Document Number: DSP0842

Date: 2009-06-04

Version: 1.0.0

OS Status Profile SM CLP Command Mapping Specification

Document Type: Specification

Document Status: DMTF Standard

Document Language: E

CONTENTS

Tables

58

Foreword

The *OS Status Profile SM CLP Command Mapping Specification* (DSP0842) was prepared by the Server Management Working Group.

Conventions

The pseudo-code conventions utilized in this document are the Recipe Conventions as defined in SNIA SMI-S 1.1.0, section 7.6.

Acknowledgements

- Aaron Merkin – IBM
- Jon Hass – Dell
- Khachatur Papanyan – Dell
- Jeff Hilland – HP
- Joel Clark – Intel

Introduction

This document defines the SM CLP mapping for CIM elements described in the *OS Status Profile*. The information in this specification, combined with the *SM CLP-to-CIM Common Mapping Specification 1.0* (DSP0216), is intended to be sufficient to implement SM CLP commands relevant to the classes, properties, and methods described in the *OS Status Profile* using CIM operations.

The target audience for this specification is implementers of the SM CLP support for the *OS Status Profile*.

OS Status Profile SM CLP Command Mapping Specification

1 Scope

This specification contains the requirements for an implementation of the SM CLP to provide access to, and implement the behaviors of, the *OS Status Profile*.

2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

2.1 Approved References

DMTF DSP1029, *OS Status Profile 1.0*,
http://www.dmtf.org/standards/published_documents/DSP1029_1.0.pdf

DMTF DSP0216, *SM CLP-to-CIM Common Mapping Specification 1.0*,
http://www.dmtf.org/standards/published_documents/DSP0216_1.0.pdf

SNIA, *Storage Management Initiative Specification (SMI-S) 1.1.0*,
http://www.snia.org/tech_activities/standards/curr_standards/smi

2.2 Other References

ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype

3 Terms and Definitions

For the purposes of this document, the following terms and definitions apply.

3.1
can
used for statements of possibility and capability, whether material, physical, or causal

3.2
cannot
used for statements of possibility and capability, whether material, physical or causal

3.3
conditional
indicates requirements to be followed strictly in order to conform to the document when the specified conditions are met

3.4
mandatory
indicates requirements to be followed strictly in order to conform to the document and from which no deviation is permitted

3.5
may
indicates a course of action permissible within the limits of the document

3.6
need not
indicates a course of action permissible within the limits of the document

3.7
optional
indicates a course of action permissible within the limits of the document

3.8
shall
indicates requirements to be followed strictly in order to conform to the document and from which no deviation is permitted

3.9
shall not
indicates requirements to be followed strictly in order to conform to the document and from which no deviation is permitted

3.10
should
indicates that among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required

3.11
should not
indicates that a certain possibility or course of action is deprecated but not prohibited

4 Symbols and Abbreviated Terms

The following symbols and abbreviations are used in this document.

4.1
CIM
Common Information Model

4.2
CLP
Command Line Protocol

4.3
DMTF
Distributed Management Task Force

148 **4.4**
149 **SM**
150 Server Management

151 **4.5**
152 **SMI-S**
153 Storage Management Initiative Specification

154 **4.6**
155 **SNIA**
156 Storage Networking Industry Association

157 5 Recipes

158 The following is a list of the common recipes used by the mappings in this specification. For a definition of
159 each recipe, see the *SM CLP-to-CIM Common Mapping Specification 1.0* (DSP0216).

160 • smShowInstance()
161 • smShowInstances()
162 • smSetInstance()
163 • smShowAssociationInstances()
164 • smShowAssociationInstance()

165 6 Mappings

166 The following sections detail the mapping of CLP verbs to CIM Operations for each CIM class defined in
167 the *OS Status Profile*. Requirements specified here related to support for a CLP verb for a particular class
168 are solely within the context of this profile.

169 6.1 CIM_OperatingSystem

170 The `cd` and `help` verbs shall be supported as described in DSP0216.

171 Table 1 lists each SM CLP verb, the required level of support for the verb in conjunction with instances of
172 the target class, and, when appropriate, a cross-reference to the section detailing the mapping for the
173 verb and target. Table 1 is for informational purposes only; in case of a conflict between Table 1 and
174 requirements detailed in the following sections, the text detailed in the following sections supersedes the
175 information in Table 1.

