFFh	Reserved
0000 8000h	The command that was using the collection to track processing activities was interrupted for an unknown reason.
0000 8001h	The command that was using the collection to track processing activities was interrupted an ABORT TASK task management function (see SAM-4) or another condition whose processing emulates an ABORT TASK task management function.
0000 8002h	The command that was using the collection to track processing activities was interrupted by a power on event (See SAM-4).
0000 8003h	The command that was using the collection to track processing activities was interrupted by a bus reset event (See SAM-4).
0000 8004h	The command that was using the collection to track processing activities was interrupted by a logical unit reset event (See SAM-4).
0000 8005h	The command that was using the collection to track processing activities was interrupted by a I_T nexus loss event (See SAM-4).
0000 8006h	The command that was using the collection to track processing activities was interrupted by a power loss expected event (See SAM-4).
0000 8007h to 0001 87FFh	Reserved
0001 8800h to 0001 8FFFFh	A command whose service action field contains the attribute value minus 0001 0000h is using the collection to track processing activities.
0001 9000h to FFFF FFFEh	Reserved
FFFF FFFFh	No command has used the collection to track processing activities.

If the command status attribute is set to 0000 0002h (i.e., command processing has completed with CHECK CONDITION status), then the sense data attribute (number 3h) shall contain the sense data that was, or should have been, returned to the application client. If the command status attribute is not set to 0000 0002h, the sense data attribute should be undefined (see 3.1.50).

The number of members attribute (number 10h ~~Bh~~) shall indicate the number of ~~user~~ objects that are members of the collection.

If it is defined (see 3.1.15), the objects processed attribute shall indicate the number of objects that have been removed from the collection following successful processing as specified by the command (e.g., as described for multi-object commands in 4.6.6.6).

If it is defined (see 3.1.15), the newer objects skipped attribute shall indicate the number of objects that have been removed from the collection because the creation time attribute in the User Object Timestamps attributes page

(see 7.1.2.18) is later than (i.e., greater than) the creation time attribute in the Collection Timestamps attributes page (see 7.1.2.17) (e.g., as described for multi-object commands in 4.6.6.6).

If it is defined (see 3.1.15), the missing objects skipped attribute shall indicate the number of objects that have been removed from the collection because the object was not present in the partition at the time processing was attempted (e.g., as described for multi-object commands in 4.6.6.6).

If a command attempts to set an attribute that table x220 states is not application client settable, then the command shall be terminated as described in 7.1.n {{see 08-181}}.

7.1.2.19 Collections attributes page

The Collections attributes page (4h) shall contain the attributes listed in table 150.

Table 150 — Collections attributes page contents

Attribute Number	Length (bytes)	Attribute	Application Client Settable	OSD Logical Unit Provided
0h	0 or 40	Page identification	No	Yes
1h to FFFF FF00h	0 or 8	Collection pointer	Yes~~/No~~ ~~a~~	No
FFFF FF01h to FFFF FFFEh		Reserved	No	

~~a~~ ~~If the collection type attribute in the Collection Information attributes page (see 7.1.2.10) contains 00h, the Collection pointer attribute shall be application client settable. If the collection type attribute in the Collection Information attributes page contains 01h, the Collection pointer attribute shall not be application client settable.~~

{{The above table footnote is flat-out wrong. The collection attributes page never contains pointers to collections that are not collection type 00h collections (renamed LINKED collections by this proposal).}}

…

~~If the collection type attribute in the Collection Information attributes page contains 00h,~~ For a LINKED collection (see 4.6.6.2), a user object is made a member of a collection by setting one of its collection pointer attribute values to the Collection_Object_ID of that collection.

~~If the collection type attribute in the Collection Information attributes page contains 00h,~~ For a LINKED collection (see 4.6.6.2), a user object is removed from the membership of a collection by:

a) Changing the collection pointer attribute identifying that collection to have a length of zero; or
b) Setting the collection pointer attribute identifying that collection to the Collection_Object_ID of a different collection.

The command shall be terminated as described in 7.1.n {{see 08-181}} if it attempts to set: ~~with a CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST if a set attributes list (see 5.2.4.4) contains an entry that sets:~~

a) The same Collection_Object_ID in more than one collection pointer attribute;
b) A collection pointer attribute to a value that is not a Collection_Object_ID; ~~or~~

c) A collection pointer attribute to the Collection_Object_ID of a collection in which the collection type attribute in the Collection Information attributes page (see 7.1.2.10) is set to a value other than LINKED (see table 133 in 7.1.2.10); or
d) A collection pointer attribute to any length other than zero or eight.

Change 3 – Define an IMMED bit that is usable with TRACKING collections

Description

The time required to complete many multi-object operations and especially the time required to complete a CREATE SNAPSHOT command is so great that no operating system will allow a command to be outstanding for the entire interval. Therefore, an OSD equivalent of the IMMED bit is needed.

However, REQUEST SENSE reporting of progress on background commands is not sufficient to handle all the OSD issues with IMMED commands (e.g., it is very desirable that OSD IMMED commands be restartable via application client actions).

The principle is that any command that processes an OSD command that uses a TRACKING collection (see 4.6.6.3) can be an OSD IMMED command, with all progress tracking and restart features being handled by the tracking properties of the collection, especially by the attributes in the Command Tracking attributes page (see 7.1.2.c).

Thus, the bit defined in the IMMED_TR bit, which is both more (e.g., restartable) and less (no REQUEST SENSE effects) than the traditional IMMED bit.

Note: the effects of the on multi-object processing features already defined in OSD-2 are included in change 2, since that change already substantially modifies OSD-2 r03 subclause 4.6.2.2 (now 4.6.6.6 in this proposal).

This change also includes a correction for a reported error in the handling of the FCR bit in the REMOVE COLLECTION command (see 6.25).

Proposed changes in OSD-2 r03

5.2 Fields commonly used in OSD commands

5.2.1 Overview

OSD commands employ the basic CDB structure shown in 5.1. Within the basic CDB structure, the OSD service action specific fields are organized so that the same field is in the same location in all OSD CDBs (see table 49). OSD service action specific fields that are unique to a small number of CDBs are not shown in this subclause.

Table 49 — OSD service action specific fields

Bit Byte	7	6	5	4	3	2	1	0
10	Reserved			DPO [a]	FUA [a]	ISOLATION (see 5.2.5)		
~~11~~	~~Reserved~~		~~GET/SET CDBFMT~~ [~~b~~]		~~Command specific options~~			
11	IMMED_TR [b]	Reserved	GET/SET CDBFMT [c]		Command specific options			
12	TIMESTAMPS CONTROL (see 5.2.10)							
13	{{No other changes in the body of table 49.}}							
…								

[a] See 5.2.3.
[b] See 5.2.i.
[c] See 5.2.4.

…

{{Insert the following new subclause in the proper alphabetical order.}}

5.2.i Immediate bit for TRACKING collections

The immediate bit for TRACKING collections (IMMED_TR) allows an application client to specify that the command be completed with GOOD status after the TRACKING collection (see 4.6.6.3) has been set up but before all objects in the TRACKING collection have been processed. If the IMMED_TR bit is set to zero, the device server shall process all command functions (e.g., all objects in the TRACKING collection evaluated) before completing the command. If the IMMED_TR bit is set to one, the device server shall:

1) Verify the correctness of all CDB and CDB continuation fields;
2) Perform any security checks required to validate the command (see 4.10);
3) Initialize the Command Tracking attributes page (see 7.1.2.c) in the TRACKING collection with all information necessary to process, track, and restart the command;
4) Process all command functions related to attributes as described in 4.7.4, except those command functions that are to be performed individually on objects in the TRACKING collection; and
5) If no errors have been detected, complete the command with GOOD status.

Commands completed with GOOD status due to the IMMED_TR bit being set shall not result in the REQUEST SENSE command reporting progress indication information (see SPC-4).

…

6.6 CREATE COLLECTION

The CREATE COLLECTION command (see table 62) initializes a new LINKED collection ~~(see 4.6.6)~~ (see 4.6.6.2).

…

The security parameters are defined in 5.2.8.

The collection type attribute in the Collection Information attributes page (see 7.1.2.10) shall be set to 00h (i.e., LINKED).

The assigned Collection_Object_ID shall be placed in the Collection_Object_ID or User_Object_ID attribute in the Current Command attributes page (see 7.1.2.29).

…

6.9 FLUSH COLLECTION

…

The COLLECTION_OBJECT_ID field specifies Collection_Object_ID (see 4.6.6). If the collection identified by the COLLECTION_OBJECT_ID field does not exist, the command shall be terminated with ~~a~~ CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The type of collection (see 4.6.6.1) being flushed shall not affect how the FLUSH COLLECTION command is processed (e.g., the attributes of the Command Tracking attributes page (see 7.1.2.c), if any, shall not be modified).

