

To: T10 Technical Committee and T13 Technical Committee
 From: Bill Martin (bill.martin@samsung.com), Samsung
 Date: 6 May 2025
 Subject: Clarification of Informational Exceptions

Revision history

Revision 0 (6 May 2025) First revision

References

SBC-5r8

Overview

Informational Exceptions uses three terms in reference to Informational Exceptions: process, report, and detect. In actuality Informational Exceptions are either detected or reported. This proposal details the changes necessary to remove the reference to “processing” informational exceptions.

Proposal

modify SBC-5r8 4.15.10 as follows:

4.15.10 Power loss effects on caches

The power, if any, needed to maintain a non-volatile cache may decrease to the point that the device server is unable to ensure the non-volatility of the cache for a vendor specific interval of time (e.g., the battery voltage becomes too low to sustain cache contents beyond a vendor specific time). If this occurs and the Extended INQUIRY Data VPD page (see SPC-6) indicates that the device server contains non-volatile cache (i.e., NV_SUP bit set to one), then:

- a) if the reporting of informational exceptions control warnings is enabled (i.e., the EWASC bit is set to one in the Informational Exceptions Control mode page (see 6.5.8)), then the device server shall report the degraded non-volatile cache as specified in the Informational Exceptions Control mode page with an additional sense code set to WARNING - DEGRADED POWER TO NON-VOLATILE CACHE; or
- b) if the reporting of informational exceptions control warnings is disabled (i.e., the EWASC bit is set to zero in the Informational Exceptions Control mode page), then the device server shall establish a unit attention condition (see SAM-6) for the SCSI initiator port associated with every I_T nexus with the additional sense code set to WARNING - DEGRADED POWER TO NON-VOLATILE CACHE.

Non-volatile caches may become volatile (e.g., battery voltage becomes too low to sustain cache contents when power is lost). If non-volatile caches become volatile, then logical block data transferred for read commands or write commands in which the force unit access (FUA) bit in the CDB is set to one may bypass the cache.

If a non-volatile cache becomes volatile, then the device server shall set the REMAINING NON-VOLATILE TIME field to zero in the Non-volatile Cache log page (see 6.4.7).

If non-volatile cache becomes volatile and the Extended INQUIRY Data VPD page (see SPC-6) indicates that the device server contains non-volatile cache (i.e., the NV_SUP bit is set to one), then:

- a) if the reporting of informational exceptions control warnings is enabled (i.e., the EWASC bit is set to one in the Informational Exceptions Control mode page (see 6.5.8)), then the device server shall report the change in the cache as specified in the Informational Exceptions Control mode page with the additional sense code set to WARNING - NON-VOLATILE CACHE NOW VOLATILE; or
- b) if the reporting of informational exceptions control warnings is disabled (i.e., the EWASC bit is set to zero in the Informational Exceptions Control mode page), then the device server shall establish a unit attention condition (see SAM-6) for the SCSI initiator port associated with every I_T nexus with the additional sense code set to WARNING - NON-VOLATILE CACHE NOW VOLATILE.

...

modify SBC-5r84.18.2 as follows:**4.15.11 Processing pseudo unrecovered errors**

If a pseudo unrecovered error with correction disabled is encountered on a logical block (e.g., by a command, a background scan (see 4.23.1), or a background self-test (see SPC-6)), then the device server shall:

- a) perform no error recovery on the affected logical blocks, including any read error recovery enabled by the Read-Write Error Recovery mode page (see 6.5.10) or the Verify Error Recovery mode page (see 6.5.11);
- b) perform no automatic read reassignment or automatic write reassignment for the affected logical blocks, regardless of the settings of the AWRE bit and the ARRE bit in the Read-Write Error Recovery mode page;
- c) not consider errors on the affected logical blocks to be informational exception conditions as defined in the Informational Exceptions Control mode page (see 6.5.8);
- d) not log errors on the affected logical blocks in any log page that contain error counters (see SPC-6); and
- e) in any information returned for the error (e.g., in sense data or in the Background Scan Results log page (see 6.4.2)), set the sense key to MEDIUM ERROR and either:
 - A) should set the additional sense code to READ ERROR – LBA MARKED BAD BY APPLICATION CLIENT; or
 - B) may set the additional sense code to UNRECOVERABLE READ ERROR.

