



Date: 13 January 2025  
 To: T10 Technical Committee  
 From: Ralph O. Weber  
 Subject: SBC-6 Define Formatting By LBA Ranges

## Introduction

A *Divide And Conquer* approach is proposed herein as a possible way to tame the hideously long amounts of time required to completely format a modern, high-capacity disk.

In effect, a whole-disk formatting is divided in to multiple user data format operations (see 4.33.4) that are performed separately on 'formatting LBA ranges'.

By performing opportunistic formatting operations on LBA ranges where the host has requested data to be written, increased write times replace the existing long time intervals during which the disk is veritably unusable as the penalty associated with disk formatting.

A noticeable difference between T13/g24143 and this proposal is that the process which ACS calls 'initialization' SBC describes as 'formatting'. As one SBC-5 example, a FORMAT UNIT command requests an initialization, not vice versa.

Furthermore, formatting has a format operation but there are no initialization operations (except for a few fuzzy-logic cases in depopulation an depopulation restoration, which this proposal will replace with suitable format operation stuff).

Regrettably, these differences increase the difficulty of comparing the SCSI and SATA proposals. On the plus side, however, the proposed formatting model dovetails nicely with the capabilities defined for the FFMT field (see table 40)

## Revision History

r0 Initial revision

Unless otherwise indicated additions are shown in underlined blue, deletions in ~~red-strikethrough~~, and comments in **green**. Differences between this revision and the previous revision, if any, are highlighted with change bars.

## Proposed Changes in SBC-6 (based on SBC-5 r08)

<<<Yes, some (but not all) glossary entries are included primarily for reference.>>>

...

### 3.1.36 format operation

process by which a device server initializes the medium in a logical unit

Note 1 to entry: See 4.10 and 4.33.

...

### 3.1.114 write cache operation

process by which a device server writes logical blocks for one or more LBAs to the cache (see 4.7.1)

**3.1.115 write command**

command that requests write operations

Note 1 to entry: See 4.2.2.

**3.1.116 write medium operation**

process by which a device server writes logical blocks for one or more LBAs to the medium using the parameters specified in the Read-Write Error Recovery mode page (see 6.5.10)

**3.1.117 write operation**

process by which a device server performs operations as described in this standard

...

**4.33 Format operations****4.33.1 Format operations overview**

A format operation results in the device server:

- a) configuring the logical block length and number of logical blocks of the logical unit as specified by the block descriptor (see 6.5.2); and
- b) performing the following as specified by:
  - A) a FORMAT UNIT command (see 5.4):
    - a) configure protection information;
    - b) perform defect management;
    - c) initialize LBAs; and
    - d) vendor specific medium certification;
  - or
  - B) a FORMAT WITH PRESET command (see 5.5):
    - a) change LBA configuration for a new device type;
    - b) update peripheral device type;
    - c) update VPD pages;
    - d) perform medium defect management; and
    - e) initialize LBAs as described in 4.33.3.

The degree that the medium is altered by a format operation is vendor specific. A format operation is requested by a FORMAT UNIT command or a FORMAT WITH PRESET command.

**4.33.2 Performing a format operation**

Before performing a format operation, the device server shall stop all:

- a) enabled power condition timers (see SPC-6);
- b) timers for enabled background scan operations (see 4.23); and
- c) timers or counters enabled for device-specific background functions.

As the result of completing a format operation, the device server shall:

- a) initialize and start all enabled timers and counters for background functions; and
- b) initialize and start all operational (see SPC-6) power condition timers.

While performing a format operation, the device server shall:

- a) process commands already in a task set when a FORMAT UNIT command or a FORMAT WITH PRESET command is received in a vendor specific manner;
- b) process an INQUIRY command by returning parameter data based on the condition of the logical unit before beginning the FORMAT UNIT command or the FORMAT WITH PRESET command (i.e., INQUIRY data does not change until successful completion of a format operation);
- c) process a REQUEST SENSE command by returning parameter data containing sense data with the sense key set to NOT READY, the additional sense code set to LOGICAL UNIT NOT READY, FORMAT IN PROGRESS, and the PROGRESS INDICATION field in the sense data (see SPC-6) set to indicate the progress of the format operation;
- d) process REPORT LUNS commands;
- e) terminate all commands, except INQUIRY commands, REPORT LUNS commands, and REQUEST SENSE commands, with CHECK CONDITION status with the sense key set to NOT READY and the additional sense code set to LOGICAL UNIT NOT READY, FORMAT IN PROGRESS;
- f) remove all Background Scan Results log parameters (see 6.4.2.3) from the Background Scan Results log page, if supported; and
- g) remove all Pending Defect log parameters (see 6.4.8.3) from the Pending Defects log page, if supported.