…

6.14 GET MEMBER ATTRIBUTES

The GET MEMBER ATTRIBUTES command (see table 74) instructs the device server to return the specified attributes for the specified collection and the user object members of the collection before setting the attributes, if any, specified by the command (see 4.7.4). The GET MEMBER ATTRIBUTES command is a multi-object command (see 4.6.6.6).

Table 74 — GET MEMBER ATTRIBUTES command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)							
9	SERVICE ACTION (88A2h)							(LSB)
10	Reserved			DPO	FUA	ISOLATION		
~~11~~	~~Reserved~~		~~GET/SET CDBFMT~~		~~Command specific options~~			
11	IMMED_TR	Reserved	GET/SET CDBFMT		Command specific options			
12	TIMESTAMPS CONTROL (see 5.2.10)							
13	{{No other changes in the body of table 74.}}							
…								

…

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

…

6.16 LIST COLLECTION

…

The COLLECTION_OBJECT_ID field specifies Collection_Object_ID (see 4.6.6) to be processed. The contents of the COLLECTION_OBJECT_ID field combined with the LIST_ATTR bit value specify the information that shall be returned (see table 85). If the collection identified by a non-zero COLLECTION_OBJECT_ID field does not exist, the command shall be terminated with ~~a~~ CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The type of collection (see 4.6.6.1) being listed shall not affect how the LIST COLLECTION command is processed (e.g., the attributes of the Command Tracking attributes page (see 7.1.2.c), if any, shall not be modified).

…

6.21 QUERY {{All changes to the QUERY command are shown in change 5.}}

…

6.25 REMOVE COLLECTION

…

The FCR (force collection removal) bit specifies the actions to be taken if the collection contains user objects. If the FCR bit is set to zero and the collection contains user objects, the command shall be terminated with ~~a~~ CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST, and the additional sense code shall be set to PARTITION OR COLLECTION CONTAINS USER OBJECTS. If the FCR bit is set to one, the collection shall be removed as follows even if it contains user objects:

1) The collection type (see 4.6.6.1) shall affect the processing of user objects in the as follows:
   A) For LINKED collections (see 4.6.6.2), each ~~Each~~ user object in the collection shall be modified to indicate that the user object no longer is a member of the collection; or
   B) For all other collection types, the user objects in the collection shall not be modified;
   and
2) The collection shall be removed.

…

The contents of the COLLECTION_OBJECT_ID field specify the Collection_Object_ID (see 4.6.6) the collection to be removed. If the COLLECTION_OBJECT_ID field specifies a well known collection (see 4.6.6.5), the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

Except for the processing of the FCR bit, the type of collection (see 4.6.6.1) being removed shall not affect how the REMOVE COLLECTION command is processed (e.g., the attributes of the Command Tracking attributes page (see 7.1.2.c), if any, shall not be modified).

…

6.26 REMOVE MEMBER OBJECTS

The REMOVE MEMBER OBJECTS command (see table 110) instructs the device server to remove all the user objects that are members of the specified collection. The REMOVE MEMBER OBJECTS command is a multi-object command (see 4.6.6.6).

Table 110 — REMOVE MEMBER OBJECTS command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)							
9	SERVICE ACTION (88A1h)							(LSB)
10	Reserved			DPO	FUA	ISOLATION		
~~11~~	~~Reserved~~		~~GET/SET CDBFMT~~		~~Command specific options~~			
11	IMMED_TR	Reserved	GET/SET CDBFMT		Command specific options			
12	TIMESTAMPS CONTROL (see 5.2.10)							
13	{{No other changes in the body of table 110.}}							
…								

…

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

…

6.31 SET MEMBER ATTRIBUTES

The SET MEMBER ATTRIBUTES command (see table 119) instructs the device server to set the specified attributes for the specified collection and user object members of the collection before retrieving the attributes, if any, specified by the command (see 4.7.4). The SET MEMBER ATTRIBUTES command is a multi-object command (see 4.6.6.6).

Table 119 — SET MEMBER ATTRIBUTES command

Bit Byte	7	6	5	4	3	2	1	0
8	(MSB)							
9	SERVICE ACTION (88A3h)							(LSB)
10	Reserved			DPO	FUA	ISOLATION		
~~11~~	~~Reserved~~		~~GET/SET CDBFMT~~		~~Command specific options~~			
11	IMMED_TR	Reserved	GET/SET CDBFMT		Command specific options			
12	TIMESTAMPS CONTROL (see 5.2.10)							
13	{{No other changes in the body of table 119.}}							
…								

…

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i.

…

Change 4 – CREATE USER TRACKING COLLECTION command

Description

The usage of a TRACKING collection is defined by change 2 and change 3. However, the only TRACKING collections for which the contents are described are well known collections (see 4.6.6.5). The change defines a CREATE USER TRACKING COLLECTION, which rounds out the ways in which a TRACKING collection can be made ready for processing by a multi-object command (see 4.6.6.6).

The permissions changes that support the commands defined in change 4 appear in change 10 so that all of the permissions changes for all of the commands defined in this proposal can be reviewed as a group.

Proposed changes in OSD-2 r03

6.t CREATE USER TRACKING COLLECTION {{All of 6.t is new. Change markups suspended.}}

The CREATE USER TRACKING COLLECTION command (see table x222) creates a user tracking collection (see 3.1.a) and copies the membership of another collection of any type (see 4.6.6.1) to the newly created collection. If the specified output user tracking collection already exists and the contents of its Command Tracking attributes

page (see 7.1.2.c) indicate that it use by another command has been finished, the membership of the output user tracking collection is replaced.

Table x222 — CREATE USER TRACKING COLLECTION command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)							
9				SERVICE ACTION (8894h)				(LSB)
10	Reserved			DPO	FUA	ISOLATION		
11	Reserved		GET/SET CDBFMT		Reserved			
12	TIMESTAMPS CONTROL							
13				Reserved				
15								
16	(MSB)							
23				PARTITION_ID				(LSB)
24	(MSB)							
31				REQUESTED COLLECTION_OBJECT_ID				(LSB)
32								
39				Reserved				
40	(MSB)							
47				SOURCE COLLECTION_OBJECT_ID				(LSB)
48	(MSB)							
51				CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				(LSB)
52								
79				Get and set attributes parameters (see 5.2.4)				
80								
183				Capability (see 5.2.c) {{in 08-185}}				
184								
235				Security parameters (see 5.2.8)				

The contents of the DPO bit and the FUA bit are defined in 5.2.3.

The contents of the ISOLATION field are defined in 5.2.5.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

The contents of the PARTITION_ID field (see 5.2.7) specify the Partition_ID of the partition in which the user tracking collection is to be created.

The contents of the REQUESTED COLLECTION_OBJECT_ID field specify the Collection_Object_ID (see 4.6.6) to be assigned to the created user tracking collection. If the REQUESTED COLLECTION_OBJECT_ID field contains zero any

Collection_Object_ID may be assigned. If the REQUESTED COLLECTION_OBJECT_ID field contains any value other than zero and the device server is unable to assign the requested Collection_Object_ID to the created user tracking collection, the user tracking collection shall not be created and the command shall be terminated with a CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

Within a partition, the device server shall not allow:

a) The same Collection_Object_ID to be associated with more than one collection at any point in time; or
b) A Collection_Object_ID to have the same value as any assigned User_Object_ID.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14. The Collection_Object_ID assigned by the CREATE USER TRACKING COLLECTION command may be obtained from the Current Command attributes page (see 7.1.2.29).

The get and set attributes parameters shall affect only the created user tracking collection. The get and set attributes parameters shall not affect the attributes of the collection, if any, specified by the SOURCE COLLECTION_OBJECT_ID field.

The capability is defined in (see 5.2.c) {{in 08-185}}.

The security parameters are defined in 5.2.8.

The collection type attribute in the Collection Information attributes page (see 7.1.2.10) shall be set to 01h (i.e., TRACKING).

The assigned Collection_Object_ID shall be placed in the Collection_Object_ID or User_Object_ID attribute in the Current Command attributes page (see 7.1.2.29).

If a CREATE USER TRACKING COLLECTION command causes the value in the number of collections, user tracking collections, and user objects attribute in the Partition Information attributes page (see 7.1.2.9) to exceed the value in the object count attribute in the Partition Quotas attributes page (see 7.1.2.13), then a quota error shall be generated (see 4.10.2). The quota testing principles described in 4.10.3 apply to the testing of the object count quota.