176 **Table 1 – Command Verb Requirements for CIM_OperatingSystem**

Command Verb	Requirement	Comments
create	Not supported	
delete	Not supported	
dump	Not supported	
load	Not supported	
reset	May	See 6.1.2.
set	May	See 6.1.3.

Command Verb	Requirement	Comments
show	Shall	See 6.1.4.
start	Not supported	
stop	May	See 6.1.5.

No mapping is defined for the following verbs for the specified target: `create`, `delete`, `dump`, and `load`.

6.1.1 Ordering of Results

When results are returned for multiple instances of CIM_OperatingSystem, implementations shall utilize the following algorithm to produce the natural (that is, default) ordering:

- Results for CIM_OperatingSystem are unordered; therefore, no algorithm is defined.

6.1.2 Reset

This section describes how to implement the `reset` verb when applied to an instance of CIM_OperatingSystem. Implementations may support the use of the `reset` verb with CIM_OperatingSystem.

6.1.2.1 General Usage of Reset for a Single Property

6.1.2.1.1 Command Form

```
reset <CIM_OperatingSystem single instance>
```

6.1.2.1.2 CIM Requirements

```
uint16 EnabledState;
uint16 RequestedState;
uint32 CIM_OperatingSystem.RequestStateChange (
   [IN] uint16 RequestedState,
   [OUT] REF CIM_ConcreteJob Job,
   [IN] datetime TimeoutPeriod );
```

6.1.2.1.3 Behavior Requirements

6.1.2.1.3.1 Preconditions

$instance represents the targeted instance of CIM_OperatingSystem.

```
$instance=<CIM_OperatingSystem single instance>
```

6.1.2.1.3.2 Pseudo Code

```
&smResetRSC ( $instance.getObjectPath() );
&smEnd;
```

6.1.3 Set

This section describes how to implement the `set` verb when it is applied to an instance of CIM_OperatingSystem. Implementations may support the use of the `set` verb with CIM_OperatingSystem.

6.1.3.1 Set requestedstate

This command form is for the `set` verb applied to a single instance of CIM_OperatingSystem.

6.1.3.1.1 Command Form

```
set <CIM_OperatingSystem single instance> RequestedState=<requestedstate>
```

6.1.3.1.2 CIM Requirements

```
uint16 EnabledState;
uint16 RequestedState;
uint32 EnabledLogicalElement.RequestStateChange (
   [IN] uint16 RequestedState,
   [OUT] REF CIM_ConcreteJob Job,
   [IN] datetime TimeoutPeriod );
```

6.1.3.1.3 Behavior Requirements

6.1.3.1.3.1 Preconditions

```
$instance=<CIM_OperatingSystem single instance>
```

6.1.3.1.3.2 Pseudo Code

```
&smRequestStateChange ( $instance.getObjectPath(), <requestedstate> );
&smEnd;
```

6.1.3.2 General Usage of Set for a Single Property

This command form corresponds to the general usage of the `set` verb to modify a single property of a target instance. This is the most common case.

The requirement for supporting modification of a property using this command form shall be equivalent to the requirement for supporting modification of the property using the ModifyInstance operation as defined in the *OS Status Profile*.

6.1.3.2.1 Command Form

```
set <CIM_OperatingSystem single instance> <propertyname>=<propertyvalue>
```

6.1.3.2.2 CIM Requirements

See CIM_OperatingSystem in the “CIM Elements” section of the *OS Status Profile* for the list of modifiable properties.

6.1.3.2.3 Behavior Requirements

6.1.3.2.3.1 Preconditions

```
$instance=<CIM_OperatingSystem single instance>
```

6.1.3.2.3.2 Pseudo Code

```
#propertyNames[] = {<propertyname>};
#propertyValues[] = {<propertyvalue>};
&smSetInstance ( $instance, #propertyNames[], #propertyValues[] );
&smEnd;
```

6.1.3.3 General Usage of Set for Multiple Properties

This command form corresponds to the general usage of the `set` verb to modify multiple properties of a target instance where there is not an explicit relationship between the properties. This is the most common case.