...

modify SBC-5r8 6.5.8 as follows:**6.5.8 Informational Exceptions Control mode page**

The Informational Exceptions Control mode page (see table 246) defines the methods used by the device server to control the **processing detection** and reporting of informational exception conditions. Informational exception conditions are defined as any event that the device server reports or logs as failure predictions (i.e., with the ADDITIONAL SENSE CODE field set to 5Dh (e.g., FAILURE PREDICTION THRESHOLD EXCEEDED)) or warnings (i.e., with the ADDITIONAL SENSE CODE field set to 0Bh (e.g., WARNING)).

Informational exception conditions may occur while a logical unit is processing:

- a) a background self-test (see SPC-6);
- b) device specific background functions (see SPC-6);
- c) a command; or
- d) other device specific events.

An informational exception condition may occur at any time (e.g., the condition may be asynchronous to any commands issued by an application client).

The mode page policy for this mode page shall be shared or per I_T nexus (see SPC-6).

Storage devices that support SMART (Self-Monitoring Analysis and Reporting Technology) for predictive failure software should ~~use~~detect and report informational exception conditions.

Table 246 — Informational Exceptions Control mode page

Byte	Bit	7	6	5	4	3	2	1	0
0		PS	SPF (0b)	PAGE CODE (1Ch)					
1		PAGE LENGTH (0Ah)							
2		PERF	Reserved	EBF	EWASC	DEXCPT	TEST	EBACKERR	LOGERR
3		Reserved				MRIE			
4		(MSB) _____ INTERVAL TIMER _____ (LSB)							
...									
7									
8		(MSB) _____ REPORT COUNT _____ (LSB)							
...									
11									

The PS bit, the SPF bit, the PAGE CODE field, and the PAGE LENGTH field are described in SPC-6.

The SPF bit, the PAGE CODE field, and the PAGE LENGTH field shall be set to the values shown in table 246 for the Informational Exceptions Control mode page.

If the performance (PERF) bit is set to zero, then the device server may ~~process~~detect informational exception conditions that cause delays in processing other operations (e.g., processing a command). If the PERF bit is set to one, then the device server shall not ~~process~~detect informational exception conditions that cause delays in processing other operations. A PERF bit set to one may cause the device server to disable some or all of the ~~processing~~detecting of informational exception conditions, thereby limiting the reporting of informational exception conditions.

If device specific background functions (see SPC-6) are implemented by the logical unit, and the enable background function (EBF) bit is set to one, then the device server shall enable device specific background functions. If the EBF bit is set to zero, then the device server shall disable device specific background functions. Background functions with separate enable control bits (e.g., the background medium scan (see 4.23)) are not controlled by the EBF bit.

The enable warning (EWASC) bit specifies if the device server enables reporting of warnings (see table 247).

The disable exception control (DEXCPT) bit specifies if the device server disables reporting of failure predictions (see table 247).

The TEST bit specifies if the device server creates a test device failure prediction (see table 247).

If an informational exception condition occurs that is not the result of the logical unit processing a background self-test (see SPC-6) or device specific background function (see SPC-6), then the device server:

- shall use the definitions for the combination of the values in the EWASC bit, the DEXCPT bit, and the TEST bit shown in table 247 for ~~processing~~reporting informational exception conditions if the MRIE field is set from 2h to 6h;
- may use the definitions for the combination of the values in the EWASC bit, the DEXCPT bit, and the TEST bit shown table 247 for ~~processing~~reporting informational exception conditions if the MRIE field is set from Ch to Fh; and

- c) shall ignore the EWASC bit, the DEXCPT bit, and the TEST bit if the MRIE field is set to any value other than 2h to 6h or Ch to Fh.