If a CREATE USER TRACKING COLLECTION command causes the value in the used capacity attribute in the Partition Information attributes page (see 7.1.2.9) to exceed the value in the capacity quota attribute in the Partition Quotas attributes page (see 7.1.2.13), then a quota error shall be generated (see 4.10.2). The quota testing principles described in 4.10.3 apply to the testing of the object count quota.

Change 5 – QUERY command modifications and enhancements

Description

The current QUERY command definition is the only OSD-2 usage of the user data region in the Data-Out Buffer as a CDB continuation. This should be changed to use the CDB continuation features defined in 08-185. Also, a request for QUERY to create and populate an output TRACKING collection is honored by proposing the details of how to do it.

The permissions changes that support the commands defined in change 5 appear in change 10 so that all of the permissions changes for all of the commands defined in this proposal can be reviewed as a group.

Proposed changes in OSD-2 r03

5.y.q Query list {{5.y is defined in 08-185}}

{{All of 5.y.q is new. The use of change markups is suspended for the remainder of 5.y.q.}}

The query list CDB continuation descriptor (see table x223) specifies the criteria for selecting the user objects whose User_Object_IDs are returned as matches by a QUERY command (see 6.21).

Table x223 — Query list CDB continuation descriptor format

Bit / Byte	7	6	5	4	3	2	1	0
	CDB continuation descriptor header							
0	(MSB)			CDB CONTINUATION DESCRIPTOR TYPE (0002h)				
1								(LSB)
2	Reserved							
3	Reserved					PAD LENGTH (p-n)		
4	(MSB)			CDB CONTINUATION DESCRIPTOR LENGTH (n-7)				
7								(LSB)
	CDB continuation descriptor type specific data							
8	Reserved				QUERY TYPE			
9	Reserved							
11								
12	Query criteria entry (see table x225) [first]							
	⋮							
	Query criteria entry (see table x225) [last]							
n								
	CDB continuation descriptor alignment bytes							
n+1	Pad bytes (for eight-byte alignment)							
p								

The CDB CONTINUATION DESCRIPTOR TYPE field contains 0002h (i.e., query list CDB continuation descriptor).

The PAD LENGTH field specifies the number of bytes containing zeros that follow the CDB continuation descriptor type specific data.

The CDB CONTINUATION DESCRIPTOR LENGTH field contains the number of bytes of CDB continuation descriptor type specific data that follow in this descriptor. The contents of the CDB CONTINUATION DESCRIPTOR LENGTH field shall be validated as described in 5.y.1. {{see 08-185}}

If the sum of the pad length and the CDB continuation descriptor length is not a multiple of eight, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

The QUERY TYPE field (see table x224) specifies the format of the query criteria entries that follow.

Table x224 — QUERY TYPE field values

Code	Description
0h	A match with any query criteria entry shall cause the user object to appear in the list.
1h	Matching all query criteria entries shall cause the user object to appear in the list.
2h to Fh	Reserved

Each query criteria entry (see table x225) specifies matching criteria for one attribute.

Table x225 — Query criteria entry format

Bit / Byte	7	6	5	4	3	2	1	0
0	Reserved							
1								
2	(MSB)	QUERY ENTRY LENGTH (n-3)						
3								(LSB)
4	(MSB)	ATTRIBUTES PAGE						
7								(LSB)
8	(MSB)	ATTRIBUTE NUMBER						
11								(LSB)
12	(MSB)	MINIMUM ATTRIBUTE VALUE LENGTH (m-13)						
13								(LSB)
14	(MSB)	MINIMUM ATTRIBUTE VALUE						
m								(LSB)
m+1	(MSB)	MAXIMUM ATTRIBUTE VALUE LENGTH (n-m-2)						
m+2								(LSB)
m+3	(MSB)	MAXIMUM ATTRIBUTE VALUE						
n								(LSB)

The QUERY ENTRY LENGTH field specifies the number of bytes that follow in the query entry.

The ATTRIBUTES PAGE field specifies the page number of the attribute value. If the attributes page is not between 0h and 2FFF FFFFh, inclusive, the command shall be terminated with a CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

The ATTRIBUTE NUMBER field specifies the attribute number within the attributes page specified by the ATTRIBUTES PAGE field of the attribute value.

The MINIMUM ATTRIBUTE VALUE LENGTH field specifies the number of bytes that follow in the MINIMUM ATTRIBUTE VALUE field.

The MINIMUM ATTRIBUTE VALUE field specifies the minimum attribute value necessary for a user object to meet the criteria.

The MAXIMUM ATTRIBUTE VALUE LENGTH field specifies the number of bytes that follow in the MAXIMUM ATTRIBUTE VALUE field.

The MAXIMUM ATTRIBUTE VALUE field specifies the maximum attribute value necessary for a user object to meet the criteria.

…

6.21 QUERY

6.21.1 Introduction

The QUERY command (see table 97) instructs the device server to return a list of the user objects that are members of the specified user tracking collection (see 3.1.a) and have attributes matching the specified values. The QUERY command is a multi-object command ~~(see 4.6.6.2)~~ (see 4.6.6.6).

Table 97 — QUERY command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)			SERVICE ACTION (88A0h)				
9								(LSB)
10				Reserved		ISOLATION		
~~11~~	~~Reserved~~		~~GET/SET CDBFMT~~		~~Reserved~~			
11	IMMED_TR	Reserved	GET/SET CDBFMT		Reserved			
12	TIMESTAMPS CONTROL							
13				Reserved				
15								
16	(MSB)			PARTITION_ID				
23								(LSB)
24	(MSB)			COLLECTION_OBJECT_ID				
31								(LSB)
32	(MSB)			ALLOCATION LENGTH				
39								(LSB)
~~40~~				Reserved				
~~47~~								
40	(MSB)			MATCHES COLLECTION_OBJECT_ID				
47								(LSB)
48	(MSB)			~~QUERY LIST LENGTH~~				
51				CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-185}}				(LSB)
52				Get and set attributes parameters (see 5.2.4)				
79								
80				Capability ~~(see 4.11.2.2)~~ (see 5.2.c) {{in 08-185}}				
183								
184				Security parameters (see 5.2.8) {{note new field size from 08-185}}				
235								

The contents of the ISOLATION field are defined in 5.2.5.

The IMMED_TR bit is defined in 5.2.i. If the IMMED_TR bit is set to one, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB if any of the following are true:

a) The ALLOCATION LENGTH field is not set to zero; or
b) The OUTPUT COLLECTION_OBJECT_ID field is set to zero.

If the IMMED_TR bit is set to one, the user tracking collections (see 3.1.a) specified by the COLLECTION_OBJECT_ID field and the OUTPUT COLLECTION_OBJECT_ID field shall be initialized before GOOD status is returned.

…

The COLLECTION_OBJECT_ID field specifies Collection_Object_ID (see 4.6.6) of the user tracking collection (see 3.1.a) to be processed. The device server shall constrain the Collection_Object_ID values as defined in ~~4.6.6.2~~ 4.6.6.6.

The ALLOCATION LENGTH field …

~~The QUERY LIST LENGTH field specifies the number of bytes to be transferred of query list data (see 6.21.2) that contain the attributes query criteria~~

The MATCHES COLLECTION_OBJECT_ID field specifies Collection_Object_ID (see 4.6.6) of the user tracking collection (see 3.1.a) in which the User_Object_IDs of the user objects that are members of the collection specified by the COLLECTION_OBJECT_ID field and have attributes matching the specified values. If the MATCHES COLLECTION_OBJECT_ID field is set to zero, the matching User_Object_IDs shall not be returned in a user tracking collection.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB if any of the following are true:

a) The MATCHES COLLECTION_OBJECT_ID field contains a non-zero value that is not the Collection_Object_ID of a user tracking collection (see 3.1.a); or
b) The command status attribute in the Command Tracking attributes page (see 7.1.2.c) of the output user tracking collection contains a value other than 0000 0000h or FFFF FFFFh.

If the matches user tracking collection contains any members, they shall be removed before the user tracking collection specified by the COLLECTION_OBJECT_ID field is processed.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{see 08-185}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if CDB continuation segment (see 5.x) {{see 08-185}}:

a) Does not contain one query list CDB continuation descriptor (see 5.y.q);
b) Contains more than one extension capabilities CDB continuation descriptor (see 5.y.z) {{see 08-185}} if the OUTPUT COLLECTION_OBJECT_ID field contains a non-zero value;
c) Contains more than zero extension capabilities CDB continuation descriptor (see 5.y.z) {{see 08-185}} if the OUTPUT COLLECTION_OBJECT_ID field contains zero;
d) Contains any CDB continuation descriptors other than the following:
   A) Copy user object source CDB continuation descriptor (see 5.y.q); and
   B) Extension capabilities CDB continuation descriptor (see 5.y.z) {{see 08-185}}.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14. The get ~~Get~~ and set attributes processing requirements specific to multi-object commands ~~are~~ defined in 4.6.6.2 shall affect only the collection specified by the COLLECTION_OBJECT_ID field. The get and set attributes parameters shall not affect the attributes of the collection specified by the MATCHES COLLECTION_OBJECT_ID field.