The requirement for supporting modification of a property using this command form shall be equivalent to the requirement for supporting modification of the property using the ModifyInstance operation as defined in the *OS Status Profile*.

6.1.3.3.1 Command Form

```
set <CIM_OperatingSystem single instance> <propertyname1>=<propertyvalue1>
   <propertynamen>=<propertyvaluen>
```

6.1.3.3.2 CIM Requirements

See CIM_OperatingSystem in the “CIM Elements” section of the *OS Status Profile* for the list of supported properties.

6.1.3.3.3 Behavior Requirements

6.1.3.3.3.1 Preconditions

```
$instance=<CIM_OperatingSystem single instance>
```

6.1.3.3.3.2 Pseudo Code

```
#propertyNames[] = {<propertyname>};
for #i < n
   {
   #propertyNames[#i] = <propertname#i>
   #propertyValues[#i] = <propertyvalue#i>
   }
&smSetInstance ( $instance, #propertyNames[], #propertyValues[] );
&smEnd;
```

6.1.4 Show

This section describes how to implement the `show` verb when applied to an instance of CIM_OperatingSystem. Implementations shall support the use of the `show` verb with CIM_OperatingSystem.

The `show` verb is used to display information about the account.

6.1.4.1 Show a Single Instance

This command form is for the `show` verb applied to a single instance of CIM_OperatingSystem.

6.1.4.1.1 Command Form

```
show <CIM_OperatingSystem single instance>
```

6.1.4.1.2 CIM Requirements

See CIM_OperatingSystem in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.1.4.1.3 Behavior Requirements

6.1.4.1.3.1 Preconditions

#all is true if the “-all” option was specified with the command; otherwise, #all is false.

6.1.4.1.3.2 Pseudo Code

```
$instance=<CIM_OperatingSystem single instance>
#propertylist[] = NULL;
if (false == #all)
   {
   #propertylist[] = {//all mandatory non-key properties };
   }
&smShowInstance ( $instance.getObjectPath(), #propertylist[] );
&smEnd;
```

6.1.4.2 Show Multiple Instances

This command form is for the show verb applied to multiple instances of CIM_OperatingSystem. This command form corresponds to UfsT-based selection within a scoping system.

6.1.4.2.1 Command Form

```
show <CIM_OperatingSystem multiple instances>
```

6.1.4.2.2 CIM Requirements

See CIM_OperatingSystem in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.1.4.2.3 Behavior Requirements

6.1.4.2.3.1 Preconditions

$containerInstance contains the instance of CIM_ComputerSystem for which we are displaying scoped accounts (CIM_OperatingSystem instances). The *OS Status Profile* requires that the CIM_OperatingSystem instance be associated with its scoping system via an instance of the CIM_OperatingSystemOnSystem association.

#all is true if the “-all” option was specified with the command; otherwise, #all is false.

6.1.4.2.3.2 Pseudo Code

```
#propertylist[] = NULL;
if (false == #all)
   {
   #propertylist[] = {//all mandatory non-key properties };
   }
&smShowInstances ( “CIM_OperatingSystem”, “CIM_OperatingSystemOnSystem”,
   $containerInstance.getObjectPath(), #propertylist[] );
&smEnd;
```

6.1.5 Stop

This section describes how to implement the `stop` verb when applied to an instance of CIM_OperatingSystem. Implementations may support the use of the `stop` verb with CIM_OperatingSystem.

The `stop` verb is used to disable an account.

6.1.5.1 Stop a Single Instance

This command form is for the `stop` verb applied to a single instance of CIM_OperatingSystem.

6.1.5.1.1 Command Form

```
stop <CIM_OperatingSystem single instance>
```

6.1.5.1.2 CIM Requirements

```
uint16 EnabledState;
uint16 RequestedState;
//3 (Disabled)
uint32 EnabledLogicalElement.RequestStateChange (
   [IN] uint16 RequestedState = 3,
   [OUT] REF CIM_ConcreteJob Job,
   [IN] datetime TimeoutPeriod );
```

6.1.5.1.3 Behavior Requirements

6.1.5.1.3.1 Preconditions

```
$instance=<CIM_OperatingSystem single instance>
```

6.1.5.1.3.2 Pseudo Code

```
smStopRSC ( $instance.getObjectPath() );
&smEnd;
```

6.2 CIM_RunningOS

The `cd` and `help` verbs shall be supported as described in DSP0216.