Table 247 — Definitions for the combinations of values in EWASC, DEXCPT, and TEST

EWASC	DEXCPT	TEST	Description
0	0	0	The device server shall process <u>report</u> informational exception conditions as follows: a) failure prediction processing <u>reporting</u> shall be enabled ^a ; and b) warning processing <u>reporting</u> shall be disabled.
1	0	0	The device server shall process <u>report</u> informational exception conditions as follows: a) failure prediction processing <u>reporting</u> shall be enabled ^a ; and b) warning processing <u>reporting</u> shall be enabled ^a .
0	1	0	The device server shall process <u>report</u> informational exception conditions as follows: a) failure prediction processing <u>reporting</u> shall be disabled; and b) warning processing <u>reporting</u> shall be disabled.
1	1	0	The device server shall process <u>report</u> informational exception conditions as follows: a) failure prediction processing <u>reporting</u> shall be disabled; and b) warning processing <u>reporting</u> shall be enabled ^a .
0	0	1	The device server shall set the additional sense code to FAILURE PREDICTION THRESHOLD EXCEEDED (FALSE) ^a .
1	0	1	
0	1	1	The device server shall terminate the MODE SELECT command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.
1	1	1	
^a If applicable based on the value in the MRIE field (e.g., 2h to 6h), then the values in the LOGERR bit, the INTERVAL TIMER field, and the REPORT COUNT field determine how the informational exception condition is processed <u>reported</u> .			

If an informational exception condition occurs while the logical unit is processing a background self-test (see SPC-6) or background function (see SPC-6), then the enable background error (EBACKERR) bit determines how the device server ~~processes~~reports the informational exception as defined in the following:

- a) if the EBACKERR bit is set to zero, then the device server shall disable reporting of informational exception conditions that occur during the processing of background self-tests and background functions;
 - b) if the EBACKERR bit is set to one, then, for informational exception conditions that occur during the processing of background self-tests and background functions, the device server shall:
 - A) enable reporting of the informational exception conditions;
 - B) use the method for reporting the informational exception conditions as determined by contents of the MRIE field; and
 - C) report the informational exception conditions as soon as the method specified in the MRIE field occurs (i.e., the INTERVAL TIMER field and REPORT COUNT field do not apply for background self-test errors and errors that occur during background functions);
- and
- c) logging by the device server of informational exception conditions is determined by the value in the LOGERR bit.

A LOGERR bit set to zero specifies that the device server may log any informational exception conditions in the Informational Exceptions log page (see SPC-6). A LOGERR bit set to one specifies that the device server shall log informational exception conditions in the Informational Exceptions log page.

The method of reporting informational exceptions (MRIE) field (see table 248) specifies the method that shall be used by the device server to report:

- a) informational exception conditions if the specified code value is supported by the device server; and
- b) background self-test errors and device specific background function errors with the ADDITIONAL SENSE CODE field set to 0Bh or 5Dh if the EBACKERR bit is set to one and the specified code value is supported by the device server.

A device server that supports the Informational Exceptions Control mode page shall support at least one code value other than zero in the MRIE field.

The priority of reporting multiple informational exceptions is vendor specific.

Table 248 — Method of reporting informational exceptions (MRIE) field (part 1 of 2)

Code	Description
0h	No reporting of informational exception condition: The device server shall not report informational exception conditions.
1h	Obsolete
2h	Establish unit attention condition: The device server shall report informational exception conditions by establishing a unit attention condition (see SAM-6) for the SCSI initiator port associated with every I_T nexus, with the additional sense code set to indicate the cause of the informational exception condition. ^a
3h	Conditionally generate recovered error: The device server shall report informational exception conditions, if the reporting of recovered errors is allowed ^b , by modifying the completion of the next command processed without encountering any errors, regardless of the I_T nexus on which the command was received. The modification shall be to terminate the command with CHECK CONDITION status with the sense key set to RECOVERED ERROR and the additional sense code set to indicate the cause of the informational exception condition.
4h	Unconditionally generate recovered error: The device server shall report informational exception conditions, regardless of whether the reporting of recovered errors is allowed ^b , by modifying the completion of the next command processed without encountering any errors, regardless of the I_T nexus on which the command was received. The modification shall be to terminate the command with CHECK CONDITION status with the sense key set to RECOVERED ERROR and the additional sense code set to indicate the cause of the informational exception condition.
5h	Generate no sense: The device server shall report informational exception conditions by modifying the completion of the next command processed without encountering any errors, regardless of the I_T nexus on which the command was received. The modification shall be to terminate the command with CHECK CONDITION status with the sense key set to NO SENSE and the additional sense code set to indicate the cause of the informational exception condition.
^a The device server terminates the command to report the unit attention condition for the informational exception condition (i.e., the device server does not process the command except to report the unit attention condition) (see SAM-6). ^b This is controlled by the PER bit (see 6.5.10) or the PER bit (see 6.5.11).	

