Chapter 1  About Command Line Interface
Chapter 2  XML interface
Chapter 3  Component management
Chapter 4  Group management
Chapter 5  ExpressUpdate
Chapter 6  Log management
Chapter 7  Troubleshooting
Chapter 8  Terminology
Chapter 9  Appendix
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contents</td>
<td>1</td>
</tr>
<tr>
<td>Trademarks</td>
<td>3</td>
</tr>
<tr>
<td>About This Document</td>
<td>4</td>
</tr>
<tr>
<td><strong>Chapter 1</strong></td>
<td></td>
</tr>
<tr>
<td>Overview</td>
<td>5</td>
</tr>
<tr>
<td>System Requirements</td>
<td>5</td>
</tr>
<tr>
<td>Configuring examples</td>
<td>6</td>
</tr>
<tr>
<td>Using the command line</td>
<td>7</td>
</tr>
<tr>
<td>1.4.1 Shell mode</td>
<td>7</td>
</tr>
<tr>
<td>1.4.2 One liner mode</td>
<td>9</td>
</tr>
<tr>
<td>1.4.3 XML scripting mode</td>
<td>10</td>
</tr>
<tr>
<td>1.4.4 Encrypting of the password</td>
<td>11</td>
</tr>
<tr>
<td>1.5 Basic commands</td>
<td>12</td>
</tr>
<tr>
<td>1.5.1 Target</td>
<td>14</td>
</tr>
<tr>
<td>1.5.2 Basic options</td>
<td>14</td>
</tr>
<tr>
<td>1.5.3 Specific options</td>
<td>15</td>
</tr>
<tr>
<td>1.6 Example of command output</td>
<td>16</td>
</tr>
<tr>
<td>1.7 Diagram of the entire address space</td>
<td>17</td>
</tr>
<tr>
<td><strong>Chapter 2</strong></td>
<td></td>
</tr>
<tr>
<td>XML interface</td>
<td>20</td>
</tr>
<tr>
<td>2.1 Execution of XML</td>
<td>20</td>
</tr>
<tr>
<td>2.1.1 Overview</td>
<td>20</td>
</tr>
<tr>
<td>2.1.2 XML elements</td>
<td>20</td>
</tr>
<tr>
<td>2.1.3 Examples</td>
<td>21</td>
</tr>
<tr>
<td>2.1.4 Override process</td>
<td>23</td>
</tr>
<tr>
<td>2.1.5 Include process</td>
<td>24</td>
</tr>
<tr>
<td>2.2 Output XML data</td>
<td>25</td>
</tr>
<tr>
<td>2.2.1 Overview</td>
<td>25</td>
</tr>
<tr>
<td>2.2.2 XML elements</td>
<td>25</td>
</tr>
<tr>
<td>2.2.3 XML format for each basic commands</td>
<td>26</td>
</tr>
<tr>
<td><strong>Chapter 3</strong></td>
<td></td>
</tr>
<tr>
<td>Component management</td>
<td>29</td>
</tr>
<tr>
<td>3.1 Component information</td>
<td>29</td>
</tr>
<tr>
<td>3.1.1 Showing a list of components</td>
<td>29</td>
</tr>
<tr>
<td>3.1.2 Showing component information</td>
<td>30</td>
</tr>
<tr>
<td><strong>Chapter 4</strong></td>
<td></td>
</tr>
<tr>
<td>Group management</td>
<td>31</td>
</tr>
<tr>
<td>4.1 Group information</td>
<td>31</td>
</tr>
<tr>
<td>4.1.1 Showing a list of groupset</td>
<td>31</td>
</tr>
<tr>
<td>4.1.2 Showing information of groupset</td>
<td>32</td>
</tr>
<tr>
<td>4.1.3 Showing information of group</td>
<td>33</td>
</tr>
<tr>
<td><strong>Chapter 5</strong></td>
<td></td>
</tr>
<tr>
<td>ExpressUpdate</td>
<td>35</td>
</tr>
<tr>
<td>5.1 ExpressUpdate information</td>
<td>35</td>
</tr>
<tr>
<td>5.1.1 Displaying ExpressUpdate information</td>
<td>35</td>
</tr>
<tr>
<td>5.1.2 Displaying a list of modules supporting automatic update</td>
<td>37</td>
</tr>
<tr>
<td>5.1.3 Displaying a list of modules not supporting automatic update</td>
<td>39</td>
</tr>
<tr>
<td>5.1.4 Information of module supporting automatic update</td>
<td>41</td>
</tr>
<tr>
<td>5.1.5 Information of module not supporting automatic update</td>
<td>43</td>
</tr>
<tr>
<td>5.2 Update, install and uninstall</td>
<td>45</td>
</tr>
<tr>
<td>5.2.1 Update modules supporting automatic update</td>
<td>45</td>
</tr>
<tr>
<td>5.2.2 Install commands</td>
<td>48</td>
</tr>
<tr>
<td>5.2.3 Uninstall commands</td>
<td>49</td>
</tr>
<tr>
<td>5.2.4 Cancelling update commands</td>
<td>50</td>
</tr>
<tr>
<td>5.2.5 Update modules not supporting automatic update</td>
<td>51</td>
</tr>
<tr>
<td>5.3 Repository and Update package management</td>
<td>54</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.3.1 Repository settings</td>
<td>54</td>
</tr>
<tr>
<td>5.3.2 Adding update packages to repository</td>
<td>58</td>
</tr>
<tr>
<td>5.3.3 Removing update packages from repository</td>
<td>59</td>
</tr>
<tr>
<td>5.3.4 Saving update packages</td>
<td>61</td>
</tr>
<tr>
<td>5.3.5 Update packages information</td>
<td>62</td>
</tr>
<tr>
<td>Chapter 6 Log management</td>
<td>66</td>
</tr>
<tr>
<td>6.1 Logging</td>
<td>66</td>
</tr>
<tr>
<td>6.1.1 Application log list</td>
<td>66</td>
</tr>
<tr>
<td>6.1.2 Application log</td>
<td>67</td>
</tr>
<tr>
<td>6.1.3 NEC ExpressUpdate Agent log</td>
<td>68</td>
</tr>
<tr>
<td>Chapter 7 Troubleshooting</td>
<td>69</td>
</tr>
<tr>
<td>7.1 Error message</td>
<td>69</td>
</tr>
<tr>
<td>Chapter 8 Terminology</td>
<td>70</td>
</tr>
<tr>
<td>Chapter 9 Appendix</td>
<td>71</td>
</tr>
<tr>
<td>9.1 XML Schema</td>
<td>71</td>
</tr>
<tr>
<td>9.1.1 XML Schema for request file</td>
<td>71</td>
</tr>
<tr>
<td>9.1.2 XML Schema for response file</td>
<td>74</td>
</tr>
</tbody>
</table>
Trademarks

NEC EXPRESSBUILDER and NEC ESMPRO are registered trademarks of NEC Corporation.
Microsoft, Windows, Windows Vista, Windows Server are registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.

All other company, or product names used in this document are registered trademarks or trademarks of their respective trademark owners.

Windows 8 stands for Windows® 8 Pro, and Windows® 8 Enterprise.
Windows 7 stands for Windows® 7 Professional operating system, and Windows® 7 Ultimate operating system.
Windows Vista stands for Windows Vista® Business operating system, Windows Vista® Enterprise operating system, and Windows Vista® Ultimate operating system.
Windows XP stands for Windows® XP Professional operating system, and Windows® XP Professional x64 Edition operating system.

All names used in sample applications are fictitious. They are unrelated to existing product, organization, or individual names.

Notes
(1) No part of this document may be reproduced in any form without the prior written permission of NEC Corporation.
(2) The contents of this document may be revised without prior notice.
(3) The contents of this document shall not be copied or altered without the prior written permission of NEC Corporation
(4) All efforts have been made to ensure the accuracy of all information in this document. If you notice any part unclear, incorrect, or omitted in the document, contact your authorized NEC sales representative.
(5) NEC assumes no liability for damages arising from the use of this product, nor any liability for incidental or consequential damages arising from the use of this document regardless of (4)
About This Document

This document introduces command line interface of NEC ExpressUpdate. It is a function of the component management utility "NEC ESMPRO Manager".

Before attempting to operate the command line interface, read this document so as to gain an adequate understanding of the contents.

Attention
This document is intended for persons who are familiar with the operating system's functions and operations and the network's functions and setup. For operations and inquiries about the operating system, see its online help information.

This document covers universal information about generally managed components. The notes and restrictions on use of each product as a managed component are explained in the user's guide provided with the managed component.

Names used with screen images in this document are fictitious. They are unrelated to existing product names, names of organizations, or individual names. The setting values on the screen images are shown as examples, so setting values such as IP addresses on screen images are not guaranteed for operation.

About Symbols in This Document
The following explains three symbols that are used in this document:

IMPORTANT: Points that are mandatory or require attention when using the software or the component.
CHECK: Points that are require confirmation when using the software or the component.
TIP: Helpful and convenient piece of information.

About Font in This Document
The Italic font shows the option of command in this document.

For other information about the NEC ESMPRO Manager
See the documents below.

NEC ESMPRO Manager Ver.6 Installation Guide
NEC ESMPRO Manager Ver.6 Setup Guide
NEC ESMPRO Manager Ver.6 Command Line Interface
Chapter 1  About Command Line Interface

1.1 Overview
This document provides information about a Command Line Interface to NEC ExpressUpdate. This interface is executed by 'esmcli' command.

IMPORTANT:
- 'esmcli' command does not have a function to registering a component to NEC ESMPRO Manager. In case of registering a component, please use NEC ESMPRO Manager's Web interface.
- In case of CLI, following characters are not available in group name.
  "' ¥ < > & " ( ) ^"
- Using NEC ExpressUpdate, "Updates via NEC ExpressUpdate Agent" or "Updates via Management controller" of the component must be enabled.

1.2 System Requirements
'esmcli' command can be executed only on a management PC, which NEC ESMPRO Manager Ver. 5.4 or later is installed on.

NEC ESMPRO Manager command line interface requires following user level of operating system:
On Windows: Administrator
On Linux: root

CHECK:

TIP:
- See "NEC ESMPRO Manager Ver.6 Installation Guide" for information about system requirements of NEC ESMPRO Manager.
1.3 Configuring examples
When the client PC is different from the management PC, log in to the management PC using remote desktop, Telnet/SSH clients, etc.

Examples
When NEC ESMPRO Manager is installed on Windows environments

Examples
When NEC ESMPRO Manager is installed on Linux environments
1.4 Using the command line
All commands in this document are executed by 'esmcli' command. When you install NEC ESMPRO Manager, 'esmcli' command is also installed on the following directories.

Windows OS:
C:\Program Files\ESMPRO\ESMMNG\bin
This path is added to the system environment variable "PATH".

CHECK:
- The Windows path varies if NEC ESMPRO Manager is not installed in the default location.

Linux OS:
/opt/nec/es_manager/bin
A symbolic link to 'esmcli' is created on /usr/bin directory.

'esmcli' has two mode, interactive "shell mode" and non-interactive "one liner mode".

1.4.1 Shell mode
In this mode, you can execute CLI commands interactively.

1.4.1.1 Logging in to the shell mode
To start the shell mode, enter the following command from a command line. Then input a user name and password of NEC ESMPRO Manager.
Please refer to "1.5 Basic commands" for information about CLI commands.

```
esmcli [Option]
esmcli [Option]
```

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Display the syntax of esmcli commands. When this option is specified, the shell mode does not start.</td>
</tr>
<tr>
<td>-u &lt;user name&gt;</td>
<td>Specifies an user name of NEC ESMPRO Manager. At the time of login, the input of the user name is omitted.</td>
</tr>
<tr>
<td>-p &lt;password&gt;</td>
<td>Specifies an user password of NEC ESMPRO Manager. At the time of login, the input of the user password is omitted.</td>
</tr>
</tbody>
</table>

TIP:
- The password at the time of the login can input the password that encrypted in "1.4.4 Encrypting of the password".