Table 2 lists each SM CLP verb, the required level of support for the verb in conjunction with instances of the target class, and, when appropriate, a cross-reference to the section detailing the mapping for the verb and target. Table 2 is for informational purposes only; in case of a conflict between Table 2 and requirements detailed in the following sections, the text detailed in the following sections supersedes the information in Table 2.

Table 2 – Command Verb Requirements for CIM_RunningOS

Command Verb	Requirement	Comments
create	Not supported	
delete	Not supported	
dump	Not supported	
load	Not supported	
reset	Not supported	

Command Verb	Requirement	Comments
set	Not supported	
show	Shall	See 6.2.2.
start	Not supported	
stop	Not supported	

No mappings are defined for the following verbs for the specified target: `create`, `delete`, `dump`, `load`, `reset`, `set`, `start`, and `stop`.

6.2.1 Ordering of Results

When results are returned for multiple instances of CIM_RunningOS implementations shall utilize the following algorithm to produce the natural (that is, default) ordering:

- Results for CIM_RunningOS are unordered; therefore, no algorithm is defined.

6.2.2 Show

This section describes how to implement the `show` verb when applied to an instance of CIM_RunningOS. Implementations shall support the use of the `show` verb with CIM_RunningOS.

The `show` command is used to display information about the CIM_RunningOS instance or instances.

6.2.2.1 Show Multiple Instances

This command form is for the `show` verb applied to multiple instances. This command form corresponds to a `show` command issued against CIM_RunningOS where only one reference is specified and the reference is to an instance of CIM_ComputerSystem.

6.2.2.1.1 Command Form

```
show <CIM_RunningOS multiple instances>
```

6.2.2.1.2 CIM Requirements

See CIM_RunningOS in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.2.2.1.3 Behavior Requirements

6.2.2.1.3.1 Preconditions

$instance contains the instance of CIM_ComputerSystem which is referenced by CIM_RunningOS.

6.2.2.1.3.2 Pseudo Code

```
&smShowAssociationInstances ( “CIM_RunningOS”, $instance.getObjectPath() );
&smEnd;
```

6.2.2.2 Show a Single Instance – CIM_OperatingSystem Reference

This command form is for the `show` verb applied to a single instance. This command form corresponds to a `show` command issued against CIM_RunningOS where the reference specified is to an instance of CIM_OperatingSystem. An instance is referenced by exactly one instance of CIM_RunningOS. Therefore, a single instance will be returned.

6.2.2.2.1 Command Form

```
show <CIM_RunningOS single instance>
```

6.2.2.2.2 CIM Requirements

See CIM_RunningOS in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.2.2.2.3 Behavior Requirements

6.2.2.2.3.1 Preconditions

$instance contains the instance of CIM_OperatingSystem which is referenced by CIM_RunningOS.

6.2.2.2.3.2 Pseudo Code

```
&smShowAssociationInstances ( “CIM_RunningOS”, $instance.getObjectPath() );
&smEnd;
```

6.2.2.3 Show a Single Instance – Both References

This command form is for the `show` verb applied to a single instance. This command form corresponds to a `show` command issued against CIM_RunningOS where both references are specified and therefore the desired instance is unambiguously identified.

6.2.2.3.1 Command Form

```
show <CIM_RunningOS single instance>
```

6.2.2.3.2 CIM Requirements

See CIM_RunningOS in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.2.2.3.3 Behavior Requirements

6.2.2.3.3.1 Preconditions

$instanceA contains the instance of CIM_ComputerSystem which is referenced by CIM_RunningOS.

$instanceB contains the instance of CIM_OperatingSystem which is referenced by CIM_RunningOS.

6.2.2.3.3.2 Pseudo Code

```
&smShowAssociationInstance ( “CIM_RunningOS”, $instanceA.getObjectPath(),
   $instanceB.getObjectPath() );
&smEnd;
```

6.3 CIM_InstalledOS

The `cd` and `help` verbs shall be supported as described in DSP0216.