For a FORMAT UNIT command, the application client may specify:

- a) that the device server clear the existing GLIST;
- b) a list of address descriptors that the device server adds to the GLIST;
- c) that the device server enable a certification operation that adds address descriptors for physical blocks with medium defects discovered during the certification operation to the GLIST; and
- d) the behavior of the device server if it is not able to:
  - A) access the PLIST or GLIST; or
  - B) determine whether the PLIST or GLIST exists.

For a FORMAT WITH PRESET command, the device server manages all defect information. No information is specified by the application client (e.g., GLIST and PLIST).

[If the UDRFO\\_EN bit \(see 6.5.10\) is set to one and a format operation is performed with the FFMT field that is set to 01b \(see table 40\), then performing the format operation includes actions described in 4.33.4.2.](#)

### 4.33.3 Completing a format operation

#### 4.33.3.1 Completing a format operation overview

If a format operation completes without error, then:

- a) stream resources, if any, shall be released;
- b) if the logical unit is a zoned block device, then all LBAs in the logical unit are as defined in ZBC-3;
- c) if the logical unit is fully provisioned (i.e., the LBPME bit (see 5.21.2) is set to zero), then all LBAs in the logical unit are mapped (see 4.7.2); or
- d) if the logical unit supports logical block provisioning management (i.e., the LBPME bit is set to one), then if the LBPRZ field (see 6.6.9) is set to:
  - A) 000b, then each LBA in the logical unit shall be either:
    - a) mapped, if an initialization pattern was specified that does not match the vendor-specific data returned by a read command for an unmapped LBA (see 4.7.4.4); or
    - b) unmapped, if no initialization pattern was specified or an initialization pattern was specified that matches the vendor-specific data returned by a read command for an unmapped LBA (see 4.7.4.4);
  - B) xx1b, then each LBA in the logical unit:

- a) shall be mapped, if the format operation did not initialize the user data to all zeroes for the logical block referenced by that LBA;
  - b) shall be unmapped, if the format operation initialized the user data to all zeroes for the logical blocks referenced by all valid LBAs in the logical unit; or
  - c) may be unmapped, if the format operation initialized the user data to all zeroes for the logical block referenced by that LBA, and the format operation did not initialize the user data to all zeroes for the logical blocks referenced by all valid LBAs in the logical unit;
- and
- C) 010b, then each LBA in the logical unit:
    - a) shall be mapped, if an initialization pattern was specified that does not match the provisioning initialization pattern; or
    - b) shall be unmapped, if no initialization pattern was specified or an initialization pattern was specified that matches the provisioning initialization pattern;
- and
- e) if the format operation was performed for a FORMAT WITH PRESET command, then until Power On condition (see SAM-6) is detected, the device server shall:
    - A) terminate all commands except REQUEST SENSE with CHECK CONDITION status, with the sense key set to NOT READY and the additional sense code set to LOGICAL UNIT NOT READY, POWER CYCLE REQUIRED; and
    - B) for the REQUEST SENSE command, return sense data with the sense key set to NOT READY and the additional sense code set to LOGICAL UNIT NOT READY, POWER CYCLE REQUIRED.

If a format operation is aborted (e.g., by a power on condition or a hard reset condition (see SAM-6)) or completes with an error, then the logical unit may become format corrupt. Format corrupt may be cleared by a format operation that completes without error. If the logical unit is format corrupt, then the device server shall terminate any:

- a) logical block access command other than commands that may correct the format corrupt condition (e.g., FORMAT UNIT command); and
- b) any STREAM CONTROL command with the STR\_CTRL field set to 01b (i.e., open stream), with CHECK CONDITION status,

with the sense key set to MEDIUM ERROR and the additional sense code set to MEDIUM FORMAT CORRUPTED.