…

If a QUERY command causes the value in the used capacity attribute in the Partition Information attributes page (see 7.1.2.9) to exceed the value in the capacity quota attribute in the Partition Quotas attributes page (see 7.1.2.13), then a quota error shall be generated (see 4.10.2). The quota testing principles described in 4.10.3 apply to the testing of the object count quota.

~~6.21.2 Query list format~~

~~The query list (see table 98) specifies the criteria for selecting the user objects whose User_Object_IDs are returned in the matches list.~~

~~**Table 98 — Query list format**~~

~~Bit~~ / ~~Byte~~	~~7~~	~~6~~	~~5~~	~~4~~	~~3~~	~~2~~	~~1~~	~~0~~
~~0~~	~~Reserved~~				~~QUERY TYPE~~			
~~1~~ – ~~3~~	~~Reserved~~							
	~~Query criteria entries~~							
~~4~~	~~Query criteria entry 0 (see table 100)~~							
	⋮							
~~n~~	~~Query criteria entry x (see table 100)~~							

~~The QUERY TYPE field (see table 99) specifies the format of the query criteria entries that follow.~~

~~**Table 99 — QUERY TYPE field values**~~

~~Code~~	~~Description~~
~~0h~~	~~A match with any query criteria entry shall cause the user object to appear in the list.~~
~~1h~~	~~Matching all query criteria entries shall cause the user object to appear in the list.~~
~~2h to Fh~~	~~Reserved~~

~~Each query criteria entry (see table 100) specifies matching criteria for one attribute.~~

~~The QUERY ENTRY LENGTH field specifies the number of bytes that follow in the query entry.~~

~~The ATTRIBUTES PAGE field specifies the page number of the attribute value. If the attributes page is not between 0h and 2FFF FFFFh, inclusive, the command shall be terminated with a CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.~~

Table 100 — Query criteria entry format

Bit / Byte	7	6	5	4	3	2	1	0
0	Reserved							
1								
2	(MSB)	QUERY ENTRY LENGTH (n-3)						
3								(LSB)
4	(MSB)	ATTRIBUTES PAGE						
7								(LSB)
8	(MSB)	ATTRIBUTE NUMBER						
11								(LSB)
12	(MSB)	MINIMUM ATTRIBUTE VALUE LENGTH (m-13)						
13								(LSB)
14	(MSB)	MINIMUM ATTRIBUTE VALUE						
m								(LSB)
m+1	(MSB)	MAXIMUM ATTRIBUTE VALUE LENGTH (n-m-2)						
m+2								(LSB)
m+3	(MSB)	MAXIMUM ATTRIBUTE VALUE						
n								(LSB)

The ATTRIBUTE NUMBER field specifies the attribute number within the attributes page specified by the ATTRIBUTES PAGE field of the attribute value.

The MINIMUM ATTRIBUTE VALUE LENGTH field specifies the number of bytes that follow in the MINIMUM ATTRIBUTE VALUE field.

The MINIMUM ATTRIBUTE VALUE field specifies the minimum attribute value necessary for a user object to meet the criteria.

The MAXIMUM ATTRIBUTE VALUE LENGTH field specifies the number of bytes that follow in the MAXIMUM ATTRIBUTE VALUE field.

The MAXIMUM ATTRIBUTE VALUE field specifies the maximum attribute value necessary for a user object to meet the criteria.

6.21.3 Matches list parameter data format

…

Change 6 – DIFF_READ …

Description

The OSD TWG Snapshots proposal includes a DIFF_READ function, and this proposal instantiates that as the READ MAPS AND COMPARE command. The name change is needed to emphasize the similarities in the data processed and returned by the READ MAP and READ MAPS AND COMPARE commands.

Proposed changes in OSD-2 r03

5.y.u User object {{5.y is defined in 08-185}}

{{All of 5.y.u is new. The use of change markups is suspended for the remainder of 5.y.u.}}

The user object CDB continuation descriptor (see table x226) specifies a user object to be processed by a command (e.g., a user object input to the READ MAPS AND COMPARE command (see 6.n)).

Table x226 — User object CDB continuation descriptor format

Bit / Byte	7	6	5	4	3	2	1	0
	CDB continuation descriptor header							
0	(MSB)			CDB CONTINUATION DESCRIPTOR TYPE (0100h)				
1								(LSB)
2	Reserved							
3	Reserved					PAD LENGTH (000b)		
4	(MSB)			CDB CONTINUATION DESCRIPTOR LENGTH (n-7)				
7								(LSB)
	CDB continuation descriptor type specific data							
8	(MSB)			PARTITION_ID				
15								(LSB)
16	(MSB)			USER_OBJECT_ID				
23								(LSB)

The CDB CONTINUATION DESCRIPTOR TYPE field contains 0100h (i.e., user object CDB continuation descriptor).

The PAD LENGTH field is set to zero to indicate that no pad bytes are needed to eight byte align a user object CDB continuation descriptor. If the PAD LENGTH field is not set to zero in a user object CDB continuation descriptor, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

The CDB CONTINUATION DESCRIPTOR LENGTH field contains the number of bytes that follow in this descriptor. The contents of the CDB CONTINUATION DESCRIPTOR LENGTH field shall be validated as described in 5.y.1 {{see 08-185}}.

The PARTITION_ID field specifies the Partition_ID (see 4.6.4) of the partition that contains the user object. If the partition identified by the PARTITION_ID field does not exist, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

The USER_OBJECT_ID field specifies the User_Object_ID of the user object (see 4.6.5). If the user object identified by the USER_OBJECT_ID field does not exist, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

…

6.22 READ MAP

6.22.1 Introduction

The READ MAP command (see table 102) requests …

…

The REQUESTED MAP TYPE field (see table 103) specifies the map descriptor type values (see table 106) that shall be returned in the parameter data.

Table 103 — REQUESTED MAP TYPE field

Code	Description
0000h	Return all map type values.
0001h	Return only WRITTEN_DATA map type values.
0002h	Return only DATA_HOLE map type values.
0003h	Return only DAMAGED_DATA map type values.
0004h to 7FFFh	Reserved
~~8000h~~	~~Return only DAMAGED_ATTRIBUTES map type values.~~
8000h	Return only attributes map type values.
8001h to 8002h	Reserved
8003h	Return only DAMAGED_ATTRIBUTES map type values.
8004h to FFFFh	Reserved
~~8001h to FFFFh~~	~~Reserved~~

…

The security parameters are defined in 5.2.8.

The parameter data returned by a READ MAP command is described in 6.22.2.

The map descriptors in the parameter data with a map descriptor type (see 6.22.2) less than 8000h shall be sorted from smallest to largest byte offset

6.22.2 READ MAP command and READ MAPS AND COMPARE command parameter data

The parameter data returned by a READ MAP command and a READ MAPS AND COMPARE command (see table 104) contains descriptors that describe the user data and attributes associated with ~~the~~ a user object.

Table 104 — READ MAP command and READ MAPS AND COMPARE command parameter data

{{no other changes in table 104}}

The ADDITIONAL LENGTH field indicates the number of bytes of READ MAP command or READ MAPS AND COMPARE command (see 6.n) parameter data that follow. If the parameter data is truncated due to insufficient allocation length, the ADDITIONAL LENGTH field shall not be altered to reflect the truncation (i.e., the additional length indicates the number of bytes that would follow if the allocation length had been infinite). If the untruncated number of bytes that follow is greater than FFFF FFFF FFFF FFFFh the additional length shall be set to FFFF FFFF FFFF FFFFh.

Each map descriptor (see table 105) contains 16 bytes and provides information about user object attributes or one range of bytes within ~~the~~ a user object's user data.

Table 105 — Map descriptor format

Bit / Byte	7	6	5	4	3	2	1	0
~~0~~				~~Reserved~~				
~~1~~								
0	(MSB)			MAP DESCRIPTOR INDEX				
1								(LSB)
2	(MSB)			MAP DESCRIPTOR TYPE				
3								(LSB)
4	(MSB)			DATA LENGTH				
7								(LSB)
8	(MSB)			BYTE OFFSET				
15								(LSB)

In the parameter data for a READ MAP command the MAP DESCRIPTOR INDEX field is reserved. In the parameter data for the READ MAPS AND COMPARE command, the MAP DESCRIPTOR INDEX field indicates which user object the map descriptor represents as shown in table x227.

