To: T10 Technical Committee
From: Rob Elliott, HP (elliott@hp.com)
Date: 4 July 2006
Subject: 06-221r1 SPC-4 Peripheral device informational identifier

<u>**Revision history**</u>
Revision 0 (3 May 2006) First revision
Revision 1 (4 July 2006) Incorporated comments from May 2006 CAP WG - changed to extend the
REPORT/SET DEVICE IDENTIFIER commands instead of expand the Device Identification VPD page 83h.

<u>**Related documents**</u>
spc4r04 - SCSI Primary Commands - 4 (SPC-4) revision 4
scc2r04 - SCSI Controller Commands - 2 (SCC-2) revision 4
06-278 - SPC-4 Fix persistence of SET DEVICE IDENTIFIER (Paul Entzel, Quantum)

<u>**Overview**</u>
Worldwide unique logical unit names in 64-bit and 128-bit binary NAA and EUI-64-based formats or (up to)
256 byte UTF-8 SCSI name string formats are great for applications, but are unwieldly for users.

This proposal defines a new informational identifier in UTF-8 string format that can be used to provide a
descriptive string about the logical unit. The intent is that it be used for display purposes in management
applications. UTF-8 rather than ASCII allows characters from other languages to be used if appropriate.

Examples:

   a) "Database log storage"
   b) "Database table storage"
   c) "Server 34 boot drive"

Unlike the NAA, EUI-64, and SCSI name string identifiers in the Device Identification VPD page 83h, there is
no requirement that this new identifier be worldwide unique (or even unique within the target device).

The REPORT DEVICE IDENTIFIER and SET DEVICE IDENTIFIER commands in SPC-4 allow software to
maintain an arbitrary identifier. However, the format is not required to be a string, and it is being used for a
different purpose than that described here by at least one operating system.

For logical units with an SCC device type or an SCCS bit set to one, the commands are interpreted as the
SCC-2 REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER and SET PERIPHERAL
DEVICE/COMPONENT DEVICE IDENTIFIER commands. These have two additional fields: a LUN field in
bytes 4-5 and a PORCLU or IDPORC bit in byte 10 bit 1. The identifier is supposed to be ASCII. It returns the
"position of theperipheral device or component device within the target."

Proposal:

   a) The REPORT DEVICE IDENTIFIER command is renamed to REPORT ADDITIONAL IDENTIFIERS,
       and the SET DEVICE IDENTIFIER command is renamed to SET ADDITIONAL IDENTIFIERS
   b) The old identifier is renamed the "peripheral device identifier"
   c) The new identifier is called the "peripheral device informational identifier"
   d) The new identifier is required to be in UTF-8 format
   e) The new informational identifier may be set with the SET command, and may be set with a
       vendor-specific mechanism.
   f) The PORCLU/IDPORC bit (byte 10 bit 1), previously just marked Restricted, is consumed by a two-bit
       IDENTIFIER TYPE field. This ensures that it must not be set to one. If the new identifier were indicated by
       a new bit elsewhere in the CDB, then there would be no requirement to verify that the PORCLU/IDPORC
       bit is set to zero (since reserved bit checking is optional).
   g) The LUN field is still restricted; this means for a device complying with SPC-4, it is just treated as
       reserved.

<u>**Suggested changes**</u>

3.1.31 device type: The type of peripheral device (i.e., device model) implemented by the device server and indicated by the contents of the PERIPHERAL DEVICE TYPE field in the standard INQUIRY data (see 6.4.2).

3.1.xx peripheral device: An object that that connects to a logical unit.

3.1.xx peripheral device identifier: An identifier for a peripheral device accessed via the REPORT ADDITIONAL IDENTIFIERS command (see 6.20) and SET ADDITIONAL IDENTIFIERS command (see 6.31).

3.1.xx peripheral device informational identifier: An informational identifier for a peripheral device accessed via the REPORT ADDITIONAL IDENTIFIERS command (see 6.20) and SET ADDITIONAL IDENTIFIERS command (see 6.31).