#### 4.33.3.2 Completing read commands **after** a successful format operation

##### 4.33.3.2.1 Completing read commands overview

Following a successful format operation [that leaves the Percent of LBA Ranges to be Formatted log parameter \(see 6.4.4.plri\) set to zero](#) and before a write operation to an LBA, a read command or verify command that specifies that LBA shall be processed by the device server as described in:

- a) 4.33.3.2.2, 4.33.3.2.3, and 4.33.3.2.4 for a mapped LBA; and
- b) 4.7.4.4 for an unmapped LBA.

##### 4.33.3.2.2 With FFMT field set to 00b

If:

- a) the FFMT field (see table 40) was set to 00b in the most recent successful FORMAT UNIT command [\(see 5.4\), or](#)

[b\)](#) the most recent successful command was a FORMAT WITH PRESET command (see 5.5),

then subsequent read commands or verify commands that complete without error are processed using:

- a) the user data set as specified by:
  - A) the initialization pattern, if any;
  - B) the provisioning initialization pattern, if applicable; or
  - C) the manufacturer's default initialization pattern;
 and
- b) the protection information, if any, set to FFFF\_FFFF\_FFFF\_FFFFh.

#### **4.33.3.2.3 With FFMT field set to 01b**

If the FFMT field (see table 40) was set to 01b in the most recent successful FORMAT UNIT command, then subsequent read commands or verify commands:

- a) with unrecovered medium errors are processed as described in 4.18.1;
- b) with pseudo unrecovered errors are processed as described in 4.18.2;
- c) should be processed using unspecified logical block data and complete without error, if protection information is disabled; or
- d) if protection information is enabled, then:
  - A) may be processed using unspecified logical block data and complete without error; or
  - B) may terminate with CHECK CONDITION status with sense data that indicates that the protection information check fails as defined in 5.16 or 5.36.

[If the UDRFO\\_EN bit \(see 6.5.10\) is set to one, then the processing time for write commands may be increased as described in 4.33.4.3.](#)

#### **4.33.3.2.4 With FFMT field set to 10b**

If the FFMT field (see table 40) was set to 10b in the most recent successful FORMAT UNIT command, then the device server may:

- a) return unspecified logical block data and complete subsequent read commands without error;
- b) complete subsequent verify commands without error; or
- c) terminate subsequent read commands or verify commands with CHECK CONDITION status with the sense key set to HARDWARE ERROR, MEDIUM ERROR, or ABORTED COMMAND.

### **4.33.4 User data LBA range format operations**

<<<All of 4.33.4 is new. For better readability, use of change markups is suspended for the remainder of 4.33.4.>>>

#### **4.33.4.1 User data LBA range format operations overview**

User data LBA range format operations substitute numerous small formatting operations in place of a single format operation that affects the entire disk. By performing opportunistic format operations on LBA ranges where the application client has requested data to be written, increased write times become the penalty associated with disk formatting, instead of long time intervals during when the disk is veritably unusable.

A UDRFO\_EN bit (see 6.5.10) set to one specifies that user data LBA range format operations are being used whenever possible.

The following fields indicate the characteristics for user data LBA range format operations:

- a) the MAXIMUM FORMAT RANGE SIZE field (see 6.6.2) indicates, as a power of two, the maximum number of LBAs in a user data formatting LBA range; and
- b) the FORMAT RANGE ALIGNMENT field (see 6.6.2) indicates, as a power of two, the first LBA in a user data formatting LBA range.

If the application client writes an entire user data formatting LBA range, then the device server shall treat that user data formatting LBA range as having been formatted. If the application client writes less than an entire user data formatting LBA range starting near the first LBA in that range, then the device server:

- a) completes that write command as completed, even if that write command requires forced unit access; and
- b) finishes the user data LBA range format operation while awaiting the next write command.

In this way, the application client may interleave parts of the processing that creates write commands with the device server's processing of the user data LBA range format operation.