Table x227 — MAP DESCRIPTOR INDEX field

Value	Description
0000h	The user object specified by fields in the CDB
0001h	The user object specified by the first user object CDB continuation descriptor (see 5.y.u) in the CDB continuation segment (see 5.x) {{in 08-158}}
all other values	Reserved

The MAP DESCRIPTOR TYPE field (see table 106) indicates the type of information this map descriptor contains.

Table 106 — MAP DESCRIPTOR TYPE field

Code	Name	Description
0000h		Reserved
0001h	WRITTEN_DATA	This map descriptor indicates the byte offset and data length of user data that has been written to stable storage (see 4.13) and is available for reading.
0002h	DATA_HOLE	This map descriptor indicates the byte offset and data length of a user data that lies between two WRITTEN_DATA regions, but for which no user data has been written.
0003h	DAMAGED_DATA	This map descriptor indicates the byte offset and data length of user data in which uncorrectable damage has been detected (see 4.11.3).
0004h	PAST_LAST_BYTE	This map descriptor is used by the READ MAPS AND COMPARE command (see 6.n) to indicate bytes that have other map descriptor types in other user objects but are beyond the user object logical length attribute value in the User Object Information attributes page (see 7.1.2.11) for the user object indicated by the MAP DESCRIPTOR INDEX field.
0005h to 8000h		Reserved
8001h	NORMAL_ATTRIBUTES	This map descriptor indicates that one or more user object attributes contain are undamaged. [a]
8002h		Reserved
8003h	DAMAGED_ATTRIBUTES	This map descriptor indicates that one or more user object attributes contain uncorrectable damage. [a]
8004h to FFFFh		Reserved
~~0004h to 7FFFh~~		~~Reserved~~
~~8000h~~	~~DAMAGED_ATTRIBUTES~~	~~This map descriptor indicates that one or more user object attributes contain uncorrectable damage.~~
~~8001h to FFFFh~~		~~Reserved~~

[a] All the attributes in a user object are represented by a single map descriptor.

If the map descriptor type is greater than 7FFFh, the BYTE OFFSET field is reserved. If the map descriptor type is less than 8000h, the BYTE OFFSET field indicates the starting byte address of the user data that this map descriptor represents. The byte offset in the first map descriptor shall be equal to or greater than the contents of the DATA MAP BYTE OFFSET field in the CDB. The byte offset in any map descriptor after the first shall be greater than or equal to the ~~sum of~~ the byte offset ~~and data length~~ in the preceding map descriptor.

If the map descriptor type is greater than 7FFFh, the DATA LENGTH field is reserved. If the map descriptor type is less than 8000h, the DATA LENGTH field indicates the number of bytes of user data, starting at byte offset, that this map descriptor represents.

~~If the READ MAP parameter data contains a map descriptor with the MAP DESCRIPTOR TYPE field set to 7FFFh DAMAGED_ATTRIBUTES, then that map descriptor shall be the last one in the parameter data.~~

The parameter data shall not contain any map descriptors in which the MAP DESCRIPTOR TYPE field is set to a value that is less than 8000h following the first map descriptor in which MAP DESCRIPTOR TYPE field is set to a value that is greater than 8000h.

6.n READ MAPS AND COMPARE {{All of 6.n is new. Change markups suspended.}}

The READ MAPS AND COMPARE command (see table x228) requests that the device server compare the map information that would be returned by a READ MAP command (see 6.22) for two user objects and return information about where the maps are different.

Table x228 — READ MAPS AND COMPARE command

Bit / Byte	7	6	5	4	3	2	1	0
8	(MSB)							
9				SERVICE ACTION (88B2h)				(LSB)
10				Reserved			ISOLATION	
11	Reserved			GET/SET CDBFMT	Reserved		COMPARISON SCOPE	
12				TIMESTAMPS CONTROL				
13								
15				Reserved				
16	(MSB)							
23				PARTITION_ID (see 5.2.7)				(LSB)
24	(MSB)							
31				USER_OBJECT_ID (see 5.2.11)				(LSB)
32	(MSB)							
39				ALLOCATION LENGTH (see 5.2.2)				(LSB)
40	(MSB)							
47				DATA MAP BYTE OFFSET				(LSB)
48	(MSB)							
51				CDB CONTINUATION LENGTH (see 5.2.x) {{in 08-158}}				(LSB)
52								
79				Get and set attributes parameters (see 5.2.4)				
80								
183				Capability (see 5.2.c) {{in 08-185}}				
184								
235				Security parameters (see 5.2.8)				

The contents of the ISOLATION field are defined in 5.2.5.

The GET/SET CDBFMT field specifies the format of the get and set attributes parameters as described in 5.2.4.

The COMPARISON SCOPE field (see table x229) specifies the scope of the map comparison to be performed.

Table x229 — COMPARISON SCOPE field

Value	Description
000b	Only differences in the read map data shall be returned
001b	Differences in the read map data, and WRITTEN_DATA map descriptor types (see 6.22.2) where the data is not shared between the user objects shall be returned
010b to 111b	Reserved

The contents of the TIMESTAMPS CONTROL field are defined in 5.2.10.

The contents of the PARTITION_ID field are defined in 5.2.7.

The contents of the USER_OBJECT_ID field are defined in 5.2.11.

The contents of the ALLOCATION LENGTH field are defined in 5.2.2.

The DATA MAP BYTE OFFSET field specifies the first byte of user data to be compared. If the DATA MAP BYTE OFFSET field specifies a byte that is beyond the user object logical length attribute value in the User Object Information attributes page (see 7.1.2.11) of any user object involved in the comparison, then the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The contents of the CDB CONTINUATION LENGTH field are defined in 5.2.x {{in 08-158}}. If the CDB CONTINUATION LENGTH field contains zero, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

The command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST, if the CDB continuation segment (see 5.x) {{in 08-158}}:

a) Does not contain one extension capabilities CDB continuation descriptor (see 5.y.z) {{in 08-158}};
b) Does not contain one user object CDB continuation descriptor (see 5.y.u); or
c) Contains any CDB continuation descriptors other than:
   A) The extension capabilities CDB continuation descriptor; and
   B) The user object CDB continuation descriptor.

The get and set attributes parameters are defined in 5.2.4. The format of the Data-In Buffer and Data-Out Buffer when attributes are being retrieved or set is described in 4.14.

The capability is defined in 5.2.c {{see 08-185}}.

The security parameters are defined in 5.2.8.

The parameter data returned by a READ MAPS AND COMPARE command is described in 6.22.2.

The map descriptors in the parameter data shall represent only bytes or attributes where the contents of the MAP DESCRIPTOR TYPE field (see 6.22.2) have different values among the user objects being compared under the requirements established by the COMPARISON SCOPE field.

The map descriptors in the parameter data with a map descriptor type (see 6.22.2) less than 8000h shall be sorted as follows:

1) Smallest to largest map descriptor index; and
2) Smallest to largest byte offset.

The map descriptors in the parameter data with a map descriptor type greater than 8000h shall be sorted from smallest to largest map descriptor index.

Change 7 – REMOVE command and collections

Description

Nowhere does the REMOVE command definition mention collections. This oversight should be remedied.

Proposed changes in OSD-2 r03

6.24 REMOVE

The REMOVE command (see table 108) removes a user object from any LINKED collections (see 4.6.6.2) in which it is a member, and deletes ~~a~~ the user object.

…

Change 8 – Quota changes

Description

The user tracking collections described in this proposal constitute a new source of quota consumption possibilities that need to be reflected in a minor changes the Partition information attributes page.

Proposed changes in OSD-2 r03

7.1.2.9 Partition Information attributes page

…

For all partitions except partition zero, the used capacity attribute (number 81h) shall contain the number of allocated bytes for the partition as described in this subclause. For partition zero, the used capacity attribute shall contain the number of allocated bytes for partition zero and all other partitions described in this subclause. The number of allocated bytes shall be computed as the sum of the following:

a) The number of bytes used by:
   A) The partition or partitions;
   B) All collections within the partition or partitions; ~~and~~
   C) All user tracking collections within the partition or partitions; and
   D) All user objects within the partition or partitions including attributes bytes;
   and
b) The number of unused reserved bytes computed as:
   A) Value in the reserved data space attribute in this Partition Information attributes page minus the value in the actual data space attribute in this Partition Information attributes page; or
   B) Zero if the value in the actual data space attribute in this Partition Information attributes page is larger than the value in the reserved data space attribute in this Partition Information attributes page.