3.1.xx peripheral device type: Synonym for device type (see 3.1.31).

3.3.10 restricted: A keyword referring to bits, bytes, words, and fields that are set aside for use in other SCSI standards. A restricted bit, byte, word, or field shall be treated as a reserved bit, byte, word or field for the purposes of the requirements defined in this standard.

6.20 REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command

The REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command (see table 146) requests that the device server send ~~device identification information~~a peripheral device identifier or peripheral device informational identifier to the application client. ~~As defined in the SCC-2 standard, t~~The REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command is an extension to the REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER service action of the MAINTENANCE IN command defined in SCC-2. Additional MAINTENANCE IN and MAINTENANCE OUT service actions are defined in SCC-2 and in this standard.

The MAINTENANCE IN service actions defined only in SCC-2 shall apply only to ~~SCSI devices~~logical units that return a device type of 0Ch (i.e., storage array controller device) or the SCCS bit set to one in their standard INQUIRY data (see 6.4.2). When a ~~SCSI device~~logical unit returns a device type of 0Ch or the SCCS bit set to one in its standard INQUIRY data, the implementation requirements for the SCC-2 MAINTENANCE IN service actions shall be as specified in SCC-2. Otherwise the MAINTENANCE IN service action definitions and implementation requirements stated in this standard shall apply.

Table 146 — REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	Reserved			SERVICE ACTION (05h)				
2	Reserved							
3								
4	Restricted (see SCC-2)							
5								
6	(MSB)	ALLOCATION LENGTH						
9								(LSB)
10	Reserved						~~Restricted~~ IDENTIFIER TYPE	~~Reserved~~
11	CONTROL							

~~SCC-2 defines specific usages for bytes 4 and 5, and bit 1 in byte 10, however these fields are reserved for the REPORT DEVICE IDENTIFIER command defined by this standard.~~

The ALLOCATION LENGTH field is defined in 4.3.4.6.

The IDENTIFIER TYPE field and the parameter data IDENTIFIER field is defined in table 147.

Table 147 — IDENTIFIER TYPE field and parameter data IDENTIFIER field

Code	Name	Description of parameter data IDENTIFIER field
00b	Peripheral device identifier	The parameter data IDENTIFIER field contains a value set by the application client with the SET ADDITIONAL IDENTIFIERS command (see 6.31) with the IDENTIFIER TYPE field set to 00b.
01b	Peripheral device informational identifier	The parameter data IDENTIFIER field contains a value set by the application client with the SET ADDITIONAL IDENTIFIERS command (see 6.31) with the IDENTIFIER TYPE field set to 01b.
10b	Restricted (see SCC-2)	
11b		

The REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS parameter data ~~(see~~is defined in table 148~~)~~ ~~contains a four-byte field that contains the length in bytes of the parameter data and the logical unit's identifier~~.

Table 148 — REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS parameter data

Byte\Bit	7	6	5	4	3	2	1	0
0	(MSB)			IDENTIFIER LENGTH (n-3)				
3								(LSB)
4				IDENTIFIER				
n								

The IDENTIFIER LENGTH field indicates the length in bytes of the IDENTIFIER field. The relationship between the IDENTIFIER LENGTH field and the CDB ALLOCATION LENGTH field is defined in 4.3.4.6. ~~The identifier length shall initially equal zero, and shall be changed only by a successful SET DEVICE IDENTIFIER command.~~

The IDENTIFIER field is defined in table 147.~~shall contain a vendor specific value. The value reported shall be the last value written by a successful SET DEVICE IDENTIFIER command. The value of the identifier shall be changed only by a SET DEVICE IDENTIFIER command. The identifier value shall persist through logical unit resets, I_T nexus losses, media format operations, and media replacement.~~

The logical unit shall return the same identifier to all application clients (i.e., through all I_T nexuses).