The device server shall finish necessary parts of a user data LBA range format operation before allowing other read commands or write commands to access affected LBAs in a user data formatting LBA range that is being processed.

As part of treating a user data formatting LBA range as having been formatted, the device server updates the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) so that it indicates how many of the established user data LBA range format operations have been completed. The Percent of LBA Ranges to be Formatted log parameter:

- a) is set to 100 by the device server as part of preparing for a user data range initialization operations (see 4.33.4.2); and
- b) shall not be set to zero by the device server until all user data formatting LBA ranges have been formatted.

#### **4.33.4.2 Managing user data LBA range format operations**

If a format operation is performed (see 4.33.2), the FFMT field, or equivalent, is set to 01b (see table 40), and the UDRFO\_EN bit (see 6.5.10) is set to one, then as part of performing the format operation, the device server shall:

- a) set the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) to 100;
- b) initialize its method for tracking which user data formatting LBA ranges have been formatted in a way that indicates that every user data formatting LBA range has not been formatted; and
- c) complete the format operation as described in 4.33.2.

Upon completion of the format operation and while the Percent of LBA Ranges to be Formatted log parameter is not set to zero, the device server shall:

- a) include the processing of a user data LBA range format operation (see 4.33.4.3) in the processing of every write command; and
- b) process read commands and verify commands as described in 4.33.4.4.

The completion of the work initiated by the original format operation is associated with:

- a) the Percent of LBA Ranges to be Formatted log parameter being set to zero; and
- b) the device server's method for tracking which user data formatting LBA ranges have been formatted indicating that every user data formatting LBA range has been formatted.

#### 4.33.4.3 Performing a user data LBA range format operation

While the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) is not set to zero, if the device server's method for tracking which user data formatting LBA ranges have been formatted indicates that any logical block being written has not been formatted, then the device server shall:

- a) process that write command as specified; and
- b) perform a format operation whose scope is limited to those LBAs in that user data formatting LBA range that have not been written by that write command.

Upon completing the format operation, the device server shall update:

- a) its method for tracking which user data formatting LBA ranges have been formatted to indicate that the affected user data formatting LBA range has been formatted; and
- b) the Percent of LBA Ranges to be Formatted log parameter based on the updated user data formatting LBA ranges information.

While the Percent of LBA Ranges to be Formatted log parameter is not set to zero, the application client may:

- a) increase the time anticipated to complete write commands based information indicated by the MAXIMUM FORMAT RANGE SIZE field (see 6.6.2); and
- b) optimize the performance of write commands based on information provided by the FORMAT RANGE ALIGNMENT field (see 6.6.2).

If a utilization management capability such as Command Duration Limits (see SPC-7) causes the termination of a command that started a user data LBA range format operation, then the device server may:

- a) finish the initialization; or
- b) treat the affected user data formatting LBA range as if no user data LBA range format operation had ever been started.

#### 4.33.4.4 Read command and verify command considerations for user data LBA range format operations

While the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) is not set to zero, if the device server's method for tracking which user data formatting LBA ranges have been formatted indicates that logical block being read or verified has:

- a) not been formatted, then the device server shall perform the read or verify processing described in 4.33.3.2.3 shall be performed; or
- b) been formatted, then normal read or verify processing is be performed.

<<<This is the end of the change markups suspension.>>>

---

**Editors Note 1 - ROW:** The following changes in the *storage element depopulation and restoration* model (see 4.36 in [SBC-5 r08](#)) rectify ancient the gaffe from the days when the SCSI Re/Depop changes were carried over from ACS-x based on the assumption that SCSI 'initialization' was the same as SATA 'initialization'. The anomaly that hides behind this bogus assumption amounts to a failure to heed the inter-committee fact that, although 'format' is a dirty word in the SATA realm, 'format' is the concept that (since SCS-2) SCSI has fitted between 'device initialization' and the application-client-to-device-server interface (to use modern terminology).