Examples
If username and password are not specified at command line options, enter them at following prompts.

```
> esmcli
user: passwd:
```

Examples
When you specify the user name and the password at command line, please input as follows.
```
esmcli -u Administrator -p password
```
Examples
When you specify the encrypted password, please input as follows. See 1.4.4 Encrypting of the password.

```
esmcli -u Administrator -p {ENC}c10f259c9f7d203a4424bf06b6713
```

When the login is successful, 'esmcli' prompt is displayed.

```
NEC ESM PRO Manager Version 6
Copyright (C) 2004-2015 NEC Corporation. All Rights Reserved.
->
```

1.4.1.2 Logging out of the shell mode
To exit the shell mode, enter the exit command or input Ctrl + C.

```
- -> exit
```

1.4.1.3 Keyboard Shortcuts
This section describes keyboard shortcuts list.

### Table 1-1 Keyboard Shortcuts

<table>
<thead>
<tr>
<th>Keyboard Shortcuts</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter</td>
<td>Decision.</td>
</tr>
<tr>
<td>BackSpace</td>
<td>Erase one character.</td>
</tr>
<tr>
<td>Tab</td>
<td>Autocompletes from the cursor position.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Ctrl + B</td>
<td>Moves the cursor backward one character. (equivalent to the key &quot; &quot;)</td>
</tr>
<tr>
<td>Ctrl + F</td>
<td>Moves the cursor forward one character. (equivalent to the key &quot; &quot; )</td>
</tr>
<tr>
<td>Ctrl + A</td>
<td>Moves the cursor to the line start.</td>
</tr>
<tr>
<td>Ctrl + E</td>
<td>Moves the cursor to the line end.</td>
</tr>
<tr>
<td>Ctrl + G</td>
<td>Move the cursor to the top of the word before one. Because there are space and hyphen (&quot;-&quot;) and directory separator (&quot;/&quot; or &quot;&quot;), the word is discerned.</td>
</tr>
<tr>
<td>Ctrl + P</td>
<td>Recalls the prior command. (equivalent to the key &quot; &quot;)</td>
</tr>
<tr>
<td>Ctrl + N</td>
<td>Recalls the next command. (equivalent to the key &quot; &quot;)</td>
</tr>
<tr>
<td>Ctrl + I</td>
<td>Autocompletes from the cursor position. (equivalent to the key &quot;Tab&quot;)</td>
</tr>
<tr>
<td>Ctrl + V</td>
<td>Paste. (Supported only in Windows OS.)</td>
</tr>
<tr>
<td>Ctrl + J</td>
<td>Decision. (equivalent to the key &quot;Enter&quot;)</td>
</tr>
<tr>
<td>Ctrl + M</td>
<td>Decision. (equivalent to the key &quot;Enter&quot;)</td>
</tr>
<tr>
<td>Ctrl + H</td>
<td>Erase one character. (equivalent to the key &quot;BackSpace&quot;)</td>
</tr>
<tr>
<td>Ctrl + L</td>
<td>Clear the screen. (Supported only in Linux OS.)</td>
</tr>
<tr>
<td>Ctrl + K</td>
<td>Delete the line after the cursor position.</td>
</tr>
<tr>
<td>Ctrl + U</td>
<td>Delete the line before the cursor position.</td>
</tr>
<tr>
<td>Ctrl + W</td>
<td>Delete all the letters from the cursor to the directory separator before one.</td>
</tr>
<tr>
<td>Ctrl + D</td>
<td>When there is an input, delete one character of the position of the cursor. When there is not an input, finish the shell mode.</td>
</tr>
<tr>
<td>Ctrl + C</td>
<td>Finish the shell mode.</td>
</tr>
</tbody>
</table>
1.4.2 One liner mode

The one liner mode executes only specified CLI command without starting the shell function of esmcli. To execute the one liner mode, enter a CLI command following to a username and password from a command line as shown below. And please input user name and password of NEC ESMPRO Manager. Please refer to "1.5 Basic commands" for the CLI command to execute.

```
esmcli [Option] '{CLI Command}'
```

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>-help</td>
</tr>
<tr>
<td>-u</td>
<td>-user &lt;user name&gt;</td>
</tr>
<tr>
<td>-p</td>
<td>-pswd &lt;password&gt;</td>
</tr>
</tbody>
</table>

TIP:
- The password at the time of the login can input the password that encrypted in "1.4.4 Encrypting of the password".

Examples
Following command enables you to execute 'show /' command.
```
esmcli -u Administrator -p password 'show /
```

Examples
Using an encrypted password, please input it as follows.
```
esmcli -u Administrator -p {ENC}c10f239c9f7d203fa4424bffb06b6713 'show /
```

1.4.2.1 Notes on executing one liner mode

(1) When entering special characters
When input double quotation (") in CLI command, please set ¥ before double quotation. The following shows examples.
```
esmcli 'show /cmps/¥"server01¥"/map/expup'
```
1.4.3 XML scripting mode
XML scripting mode enables you to write esmcli commands and some arguments in an XML file and execute it. To obtain the information of XML format, refer to "9.1.1XML Schema for request file".

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>esmcli -f &lt;XML file name&gt; [Option]</td>
<td>Indicates the NEC ESM PRO Manager command line interface command</td>
</tr>
<tr>
<td>&lt;XML file name&gt;</td>
<td>Specifies XML file in which esmcli commands are written.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>-h</td>
<td>-help</td>
</tr>
<tr>
<td>-u</td>
<td>-user &lt;user name&gt;</td>
</tr>
<tr>
<td>-p</td>
<td>-pswd &lt;password&gt;</td>
</tr>
<tr>
<td>-x</td>
<td>examine</td>
</tr>
<tr>
<td>-override &lt;Name&gt;=&lt;Value&gt;</td>
<td>Replaces the value of XML element whose name equals &lt;Name&gt; with &lt;Value&gt;. To obtain more information, refer to &quot;2.1.4Override process&quot;.</td>
</tr>
</tbody>
</table>

TIP:
- The password at the time of the login can input the password that encrypted in "1.4.4Encrypting of the password".

Examples
Following command enables you to execute sample.xml.
```
esmcli -f sample.xml -u Administrator -p password
```

Examples
Using an encrypted password, please input it as follows.
```
esmcli -f sample.xml -u Administrator -p {ENC}c10f239c9f7d203fa4424bfb7b06b6713
```
### 1.4.4 Encrypting of the password

The esmclipasswd command is used for encrypting of the password. To encrypt the password, enter the command following the command prompt as shown below.

```
esmclipasswd [Option] <Password>
```

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>-help</td>
</tr>
</tbody>
</table>

| <Password> | Input a password to encrypt.                                               |

#### Examples

When you encrypt the password, please input it as follows.

```
> esmclipasswd password
{ENC}c10f239c9f7d203fa4424bfb06b6713
```

The encrypted password is displayed on the screen.
1.5 Basic commands

This section describes basic commands. These basic commands are based on SMASH style proposed by DMTF (Distributed Management Task Force).

Help string, command syntax, of each command appears when "-h | -help" option is specified as the <options> of the command. The argument placed between "[" and "]" is omissible.

**TIP:**
- In the case of the user authority is Administrator, the user can execute all basic commands.
- In the case of the user authority is Operator, the user can execute cd, exit, help and show commands. Other commands become executable by setting of the user level. The details please identify a chapter of each operation.
- help, cd, exit and show commands are supported at all of targets.

**help**

**Syntax**
```
help [<options>] [<target>]
```

**Description**
The `help` command is used to request information related to the use of the CLP. When you omit <target>, the `help` command will display information about the use of current target.

**cd**

**Syntax**
```
cd [<options>] [<target>]
```

**Description**
The `cd` command is used to change the current default target to the target specified by the <target> argument. You can shorten a command by changing current target. When you omit <target>, the `cd` command will display current target.

**exit**

**Syntax**
```
exit [<options>]
```

**Description**
The `exit` command terminates and logs out the user session.

**show**

**Syntax**
```
show [<options>] [<target>]
```

**Description**
The `show` command is used to display information about <target>. When you omit <target>, the `show` command will display information about current target.

**create**

**Syntax**
```
create [<options>] <target>
```

**Description**
The `create` command is used to create new target objects.
delete
Syntax
delete [options] [target]
Description
The delete command is used to remove a target.
When you omit target, the delete command will delete current target.

load
Syntax
load [options] [target]
Description
The load command is used to take a binary image from a specific source location and place it at the specified target address.

reset
Syntax
reset [options] [target]
Description
The reset command resets the target’s state.

set
Syntax
set [options] [target] <propertyname>=<value>...
Description
The set command is used to set the value of one or more of a target’s properties.

start
Syntax
start [options] [target]
Description
The start command starts the target.

stop
Syntax
stop [options] [target]
Description
The stop command stops the target.

dump
Syntax
dump -destination <path> [options] [target]
Description
The dump command is used to take a binary image from the target and send it to a specific location.
1.5.1 Target
Each basic command functions to a specified target. The target points the managed element by address much like the path to a file in a file system.

Both absolute path, which is started from "/", and relative path are available for pointing the target. Specifically, "." and ".." are supported. The "." means the current default target and the ".." means the parent target.

Specifying the target, enter <target> following to each basic command. If it is not specified, commands functions to a current default target. The current default target can be changed by the cd command. The current default target is "/"(root) when the command line interface session is started.

1.5.2 Basic options
This section describes basic options based on SMASH style.

-h | -help
Description
Displays documentation about the command verb.
When this option is specified, the command is not executed.
This option is supported by all basic commands.

-x | -examine
Description
Checks the syntax of the command.
When this option is specified, the command is not executed.
This option is supported by all basic commands.

-d | -display <type>,<type>,...
Description
Shows information of the selected <type>.
This option is supported by show command.
Type can specify the following items. If there are multiple types, they must be separated by commas.

targets=(<name>, …)
Shows the target.
When <name> is specified, the target that matches <name> is displayed.
If there are multiple <name>, they must be surrounded by parenthesis and separated by commas.

properties=(<name>, …)
Shows properties.
When <name> is specified, the property that matches <name> is displayed.
If there are multiple <name>, they must be surrounded by parenthesis and separated by commas.

verbs
Shows supported commands.
Examples
(1) Shows only targets

```
show -d Targets
```

(2) Shows target whose name is “server01” and supported commands.

```
show -d targets=server01,verbs
```

(3) Shows property whose name is “Name” or “Status”, and shows supported commands.

```
show -d properties=(Name,Status),verbs
```

```
-o | -output <arg>,<arg>...
```

Description
Specifies a format of output data.

```
format=text | clxml
```

If set to "text", output is in text format.
If set to "clxml", output is in XML format.

1.5.3 Specific options
This section describes NEC ESMPRO Manager specific options.

```
-exclude <arg>,<arg>...
```

Description
Specifies exceptional items. This option is effective when the target is group.

```
cmp="("<name>,<name>, ...,<name>")"
```

Specifies exceptional items using component names.
If there are some <name>, they must be separated by comma and surrounded by parentheses.

```
grp="("<name>,<name>, ...,<name>")"
```

Specifies exceptional items using group names.
If there are some <name>, they must be separated by comma and surrounded by parentheses.

```
ufit="("<name>,<name>, ...,<name>")"
```

Specifies exceptional items using UFit.
If there are some <name>, they must be separated by comma and surrounded by parentheses.

TIP:
• UFit is displayed at the result of show command.

```
-outputfile <path>
```

Description
Saves command result to a file of specified path.
When the file of specified path is not existing, new file is created and output it.
When the file of specified path exists and the file is not an output file of esmcli, the command result will be error.
This option is supported by all basic commands.
1.6 Example of command output

All commands shows results as following format.

```
-> <CLI command>
<status>
results

->
```

Each status is listed in below table.

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND COMPLETED</td>
<td>Command was successful. Following commands skips to show this status.</td>
</tr>
<tr>
<td></td>
<td>• cd</td>
</tr>
<tr>
<td></td>
<td>• exit</td>
</tr>
<tr>
<td></td>
<td>• help</td>
</tr>
<tr>
<td></td>
<td>• how</td>
</tr>
<tr>
<td>COMMAND PROCESSING FAILED</td>
<td>Syntax error was occurred.</td>
</tr>
<tr>
<td>COMMAND EXECUTION FAILED</td>
<td>Command was unsuccessful.</td>
</tr>
</tbody>
</table>
1.7 Diagram of the entire address space

A diagram of the entire address space is as follows.
Chapter 2 XML interface

2.1 Execution of XML

2.1.1 Overview
This function enables you to write a command and arguments into a request XML file and execute the file instead of "Shell mode" or "One liner mode". An XML schema of the request XML is described in "9.1.1 XML Schema for request file".