Table 3 lists each SM CLP verb, the required level of support for the verb in conjunction with instances of the target class, and, when appropriate, a cross-reference to the section detailing the mapping for the verb and target. Table 3 is for informational purposes only; in case of a conflict between Table 3 and requirements detailed in the following sections, the text detailed in the following sections supersedes the information in Table 3.

412 **Table 3 – Command Verb Requirements for CIM_InstalledOS**

Command Verb	Requirement	Comments
create	Not supported	
delete	Not supported	
dump	Not supported	
load	Not supported	
reset	Not supported	
set	Not supported	
show	Shall	See 6.2.2.
start	Not supported	
stop	Not supported	

413 No mappings are defined for the following verbs for the specified target: `create`, `delete`, `dump`, `load`,
414 `reset`, `set`, `start`, and `stop`.

415 ### 6.3.1 Ordering of Results

416 When results are returned for multiple instances of CIM_InstalledOS implementations shall utilize the
417 following algorithm to produce the natural (that is, default) ordering:

418 - Results for CIM_InstalledOS are unordered; therefore, no algorithm is defined.

419 ### 6.3.2 Show

420 This section describes how to implement the `show` verb when applied to an instance of CIM_InstalledOS.
421 Implementations shall support the use of the `show` verb with CIM_InstalledOS.

422 The `show` command is used to display information about the CIM_InstalledOS instance or instances.

423 #### 6.3.2.1 Show Multiple Instances

424 This command form is for the `show` verb applied to multiple instances. This command form corresponds
425 to a `show` command issued against CIM_InstalledOS where only one reference is specified and the
426 reference is to an instance of CIM_ComputerSystem.

427 ##### 6.3.2.1.1 Command Form

428 **`show <CIM_InstalledOS`** ***`multiple instances`*****`>`**

429 ##### 6.3.2.1.2 CIM Requirements

430 See CIM_InstalledOS in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory
431 properties.

432 ##### 6.3.2.1.3 Behavior Requirements

433 ###### 6.3.2.1.3.1 Preconditions

434 $instance contains the instance of CIM_ComputerSystem which is referenced by CIM_InstalledOS.

435 #### 6.3.2.1.3.2 Pseudo Code

436 ```
&smShowAssociationInstances (“CIM_InstalledOS”, $instance.getObjectPath());
437 &smEnd;
```

438 ### 6.3.2.2 Show a Single Instance – CIM_OperatingSystem Reference

439 This command form is for the `show` verb applied to a single instance. This command form corresponds to
440 a `show` command issued against CIM_InstalledOS where the reference specified is to an instance of
441 CIM_OperatingSystem. An instance is referenced by exactly one instance of CIM_InstalledOS. Therefore,
442 a single instance will be returned.

443 #### 6.3.2.2.1 Command Form

444 **`show <CIM_InstalledOS`** ***`single instance`*****`>`**

445 #### 6.3.2.2.2 CIM Requirements

446 See CIM_InstalledOS in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory
447 properties.

448 #### 6.3.2.2.3 Behavior Requirements

449 ##### 6.3.2.2.3.1 Preconditions

450 $instance contains the instance of CIM_OperatingSystem which is referenced by CIM_InstalledOS.

451 ##### 6.3.2.2.3.2 Pseudo Code

452 ```
&smShowAssociationInstances ( “CIM_InstalledOS”, $instance.getObjectPath() );
453 &smEnd;
```

454 ### 6.3.2.3 Show a Single Instance – Both References

455 This command form is for the `show` verb applied to a single instance. This command form corresponds to
456 a `show` command issued against CIM_InstalledOS where both references are specified and therefore the
457 desired instance is unambiguously identified.

458 #### 6.3.2.3.1 Command Form

459 **`show <CIM_InstalledOS`** ***`single instance`*****`>`**

460 #### 6.3.2.3.2 CIM Requirements

461 See CIM_InstalledOS in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory
462 properties.

463 #### 6.3.2.3.3 Behavior Requirements

464 ##### 6.3.2.3.3.1 Preconditions

465 $instanceA contains the instance of CIM_ComputerSystem which is referenced by CIM_InstalledOS.