From the days of [17-014r3](#) onward, one of the nagging problems was the absence of a defined 'initialization operation' in SBC-4 and later SBC-5. As will be seen in the proposed changes, the 'operation' that SCSI would want to be used in a 'format operation'. FWIW [17-014r3](#) names and defines 'depopulate operations' and 'truncate operations'; however, 'initialization' is simply 'initialization'.

---

...

#### 4.36.4 Storage element depopulation

...

A REMOVE ELEMENT AND TRUNCATE command specifies that the device server shall perform:

- a) ~~shall perform~~ a depopulate operation (see 4.36.4.2);
- b) ~~shall perform~~ a truncate operation (see 4.36.4.3); and
- c) ~~may perform an initialization~~: a format operation (see 4.33) that is processed as if the FFMT field (see table 40) is set to one of the codes shown in table x1.

**Table x1 – FFMT codes for storage element depopulation and storage element restoration**

<u>Code <sup>a</sup></u>	<u>Command completion time</u>	<u>Read operation and write operation processing while some logical blocks have not been formatted</u>
<u>00b</u>	<u>sufficient additional time to format all logical blocks</u>	<u>see 4.33.2</u>
<u>01b</u>	<u>sufficient additional time to configure any special processing for read operations and write operations</u>	<u>For read operations, the requirements in 4.33.3.2.3 apply. <sup>b</sup></u>
<u>10b</u>	<u>very minimal additional time</u>	<u>For read operations, the requirements in 4.33.3.2.4 apply. <sup>b</sup></u>
<sup>a</sup> <u>Which code is used is vendor specific.</u> <sup>b</sup> <u>For write operations, the requested data transfers are performed as if the format operation has completed without error.</u>		

The depopulate operation, truncate operation, and ~~initialization, if any,~~ format operation may continue after the successful completion of the REMOVE ELEMENT AND TRUNCATE command.

If ~~an initialization~~ a format operation is not performed, then user data written before the depopulate operation may be readable in any accessible logical block.



The processing of a REMOVE ELEMENT AND TRUNCATE command shall not change:

- a) the logical block length (see 4.5);
- b) the lowest aligned logical block address (see 4.6.1); and
- c) the protection type (see 4.21.2).

A REMOVE ELEMENT AND TRUNCATE command may be issued for each storage element that is to be removed from the current operating configuration. The effect of the processing of multiple REMOVE ELEMENT AND TRUNCATE commands shall be cumulative (see 5.26).

A device server may have a limit on the number of storage elements that may be depopulated. If the device server is requested to depopulate a storage element in excess of this limit, the device server may terminate that command with a sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.

Upon successful completion of the REMOVE ELEMENT AND TRUNCATE command; the depopulate operation, the truncate operation, and the ~~initialization, if any;~~ format operation may continue as background operations.

<<<This sentence appears to repeat the contents of the first paragraph after table x1. Are both instances needed?>>> While those operations are in progress, the device server shall:

- a) provide pollable sense data (see SPC-6) with the sense key set to NOT READY, the additional sense code set to DEPOPULATION IN PROGRESS and the PROGRESS INDICATION field in the sense data set to indicate the progress of those operations; and
- b) process other commands as described in 4.36.6.

Upon the completion of the depopulate operation, the truncate operation, and the ~~initialization, if any;~~ format operation the contents of the user data area may have no relation to the contents of the user data area before the processing of the REMOVE ELEMENT AND TRUNCATE command.

Any depopulate operation, truncate operation, and ~~initialization~~ format operation initiated by a REMOVE ELEMENT AND TRUNCATE command shall resume after any hard reset or logical unit reset. If any depopulate operation, truncate operation, and ~~initialization~~ format operation requested initiated by a REMOVE ELEMENT AND TRUNCATE command is interrupted by a power cycle then that operation shall be terminated and the logical unit may become format corrupt.

If a depopulate operation, a truncate operation, or ~~an initialization~~ a format operation initiated by the REMOVE ELEMENT AND TRUNCATE command does not complete successfully, then the logical unit may become format corrupt.