2.1.2 XML elements
Standard contents of the request XML are as below. Details of those elements are described in Table 2-1.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <COMMAND>
    <abort>true</abort>
    <instance>
      <ufip>Target of this COMMAND</ufip>
      <options>
        <option>
          <name>Name of option</name>
          <value>
            <val>Value of option</val>
          </value>
        </option>
      </options>
      <properties>
        <property>
          <name>Name of property</name>
          <value>
            <val>Value of property</val>
          </value>
        </property>
      </properties>
    </instance>
  </COMMAND>
</request>
```
### Table 2-1 Elements of request XML

<table>
<thead>
<tr>
<th>Name of element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>request</td>
<td>A root element of request XML.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>An element specifying a basic command.</td>
</tr>
<tr>
<td>abort</td>
<td>An element specifying a behavior in case of error. If set to true, the process is stopped. If set to false, the process runs on. If this element does not exist, true is used. This setting is only used when multi commands are defined in a request XML.</td>
</tr>
<tr>
<td>instance</td>
<td>An element grouping a target, options and properties.</td>
</tr>
<tr>
<td>ufp</td>
<td>An element specifying a target of the command.</td>
</tr>
<tr>
<td>options</td>
<td>An element grouping option elements.</td>
</tr>
<tr>
<td>option</td>
<td>An element corresponding to option.</td>
</tr>
<tr>
<td>properties</td>
<td>An element grouping property elements.</td>
</tr>
<tr>
<td>property</td>
<td>An element corresponding to property.</td>
</tr>
<tr>
<td>name</td>
<td>An element holding a name of option or property.</td>
</tr>
<tr>
<td>value</td>
<td>An element holding a value.</td>
</tr>
<tr>
<td>val</td>
<td>An element holding a value of option or property.</td>
</tr>
<tr>
<td>include</td>
<td>An element specifying an included XML file.</td>
</tr>
<tr>
<td>file</td>
<td>An element specifying a file.</td>
</tr>
</tbody>
</table>

### 2.1.3 Examples

Followings are typical command's input method using shell mode and XML interface.

**Example**

Execute 'show' command to '/'.

```
->show /
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <show>
    <instance>
      <ufip>/</ufip>
    </instance>
  </show>
</request>
```
Example
Execute 'dump' command to /logs and save logs to C:\temp.
-> dum p -destination C:\temp /logs

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <dump>
    <instance>
      <ufip>/logs</ufip>
      <options>
        <option>
          <name>destination</name>
          <value>
            <val>C:\temp</val>
          </value>
        </option>
      </options>
    </instance>
  </dump>
</request>
```

Example
Execute 'set' command to /repository and change RepositoryLocation to LOCAL.
-> set/repository RepositoryLocation=LOCAL

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <set>
    <instance>
      <ufip>/repository</ufip>
      <properties>
        <property>
          <name>RepositoryLocation</name>
          <value>
            <val>LOCAL</val>
          </value>
        </property>
      </properties>
    </instance>
  </set>
</request>
```

Example
Execute 'load' command to /cm ps/<Component Name>/map/expup and applies latest update packages to <Component Name>.
-> load /cm ps/<Component Name>/map/expup

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <load>
    <instance>
      <ufip>/cm ps/<Component Name>/map/expup</ufip>
    </instance>
  </load>
</request>
```
2.1.4 Override process

Specifying ‘override’ option with the esmcli, following elements of an XML file will be overridden.

- ufip element
- val element belonging to option element
- val element belonging to property element

The overridden element is specified by absolute path separated by '/'. To specify some elements, describe them separated by '!' if there is more than one element corresponding to the specified path, specify the order of the element by "ElementName[number]". If the number is not specified, all elements corresponding to the path will be overridden.

In the following example,

- /request/show/instance/options/option/value/val matches both [1] and [2]

```xml
<request>
  <show>
    <instance>
      <ufip/>
      <options>
        <option>
          <name>name</name>
          <value>
            <val>val</val> <!-- [1] -->
          </value>
        </option>
        <option>
          <name>name</name>
          <value>
            <val>val</val> <!-- [2] -->
          </value>
        </option>
      </options>
    </instance>
  </show>
</request>
```
2.1.5 Include process

A request XML can include other request XML and execute its commands. To use this function, create an `<include>` element under `<request>` element and specify included file using `<file>` element.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <include>
    <file>filename.xml</file>
    <abort>true</abort>
  </include>
</request>
```

More than one element can be specified. In the following case, filename1.xml is executed and then filename2.xml is executed.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <include>
    <file>filename1.xml</file>
    <abort>true</abort>
  </include>
  <include>
    <file>filename2.xml</file>
    <abort>true</abort>
  </include>
</request>
```
2.2 Output XML data

2.2.1 Overview
If an argument "-o | -output format=clp|xml" is specified, output is in XML format. An XML schema of the XML is described in "9.1.2XML Schema for response file".

2.2.2 XML elements
Standard contents of the response XML are as below. A format of <COMMAND> element is different from each basic commands. Details of other elements are described in Table 2-2.

```
<!-- In the case of successful -->
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <command>
    <inputline>…</inputline>
  </command>
  <cmdstat>
    <status>0</status>
    <status_tag>COMMAND COMPLETED</status_tag>
  </cmdstat>
  <COMMAND>
  </COMMAND>
  <oemdata>
  </oemdata>
</response>

<!-- In case of error -->
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <command>
    <inputline>…</inputline>
  </command>
  <cmdstat>
    <status>3</status>
    <status_tag>COMMAND EXECUTION FAILED</status_tag>
    <error>255</error>
    <error_tag>COMMAND ERROR UNSPECIFIED</error_tag>
  </cmdstat>
  <COMMAND>
  </COMMAND>
  <oemdata>
  </oemdata>
</response>
```
<table>
<thead>
<tr>
<th>Name of element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>response</td>
<td>A root element of response XML.</td>
</tr>
<tr>
<td>command</td>
<td>An element holding information of the command.</td>
</tr>
<tr>
<td>inputline</td>
<td>An element holding an input command line.</td>
</tr>
<tr>
<td>cmdstat</td>
<td>An element grouping a status of command.</td>
</tr>
<tr>
<td>status</td>
<td>An element holding a status code.</td>
</tr>
<tr>
<td>status_tag</td>
<td>An element holding a status tag.</td>
</tr>
<tr>
<td>error</td>
<td>An element holding an error code.</td>
</tr>
<tr>
<td>error_tag</td>
<td>An element holding an error tag.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>An element specifying a basic command.</td>
</tr>
<tr>
<td>target</td>
<td>An element holding a target of the command.</td>
</tr>
<tr>
<td>instance</td>
<td>An element grouping a target, options and properties.</td>
</tr>
<tr>
<td>ufit</td>
<td>An element holding target's UFiT.</td>
</tr>
<tr>
<td>properties</td>
<td>An element grouping property elements.</td>
</tr>
<tr>
<td>property</td>
<td>An element corresponding to property.</td>
</tr>
<tr>
<td>name</td>
<td>An element holding a name of property.</td>
</tr>
<tr>
<td>value</td>
<td>An element holding a value of property.</td>
</tr>
<tr>
<td>val</td>
<td>An element holding a value of property.</td>
</tr>
<tr>
<td>source</td>
<td>An element holding source information of dump and load command.</td>
</tr>
<tr>
<td>destination</td>
<td>An element holding destination information of dump and load command.</td>
</tr>
<tr>
<td>uri</td>
<td>An element holding a path of the file system.</td>
</tr>
<tr>
<td>help</td>
<td>An element holding a help message.</td>
</tr>
<tr>
<td>text</td>
<td>An element holding a text message.</td>
</tr>
<tr>
<td>examine</td>
<td>An element holding a result of examination.</td>
</tr>
<tr>
<td>verbs</td>
<td>An element holding a list of supported commands.</td>
</tr>
<tr>
<td>standardverbs</td>
<td>An element holding standard verbs.</td>
</tr>
<tr>
<td>oemdata</td>
<td>An element holding esmcli specific information.</td>
</tr>
<tr>
<td>exitcode</td>
<td>An element holding an exit code.</td>
</tr>
<tr>
<td>result</td>
<td>An element holding a result of the command.</td>
</tr>
</tbody>
</table>

### 2.2.3 XML format for each basic commands

Followings are XML format of each basic commands. A XML schema of the XML is described in "9.1.2XML Schema for response file".

#### 2.2.3.1 cd

```xml
<cd>
  <ufit>UFiP of the current default target.</ufit>
</cd>
```
2.2.3.2 delete

```xml
<delete>
  <target>
    <instance>
      <ufip>UFiP identifying target.</ufip>
    </instance>
  </target>
  <target>
    Recursive target elements representing Managed Elements contained in the initial target.
  </target>
</delete>
```

2.2.3.3 dump

```xml
<dump>
  <source>
    <ufip>Full path of source of dump</ufip>
  </source>
  <destination>
    <ufip>Full path of destination of dump</ufip>
  </destination>
</dump>
```

2.2.3.4 exit

```xml
<exit>
</exit>
```

2.2.3.5 help

```xml
<help>
  <text>Free-form text</text>
</help>
```

2.2.3.6 load

```xml
<load>
  <source>
    <uri>Full path of source of load</uri>
  </source>
  <destination>
    <ufip>Full path of destination of load</ufip>
  </destination>
</load>
```
2.2.3.7 set

<set>
  <instance>
    <ufip>UFiP identifying target</ufip>
    <properties>
      <property>A modified property of Managed Element</property>
    </properties>
  </instance>
</set>

2.2.3.8 show

<show>
  <target>
    <instance>
      <ufip>UFiP identifying target</ufip>
      <properties>
        <property>A property of Managed Element</property>
      </properties>
      <verbs>
        <standardverbs>Standard commands supported at the target</standardverbs>
        <oemverbs>OEM commands supported at the target</oemverbs>
      </verbs>
    </instance>
    <target>Recursive target elements representing Managed Elements contained in initial target</target>
  </target>
</show>

2.2.3.9 stop

<stop>
  <ufip>UFiP identifying target to stop</ufip>
</stop>
Chapter 3  Component management

3.1 Component information

3.1.1 Showing a list of components

'show' command at following target shows a list of all managed components.

- /cmps

Properties of this target are as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of registered components.</td>
</tr>
<tr>
<td>Status</td>
<td>read</td>
<td>Displays a representative status of all of the components.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays a representative ExpressUpdate status of all of the components. Refer to Table 5-2 ExpressUpdate statuses to get information about ExpressUpdate status.</td>
</tr>
</tbody>
</table>

Example

'show' command shows a list of managed components.
More concrete examples are as follows.

```
$ show
ufip=/cmps
ufit=cmps
Targets:
    Server01
    Server02
Properties:
    EntryCount=2
    Status=NORMAL
    ExpUpStatus=LATEST_CONDITION
Verbs:
    cd
    exit
    help
    show
```

Tip:
- 'show /cmps' command also shows above information even if the current target is not '/cmps'.
### 3.1.2 Showing component information

'Show' command at following target shows information of managed component.

- `/cmps/<Component Name>`

Properties of this target are as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>read</td>
<td>Displays a name of the component.</td>
</tr>
<tr>
<td>Status</td>
<td>read</td>
<td>Displays a representative status of all of the components.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays a representative ExpressUpdate status of all of the components.</td>
</tr>
<tr>
<td>Group</td>
<td>read</td>
<td>Displays a name of a group which the component belongs to.</td>
</tr>
<tr>
<td>IpAddress</td>
<td>read</td>
<td>Displays an IP address of the component.</td>
</tr>
<tr>
<td>BmcIpAddress</td>
<td>read</td>
<td>Displays a BMC's IP address of the component.</td>
</tr>
<tr>
<td>Model</td>
<td>read</td>
<td>Displays a model name of the component.</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>read</td>
<td>Displays a serial number of the component.</td>
</tr>
<tr>
<td>Guid</td>
<td>read</td>
<td>Displays a GUID of the component.</td>
</tr>
<tr>
<td>OsVersion</td>
<td>read</td>
<td>Displays an OS information of the component.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>'show' command</td>
<td>Shows information of the component.</td>
</tr>
<tr>
<td>More concrete examples are as follows.</td>
<td></td>
</tr>
</tbody>
</table>

```
-> show /cmps/Server01
ufip=/cmps/Server01
ufit=Server01
Targets:
  system
  map
Properties:
  Name=Server01
  Status=NORMAL
  ExpUpStatus=LATEST_CONDITION
  Group=group01
  IpAddress=192.168.14.18
  BmcIpAddress=192.168.14.19
  Model=Express5800/51Ma [N8000-2001]
  SerialNumber=1234567
  Guid=AAAAAAAA-0000-BBBB-1111-CCCCCCCCCCCC
  OsVersion=Microsoft Windows Vista Business Service Pack 2
Verbs:
  cd
  exit
  help
  show
```
Chapter 4  Group management

4.1  Group information

4.1.1  Showing a list of groupset

'Show' command at the following target shows a list of groupset.