466 $instanceB contains the instance of CIM_OperatingSystem which is referenced by CIM_InstalledOS.

467 ##### 6.3.2.3.3.2 Pseudo Code

468 ```
&smShowAssociationInstance (“CIM_InstalledOS”, $instanceA.getObjectPath(),
469    $instanceB.getObjectPath());
470 &smEnd;
```
```

6.4 CIM_ElementCapabilities

The `cd` and `help` verbs shall be supported as described in DSP0216.

Table 4 lists each SM CLP verb, the required level of support for the verb in conjunction with instances of the target class, and, when appropriate, a cross-reference to the section detailing the mapping for the verb and target. Table 4 is for informational purposes only; in case of a conflict between Table 4 and requirements detailed in the following sections, the text detailed in the following sections supersedes the information in Table 4.

Table 4 – Command Verb Requirements for CIM_ElementCapabilities

Command Verb	Requirement	Comments
create	Not supported	
delete	Not supported	
dump	Not supported	
load	Not supported	
reset	Not supported	
set	Not supported	
show	Shall	See 6.4.2.
start	Not supported	
stop	Not supported	

No mapping is defined for the following verbs for the specified target: `create`, `delete`, `dump`, `load`, `reset`, `set`, `start`, and `stop`.

6.4.1 Ordering of Results

When results are returned for multiple instances of CIM_ElementCapabilities, implementations shall utilize the following algorithm to produce the natural (that is, default) ordering:

- Results for CIM_ElementCapabilities are unordered; therefore, no algorithm is defined.

6.4.2 Show

This section describes how to implement the `show` verb when applied to an instance of CIM_ElementCapabilities. Implementations shall support the use of the `show` verb with CIM_ElementCapabilities.

The `show` command is used to display information about the CIM_ElementCapabilities instance or instances.

6.4.2.1 Show Multiple Instances – CIM_OperatingSystemCapabilities Reference

This command form is for the `show` verb applied to multiple instances. This command form corresponds to a `show` command issued against CIM_ElementCapabilities where only one reference is specified and the reference is to an instance of CIM_OperatingSystemCapabilities.

6.4.2.1.1 Command Form

```
show <CIM_ElementCapabilities multiple instances>
```

6.4.2.1.2 CIM Requirements

See CIM_ElementCapabilities in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.4.2.1.3 Behavior Requirements

6.4.2.1.3.1 Preconditions

$instance contains the instance of CIM_OperatingSystemCapabilities which is referenced by CIM_ElementCapabilities.

6.4.2.1.3.2 Pseudo Code

```
&smShowAssociationInstances ( “CIM_ElementCapabilities”, $instance.getObjectPath() );
&smEnd;
```

6.4.2.2 Show a Single Instance – CIM_OperatingSystem Reference

This command form is for the show verb applied to a single instance. This command form corresponds to a show command issued against CIM_ElementCapabilities where the reference specified is to an instance of CIM_OperatingSystem.

6.4.2.2.1 Command Form

```
show <CIM_ElementCapabilities single instance>
```

6.4.2.2.2 CIM Requirements

See CIM_ElementCapabilities in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.4.2.2.3 Behavior Requirements

6.4.2.2.3.1 Preconditions

$instance contains the instance of CIM_OperatingSystem, which is referenced by CIM_ElementCapabilities.

6.4.2.2.3.2 Pseudo Code

```
&smShowAssociationInstances ( “CIM_ElementCapabilities”, $instance.getObjectPath() );
&smEnd;
```

6.4.2.3 Show a Single Instance – CIM_OperatingSystemCapabilities and CIM_OperatingSystem References

This command form is for the show verb applied to a single instance. This command form corresponds to a show command issued against CIM_ElementCapabilities where both references are specified and therefore the desired instance is unambiguously identified.

6.4.2.3.1 Command Form

```
show <CIM_ElementCapabilities single instance>
```

6.4.2.3.2 CIM Requirements

See CIM_ElementCapabilities in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.4.2.3.3 Behavior Requirements

6.4.2.3.3.1 Preconditions

$instanceA contains the instance of CIM_OperatingSystemCapabilities which is referenced by CIM_ElementCapabilities.