If a REMOVE ELEMENT AND TRUNCATE command ~~requests~~ initiates a depopulate operation, a truncate operation, or ~~an initialization~~ a format operation and that ~~requested~~ action is terminated by a power cycle as described in this subclause (i.e., 4.36.4.1) with the result that the logical unit:

- a) becomes format corrupt, then the device server shall terminate any medium access command with CHECK CONDITION status, with the sense key set to MEDIUM ERROR and the additional sense code set to DEPOPULATION FAILED; or
- b) does not become format corrupt, then the device server should terminate any medium access command with CHECK CONDITION status, with the sense key set to NOT READY and the additional sense code set to DEPOPULATION INTERRUPTED.

#### 4.36.4.2 Depopulate operations

...

#### 4.36.5 Storage element restoration

...

A RESTORE ELEMENTS AND REBUILD command specifies that the device server shall perform:

- 1) ~~shall perform~~ a depopulation revocation operation (see 4.36.5.2);
- 2) ~~shall perform~~ a rebuild operation (see 4.36.5.3); and
- 3) ~~may perform an initialization~~: a format operation (see 4.33) that is processed as if the FFMT field (see table 40) is set to one of the codes shown in table x1.

If ~~an initialization~~ a format operation is ~~not~~ performed as if the FFMT field is set to 10b (see table x1 and table 40), user data written before the depopulation revocation operation may be readable in any accessible logical block.

The depopulation revocation operation, rebuild operation, and ~~initialization~~ format operation, if any, may continue after the successful completion of the RESTORE ELEMENTS AND REBUILD command.

The processing of a RESTORE ELEMENTS AND REBUILD command shall not change:

- a) the logical block length (see 4.5);
- b) the lowest aligned logical block address (see 4.6.1); or
- c) the protection type (see 4.21.2).

Upon successful completion of the RESTORE ELEMENTS AND REBUILD command; the depopulation revocation operation, rebuild operation, and the ~~initialization, if any~~, format operation shall continue as background operations. While those operations are in progress, the device server shall:

- a) provide pollable sense data (see SPC-6) with the sense key set to NOT READY, the additional sense code set to DEPOPULATION RESTORATION IN PROGRESS and the PROGRESS INDICATION field in the sense data set to indicate the progress of those operations; and
- b) process other commands as described in 4.36.6.

Upon the completion of the depopulation revocation operation, rebuild operation, and ~~initialization, if any~~, format operation the contents of the user data area may have no relation to the contents of the user data area before the processing of the RESTORE ELEMENTS AND REBUILD command.

Any depopulation revocation operation, rebuild operation, and ~~initialization~~ format operation ~~requested~~ initiated by a RESTORE ELEMENTS AND REBUILD command shall resume after any interruption hard reset or logical unit reset. If any depopulate operation, truncate operation, and ~~initialization~~ format operation ~~requested~~ initiated by a RESTORE ELEMENTS AND REBUILD command is interrupted by a power cycle then that operation shall be terminated and the logical unit may become format corrupt.

If a depopulation revocation operation, a rebuild operation, or ~~an initialization~~ a format operation initiated by the RESTORE ELEMENTS AND REBUILD command does not complete successfully, then the logical unit may become format corrupt.

If a RESTORE ELEMENTS AND REBUILD command ~~requests~~ initiates a depopulation revocation operation, rebuild operation, or ~~initialization~~ format operation and that ~~requested~~ action is terminated by a power cycle interruption as described in this subclause (i.e., 4.36.5.1) with the result that the logical unit:

- a) becomes format corrupt, then the device server shall terminate any medium access command with CHECK CONDITION status, with the sense key set to MEDIUM ERROR and the additional sense code set to DEPOPULATION RESTORATION FAILED; or

- b) does not become format corrupt, then the device server should terminate any medium access command with CHECK CONDITION status, with the sense key set to NOT READY and the additional sense code set to DEPOPULATION RESTORATION INTERRUPTED.

#### **4.36.5.2 Depopulation revocation operation**

...

### 5.4.1 FORMAT UNIT command overview

...