- /grps

Property of this target is as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>A number of registered groupset.</td>
</tr>
</tbody>
</table>

Example

'Show' command shows a list of groupset.

More concrete examples are as follows.

```bash
-> show /grps
ufip=/grps
ufit=grps
Targets:
grps
chassisset
Properties:
EntryCount=2
Verbs:
cd
exit
help
show
```
4.1.2 Showing information of groupset

'show' command at the following target shows information of groupset.

- /grps/<GroupSet Name>

Properties of this target are as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>read</td>
<td>Displays a name of groupset.</td>
</tr>
<tr>
<td>RootGroup</td>
<td>read</td>
<td>Displays root group's name.</td>
</tr>
</tbody>
</table>

Example

'show' command shows information of a groupset.

More concrete examples are as follows.

```
-> show /grps/grpset
ufip=/grps
ufit=grps
Targets:
  root
Properties:
  Name=grpset
  RootGroup=root
Verbs:
  cd
  exit
  help
  show
```
4.1.3 Showing information of group

'Show' command at the following target shows information of a group.

- `/grps/<GroupSet Name>/<Group Name>`

Properties of this target are as follows. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>read</td>
<td>Displays a name of the group.</td>
</tr>
<tr>
<td>Status</td>
<td>read</td>
<td>Displays a status of the group.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays an ExpressUpdate status of the group.</td>
</tr>
<tr>
<td>GroupCount</td>
<td>read</td>
<td>Displays a number of groups belonging to the group.</td>
</tr>
<tr>
<td>ComponentCount</td>
<td>read</td>
<td>Displays a number of components belonging to the group.</td>
</tr>
<tr>
<td>Comment</td>
<td>read</td>
<td>Displays a comment of the group.</td>
</tr>
<tr>
<td>Components</td>
<td>read</td>
<td>Displays a list of components belonging to the group.</td>
</tr>
</tbody>
</table>

**Example**

'Show' command shows information of the group.

More concrete examples are as follows.

```shell
$ show /grps/grpset/root
ufip=/grps/grpset/root
ufit=root
Targets:
  group01
  group02
  expup
Properties:
  Name=grpset
  Status=NORMAL
  ExpUpStatus=LATEST_CONDITION
  GroupCount=2
  ComponentCount=3
  Comment=-
  Components=server01
              server02
              server03
Verbs:
  cd
  exit
  help
  show
```

When show command is executed to groups, 'grpall' is able to specify as a `<type>` of `--display` option. If it is specified, all groups are show as hierarchy style.

Following command shows all groups as hierarchy.

```shell
show --display targets.grpall
```
Followings are example of the below hierarchy diagram.

```
-> show -display targets.grpall/grps/grpset/root
ufip=/grps/grpset/root
ufit=root
Targets:
  group01
    group01-01
    group01-02
      group01-02-01
  group02
    group02-01
      group02-01-01
expup
```
Chapter 5  ExpressUpdate

5.1  ExpressUpdate information
5.1.1  Displaying ExpressUpdate information
5.1.1.1  Information of components

'show' command at the below target enables you to view ExpressUpdate information about the specified component.

- `/cmps/<Component Name>/map/expup`

Properties and specific commands of this target are as follows.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays ExpressUpdate statuses. Each status is listed in Table 5-2.</td>
</tr>
<tr>
<td>NextUpdateDate</td>
<td>read</td>
<td>Displays time and date of the next update if the update is scheduled.</td>
</tr>
<tr>
<td>RepositoryLocation</td>
<td>read</td>
<td>Displays a location of the repository. (local/remote)</td>
</tr>
<tr>
<td>UpdatePkgLatestDownloaded</td>
<td>read</td>
<td>Displays the most recent time when the repository was updated.</td>
</tr>
<tr>
<td>NextDownloadDate</td>
<td>read</td>
<td>Displays time and date of the next repository update if it is scheduled.</td>
</tr>
</tbody>
</table>

Table 5-2  ExpressUpdate statuses

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOT_LATEST</td>
<td>The repository has update packages newer than current version.</td>
</tr>
<tr>
<td>UNDER_INSTALLATION</td>
<td>An updating is in progress.</td>
</tr>
<tr>
<td>UNDER_UNINSTALLING</td>
<td>An uninstallation is in progress.</td>
</tr>
<tr>
<td>REBOOTING</td>
<td>The update which requires rebooting the system is executed and rebooting is in progress. It is not possible to update a module under this status.</td>
</tr>
<tr>
<td>REBOOT_REQUIRED</td>
<td>This status shows that the system has not been rebooted yet although the update requires rebooting. In this status, the update has not been completed yet. This status is cleared by rebooting the system. It is not possible to update a module under this status.</td>
</tr>
<tr>
<td>LATEST_CONDITION</td>
<td>The repository has no update package newer than current version.</td>
</tr>
<tr>
<td>INSTALLATION_FAILED</td>
<td>This status shows that the last update command was unsuccessful. This status is cleared by rebooting the system. A reason of the failure is shown in application logs. Refer to section 6.1.1 and 6.1.2 to get more information about the application log. It is not possible to update a module under this status.</td>
</tr>
<tr>
<td>ACCESS_FAILED</td>
<td>There is no communication to NEC ExpressUpdate Agent.</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>Failed to get version information because the repository has incompatible packages. Updating ESMPRO/ServerManager is required.</td>
</tr>
<tr>
<td>NO_PACKAGE</td>
<td>The repository has no package.</td>
</tr>
<tr>
<td>INSTALLABLE</td>
<td>The specified software is installable.</td>
</tr>
</tbody>
</table>
Table 5-3 Specific commands of '/cmps/<Component Name>/map/expup'

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>load</td>
<td>Updates all modules of &lt;Component Name&gt; to the latest version. Refer to section 5.2.1.1 for more information.</td>
</tr>
<tr>
<td>stop</td>
<td>Cancels all update command to &lt;Component Name&gt;. Refer to section 5.2.4.1 for more information.</td>
</tr>
</tbody>
</table>

Example

'`show /cmps/<Component Name>/map/expup`' command enables you to view ExpressUpdate information of the component <Component Name>.

```
-> show /cmps/Server01/map/expup
ufip=/cmps/Server01/map/expup
ufit=expup
Targets:
   mods       (#1)
   uninstalled (#2)
modules
Properties:
   RepositoryLocation=LOCAL
   ExpUpStatus=NOT_LATEST
   UpdatePkgLatestDownloaded=2011/04/01 01:00:00
   NextDownloadDate=2011/04/02 01:00:00
   Model=Express5800/110Ge-S
   OsVersion=Microsoft Windows Vista Business Service Pack 2 x64
   NextUpdateDate=2011/05/01 12:00
Verbs:
   cd
   exit
   help
   show
   load
   stop
```

#1: 'mods' element is deprecated. Use '/cmps/Server01/map/expup/modules/supportedmods/managedmods' instead of this.
#2: 'uninstalled' element is deprecated. Use '/cmps/Server01/map/expup/modules/supportedmods/uninstalledmods' instead of this.

5.1.1.2 Information of groups

'show' command at the below target enables you to view ExpressUpdate information about the specified group.

- '/grps/grpset/<Group Name>/expup'

Properties and specific commands of this target are as follows.

Table 5-4 Properties of '/grps/grpset/<Group Name>/expup'

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays ExpressUpdate statuses. Each status is listed in Table 5-2.</td>
</tr>
<tr>
<td>RepositoryLocation</td>
<td>read</td>
<td>Displays a location of the repository. (local/remote)</td>
</tr>
<tr>
<td>UpdatePkgLatestDownloaded</td>
<td>read</td>
<td>Displays the most recent time when the repository was updated. If updating has never been executed, it shows 'None'.</td>
</tr>
<tr>
<td>NextDownloadDate</td>
<td>read</td>
<td>Displays time and date of the next repository update if it is scheduled.</td>
</tr>
</tbody>
</table>
Table 5-5 Specific commands of '/grps/grpset/<Group Name>/expup'

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>load</td>
<td>Updates all modules of components belonging to the specified group to the latest version. Refer to section 5.2.1.3 for more information.</td>
</tr>
<tr>
<td>stop</td>
<td>Cancels all update command to &lt;Group Name&gt;. Refer to section 5.2.4.2 for more information.</td>
</tr>
</tbody>
</table>

Example
'show /grps/grpset/<Group Name>/expup' command enables you to view ExpressUpdate information of the group <Group Name>.

```bash
-> show /grps/grpset/root/expup
ufip=/grps/grpset/root/expup
ufit=expup
Targets:
   mods   (#1)
   modules
Properties:
   RepositoryLocation=LOCAL
   ExpUpStatus=NOT_LATEST
   UpdatePkgLatestDownloaded=2011/04/01 01:00:00
   NextDownloadDate=2011/04/02 01:00:00
Verbs:
   cd
   exit
   help
   show
   load
   stop
```

#1: 'mods' element is deprecated. Use '/grps/grpset/root/expup/modules/supportedmods/managedmods' instead of this.

5.1.2 Displaying a list of modules supporting automatic update
5.1.2.1 Information of components
A list of modules supporting automatic update for the specified component is displayed at the below target.

- '/cmps/ <Component Name>/map/expup/modules/supportedmods/managedmods

Property of this target is as follow. There is no specified command for this target.

Table 5-6 Property of '/cmps/ <Component Name>/map/expup/modules/supportedmods/managedmods'

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of modules managed by ExpressUpdate.</td>
</tr>
</tbody>
</table>
Example
'show /cmps/<Component Name>/map/expup/mods' command enables you to view a list of modules for the specified component.<Component Name>.

```
-> show /cmps/Server01/map/expup/modules/supportedmods/managedmods
ufip=/cmps/Server01/map/expup/modules/supportedmods/managedmods
ufit=managedmods
Targets:
"System BIOS"
"BMC Firmware"
"NEC ExpressUpdate Agent"
Properties:
  EntryCount=3
Verbs:
  cd
  exit
  help
  show
```

5.1.2.2 Information of groups
A list of modules supporting automatic update for components belongings to the specified group is displayed at the below target.

- /grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods

Property of this target is as follow. There is no specified command for this target.

Table 5-7 Property of '/grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods'

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of modules managed by ExpressUpdate.</td>
</tr>
</tbody>
</table>

Example
'show /grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods' command enables you to view a list of modules for components belongings to the specified group.<Group Name>.

```
-> show /grps/grpset/root/expup/modules/supportedmods/managedmods
ufip=/grps/grpset/root/expup/modules/supportedmods/managedmods
ufit=managedmods
Targets:
"System BIOS"
"BMC Firmware"
"NEC ExpressUpdate Agent"
Properties
  EntryCount=3
Verbs
  cd
  exit
  help
  show
```
5.1.3 Displaying a list of modules not supporting automatic update

5.1.3.1 Information of components

A list of modules not supporting automatic update for the specified component is displayed at the below target.

- `/cmps/<Component Name>/map/expup/modules/unsupportedmods`

Property of this target is as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of modules managed by ExpressUpdate.</td>
</tr>
</tbody>
</table>

Example

`show /cmps/<Component Name>/map/expup/unsupportedmods` command enables you to view a list of modules for the specified component<Component Name>.

```bash
-> show /cmps/Server01/map/expup/modules/unsupportedmods
ufip=/cmps/Server01/map/expup/modules/unsupportedmods
ufit=unsupportedmods
Targets:
"System BIOS"
"BMC Firmware"
"NEC ExpressUpdate Agent"
Properties:
EntryCount=3
Verbs:
cd
exit
help
show
```
5.1.3.2 Information of groups
A list of modules not supporting automatic update for the specified group is displayed at the below target.