…

Change 9 – Permissions to access the snapshot/clone tracking well known collection

Description

The entity that successfully performs a CREATE SNAPSHOT or CREATE CLONE command with the IMMED_TR bit set to one should be able to track the progress of the create process via the snapshot/clone tracking well known collection with needing a new capability. This is all the more important because substantial effort is require to make any capabilities that access the destination partition, beginning with a SET KEY command.

Also, the PAR capability descriptor needs to be updated to allow OBSD selected Partition_IDs in CREATE SNAPSHOT and CREATE CLONE commands.

Proposed changes in OSD-2 r03

4.11.2.2.3 PAR capability object descriptor

… {{The text shown in black here comes from 08-158, not OSD-2 r03.}}

The ALLOWED PARTITION_ID field specifies the partition to which access is allowed. The command shall be terminated as described in 4.11.2.2.n, if any of the following are true:

a) If the OBJECT TYPE field contains 02h (i.e., PARTITION), ~~the command is not CREATE PARTITION (see 6.7), and~~ the ALLOWED PARTITION_ID field contains zero, and the command is not one of the following:~~; or~~
   A) CREATE PARTITION (see 6.7);
   B) CREATE SNAPSHOT (see 6.e); or
   C) CREATE CLONE (see 6.d);
   or
b) If the OBJECT TYPE field contains 01h (i.e., ROOT) and the ALLOWED PARTITION_ID field contains a value other than zero.

The command that accesses a partition or a well known collection (see 4.6.6.5) in a partition shall be terminated as described in 4.11.2.2.n, if none of the capabilities associated with the command (i.e., the capability in the CDB (see 5.2.1) and the capabilities, if any, in the CDB continuation segment (see 5.x)) match all of the following criteria:

a) If the OBJECT TYPE field contains:
   A) 02h (i.e., PARTITION), then the partition being accessed (e.g., the partition specified by the PARTITION_ID field in the CDB of a LIST command) matches the ALLOWED PARTITION_ID field in the capability; or
   B) 01h (i.e., ROOT), then the partition being accessed (e.g., the partition specified by the PARTITION_ID field in the CDB of a LIST command) is zero;
   and
b) The User_Object_ID (see 4.6.2) associated with the object being accessed, if any, is:
   A) Zero ~~zero.~~; or
   B) The Collection_Object_ID of a well known collection (see 4.6.6.5).

…

Table 23 — Commands allowed by specific capability field values

Commands allowed and CDB fields whose contents are restricted by capability field contents, if any	Capability Field values that allow a command		
	Object Type Name	Permission Bits That Are Set To One	Object Descriptor Name
…	…	…	…
A LIST COLLECTION command addressed to a partition	PARTITION	READ and M_OBJECT	PAR
A LIST COLLECTION command addressed to a well known collection with the LIST_ATTR bit set to zero	PARTITION	READ	PAR
…	…	…	…

…

Table 24 — Attribute retrieving and setting function allowed by specific capability field values

Attribute-Related Functions Allowed	Capability Field values that allow attribute-related functions		
	Object Type Name	Permission Bits That Are Set To One	Object Descriptor Name
…	…	…	…
Retrieval of attributes from the Current Command attributes page	PARTITION or ROOT	GET_ATTR	PAR
Retrieval of attributes from an attributes page associated with a well known collection (see 4.6.6.5)	PARTITION	GET_ATTR	PAR
…	…	…	…

…

Change 10 – Permissions, reservations, etc. changes

Description

The changes in this group might belong in the miscellaneous changes (see change 11), but they have sufficient importance to some reviewers that they have been separated from the others.

Proposed changes in OSD-2 r03

…

Table 16 — Permissions bit mask format

Bit / Byte	7	6	5	4	3	2	1	0
49	READ	WRITE	GET_ATTR	SET_ATTR	CREATE	REMOVE	OBJ_MGMT	APPEND
50	DEV_MGMT	GLOBAL	POL/SEC	M_OBJECT	QUERY	GBL_REM	Reserved	
51	Reserved							
52	Reserved							
53	Reserved							

…

A GBL_REM (global remove) bit set to one allows all the user objects, collections, and partitions referenced by a single command to be removed (e.g., the wholesale removal of objects performed by a REMOVE PARTITION command (see 6.27) with the remove scope field set to 111b). A GBL_REM bit set to zero prohibits the removal of all the user objects, collections, and partitions referenced by a single command.

…

Table 23 — Commands allowed by specific capability field values (part 1 of 3)

Commands allowed and CDB fields whose contents are restricted by capability field contents, if any	Capability Field values that allow a command		
	Object Type Name	Permission Bits That Are Set To One	Object Descriptor Name
…	…	…	…
A CREATE CLONE command [a]			
Source partition	PARTITION	READ	PAR
Destination partition	PARTITION	WRITE	PAR
A CREATE PARTITION command	PARTITION	CREATE	PAR
Combinations of OBJECT TYPE field, PERMISSION BITS field, and OBJECT DESCRIPTOR TYPE field values not shown in this table and table 24 are reserved. The capability fields not shown in this table may place additional limits on the objects that are allowed to be accessed.			
[a] This command accesses multiple objects. … {{see 08-185}}			

Table 23 — Commands allowed by specific capability field values (part 2 of 3)

Commands allowed and CDB fields whose contents are restricted by capability field contents, if any	Capability Field values that allow a command		
	Object Type Name	**Permission Bits That Are Set To One**	**Object Descriptor Name**
A CREATE SNAPSHOT command [a]			
Source partition	PARTITION	READ	PAR
Destination partition	PARTITION	WRITE	PAR
A CREATE USER TRACKING COLLECTION command with the SOURCE COLLECTION_OBJECT_ID field set to zero	COLLECTION	CREATE	COL
A CREATE USER TRACKING COLLECTION command with the SOURCE COLLECTION_OBJECT_ID field set to a non-zero value [a]			
The collection specified by the COLLECTION_OBJECT_ID field in the CDB	COLLECTION	CREATE and WRITE	COL
The collection specified by the SOURCE COLLECTION_OBJECT_ID field in the CDB	COLLECTION	READ	COL
…	…	…	…
A DETACH CLONE command	PARTITION	WRITE	PAR
…	…	…	…
A QUERY command ~~addressed to a collection~~ with the MATCHES COLLECTION_OBJECT_ID field set to zero	COLLECTION	QUERY	COL
A QUERY command with the MATCHES COLLECTION_OBJECT_ID field set to a non-zero value [a]			
The collection specified by the COLLECTION_OBJECT_ID field in the CDB	COLLECTION	QUERY	COL
The collection specified by the MATCHES COLLECTION_OBJECT_ID field in the CDB	COLLECTION	WRITE	COL
…	…	…	…
A READ MAPS AND COMPARE command [a]			
Each user object that participates in the comparison	USER	DEV_MGMT	USER
…	…	…	…

Combinations of OBJECT TYPE field, PERMISSION BITS field, and OBJECT DESCRIPTOR TYPE field values not shown in this table and table 24 are reserved.
The capability fields not shown in this table may place additional limits on the objects that are allowed to be accessed.

[a] This command accesses multiple objects. … {{see 08-185}}

Table 23 — Commands allowed by specific capability field values (part 3 of 3)

Commands allowed and CDB fields whose contents are restricted by capability field contents, if any	Capability Field values that allow a command		
	Object Type Name	Permission Bits That Are Set To One	Object Descriptor Name
A REFRESH SNAPSHOT OR CLONE command [a]			
Source partition	PARTITION	READ	PAR
Destination partition	PARTITION	APPEND	PAR
…	…	…	…
A REMOVE PARTITION command with the REMOVE SCOPE field set to 000b or 001b	PARTITION	REMOVE	PAR
A REMOVE PARTITION commanders the REMOVE SCOPE field set to 111b	PARTITION	REMOVE and GBL_REM	PAR
A RESTORE PARTITION FROM SNAPSHOT command [a]			
Main partition	PARTITION	READ	PAR
Snapshot partition	PARTITION	WRITE	PAR
…	…	…	…
Combinations of OBJECT TYPE field, PERMISSION BITS field, and OBJECT DESCRIPTOR TYPE field values not shown in this table and table 24 are reserved. The capability fields not shown in this table may place additional limits on the objects that are allowed to be accessed.			
[a] This command accesses multiple objects. … {{see 08-185}}			

…

Table 47 — OSD commands that are allowed in the presence of various reservations (part 1 of 2)