Processing a REPORT ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS may require the enabling of a nonvolatile memory within the logical unit. If the nonvolatile memory is not ready, the command shall be terminated with CHECK CONDITION status, rather than wait for the nonvolatile memory to become ready. The sense key shall be set to NOT READY and the additional sense code shall be set as described in table 193 (see 6.35). This information should allow the application client to determine the action required to cause the device server to become ready.

6.31 SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command

The SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command (see table 181) requests that the ~~device identifier information~~ peripheral device identifier or peripheral device informational identifier in the logical unit be set to the value received in the SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS parameter list. ~~As defined in the SCC-2 standard, t~~The SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command is an extension to the SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER service action of the MAINTENANCE OUT command defined in SCC-2. Additional MAINTENANCE IN and MAINTENANCE OUT service actions are defined in SCC-2 and in this standard.

The MAINTENANCE OUT service actions defined only in SCC-2 shall apply only to ~~SCSI devices~~logical units that return a device type of 0Ch (i.e., storage array controller device) or the SCCS bit set to one in their standard INQUIRY data (see 6.4.2). When a ~~SCSI devices~~logical unit returns a device type of 0Ch or the SCCS bit set to one in its standard INQUIRY data, the implementation requirements for the SCC-2 MAINTENANCE OUT service actions shall be as specified in SCC-2. Otherwise the MAINTENANCE OUT service action definitions and implementation requirements stated in this standard shall apply.

On successful completion of a SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command that changes ~~the device~~an identifier saved by the logical unit, the device server shall establish a unit attention condition (see SAM-3) for the initiator port associated with every I_T nexus except the I_T nexus on which the SET ~~IDENTIFIER~~ADDITIONAL IDENTIFIERS command was received, with the additional sense code set to DEVICE IDENTIFIER CHANGED.

Table 181 — SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	Reserved			SERVICE ACTION (06h)				
2	Reserved							
3								
4	Restricted (see SCC-2)							
5								
6	(MSB)	PARAMETER LIST LENGTH						
9								(LSB)
10	Reserved						~~Restricted~~ IDENTIFIER TYPE	~~Reserved~~
11	CONTROL							

~~SCC-2 defines specific usages for bytes 4 and 5, and bit 1 in byte 10, however these fields are reserved for the SET DEVICE IDENTIFIER command defined by this standard.~~

The PARAMETER LIST LENGTH field specifies the length in bytes of the identifier that shall be transferred from the application client to the device server. ~~The maximum value for this field shall be 512 bytes. A parameter list length of zero specifies that no data shall be transferred, and that subsequent REPORT DEVICE IDENTIFIER commands shall return an Identifier length of zero. Logical units that implement this command shall be capable of accepting a parameter list length of 64 bytes or less. If the parameter list length exceeds 64 bytes and the logical unit is not capable of storing the requested number of bytes, 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 IDENTIFIER TYPE field and the parameter list IDENTIFIER field are defined in table 147.

Table 182 — IDENTIFIER TYPE field and parameter list IDENTIFIER field

Code	Name	Definition of the parameter data IDENTIFIER field
00b	Peripheral device identifier	The IDENTIFIER field contains a value selected by the application client (e.g., an operating system volume label). The device server shall support a PARAMETER LIST LENGTH field set to 0 through 64 and may support a PARAMETER LIST LENGTH field set to 65 through 512. If the PARAMETER LIST LENGTH field is set to greater than 512 or the maximum value supported by the device server, the device server shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB. The peripheral device identifier shall persist through power cycles, hard resets, logical unit resets, I_T nexus losses, media format operations, and media replacement. The logical unit may allow a system administrator to change the peripheral device identifier by means outside the scope of this standard.
01b	Peripheral device informational identifier	The IDENTIFIER field shall contain a null-terminated (see 4.4.2) UTF-8 format string providing an informational description (e.g., a descriptive string entered by a system administrator) describing the peripheral device (i.e., the logical unit). The device server shall support a PARAMETER LIST LENGTH field set to 0 through 256. If the PARAMETER LIST LENGTH field is set to greater than 256, the device server shall terminate the command 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 device server detects the format of the IDENTIFIER field is incorrect, it shall terminate the 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. The peripheral device informational identifier shall persist through power cycles, hard resets, logical unit resets, I_T nexus losses, media format operations, and media replacement. The logical unit may allow a system administrator to change the peripheral device informational identifier by means outside the scope of this standard.
10b 11b	Restricted (see SCC-2)	

The SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS parameter list ~~(see~~is defined in table 183~~)~~ ~~contains the identifier to be set by the addressed logical unit~~.