- `/grps/grpset/<Group Name>/expup/modules/unsupportedmods`

Property of this target is as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of modules managed by ExpressUpdate.</td>
</tr>
</tbody>
</table>

**Example**

`show/grps/grpset/<Group Name>/expup/modules/unsupportedmods` command enables you to view a list of modules for the specified group `<Group Name>`.

```
-> show /grps/grpset/root/expup/modules/unsupportedmods
ufip=/grps/grpset/root/expup/modules/unsupportedmods
ufit=unsupportedmods
Targets:
"System BIOS"
"BMC Firmware"
"NEC ExpressUpdate Agent"
Properties:
 EntryCount=3
 Verbs:
  cd
  exit
  help
  show
```
5.1.4 Information of module supporting automatic update
5.1.4.1 Information of components

Information about each module supporting automatic update is displayed at the below target.

- `/cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>`

Properties and specific commands of this target are as follows.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>read</td>
<td>Displays a name of this module.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays ExpressUpdate status of this module. Each status is listed in Table 5-2.</td>
</tr>
<tr>
<td>Version</td>
<td>read</td>
<td>Displays current version of this module.</td>
</tr>
<tr>
<td>Severity</td>
<td>read</td>
<td>Displays a severity of the update. Severities are as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· HIGH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· MEDIUM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· LOW</td>
</tr>
<tr>
<td>LatestApplyResult</td>
<td>read</td>
<td>Displays result of the most recent update. If updates have never been executed, it shows hyphen (-).</td>
</tr>
</tbody>
</table>

Table 5-11 Specific commands of `/cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>`

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>load</td>
<td>Updates or installs the specified module. Refer to section 5.2.1.2 and 5.2.2 for more information.</td>
</tr>
<tr>
<td>delete</td>
<td>Uninstalls the specified module. Refer to section 5.2.3 for more information.</td>
</tr>
</tbody>
</table>

Example

`show /cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>` command enables you to view information about the module<Module Name> of the component<Component Name>.

```
-> show /cmps/Server01/map/expup/modules/supportedmods/managedmods/"NEC ExpressUpdate Agent"
ufip=/cmps/Server01/map/expup/modules/supportedmods/managedmods/"NEC ExpressUpdate Agent"
ufit="NEC ExpressUpdate Agent"
Targets:
  uppkg
Properties:
  Module= NEC ExpressUpdate Agent
  ExpUpStatus=NOT_LATEST
  Version=3.0
  Severity=HIGH
  LatestApplyResult=2011/04/01 12:00:00

  Applying the update package has been completed successfully.
  (NEC ExpressUpdate Agent: 1.0 -> 2.0)
```

Verbs:

  cd
  exit
  help
  show
  load
  delete
5.1.4.2 Information of groups

Information about each module is displayed at the below target.

- `/grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods/<Module Name>`

Properties and specific commands of this target are as follows.

**Table 5-12 Properties of `/grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods/<Module Name>`**

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>read</td>
<td>Displays a name of this module.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>Displays ExpressUpdate status of this module.</td>
</tr>
</tbody>
</table>

Each status is listed in Table 5-2.

**Table 5-13 Specific command of `/grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods/<Module Name>`**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>load</td>
<td>Updates the specified module.</td>
</tr>
<tr>
<td></td>
<td>Refer to section 5.2.1.4 for more information.</td>
</tr>
</tbody>
</table>

**Example**

`show /grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods/<Module Name>` command enables you to view information about the module `<Module Name>` of the component `<Group Name>`.

```
-> show /grps/grpset/root/expup/modules/supportedmods/managedmods":"NEC ExpressUpdate Agent"
ufip="/grps/grpset/root/expup/modules/supportedmods/managedmods":"NEC ExpressUpdate Agent"
ufit="NEC ExpressUpdate Agent"
Targets:
  uppkg
Properties:
  Module= NEC ExpressUpdate Agent
  ExpUpStatus=NOT_LATEST
  Version=3.0
  Severity=HIGH
  LatestApplyResult=2011/04/01 12:00:00
  Applying the update package has been completed successfully.
  (NEC ExpressUpdate Agent: 1.0 -> 2.0)

Verbs:
  cd
  exit
  help
  show
  load
  delete
```
5.1.5 Information of module not supporting automatic update

5.1.5.1 Information of components

Information about each module not supporting automatic update is displayed at the below target.

- `/cmps/<Component Name>/map/expup/modules/unsupportedmod/<Module Name>

Properties and specific commands of this target are as follows.

### Table 5-14 Properties of `/cmps/<Component Name>/map/expup/modules/unsupportedmod/<Module Name>`

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>read</td>
<td>Displays a name of this module.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>An ExpressUpdate status of the modules. Statuses are as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BLUE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• RED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BLACK</td>
</tr>
<tr>
<td>Version</td>
<td>read</td>
<td>Displays current version of this module.</td>
</tr>
<tr>
<td>Severity</td>
<td>read</td>
<td>Displays a severity of the update. Severities are as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• HIGH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MEDIUM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• LOW</td>
</tr>
<tr>
<td>LatestApplyResult</td>
<td>read</td>
<td>Displays result of the most recent update. If updates have never been executed, it shows hyphen (-).</td>
</tr>
</tbody>
</table>

### Table 5-15 Specific commands of `/cmps/<Component Name>/map/expup/modules/unsupportedmod/<Module Name>`

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td>Changes ExpUpStatus of the module. &quot;BLUE&quot;, &quot;RED&quot; and &quot;BLACK&quot; are able to be specified.</td>
</tr>
</tbody>
</table>

**Example**

`show /cmps/<Component Name>/map/expup/modules/unsupportedmod/<Module Name>` command enables you to view information about the module `<Module Name>` of the component `<Component Name>`.

```
-> show /cmps/Server01/map/expup/modules/unsupportedmod/"BMC Firmware"
ufit=/cmps/Server01/map/expup/modules/unsupportedmod/"BMC Firmware"
ufit="BMC Firmware"
Targets:
  uppkgs
Properties:
  Module=BMC Firmware
  ExpUpStatus=BLUE
  Version=3.0
  Severity=HIGH
  LatestApplyResult=-
Verbs:
  cd
  exit
  help
  show
  set
```
5.1.5.2 Information of groups

Information about each module is displayed at the below target.

- `/grps/grpset/<Group Name>/expup/modules/unsupportedmods/<Module Name>`

Properties and specific commands of this target are as follows.

Table 5-16 Properties of `/grps/grpset/<Group Name>/expup/modules/unsupportedmods/<Module Name>`

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>read</td>
<td>Displays a name of this module.</td>
</tr>
<tr>
<td>ExpUpStatus</td>
<td>read</td>
<td>An ExpressUpdate status of the module. Statuses are as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BLUE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• RED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BLACK</td>
</tr>
</tbody>
</table>

Example

`show /grps/grpset/<Group Name>/expup/modules/unsupportedmods/<Module Name>` command enables you to view information about the module<Module Name> of the component<Group Name>.

```
-> show /grps/grpset/root/expup/modules/unsupportedmods/"BMC Firmware"
ufip=/grps/grpset/root/expup/modules/unsupportedmods/"BMC Firmware"
ufit="BMC Firmware"
Targets:
  uppkg
Properties:
  Module=BMC Firmware
  ExpUpStatus=BLUE
  Severity=HIGH
Verbs:
  cd
  exit
  help
  show
```
5.2 Update, install and uninstall

5.2.1 Update modules supporting automatic update

5.2.1.1 For the specified component

`load` command at the below target enables you to update all modules to the latest version.

- `/cmps/<Component Name>/map/expup`

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-reboot</td>
<td>Enables ExpressUpdate to reboot the system if the update requires it.</td>
</tr>
</tbody>
</table>

Table 5-17 Specific option of `/cmps/<Component Name>/map/expup`

CHECK:

- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

Example

`load /cmps/<Component Name>/map/expup` command enables you to update all modules of the specified component <Component Name>.

```bash
-> load -reboot /cmps/Server01/map/expup
COMMAND COMPLETED

Updating was started. Run the show command to the each module element and you can confirm this command's progress from ExpUpStatus property. Run the show command to the each module element and you can confirm this command's result from LatestApplyResult property.
```

| ufip=/cmps/Server01/map/expup/modules/supportedmods/managedmods/"NEC ExpressUpdate Agent" |
| ufip="NEC ExpressUpdate Agent" |
| Targets |
| uppkg |
| Properties |
| Module= NEC ExpressUpdate Agent |
| ExpUpStatus=UNDER_INSTALLATION (5%) |
| Version=2.0 |
| Severity=HIGH |
| LatestApplyResult=2011/04/01 12:00:00 |

```
Applying the update package has been completed successfully.
(NEC ExpressUpdate Agent: 1.0 -> 2.0)
```

TIP:

- If the update is in progress, ExpUpStatus property of `show /cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>` command shows the progress.

```bash
-> show /cmps/Server01/map/expup/modules/supportedmods/managedmods/"NEC ExpressUpdate Agent" |
```

Verbose

cd
exit
help
show
load
delete

45
TIP:
- If the update has been finished, the result is shown in LatestApplyResult property of `show /cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>` command.

LatestApplyResult=2011/04/01 12:00:00
Applying the update package has been completed successfully.
(NEC ExpressUpdate Agent: 2.0 → 3.0)

5.2.1.2 For the specified component and module
'load' command at the below target enables you to update the module `<Module Name>` of the component `<Component Name>` using the specified update package.
- `/cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>`
Specific option of this command is as follow.

Table 5-18 Specific options of `/cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>`

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-source</td>
<td>Specifies the update package.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
<tr>
<td>-reboot</td>
<td>Enables ExpressUpdate to reboot the system if the update requires it.</td>
</tr>
</tbody>
</table>

CHECK:
- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

TIP:
- UFiT of the update package that is referred by `-source` option is shown in the result of `show /cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>/uppkg`s` command.

Example
'load /cmps/ <Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name> ' command enables you to update the module `<Module Name>` of the component `<Component Name>` using the specified update package.

`> load -force -source "2.0" /cmps/Server01/map/expup/modules/supportedsmods/managedmods/"NEC ExpressUpdate Agent" COMMAND COMPLETED`
Updating was started. Run the show command to the each module element and you can confirm this command's progress from ExpUpStatus property. Run the show command to the each module element and you can confirm this command's result from LatestApplyResult property.

/cmps/Server01/map/expup/modules/supportedsmods/managedmods/"NEC ExpressUpdate Agent"
5.2.1.3 For the specified group

The `load` command at the below target enables you to update all modules of components belonging to the specified group `<Group Name>`. Refer to the section 5.1.4.1 for more information about this target.

- `/grps/grpset/<Group Name>/expup`

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-exclude</td>
<td>Specifies components or groups that are not included in this operation.</td>
</tr>
<tr>
<td>-reboot</td>
<td>Enables ExpressUpdate to reboot the system if the update requires it.</td>
</tr>
</tbody>
</table>

**Example**

The `load /grps/grpset/<Group Name>/expup` command enables you to update all modules of components belonging to the specified group `<Group Name>`.

```bash
-> load  /exclude cmp=Server02 /grps/grpset/root/expup
COMMAND COMPLETED
Updating was started. Run the show command to the each module element and you can confirm this command's progress from ExpUpStatus property. Run the show command to the each module element and you can confirm this command's result from LatestApplyResult property.
```

5.2.1.4 For the specified group and module

The `load` command at below targets enables you to update the specified module `<Module Name>` of components belonging to the specified group `<Group Name>`. Refer to the section 5.1.1.2 for more information about this target.

- `/grps/grpset/<Group Name>/expup/modules/supportedmods/managedmods/<Module Name>`

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-exclude</td>
<td>Specifies components or groups that are not included in this operation.</td>
</tr>
<tr>
<td>-reboot</td>
<td>Enables ExpressUpdate to reboot the system if the update requires it.</td>
</tr>
</tbody>
</table>

**CHECK**:
- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

**Example**

The `load /grps/grpset/<Group Name>/expup` command enables you to update the specified module `<Module Name>` of components belonging to the specified group `<Group Name>`.

```bash
-> load  /exclude cmp=Server02 /grps/grpset/root/expup/modules/supportedmods/managedmods/<Module Name>/NEC ExpressUpdate Agent
COMMAND COMPLETED
Updating was started. Run the show command to the each module element and you can confirm this command's progress from ExpUpStatus property. Run the show command to the each module element and you can confirm this command's result from LatestApplyResult property.
```

/cmps/ Server01/map/expup/modules/supportedmods/managedmods/"NEC ExpressUpdate Agent"
5.2.2 Install commands

The `load` command at below target enables you to install a module. Refer to the section 5.1.4.1 for more information about this target.

- `/cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>`

Specific option of this command is as follow.