OSD Command	Addressed logical unit has this type of persistent reservation held by another I_T nexus				
	From any I_T nexus		From registered I_T nexus (RR all types)	From not registered I_T nexus	
	Write Excl	Excl Access		Write Excl RR	Excl Access – RR
…	…	…	…	…	…
COPY USER OBJECTS {{see 08-185}}	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
CREATE CLONE	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
Key: **Excl**=Exclusive, **RR**=Registrants Only or All Registrants					

Table 47 — OSD commands that are allowed in the presence of various reservations (part 2 of 2)

OSD Command	Addressed logical unit has this type of persistent reservation held by another I_T nexus				
	From any I_T nexus		From registered I_T nexus (RR all types)	From not registered I_T nexus	
	Write Excl	Excl Access		Write Excl RR	Excl Access – RR
CREATE SNAPSHOT	Conflict	Conflict	Allowed	Conflict	Conflict
CREATE USER TRACKING COLLECTION	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
DETACH CLONE	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
READ MAPS AND COMPARE	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
REFRESH SNAPSHOT OR CLONE	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
RESTORE PARTITION FROM SNAPSHOT	Conflict	Conflict	Allowed	Conflict	Conflict
…	…	…	…	…	…
Key: **Excl**=Exclusive, **RR**=Registrants Only or All Registrants					

…

Table x4 — CDB CONTINUATION DESCRIPTOR TYPE field {{see 08-185}}

Value	Description	Reference
0000h	No more continuation descriptors [a]	
0001h	Scatter/gather list	5.y.c {{see 08-185}}
0002h	Query list	5.y.q
0100h	User object	5.y.u
0101h	Copy user object source	5.y.h {{see 08-185}}
FFEEh	Extension capabilities	5.y.z {{see 08-185}}
all other values	Reserved	

[a] Since the CDB continuation segment pad bytes, if any, are set to zero (see 5.x) {{see 08-185}}, encountering a CDB continuation descriptor type of zero shall be processed in the same way as reaching the last byte of the CDB continuation segment.

…

Table 57 — Commands for OSD type devices

Command name	Operation code	Service action [a]	Type	Reference
…	…	…	…	…
COPY USER OBJECTS {{see 08-185}}	7Fh	8893h	M	6.h
…	…	…	…	…
CREATE COLLECTION	7Fh	8895h	O [b]	6.6
CREATE CLONE	7Fh	88A8h	O [c]	6.d
…	…	…	…	…
CREATE SNAPSHOT	7Fh	88A9h	O [d]	6.e
CREATE USER TRACKING COLLECTION	7Fh	8894h	O [b]	6.t
…	…	…	…	…
DETACH CLONE	7Fh	88AAh	O [c]	6.f
…	…	…	…	…
REFRESH SNAPSHOT OR CLONE	7Fh	88ABh	O [c, d]	6.r
…	…	…	…	…
READ MAPS AND COMPARE	7Fh	88B2h	M	6.n
…	…	…	…	…
RESTORE PARTITION FROM SNAPSHOT	7Fh	88ACh	O [d]	6.s
…	…	…	…	…

Type Key: M = Command implementation is mandatory.
O = Command implementation is optional.

[a] No entry in the service action column means that the SERVICE ACTION field does not apply to the command. Service action codes values between 8800h and 8F7Fh that are not listed in this table are reserved for future standardization. Service action code values between 8F80h and 8FFFh may have vendor specific command assignments.

[b] Support for this command is mandatory if collections are supported (see 4.6.6).

[c] Support for this command is mandatory if the maximum clones count attribute in the Root Information attributes page (see 7.1.2.8) is defined and contains a value other than zero.

[d] Support for this command is mandatory if the maximum snapshots count attribute in the Root Information attributes page is defined and contains a value other than zero.

[e] [e] Unless the security method in effect …

Change 11 – Miscellaneous changes

Description

This change shows all the modifications (e.g., updates to summary tables and updates to annexes) that are needed to fully implement the other changes in this proposal.

Proposed changes in OSD-2 r03

…

…

…

Table 121 — Attributes pages defined by this standard (part 1 of 2)

Page Number	Page Name	Page Format Defined	Support Requirements	Reference
0h	User Object Directory	No	Mandatory	7.1.2.7
1h	User Object Information	No	Mandatory	7.1.2.11
2h	User Object Quotas	Yes	Mandatory	7.1.2.14
3h	User Object Timestamps	Yes	Mandatory	7.1.2.18
4h	Collections	Yes	Optional	7.1.2.19
5h	User Object Policy/Security	Yes	Mandatory	7.1.2.24
6h	User Object Error Recovery	Yes	Mandatory	7.1.2.28
7h to 7Fh	Reserved			
C+0h	Collection Directory	No	Mandatory	7.1.2.6
C+1h	Collection Information	No	Mandatory	7.1.2.10
C+2h	Reserved			
C+3h	Collection Timestamps	Yes	Mandatory	7.1.2.17
…	…	…	…	…
~~C+4h~~	~~Reserved~~			
C+4h	Command Tracking	No	Mandatory	7.1.2.c
…	…	…	…	…
C+5h	Collection Policy/Security	Yes	Mandatory	7.1.2.23
C+6h	Collection Error Recovery	Yes	Mandatory	7.1.2.27
C+7h to C+7Fh	Reserved			
P+0h	Partition Directory	No	Mandatory	7.1.2.5
P+1h	Partition Information	No	Mandatory	7.1.2.9
P+2h	Partition Quotas	Yes	Mandatory	7.1.2.13
P+3h	Partition Timestamps	Yes	Mandatory	7.1.2.16
P+4h	Attributes Access	No	Mandatory	7.1.2.20
P+5h	Partition Policy/Security	Yes	Mandatory	7.1.2.22
P+6h	Partition Error Recovery	Yes	Mandatory	7.1.2.26
P+7h	Snapshots Information	No	Mandatory	7.1.2.e
~~P+7h~~ P+8h to P+7Fh	Reserved			

Table 121 — Attributes pages defined by this standard (part 2 of 2)

Page Number	Page Name	Page Format Defined	Support Requirements	Reference
R+0h	Root Directory	No	Mandatory	7.1.2.4
R+1h	Root Information	No	Mandatory	7.1.2.8
R+2h	Root Quotas	Yes	Mandatory	7.1.2.12
R+3h	Root Timestamps	Yes	Mandatory	7.1.2.15
R+4h	Reserved			
R+5h	Root Policy/Security	Yes	Mandatory	7.1.2.21
R+6h	Root Error Recovery	Yes	Mandatory	7.1.2.25
R+7h to R+7Fh	Reserved			
F000 0000h to FFFF FFFDh	Reserved			
FFFF FFFEh	Current Command	Yes	Mandatory	7.1.2.29

…{{Change table 128 from 'no lines between rows' to 'lined between rows'.}}

Table 128 — Object accessibility attribute values

Code	Description
0000 0000h	Allow all accesses
0000 0001h	~~Deny all write accesses and allow all read accesses~~ Allow all read accesses, but deny all write accesses
~~0000 0002h to FFFF FFFFh~~	~~Reserved~~
0000 0002h to 8000 0000h	Reserved
8000 0001h	Allow all read accesses, but deny all write accesses except those of CREATE SNAPSHOT commands (see 6.e), CREATE CLONE commands (see 6.d), REFRESH SNAPSHOT OR CLONE commands (see 6.r), RESTORE PARTITION FROM SNAPSHOT commands (see 6.s), and REMOVE PARTITION commands (see 6.27)
8000 0002h to FFFF FFFFh	Reserved

…

Table B.1 — Numerical order OSD service action codes

Service Action	Command
…	…
8890h to 8891h	Reserved
8892h	CREATE AND WRITE
~~8893h to 8894h~~	~~Reserved~~
8893h	COPY USER OBJECTS {{see 08-185}}
8894h	CREATE TRACKING COLLECTION
8895h	CREATE COLLECTION
…	…
88A3h	SET MEMBER ATTRIBUTES
~~88A4h to 88B0h~~	~~Reserved~~
88A4h to 88A7h	Reserved
88A8h	CREATE CLONE
88A9h	CREATE SNAPSHOT
88AAh	DETACH CLONE
88ABh	REFRESH SNAPSHOT OR CLONE
88ACh	RESTORE PARTITION FROM SNAPSHOT
88ADh to 88B0h	Reserved
88B1h	READ MAP
88B2h	READ MAPS AND COMPARE
88B3h to 8F7Bh	Reserved
~~88B2h to 8F7Bh~~	~~Reserved~~
8F7Ch	PERFORM SCSI COMMAND
8F7Dh	PERFORM TASK MANAGEMENT FUNCTION
8F7Eh to 8F7Fh	Obsolete
8F80h to 8FFFh	Vendor specific

… {{Table C.1 contains the new attribute definition for this proposal and for 08-185.}}