Table 183 — SET ~~DEVICE IDENTIFIER~~ADDITIONAL IDENTIFIERS parameter list

Byte\Bit	7	6	5	4	3	2	1	0
0				IDENTIFIER				
n								

The IDENTIFIER field is defined in table 182. ~~is a value selected by the application client using mechanisms outside the scope of this standard to be returned in subsequent REPORT DEVICE IDENTIFIER commands.~~

SCC-2 excerpts (for reference)

6.3.1.7 REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER service action

The REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER service action (see table 35) requests the position of the selected logical unit within the target be sent to the application client.

Table 35 — REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	Reserved			SERVICE ACTION (05h)				
2	Reserved							
3								
4	LUN							
5								
6	(MSB)			ALLOCATION LENGTH				
9								(LSB)
10	Reserved						PORCLU	Reserved
11	CONTROL							

The LUN field contains the logical unit number of the peripheral device or component device that the target shall report per table 36. If the requested logical unit has not been added to the target the command shall be terminated with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST, and the additional sense code set to LOGICAL UNIT NOT SUPPORTED.

A report physical or component logical unit bit (PORCLU) of zero indicates the LUN field shall contain the address of a peripheral device. A PORCLU bit of one indicates the LUN field shall contain the address of a component device.

The REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER parameter list (see table 36) contains a four-byte field that contains the length in bytes of the parameter list and the position of the selected logical unit within the target.

Table 36 — REPORT PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER parameter data

Byte\Bit	7	6	5	4	3	2	1	0
0	(MSB)			IDENTIFIER LENGTH (n-3)				
3								(LSB)
4	IDENTIFIER							
n								

The IDENTIFIER LENGTH field specifies the length in bytes of the IDENTIFIER field.

The IDENTIFIER field shall be an ASCII value that indicates the position of the peripheral device or component device within the target. The ASCII value within the IDENTIFIER field is vendor specific.

6.4.1.8 SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER service action

The SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER service action (see table 66) requests the target set the identifier in the addressed logical unit with the value received in the SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER parameter list. On successful completion of this service action a unit attention shall be generated for all initiators except the one that issued the service action. When

reporting the unit attention condition the additional sense code shall be set to DEVICE IDENTIFIER CHANGED.

Table 66 — SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	Reserved			SERVICE ACTION (06h)				
2	Reserved							
3								
4	LUN							
5								
6	(MSB)			PARAMETER LIST LENGTH				
9								(LSB)
10	Reserved						IDPORC	Reserved
11	CONTROL							

The LUN field contains the logical unit number of the peripheral device or component device that shall be received by the IDENTIFIER field contents in the SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER parameter list. If the requested logical unit has not been added to the target the command shall be terminated with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST, and the additional sense code set to LOGICAL UNIT NOT SUPPORTED.

A set identification peripheral device or component device bit (IDPORC) of zero indicates the LUN field shall contain the address of a peripheral device. A IDPORC bit of one indicates the LUN field shall contain the address of a component device.

The SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER parameter list (see table 67)contains the location identifier of the addressed logical unit.

Table 67 — SET PERIPHERAL DEVICE/COMPONENT DEVICE IDENTIFIER parameter list

Byte\Bit	7	6	5	4	3	2	1	0
0	IDENTIFIER							
n								

The IDENTIFIER field shall be an ASCII value that indicates the position of the peripheral device or component device within the target. The ASCII value within the IDENTIFIER field is vendor specific.