Table 5-21 Specific options of `/cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>`

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-source</code></td>
<td>Specifies the update package.</td>
</tr>
<tr>
<td><code>-reboot</code></td>
<td>Enables ExpressUpdate to reboot the system if the installation requires it.</td>
</tr>
</tbody>
</table>

CHECK:
- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

Example

`load /cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>` command enables you to install a module `<Module Name>` to the specified component `<Component Name>`.

```
-> load -source "4.4.1" /cmps/Server01/map/expup/modules/supportedsmods/managedmods/"ESMPRO/ServerAgent Ver.4.4"
COMMAND COMPLETED
```

Updating was started. Run the show command to the each module element and you can confirm this command's progress from ExpUpStatus property. Run the show command to the each module element and you can confirm this command's result from LatestApplyResult property.

TIP:
- UFiT of the update package that is referred by `-source` option is shown in the result of `show /cmps/<Component Name>/map/expup/modules/supportedsmods/managedmods/<Module Name>/uppkgs` command.
5.2.3 Uninstall commands
'delete' command enables you to uninstall a module. Refer to the section 5.1.4.1 for more information about this target.

- /cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>
Specific option of this command is as follow.

Table 5-22 Specific option of '/cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>'

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-reboot</td>
<td>Enables ExpressUpdate to reboot the system if the uninstallation requires it.</td>
</tr>
</tbody>
</table>

CHECK:
- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

Example
'load /cmps/<Component Name>/map/expup/modules/supportedmods/managedmods/<Module Name>' command enables you to uninstall a module<Module Name> from the specified component<Component Name>.

```bash
-> delete /cmps/Server01/map/expup/modules/supportedmods/managedmods/"NEC ExpressUpdate Agent"
COMMAND COMPLETED
Uninstall was started. Run the show command to the each module element and you can confirm this command's progress from ExpUpStatus property. Run the show command to the each module element under the uninstalledmods element and you can confirm this command's result from LatestApplyResult property.
```
5.2.4 Cancelling update commands

5.2.4.1 For components

'stop' command at below targets enables you to cancel updates. Refer to the section 5.1.1.1 for more information about this target.

- /cmps/<Component Name>/map/expup

There is no specific option for this command.

CHECK:
- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

TIP:
- This command effects only waiting tasks. A task which has been started is unable to be cancelled.

Example
'stop /cmps/<Component Name>/map/expup' command enables you to cancel updates of the specified component <Component Name>. If this command is successful, a list of cancelled tasks is shown.

```
-> stop/cmps/Server01/map/expup
COMMAND COMPLETED
/cmps/Server01/map/expup/mods/"NEC ExpressUpdate Agent"
Cancel of update process was executed.
```

5.2.4.2 For groups

'stop' command at below targets enables you to cancel updates. Refer to the section 5.1.1.2 for more information about this target.

- /grps/grpset/<Group Name>/expup

There is no specific option for this command.

CHECK:
- In the case of the user authority is Operator, "Install Update Packages" of the user level must be enabled.

Example
'stop /grps/grpset/<Group Name>/expup' command enables you to cancel updates of components belonging to the specified group <Group Name>. If this command is successful, a list of cancelled tasks is shown.

```
-> stop/grps/grpset/root/expup
COMMAND COMPLETED
/cmps/Server01/map/expup/mods/"BMC Firmware"
/cmps/Server01/map/expup/mods/"NEC ExpressUpdate Agent"
/cmps/Server02/map/expup/mods/"NEC ExpressUpdate Agent"
Cancel of update process was executed.
```
5.2.5 Update modules not supporting automatic update

Procedures for applying update packages not supporting automatic update are as below.

1. Copy the update package to be applied as an update package archive on your client PC.
2. Pick up the update package from an update package archive and apply it.

How to apply the update package is described in the readme file included in the update package.

5.2.5.1 For the specified component and module

'dump' command at the below target enables you to copy update packages related to specified component and module.

- /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of modules managed by ExpressUpdate.</td>
</tr>
<tr>
<td>Progress</td>
<td>read</td>
<td>If creating an update package archive is in execution, its progress is displayed.</td>
</tr>
</tbody>
</table>

Table 5-24 Specific command of '/cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs'

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dump</td>
<td>Archives update packages related to the specified module into an update package archive.</td>
</tr>
<tr>
<td>stop</td>
<td>Cancels creating an update package archive if it is in execution.</td>
</tr>
</tbody>
</table>

Table 5-25 Specific options of 'dump /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs'

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination &lt;path&gt;</td>
<td>Specifies the path to the directory or zip file where the update package archive to be copy.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Example

'dump /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>' command enables you to archive update packages into an update package archive. If starting command succeeds, update packages to be archived are listed.

```
$ dump -destination C:\temp /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs
COMMAND COMPLETED
"3.0"
"4.0"
"5.0"
```

The process to save update packages was started. To confirm its progress, execute 'show' command on update packages element and refer to Progress property.

C:\temp\UpdatePackageArchive_20120701.zip
5.2.5.2 For the specified component and update package
'dump' command at the below target enables you to copy update packages related to specified component and
update package.
- /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs/<PKG Version>

Table 5-26 Properties of '/cmps/<Component Name>/map/expup/modules/unsupportedmods/<
Module Name>/uppkgs/<PKG Version>'

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>read</td>
<td>Displays the name of firmware or software.</td>
</tr>
<tr>
<td>Version</td>
<td>read</td>
<td>Displays the version of the update package.</td>
</tr>
<tr>
<td>Release</td>
<td>read</td>
<td>Displays the release date of the update package.</td>
</tr>
<tr>
<td>Target</td>
<td>read</td>
<td>Displays the operating system name that the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>update package supports.</td>
</tr>
<tr>
<td>Model</td>
<td>read</td>
<td>Displays the model name of the component that</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the update package supports.</td>
</tr>
<tr>
<td>EstimatedTime</td>
<td>read</td>
<td>Displays the estimate of the time to install the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>update package.</td>
</tr>
<tr>
<td>Reboot</td>
<td>read</td>
<td>Displays whether that the selected update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>package requires a reboot after the installation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NECESSITY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UNNECESSITY</td>
</tr>
<tr>
<td>Severity</td>
<td>read</td>
<td>Displays the severity of the update package.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• LOW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MIDDLE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• HIGH</td>
</tr>
<tr>
<td>ExpressUpdateSupported</td>
<td>read</td>
<td>Displays whether the update package supports</td>
</tr>
<tr>
<td></td>
<td></td>
<td>automatic updating by NEC ExpressUpdate.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SUPPORTED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UNSUPPORTED</td>
</tr>
<tr>
<td>DowngradeSupported</td>
<td>read</td>
<td>Displays whether the update package supports</td>
</tr>
<tr>
<td></td>
<td></td>
<td>downgrade.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SUPPORTED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UNSUPPORTED</td>
</tr>
<tr>
<td>MultipleModelSupported</td>
<td>read</td>
<td>Displays whether the update package supports</td>
</tr>
<tr>
<td></td>
<td></td>
<td>multiple mode of component.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SUPPORTED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UNSUPPORTED</td>
</tr>
<tr>
<td>OtherUsed</td>
<td>read</td>
<td>Displays whether the update package is in use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>by other NEC ESMPRO Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UNOCCUPIED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DURING USE</td>
</tr>
<tr>
<td>Progress</td>
<td>read</td>
<td>If archiving update packages is in execution,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>displays its progress.</td>
</tr>
</tbody>
</table>

Table 5-27 Specific command of '/cmps/<Component Name>/map/expup/modules/unsupported
mods/<Module Name>/uppkgs/<PKG Version>'

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dump</td>
<td>Archives update packages related to the specified module into an update package archive.</td>
</tr>
<tr>
<td>stop</td>
<td>Cancels creating an update package archive if it is in execution.</td>
</tr>
</tbody>
</table>
Table 5-28 Specific options of 'dump /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs/<PKG Version>'

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination &lt;path&gt;</td>
<td>Specifies the path to the directory or zip file where the update package archive to be copied.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Example

'dump /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs/<PKG Version>' command enables you to archive an update package into an update package archive. If starting command succeeds, an update package to be archived is listed.

```
> dump -destination C:\temp /cmps/<Component Name>/map/expup/modules/unsupportedmods/<Module Name>/uppkgs/"5.0"
COMMAND COMPLETED
"5.0"
The process to save update packages was started. To confirm its progress, execute 'show' command on update packages element and refer to Progress property.
C:\temp\UpdatePackageArchive_20120701.zip
```
5.3 Repository and Update package management

5.3.1 Repository settings

Repository settings are located at the following target:

- `/repository`

Properties and specific commands of this target are as follows.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RepositoryLocation</td>
<td>read/write</td>
<td>Displays and sets location of the repository. The valid values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• LOCAL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• REMOTE</td>
</tr>
<tr>
<td>UpdatePackageServerAddress</td>
<td>read</td>
<td>Displays address of update package server.</td>
</tr>
<tr>
<td>UpdatePkgLatestDownloaded</td>
<td>read</td>
<td>Displays the date when the repository downloaded update packages last.</td>
</tr>
<tr>
<td>TotalSizeOfUpdatePackage</td>
<td>read</td>
<td>Displays total size of update packages. The unit is MB.</td>
</tr>
<tr>
<td>FreeSpace</td>
<td>read</td>
<td>Displays free space of the disk which there is the preservation folder of the update packages.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td>Sets repository settings. This command enables you to modify location of the repository.</td>
</tr>
<tr>
<td>load</td>
<td>Downloads or adds update packages to repository.</td>
</tr>
<tr>
<td>stop</td>
<td>Cancels downloading update packages from the update package server.</td>
</tr>
</tbody>
</table>

Examples

'show' command enables you to view the repository settings.

```
> show /repository
ufip=/repository
ufit=repository
Targets:
     uppkgs
     localsetting
     remotesetting
Properties:
     RepositoryLocation=LOCAL
     UpdatePackageServerAddress=http://www.exifs.nec.co.jp/
     UpdatePkgLatestDownloaded=2011/04/01 12:00:00
     TotalSizeOfUpdatePackage=34.2MB
     FreeSpace=100GB
Verbs:
     cd
     exit
     help
     show
     load
     set
     stop
```

Examples

'set' command enables you to modify location of the repository.
When you modify location of the repository in local, run the following command.

`set/repository RepositoryLocation=LOCAL`

**Examples**

When you modify location of the repository in remote, run the following command in the remote repository settings have been set.

`set/repository RepositoryLocation=REMOTE`

**CHECK:**
- When you set `RepositoryLocation` in `REMOTE`, you should set remote repository settings beforehand. See 5.3.1.2 Remote repository settings.
- In the case of the user authority is Operator, "Change Environment Setting" of the user level must be enabled.

### 5.3.1.1 Local repository settings

You can view and change Manager PC repository settings. Local repository settings are located at the following target.

- `/repository/localsetting`

Properties and specific commands of this target are as follows.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RepositoryPassword</td>
<td>read/write</td>
<td>Sets the password to access repository. This is displayed in &quot;*****&quot;.</td>
</tr>
<tr>
<td>AutoUpdate</td>
<td>read/write</td>
<td>Sets the automatic updating of the repository. The valid values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• INVALID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• VALID – the repository downloads update packages.</td>
</tr>
<tr>
<td>UpdateInterval</td>
<td>read/write</td>
<td>Sets the schedule type when &quot;AutoUpdate&quot; is &quot;VALID&quot;. The valid values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DAILY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_SUNDAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_MONDAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_TUESDAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_WEDNESDAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_THURSDAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_FRIDAY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EVERY_SATURDAY</td>
</tr>
<tr>
<td>UpdateStartTime</td>
<td>read/write</td>
<td>Sets downloading time when &quot;AutoUpdate&quot; is &quot;VALID&quot;. The valid ranges are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 00:00 – 23:50 Time must be set by the 10 minute.</td>
</tr>
<tr>
<td>ProxyAddress</td>
<td>read/write</td>
<td>Sets the IP address of the proxy server.</td>
</tr>
<tr>
<td>ProxyPort</td>
<td>read/write</td>
<td>Sets the port number of the proxy server.</td>
</tr>
<tr>
<td>ProxyUser</td>
<td>read/write</td>
<td>Sets the user name with which the repository connects to the proxy server.</td>
</tr>
<tr>
<td>ProxyPassword</td>
<td>read/write</td>
<td>Sets the user password with which the repository connects to the proxy server. This is displayed in &quot;*****&quot;.</td>
</tr>
<tr>
<td>DefaultStatusIcon</td>
<td>read/write</td>
<td>Sets the default status icon of a downloaded module which does not support the automatic update. The valid ranges are:</td>
</tr>
</tbody>
</table>
Table 5-32 Specific commands of '/repository/localsetting'

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td>Sets local repository settings properties.</td>
</tr>
</tbody>
</table>

Examples
'show' command enables you to view local repository settings.

```
-> show /repository/localsetting
ufip=/repository/localsetting
utfit=localsetting
Targets:
Properties:
 RepositoryPassword=*****
 AutoUpdate=VALID
 UpdateInterval=EVERY_SATURDAY
 UpdateStartTime=23:50
 ProxyAddress=-
 ProxyPort=0
 ProxyUser=-
 ProxyPassword=-
 DefaultStatusIcon=BLUE
 Verbs:
 cd
 exit
 help
 show
 set
```

Examples
'set' command enables you to set one or more local repository settings properties. If there are multiple properties on the same command line, they must be separated by a space.