Table 248 — Method of reporting informational exceptions (MRIE) field (part 2 of 2)

Code	Description
6h	Only report informational exception condition on request: The device server shall provide pollable sense data (see SPC-6) with the sense key set to NO SENSE and the additional sense code set to indicate the cause of the informational exception condition. To find out about informational exception conditions, the application client polls the device server by issuing a REQUEST SENSE command.
7h to Bh	Reserved
Ch to Fh	Vendor specific
^a The device server terminates the command to report the unit attention condition for the informational exception condition (i.e., the device server does not process the command except to report the unit attention condition) (see SAM-6). ^b This is controlled by the PER bit (see 6.5.10) or the PER bit (see 6.5.11).	

The INTERVAL TIMER field specifies the period in 100 millisecond increments that the device server shall use for reporting that an informational exception condition has occurred (see table 249). After an informational exception condition has been reported, the interval timer shall be started. An INTERVAL TIMER field set to zero or FFFF_FFFFh specifies that the period for reporting an informational exception condition is vendor specific.

The REPORT COUNT field specifies the maximum number of times the device server may report an informational exception condition to the application client. A REPORT COUNT field set to zero specifies that there is no limit on the number of times the device server may report an informational exception condition.

The device server shall use the values in the INTERVAL TIMER field and the REPORT COUNT field based on the value in the MRIE field as shown in table 249.

Table 249 — Use of the INTERVAL TIMER field and the REPORT COUNT field based on the MRIE field

MRIE ^a	Description
2h to 6h	<p>If reporting of an informational exception condition is enabled (see table 248), then the device server shall:</p> <ol style="list-style-type: none"> 1) report an informational exception condition when the condition is first detected; and 2) if the value in the REPORT COUNT field is not equal to one, then: <ol style="list-style-type: none"> 1) if the INTERVAL TIMER field is not set to zero or FFFF_FFFFh, then wait the time specified in the INTERVAL TIMER field, and, if that informational exception condition still exists, report the informational exception again; and 2) while the informational exception condition exists, continue to report the informational exception condition after waiting the time specified in the INTERVAL TIMER field until the condition has been reported the number of times specified by the REPORT COUNT field.
Ch to Fh	The device server may use or may ignore the values in the INTERVAL TIMER field and the REPORT COUNT field to report the informational exception condition based on the device specific implementation.
^a For values in the MRIE field (see table 248) not shown in this table, the INTERVAL TIMER field and the REPORT COUNT field shall be ignored.	

Maintaining the interval timer and the report counter across power cycles, hard resets, logical unit resets, and I_T nexus losses by the device server is vendor specific.

modify SBC-5r8 Table A.6 in section A.3.1 as follows:

Table A.6 lists the SBC Base 2016 mandatory block descriptors and mode pages.

Table A.6 — Block descriptor and mode pages mandatory for the SBC Base 2010 feature set

Mode page	Additional requirements reference	Reference
Block descriptors		
Mode parameter block descriptor	A.3.4.1	6.5.2
Mode pages		
Caching	A.3.4.2	6.5.6
Control	A.3.4.3	SPC-6
Control Extension	n/a	SPC-6
Informational Exceptions Control	A.3.4.4	6.5.8
Read-Write Error Recovery	A.3.4.5	6.5.10