$instanceB contains the instance of CIM_OperatingSystem which is referenced by CIM_ElementCapabilities.

6.4.2.3.3.2 Pseudo Code

```
&smShowAssociationInstance ( “CIM_ElementCapabilities”, $instanceA.getObjectPath(),
   $instanceB.getObjectPath() );
&smEnd;
```

6.5 CIM_OperatingSystemCapabilities

The `cd` and `help` verbs shall be supported as described in DSP0216.

Table 5 lists each SM CLP verb, the required level of support for the verb in conjunction with instances of the target class, and, when appropriate, a cross-reference to the section detailing the mapping for the verb and target. Table 5 is for informational purposes only; in case of a conflict between Table 5 and requirements detailed in the following sections, the text detailed in the following sections supersedes the information in Table 5.

Table 5 – Command Verb Requirements for CIM_OperatingSystemCapabilities

Command Verb	Requirement	Comments
create	Not supported	
delete	Not supported	
dump	Not supported	
load	Not supported	
reset	Not supported	
set	Not supported	
show	Shall	See 6.5.2.
start	Not supported	
stop	Not supported	

No mapping is defined for the following verbs for the specified target: `create`, `delete`, `dump`, `load`, `reset`, `set`, `start`, and `stop`.

6.5.1 Ordering of Results

When results are returned for multiple instances of CIM_OperatingSystemCapabilities, implementations shall utilize the following algorithm to produce the natural (that is, default) ordering:

- Results for CIM_OperatingSystemCapabilities are unordered; therefore, no algorithm is defined.

6.5.2 Show

This section describes how to implement the `show` verb when applied to an instance of CIM_OperatingSystemCapabilities. Implementations shall support the use of the `show` verb with CIM_OperatingSystemCapabilities.

The `show` verb is used to display information about an instance or instances of the CIM_OperatingSystemCapabilities class.

6.5.2.1 Show a Single Instance

This command form is for the `show` verb applied to a single instance of CIM_OperatingSystemCapabilities.

6.5.2.1.1 Command Form

```
show <CIM_OperatingSystemCapabilities single instance>
```

6.5.2.1.2 CIM Requirements

See CIM_OperatingSystemCapabilities in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.5.2.1.3 Behavior Requirements

6.5.2.1.3.1 Preconditions

`#all` is true, if the “-all” option was specified with the command; otherwise, `#all` is false.

6.5.2.1.3.2 Pseudo Code

```
$instance=<CIM_OperatingSystemCapabilities single instance>
#propertylist[] = NULL;
if (false == #all)
   {
   #propertylist[] = {//all mandatory non-key properties}
   }
&smShowInstance ( $instance.getObjectPath(), #propertylist[] );
&smEnd;
```

6.5.2.2 Show Multiple Instances

This command form is for the `show` verb applied to multiple instances of CIM_OperatingSystemCapabilities. This command form corresponds to UfsT-based selection within a capabilities collection.

6.5.2.2.1 Command Form

```
show <CIM_OperatingSystemCapabilities multiple instances>
```

6.5.2.2.2 CIM Requirements

See CIM_OperatingSystemCapabilities in the “CIM Elements” section of the *OS Status Profile* for the list of mandatory properties.

6.5.2.2.3 Behavior Requirements

6.5.2.2.3.1 Preconditions

$containerInstance contains the instance of CIM_Group for which the contained CIM_Capabilities instances are displayed. CIM_Capabilities instances are addressed via an aggregating instance of CIM_Group.

`#all` is true, if the “-all” option was specified with the command; otherwise, `#all` is false.

6.5.2.2.3.2 Pseudo Code

```
#propertylist[] = NULL;
if (false == #all)
   {
   #propertylist[] = {//all mandatory non-key properties }
   }
&smShowInstances ( “CIM_OperatingSystemCapabilities”, “CIM_MemberOfCollection”,
   $containerInstance.getObjectPath(), #propertylist[] );
&smEnd;
```

608 # ANNEX A
609 (informative)
610
611
612 # Change Log

Version	Date	Author	Description
1.0.0	2009-06-04		DMTF Standard Relase

613