When you enable automatic downloading, run the following command.

```
set /repository/localsetting AutoUpdate=VALID
```

When you set the schedule type, run the following command.

```
set /repository/localsetting UpdateInterval=EVERY_SATURDAY
```

When you set downloading time, run the following command.

```
set /repository/localsetting UpdateStartTime=23:50
```

When you set the IP address and the port number of the proxy server on the same command line, run the following command.

```
set /repository/localsetting ProxyAddress=192.168.0.200 ProxyPort=8080
```

CHECK:
• In the case of the user authority is Operator, "Change Environment Setting" of the user level must be enabled.
5.3.1.2 Remote repository settings
Remote repository settings are located at the following target.
- /repository/remotesetting
Properties and specific commands of this target are as follows.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoteRepositoryAddress</td>
<td>read/write</td>
<td>Sets OS IP address of the other Manager PC.</td>
</tr>
<tr>
<td>RemoteRepositoryPort</td>
<td>read/write</td>
<td>Sets port number of the other Manager PC.</td>
</tr>
<tr>
<td>RemoteRepositoryPassword</td>
<td>read/write</td>
<td>Sets password that is configured in [RepositoryPassword] on the other Manager PC.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td>Sets remote repository settings properties.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-f</td>
<td>force</td>
</tr>
</tbody>
</table>

Examples
'show' command enables you to view remote repository settings.

```
> show /repository/remotesetting
ufip=/repository/remotesetting
ufit=remotesetting
Targets:
Properties:
    RemoteRepositoryAddress=192.168.0.100
    RemoteRepositoryPort=8080
    RemoteRepositoryPassword=*****
Verbs:
    cd
    exit
    help
    show
    set
```

Examples
'set' command enables you to set one or more remote repository settings properties. If there are multiple properties on the same command line, they must be separated by a space.

If you want to set repository location in remote, it is necessary to set RemoteRepositoryAddress, RemoteRepositoryPort, and RemoteRepositoryPassword properties.

```
set /repository/remotesetting RemoteRepositoryAddress=192.168.0.100 RemoteRepositoryPort=8080 RemoteRepositoryPassword=password
```

When the correct value is set, repository location is changed in remote.

CHECK:
- The following properties must be specified when repository location is set to remote.
  - RemoteRepositoryAddress
  - RemoteRepositoryPort
RemoteRepositoryPassword

In the case of the user authority is Operator, "Change Environment Setting" of the user level must be enabled.

Examples
When you add -force option, properties are set by force.

```
set-force /repository/remotesetting RemoteRepositoryAddress=192.168.0.111
```

5.3.2 Adding update packages to repository

'load' command at below targets enables you to add or download update packages.

- /repository

Specific command of this target is as follow. There is no property for this target.

Table 5-36 Specific command of '/repository'

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>load</td>
<td>Downloads update packages from update package server. Adds update packages to repository.</td>
</tr>
</tbody>
</table>

Specific option of this command is as follow.

Table 5-37 Specific options of '/repository'

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-wait</td>
<td>Watches it until the downloading of the update packages is completed. This option is valid only when you download without adding -source option.</td>
</tr>
<tr>
<td>-source</td>
<td>When you add the specified update package to repository, you specify the update package file path.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Examples

'load' command enables you to download update packages.

```
> load /repository
COMMAND COMPLETED

Downloading update packages. Run the show command to the repository and you can confirm the command's result from UpdatePkgLatestDownloaded property.
```

The download process will start and return to the input prompt. To confirm this command results, run 'show' command and confirm UpdatePkgLatestDownloaded property. This value is updated at the downloading end time.

```
UpdatePkgLatestDownloaded=2011/04/01 12:00:00
```

Examples

'load -wait' command enables you to watch it until downloading is completed.

```
> load -wait /repository

Downloading update packages.
Press Ctrl + D to stop monitoring.

............

Downloading update packages was finished. Please press Ctrl + D.
```

The download process will start and progress "..." will be displayed. When downloading is finished, an end message is displayed. After the end message, you need to press "Ctrl + D" key and return to the input prompt.
Examples
When you stop monitoring during downloading, press "Ctrl + D" key and return to the input prompt.

```
-> load -wait /repository
Downloading update packages.
Press Ctrl + D to stop monitoring.
```

Monitoring was stopped. Run the show command to the repository and you can confirm the command's result from UpdatePkgLatestDownloaded property.

Examples
'load -source' command enables you to add update packages to repository. Specify the path of the update package file. When you add -source option, you can add update package which is not necessary for components.

```
-> load -source C:\temp\package_634335370718834422.zip /repository
COMMAND COMPLETED

  Added specified update packages.
  "System BIOS Windows 4.0 02172011105550"
```

5.3.3 Removing update packages from repository
5.3.3.1 Removing all update packages
Removing all update packages is located at the following target.

- /repository/uppkg

Properties and specific commands of this target are as follows.

**Table 5-38 Property of '/repository/uppkg'**

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays a number of update packages managed by repository.</td>
</tr>
<tr>
<td>Progress</td>
<td>read</td>
<td>If archiving update packages is in execution, displays its progress.</td>
</tr>
</tbody>
</table>

**Table 5-39 Specific command of '/repository/uppkg'**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>delete</td>
<td>Removes all update packages.</td>
</tr>
<tr>
<td>dump</td>
<td>Archives update packages related to the specified module into an update package archive.</td>
</tr>
<tr>
<td>stop</td>
<td>Cancels creating an update package archive if it is in execution.</td>
</tr>
</tbody>
</table>

Specific option of this command is as follow.

**Table 5-40 Specific option of '/repository/uppkg'**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Examples
'delete' command enables you to remove all update packages.

```
-> delete /repository/uppkg
COMMAND COMPLETED

"NEC ExpressUpdate Agent_Windows_3.00_03022011171440" was deleted.
"NEC ExpressUpdate Agent_Linux_3.00_03f02011100740" was deleted.
```
5.3.3.2 Removing the update package

Removing update package is located at the following target.

- `/repository/uppkgs/<uppkg name>`

Specific command of this target is as follow. There is no property for this target.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>delete</td>
<td>Removes update package.</td>
</tr>
</tbody>
</table>

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-f</td>
<td>force</td>
</tr>
</tbody>
</table>

Examples

'delete' command enables you to remove the update package.

```
$ delete /repository/uppkgs""/SystemBIOS_Windows_4.0_02172011105550"
COMMAND COMPLETED
"SystemBIOS_Windows_4.0_02172011105550" was deleted.
```

---

**CHECK:**
- In the case of the user authority is Operator, "Change Environment Setting" of the user level must be enabled.

---

**TIP:**
- When you delete by force, update packages in use by other Manager PC can be removed.
5.3.4 Saving update packages

5.3.4.1 Saving update packages all together

Saving update packages all together is located at the following target.
- /repository/uppkgs

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination &lt;path&gt;</td>
<td>Specifies the path to the directory or zip file where the update package archive to be copy.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Example

'dump /repository/uppkgs' command enables you to archive all update packages into an update package archive. If starting command succeeds, all update packages to be archived are listed.

```
-> dump -destination C:\temp /repository/uppkg
COMMAND COMPLETED
"ExpressUpdate Agent/Linux_3.00_20110301010740"
"ExpressUpdate Agent/Windows_3.00_20110302171440"
The process to save update packages was started. To confirm its progress, execute 'show' command on update packages element and refer to Progress property.
C:\temp\UpdatePackageArchive_20120701.zip
```

5.3.4.2 Saving an update package

Saving an update package is located at the following target.
- /repository/uppkg/<uppkg name>

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination &lt;path&gt;</td>
<td>Specifies the path to the directory or zip file where the update package archive to be copy.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Example

'dump /repository/uppkg/<uppkg name>' command enables you to archive an update package into an update package archive. If starting command succeeds, an update package to be archived is listed.

```
-> dump -destination C:\temp /repository/uppkg/"ExpressUpdate Agent/Windows_3.00_20110302171440"
COMMAND COMPLETED
"ExpressUpdate Agent/Windows_3.00_20110302171440"
The process to save update packages was started. To confirm its progress, execute 'show' command on update package element and refer to Progress property.
C:\temp\UpdatePackageArchive_20120701.zip
```
5.3.5 Update packages information

5.3.5.1 Update packages information

Information about update packages is located at the following target:

- `/repository/uppkgs`

Specific option of this command is as follow.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-d</code> <code>-display euform=cmp</code></td>
<td>Displaying ExpressUpdate information by component.</td>
</tr>
<tr>
<td><code>-d</code> <code>-display euform=model</code></td>
<td>Displaying ExpressUpdate information by model.</td>
</tr>
</tbody>
</table>

**Examples**

'show' command enables you to view information about update packages.

```
-> show /repository/uppkgs
ufip=/repository/uppkgs
ufit=uppkgs
Targets:
"NEC ExpressUpdate Agent_Linux_3.00_20110310100740"
"NEC ExpressUpdate Agent_Windows_3.00_20110302171440"
Properties:
  EntryCount=2
Verbs:
  cd
  exit
  help
  show
  delete
```

'show' command with "-d euform=cmp" option enables you to view information about update packages by component.

```
-> show -d euform=cmp /repository/uppkgs
Server01
  "ExpressUpdate Agent_Linux_3.00_20110310100740"
Server02
  "ExpressUpdate Agent_Windows_3.00_20110302171440"
```

'show' command with "-d euform=model" option enables you to view information about update packages by model.

```
-> show -d euform=model /repository/uppkgs
"Express5800/110Ge-S"
  "ExpressUpdate Agent_Linux_3.00_20110310100740"
"Express5800/R120b-1"
  "ExpressUpdate Agent_Windows_3.00_20110302171440"
```
5.3.5.2 The update package information

Information about the update package is located at the following target.