The fast format (FFMT) field is described in table 40. <<<Insert a period.>>>

**Table 40 – FFMT field description**

Code	<a href="#">UDRFO EN bit</a> (see 6.5.10)	Description	Support
00b	<a href="#">n/a</a>	The device server initializes the medium (see 4.10) as specified in the CDB and parameter list before completing the format operation. After successful completion of the format operation, read commands and verify commands are processed as described in 4.33.3.2.1 and 4.33.3.2.2.	Mandatory
01b	<a href="#">0</a>	<p>The device server initializes the medium (see 4.10) without overwriting the medium (i.e., resources for managing medium access are initialized and the medium is not written) before completing the format operation. After successful completion of the format operation, read commands and verify commands are processed as described in 4.33.3.2.1 and 4.33.3.2.3.</p> <p>If the device server determines that the options specified in this FORMAT UNIT command are incompatible with the read command and verify command requirements described in 4.33.3.2.3, then the device server shall not perform the format operation and shall terminate the FORMAT UNIT command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FAST FORMAT.</p>	Optional
	<a href="#">1</a>	<p><u>The device server:</u></p> <ol style="list-style-type: none"> <li>1) <u>initializes:</u> <ol style="list-style-type: none"> <li>A) <u>the medium (see 4.10); and</u></li> <li>B) <u>subsequent user data LBA range format operations as described in 4.33.4.2;</u></li> </ol> </li> <li>2) <u>completes the format operation.</u></li> </ol> <p><u>While the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) is not set to zero:</u></p> <ol style="list-style-type: none"> <li>a) <u>write commands are performed as described in 4.33.4.3; and</u></li> <li>b) <u>read and verify commands are performed as described in 4.33.4.4.</u></li> </ol> <p><u>Upon the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) being set to zero, the medium is formatted.</u></p>	
10b	<a href="#">n/a</a>	The device server initializes the medium (see 4.10) without overwriting the medium (i.e., resources for managing medium access are initialized and the medium is not written) before completing the format operation. After successful completion of the format operation, read commands and verify commands are processed as described in 4.33.3.2.1 and 4.33.3.2.4.	Optional
11b		Reserved	

## 6.4.4 Format Status log page

...

Table 191 – Format Status log page parameter codes

Parameter code	Description	Resettable or Changeable <sup>a</sup>	Reference	Support
0000h	Format Data Out	Never	6.4.4.2	Mandatory
0001h	Grown Defects During Certification	Never	6.4.4.3	Mandatory
0002h	Total Blocks Reassigned During Format	Never	6.4.4.4	Mandatory
0003h	Total New Blocks Reassigned	Never	6.4.4.5	Mandatory
0004h	Power On Minutes Since Format	Never	6.4.4.6	Mandatory
<u>0005h</u>	<u>Percent of LBA Ranges to be Formatted</u>	<u>Never</u>	<u>6.4.4.plri</u>	<u>Optional <sup>b</sup></u>
<del>0005h to 7FFFh</del> <u>0006h to 7FFFh</u>	Reserved			
8000h to FFFFh	Vendor specific			Optional
<sup>a</sup> The keywords in this column – Always, Reset Only, and Never – are defined in 6.4.1.2. <sup>b</sup> <u>If the device server allows the UDRFO_EN bit to be set to one, then support is mandatory for this log parameter.</u>				

...

## 6.4.4.plri Percent of LBA Ranges to be Formatted log parameter

<<<All of 6.4.4.plri is new. For better readability, use of change markups is suspended for the remainder of 6.4.4.plri.>>>

The Percent of LBA Ranges to be Formatted log parameter for the Format Status log page has the format shown in table x2.

Table x2 – Percent of LBA Ranges to be Formatted log parameter

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB) _____							
1	PARAMETER CODE (0004h) _____ (LSB)							
2	Parameter control byte – binary format list log parameter (see SPC-6)							
	DU	Obsolete	TSD	Obsolete		FORMAT AND LINKING		
...	...							
3	PARAMETER LENGTH (04h) _____							
4	Reserved _____							
...								
6								
7	PERCENT OF LBA RANGES TO BE FORMATTED							

The `PARAMETER CODE` field is described in SPC-6 and shall be set to the value shown in table x2 for the Percent of LBA Ranges to be Formatted log parameter.

The `DU` bit, and the `FORMAT AND LINKING` field for the Percent of LBA Ranges to be Formatted log parameter shall be set for a binary format list log parameter as described in SPC-6.