Table C.1 — Numerical order attributes defined by this standard (part 1 of 6)

Page Number	Page Name	Attribute Number	Attribute
0h	User Object Directory	0h	"INCITS T10 User Object Directory"
		1h	"INCITS T10 User Object Information"
		2h	"INCITS T10 User Object Quotas"
		3h	"INCITS T10 User Object Timestamps"
		4h	"INCITS T10 Collections"
		5h	"INCITS T10 User Object Policy/Security"
		6h	"INCITS T10 User Object Error Recovery"
1h	User Object Information	0h	Page identification
		1h	Partition_ID
		2h	User_Object_ID
		9h	Username
		81h	Used capacity
		82h	User object logical length
		83h	Object accessibility
		D1h	Actual data space
		D2h	Reserved data space
2h	User Object Quotas	0h	Page identification
		1h	Maximum user object length
3h	User Object Timestamps	0h	Page identification
		1h	Created time
		2h	Attributes accessed time
		3h	Attributes modified time
		4h	Data accessed time
		5h	Data modified time
4h	Collections	0h	Page identification
		1h	Collection pointer
		…	“ “
		FFFF FF00h	“ “
5h	User Object Policy/Security	0h	Page identification
		4000 0001h	Policy access tag
6h	User Object Error Recovery	0h	Page identification
		1h	User object damage summary
		3h	Last damaged object data time
		4h	Last damaged attributes time

Table C.1 — Numerical order attributes defined by this standard (part 2 of 6)

Page Number	Page Name	Attribute Number	Attribute
3000 0000h	Partition Directory	3000 0000h	"INCITS T10 Partition Directory"
		3000 0001h	"INCITS T10 Partition Information"
		3000 0002h	"INCITS T10 Partition Quotas"
		3000 0003h	"INCITS T10 Partition Timestamps"
		3000 0004h	"INCITS T10 Attributes Access"
		3000 0005h	"INCITS T10 Partition Policy/Security"
		3000 0006h	"INCITS T10 Partition Error Recovery"
		3000 0007h	"INCITS T10 Snapshots Information"
3000 0001h	Partition Information	0h	Page identification
		1h	Partition_ID
		9h	Username
		81h	Used capacity
		83h	Object accessibility
		84h	Potential used capacity increment
		C1h	Number of collections and user objects
		D1h	Actual data space
		D2h	Reserved data space
		200h	Default snapshot duplication method
		201h	Default clone duplication method
		202h	Default copy user objects duplication method
		300h	Default snapshot time of duplication method
		301h	Default clone time of duplication method
		302h	Default copy user objects time of duplication method
3000 0002h	Partition Quotas	0h	Page identification
		1h	Default maximum user object length
		1 0001h	Capacity quota
		1 0002h	Object count
		1 0081h	Collections per user object
3000 0003h	Partition Timestamps	0h	Page identification
		1h	Created time
		2h	Attributes accessed time
		3h	Attributes modified time
		4h	Data accessed time
		5h	Data modified time
		FFFF FFFEh	Timestamp bypass
3000 0004h	Attributes Access	0h	Page identification
		1h	Allowed attributes access
		…	" " "
		FFFF FFFEh	" " "

Table C.1 — Numerical order attributes defined by this standard (part 3 of 6)

Page Number	Page Name	Attribute Number	Attribute
3000 0005h	Partition Policy/Security	0h	Page identification
		1h	Default security method
		2h	Oldest valid nonce
		3h	Newest valid nonce
		4h	Request nonce list depth
		5h	Frozen working key bit mask
		7FFFh	Partition key identifier
		8000h	Working key identifier
		…	" " "
		800Fh	" " "
		4000 0001h	Policy access tag
		4000 0002h	User object policy access tag
3000 0006h	Partition Error Recovery	0h	Page identification
		1h	Partition damage summary
		2h	Contained objects damage summary
		3h	Last damaged object data time
		4h	Last damaged object attributes time
		5h	Last damaged contained object time
		6h	Number of damaged objects
3000 0007h	Snapshots Information	0h	Page identification
		1h	Partition type
		80h	Source partition
		81h	Snapshot backward
		82h	Snapshot forward
		83h	Clone destination
		…	" "
		FFFFh	" "
		2 0001h	Snapshots count
		2 0002h	Clones count
		2 000Ch	Branch depth
		2 0011h	Create completion time
		2 0012h	Refresh completion time
		2 0013h	Restore completion time
		2 0014h	Restore Partition_ID
…	…	…	…

Table C.1 — Numerical order attributes defined by this standard (part 4 of 6)

Page Number	Page Name	Attribute Number	Attribute
6000 0000h	Collection Directory	6000 0000h	"INCITS T10 Collection Directory"
		6000 0001h	"INCITS T10 Collection Information"
		6000 0003h	"INCITS T10 Collection Timestamps"
		6000 0004h	"INCITS T10 Command Tracking"
		6000 0005h	"INCITS T10 Collection Policy/Security"
		6000 0006h	"INCITS T10 Collection Error Recovery"
6000 0001h	Collection Information	0h	Page identification
…	…	…	…
		~~Bh~~	~~Number of members~~
		~~Ch~~	~~Multi-object operation in progress~~
…	…	…	…
6000 0004h	Command Tracking	0h	Page identification
		1h	Percent complete
		2h	Command status
		3h	Sense data
		~~Bh~~ 10h	Number of members
		11h	Objects processed
		12h	Newer objects skipped
		13h	Missing objects skipped
		F000 0000h	Vendor specific
		…	" "
		FFFF FFFEh	" "
…	…	…	…
9000 0000h	Root Directory	9000 0000h	"INCITS T10 Root Directory"
		9000 0001h	"INCITS T10 Root Information"
		9000 0002h	"INCITS T10 Root Quotas"
		9000 0003h	"INCITS T10 Root Timestamps"
		9000 0005h	"INCITS T10 Root Policy/Security"
		9000 0006h	"INCITS T10 Root Error Recovery"

Table C.1 — Numerical order attributes defined by this standard (part 5 of 6)

Page Number	Page Name	Attribute Number	Attribute
9000 0001h	Root Information	0h	Page identification
		3h	OSD System ID
		4h	Vendor identification
		5h	Product identification
		6h	Product model
		7h	Product revision level
		8h	Product serial number
		9h	OSD name
		Ah	Maximum CDB continuation length
		80h	Total capacity
		81h	Used capacity
		83h	Object accessibility
		C0h	Number of partitions
		100h	Clock
		110h	Default isolation method
		111h	Supported isolation methods
		120h	Data atomicity guarantee
		121h	Data atomicity alignment
		122h	Attributes atomicity guarantee
		123h	Data/attributes atomicity multiplier
		1C1h	Maximum snapshots count
		1C2h	Maximum clones count
		1CCh	Maximum branch depth
		200h	Supported object duplication method
		…	" " " "
		2FFh	" " " "
		300h	Supported time of duplication method
		…	" " " "
		30Fh	" " " "
		310h	Support for duplicated object freezing
		0700 0001h	Supported CDB continuation descriptor type
		…	" " " " "
		0700 FFFFh	" " " " "
9000 0002h	Root Quotas	0h	Page identification
		1h	Default maximum user object length
		1 0001h	Partition capacity quota
		1 0002h	Partition object count
		1 0081h	Partition collections per user object
		2 0002h	Partition count

Table C.1 — Numerical order attributes defined by this standard (part 6 of 6)

Page Number	Page Name	Attribute Number	Attribute
9000 0003h	Root Timestamps	0h	Page identification
		2h	Attributes accessed time
		3h	Attributes modified time
		FFFF FFFEh	Timestamp bypass
9000 0005h	Root Policy/Security	0h	Page identification
		1h	Default security method
		2h	Oldest valid nonce limit
		3h	Newest valid nonce limit
		6h	Partition default security method
		7h	Supported security methods
		9h	Adjustable clock
		Ah	Boot epoch
		7FFDh	Master key identifier
		7FFEh	Root key identifier
		8000 0000h	Supported integrity check value algorithm
		…	" " " " "
		8000 000Fh	" " " " "
		8000 0010h	Supported DH group
		…	" " "
		8000 001Fh	" " "
9000 0006h	Root Error Recovery	0h	Page identification
		1h	Root damage summary
		2h	Contained objects damage summary
		3h	Last damaged object data time
		4h	Last damaged object attributes time
		5h	Last damaged contained object time
		6h	Number of damaged partitions
FFFF FFFEh	Current Command	0h	Page identification
		1h	Response integrity check value
		2h	Object Type
		3h	Partition_ID
		4h	Collection_Object_ID or User_Object_ID
		5h	Starting byte address of append