The target save disable (TSD) bit (see SPC-6) shall be set to zero for the Percent of LBA Ranges to be Initialized log parameter, indicating that the logical unit saves the Percent of LBA Ranges to be Formatted log parameter at vendor specific intervals without any request from an application client.

The `PARAMETER LENGTH` field is described in SPC-6 and shall be set to the value shown in table x2 for the Percent of LBA Ranges to be Formatted log parameter.

The `PERCENT OF LBA RANGES TO BE FORMATTED` field contains whole number percentage (i.e., 0 to 100) indication of the number of user data formatting LBA ranges that are waiting to be formatted (see 4.33.4) after the processing of a command that initiates user data formatting (e.g., a `RESTORE ELEMENTS AND REBUILD` command (see 5.29)). A percentage of zero indicates that no LBAs are waiting to be formatted, and 100 indicates that every LBA is waiting to be formatted.

The Percentage LBA Ranges to be Formatted log parameter is based on a number of formatting LBA ranges so that a meaningful progress indication is possible without using fractional percentages.

<<<This is the end of the change markups suspension.>>>

### 6.5.10 Read-Write Error Recovery mode page

The Read-Write Error Recovery mode page (see table 254) specifies the error recovery parameters the device server shall use during:

- a) read medium operations; or
- b) write medium operations.

**Table 254 – Read-Write Error Recovery mode page**

Bit	7	6	5	4	3	2	1	0
Byte								
...	...							
6	Obsolete							
7	LBP <small>ERE</small>	M <small>WR</small>	Reserved			Restricted for MMC-6		
7	LBP <small>ERE</small>	M <small>WR</small>	UDR <small>FO</small> _E <small>N</small>	Reserved			Restricted for MMC-6	
8	WRITE RETRY COUNT							
9	Reserved							
10	(MSB)							
11	RECOVERY TIME LIMIT							
	(LSB)							

...

A user data range format operations enabled (UDRFO\_EN) bit set to zero specifies that user data range format operations (see 4.33.4) are not initiated or performed as part of other operations (e.g., format operations (see 4.33)). A UDRFO\_EN bit set to one specifies that user data range format operations may be performed as part of other operations (e.g., a storage element restoration (see 4.36.5 and 4.33.3.2.3)).

If the Percent of LBA Ranges to be Formatted log parameter (see 6.4.4.plri) is not set to zero and a MODE SELECT commands attempts to change the value in the UDRFO\_EN bit, then the device server shall terminate that command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST, the additional sense code set to INVALID FIELD IN PARAMETER LIST.

... <<<No other changes are proposed in 6.5.10.>>>

## 6.6.2 Block Device Characteristics VPD page 259

...

Table 259 – Block Device Characteristics VPD page

Byte	Bit	7	6	5	4	3	2	1	0
...		...							
8		Reserved	MACT	Obsolete		RBWZ	BOCS	FUAB	VBULS
9		Reserved							
...									
44									
9		<a href="#">FORMAT RANGE ALIGNMENT</a>							
10		<a href="#">MAXIMUM FORMAT RANGE SIZE</a>							
11		<a href="#">Reserved</a>							
12		...							
...		...							

...

[The FORMAT RANGE ALIGNMENT field indicates, as a power of two, the alignment of ranges of logical blocks that the device server formats when performing a user data LBA range format operation \(see 4.33.4\) \(e.g., a FORMAT RANGE ALIGNMENT field set to 12 indicates that user data LBA range format operation are performed on ranges of logical block whose first LBA is a multiple of 4096\).](#)

[The MAXIMUM FORMAT RANGE SIZE field indicates, as a power of two, the largest number of logical blocks that are written by one user data LBA range format operation \(see 4.33.4\) \(e.g., MAXIMUM FORMAT RANGE SIZE field set to 16 indicates that the number of logical blocks written by one user data LBA range format operation is less than or equal to 65 536\).](#)

[The largest number of logical blocks written by one user data LBA range format operation may be interpreted to be an indication of how much time overhead results from performing small formatting operations as described in 4.33.4.](#)

...