- `/repository/uppkgs/<uppkg name>`

Property of this target is as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>read</td>
<td>Displays the name of firmware or software.</td>
</tr>
<tr>
<td>Version</td>
<td>read</td>
<td>Displays the version of the update package.</td>
</tr>
<tr>
<td>Release</td>
<td>read</td>
<td>Displays the release date of the update package.</td>
</tr>
<tr>
<td>Target</td>
<td>read</td>
<td>Displays the operating system name that the update package supports.</td>
</tr>
<tr>
<td>Model</td>
<td>read</td>
<td>Displays the model name of the component that the update package supports.</td>
</tr>
<tr>
<td>EstimatedTime</td>
<td>read</td>
<td>Displays the estimate of the time to install the update package.</td>
</tr>
<tr>
<td>InstallEstimatedTime</td>
<td>read</td>
<td>Displays the estimate of the time to install, when the module supports installation.</td>
</tr>
<tr>
<td>UninstallEstimatedTime</td>
<td>read</td>
<td>Displays the estimate of the time to uninstall, when the module supports uninstallation.</td>
</tr>
</tbody>
</table>
| Reboot            | read   | Displays whether the selected update package requires a reboot after the installation.  
|                   |        | • NECESSITY  
|                   |        | • UNNECESSITY                                                                  |
| Severity          | read   | Displays the severity of the update package.                              |
|                   |        | • LOW  
|                   |        | • MEDIUM  
|                   |        | • HIGH                                                                       |
| ExpressUpdateSupported | read | Displays whether the update package supports automatic updating by NEC ExpressUpdate.  
|                   |        | • SUPPORTED  
|                   |        | • UNSUPPORTED                                                                 |
| DowngradeSupported | read | Displays whether the update package supports downgrade.                    |
|                   |        | • SUPPORTED  
|                   |        | • UNSUPPORTED                                                                 |
| MultipleModelSupported | read | Displays whether the update package supports multiple mode of component.  
|                   |        | • SUPPORTED  
|                   |        | • UNSUPPORTED                                                                 |
| OtherUsed         | read   | Displays whether the update package is in use by other NEC ESMPRO Manager.  
|                   |        | • UNOCCUPIED  
|                   |        | • DURING USE                                                                 |
Examples
'show' command enables you to view information about the update package.

```bash
-> show /repository/uppkg/"NEC ExpressUpdate Agent_Windows_3.00_20110302171440"
ufp=/repository/uppkg/"NEC ExpressUpdate Agent_Windows_3.00_20110302171440"
ufit="NEC ExpressUpdate Agent_Windows_3.00_20110302171440"
Targets:
   readme
Properties:
   Module=NEC ExpressUpdate Agent
   Version=3.00
   Target=Windows Server 2003 Enterprise Edition
      Windows Server 2003 R2 Enterprise Edition
      Windows Server 2003 R2 Standard Edition
   Model=Express5800/E120b-1
   Express5800/T120a-M
   EstimatedTime=180s
   InstallEstimatedTime=120s
   UninstallEstimatedTime=300s
   Reboot=UNNECESSITY
   Severity=-
   ExpressUpdateSupported=SUPPORTED
   DowngradeSupported=UNSUPPOR TED
   MultipleModelSupported=SUPPORTED
   OtherUsed=UNOCCUPIED
Verbs:
   cd
   exit
   help
   show
   delete
```
5.3.5.3 Readme file information

After saving the readme file, you can view the readme file written notes about the update package. Information about the readme file in the update package is located at the following target:

- `/repository/uppkg/<uppkg name>/readme`

Specific command of this target is as follow. There is no property for this target.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dump</td>
<td>Saves the readme file to specified directory.</td>
</tr>
</tbody>
</table>

**Table 5-47 Specific command of '/repository/uppkg/<uppkg name>/readme'**

**Table 5-48 Specific option of '/repository/uppkg/<uppkg name>/readme'**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination</td>
<td>Specifies the directory to save the readme file. Required option.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

---

**CHECK:**

- The readme file name depends on the update package. You can't specify the readme file name when you save it.

**Examples**

'dump' command enables you to save the readme file to specified directory. Adds -destination option and specifies the path of the directory. When you add -force option, the readme file is overwritten.

```
-> dump -destination C:\Temp /repository/uppkg/*NEC ExpressUpdate Agent_Linux_3.00_20110310100740*/readme
COMMAND COMPLETED
C:\Temp\readme_en.txt
```

When the command succeeds, the readme file path is displayed.

```
C:\Temp\readme_en.txt
```
Chapter 6  Log management

6.1  Logging
You can get the following logs.
- NEC ESMPRO Manager application log
- NEC ExpressUpdate Agent log

6.1.1  Application log list
Information of NEC ESMPRO Manager application logs is located at the following target.
/\logs
Properties and specific commands of this target are as follows.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EntryCount</td>
<td>read</td>
<td>Displays the number of application log.</td>
</tr>
<tr>
<td>MaxEntryCount</td>
<td>read</td>
<td>Displays the maximum number of application log.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dump</td>
<td>Saves NEC ESMPRO Manager application log to the specified directory. File is saved in zip format. When a file name is not included in the path, LOG.zip file is saved.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination</td>
<td>Specifies the destination path of the directory or the file name. Absolute path and relative path can be specified. Required option.</td>
</tr>
<tr>
<td>-f</td>
<td>-force When there is a same name file in the destination directory, the file is overwritten.</td>
</tr>
</tbody>
</table>

Examples
'show' commands enable you to view the list of NEC ESMPRO Manager application log.

```
-> show /logs
ufip=/logs
ufit=logs
Targets:
  log0
  log1
  log2
  log3
  log4
  log5
  log6
  log7
  log8
  log9
  log10
  ...
  log74
Properties:
  EntryCount=75
  MaxEntryCount=2000
Verbs:
  cd
  exit
  help
  show
  dump
```
**Examples**
'dump' commands enable you to save application logs to the specified directory. Add `-destination` option, and specify the directory path to save file.

The following shows an example.

```
    -> dump -destination C:\temp
    COMMAND COMPLETED
    C:\temp\LOG.zip
```

If you want to save the application log file any name, specify the zip file name to the path and execute commands.

```
dump -destination C:\temp\application_log.zip /logs
```

When you add `-force` option, the zip file is overwritten.

```
dump -destination C:\temp\application_log.zip -force /logs
```

### 6.1.2 Application log

Information of NEC ESMPRO Manager application log is located at the following target.

- `/logs/<log>`

Property of this target is as follow. There is no specified command for this target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>read</td>
<td>Displays the type of application log.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• INFORMATION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WARNING</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ERROR</td>
</tr>
<tr>
<td>ComponentName</td>
<td>read</td>
<td>Displays the component name.</td>
</tr>
<tr>
<td>IpAddress</td>
<td>read</td>
<td>Displays the OS IP address of the component.</td>
</tr>
<tr>
<td>BmcIpAddress</td>
<td>read</td>
<td>Displays the BMC IP address of the component.</td>
</tr>
<tr>
<td>Date</td>
<td>read</td>
<td>Displays the date when the application log was registered.</td>
</tr>
<tr>
<td>User</td>
<td>read</td>
<td>Displays the user who executed this operation.</td>
</tr>
<tr>
<td>Contents</td>
<td>read</td>
<td>Displays the application log.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Examples**

'show' commands enable you to view the NEC ESMPRO Manager application log.

```
    -> show /logs/log10
    ulp=/logs/log10
    ulif=log10
    Targets:
    Properties:
        Type=INFORMATION
        ComponentName=Server01
        IpAddress=192.168.0.2
        BmcIpAddress=192.168.0.3
        Date=2011/03/01 12:00:00
        User=-
        Contents=The component was added.
    Verbs:
        cd
        exit
        help
        show
```
6.1.3 NEC ExpressUpdate Agent log

Information of NEC ExpressUpdate Agent log is located at the following target.

- /cmps/<component name>/map/agtlogs/expupagtlog

**IMPORTANT:**
- In <component name>, specify the component that the update function via NEC ExpressUpdate Agent is effective.
- To get NEC ExpressUpdate Agent log, NEC ExpressUpdate Agent must be running.

Specific command of this target is as follow. There is no property for this target.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dump</td>
<td>Saves NEC ExpressUpdate Agent log to the specified directory. File is saved in zip format.</td>
</tr>
</tbody>
</table>

Table 6-5 Specific commands of '/cmps/<component name>/map/agtlogs/expupagtlog'

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-destination</td>
<td>Specifies the destination path of the directory or the file name. Absolute path and relative path can be specified. Required option.</td>
</tr>
<tr>
<td>-f</td>
<td>-force</td>
</tr>
</tbody>
</table>

Examples

'show' commands enable you to view NEC ExpressUpdate Agent log.

```
-> show /cmps/Server01/map/agtlogs/expupagtlog
ufip=/cmps/Server01/map/agtlogs/expupagtlog
ufit=expupagtlog
Targets:
Properties:
Verbs:
  cd
  exit
  help
  show
  dump
```

'dump' commands enable you to save NEC ExpressUpdate Agent log to the specified directory. Add -destination option, and specify the directory path to save file.

```
-> dump -destination C:\temp /cmps/Server01/map/agtlogs/expupagtlog
COMMAND COMPLETED
C:\temp\eualog1.zip
```

If you want to save NEC ExpressUpdate Agent log file any name, specify the zip file name to the path and execute commands.

```
dump -destination C:\temp\eualog.zip /cmps/Server01/map/agtlogs/expupagtlog
```

When you add -force option, the zip file is overwritten.

```
dump -destination C:\temp\eualog.zip-force /cmps/Server01/map/agtlogs/expupagtlog
```
7.1 Error message

<table>
<thead>
<tr>
<th>Error Message</th>
<th>What to do</th>
</tr>
</thead>
<tbody>
<tr>
<td>User name or password is not correct.</td>
<td>Please confirm the user name and the password, and please input again.</td>
</tr>
<tr>
<td>Failed to connect to NEC ESMPRO Manager.</td>
<td>n   Windows OS: Please confirm that the NEC ESMPRO Manager service</td>
</tr>
<tr>
<td></td>
<td>(ESMPRO/SM Common Component) is running.</td>
</tr>
<tr>
<td></td>
<td>Linux OS: Please confirm that the following process is running by 'ps aux'</td>
</tr>
<tr>
<td></td>
<td>command. &quot;/opt/nec/es_manager/jre/bin/java com.nec.jp.dianascopc.CoreServer&quot;</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CLI</td>
<td>Command Line Interface</td>
</tr>
</tbody>
</table>
| BMC                      | Baseboard Management Controller  
The management controller to manage the interface between system management software and platform hardware.                                                                 |
| DMTF                     | Distributed Management Task Force  
An industry standard body that defines WBEM standards for the industry.                                                                                                                                   |
| SMASH                    | Systems Management Architecture for Server Hardware  
A suite of specifications that deliver industry standard protocols to increase productivity of the management of a data center.                                                                                |
| Address Space            | A set of instances that represent managed element or provided functions. These instances are accessed using a unique path.                                                                                       |
| Managed Element          | An instance of managed element or provided function on NEC ESMPRO Manager. It constructs the address space.                                                                                                   |
| UFIT                     | User Friendly instance Tag  
User-friendly identifier for a specific instance of a CIM class.                                                                                                                                           |
| UFiP                     | User Friendly instance Path  
The unique path to an instance formed by concatenating the UFiTs of each instance from the root instance to the terminating instance.                                                             |
| Component                | An object that is managed by NEC ESMPRO Manager.                                                                                                                                                           |
| NEC ExpressUpdate        | The function that manages versions of modules like firmware and software on the managed server and that updates the modules.  
Available since NEC ESMPRO Manager Ver5.1.                                                                                                      |
| NEC ExpressUpdate Agent  | Software that provides NEC ExpressUpdate function. It is installed on the managed component and communicates with NEC ESMPRO Manager.                                                                     |
| Module                   | The module like firmware and software.                                                                                                                                                                     |
| Update package           | The package which the update tool of the module is included in.                                                                                                                                           |
| Update package server    | The update package server from where the repository downloads update packages.                                                                                                                              |
| Repository               | The component of the NEC ESMPRO Manager for NEC ExpressUpdate that downloads update packages from the update package server and manages them.  
Available since NEC ESMPRO Manager Ver5.1.                                                                                                       |
Chapter 9  Appendix

9.1 XML Schema

9.1.1 XML Schema for request file

XML schema of the request XML is as below.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<!-- command request -->
<xs:element name="request">
  <xs:complexType>
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element name="cd" type="CommonRequestType"/>
      <xs:element name="create" type="CommonRequestType"/>
      <xs:element name="delete" type="CommonRequestType"/>
      <xs:element name="dump" type="CommonRequestType"/>
      <xs:element name="exit" type="CommonRequestType"/>
      <xs:element name="help" type="CommonRequestType"/>
      <xs:element name="load" type="CommonRequestType"/>
      <xs:element name="reset" type="CommonRequestType"/>
      <xs:element name="set" type="CommonRequestType"/>
      <xs:element name="show" type="CommonRequestType"/>
      <xs:element name="start" type="CommonRequestType"/>
      <xs:element name="stop" type="CommonRequestType"/>
      <xs:element name="oem verb" type="Oem verbRequestType"/>
      <xs:element name="include" type="Include" minOccurs="0" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:complexType>
</xs:element>

<!-- command request common type -->
<xs:complexType name="CommonRequestType">
  <xs:sequence>
    <xs:element name="abort" type="xsd:boolean" minOccurs="0"/>
    <xs:element name="instance" type="InstanceRequestType" maxOccurs="unbounded"/>
    <xs:element name="options" type="Options" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<!-- command request oemverb type -->
<xs:complexType name="OemverbRequestType">
  <xs:sequence>
    <xs:element name="verbnname" type="xsd:string"/>
    <xs:element name="abort" type="xsd:boolean" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="instance" type="InstanceRequestType" minOccurs="0"/>
    <xs:element name="options" type="Options" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<!-- instance request type -->
</xs:schema>
```
<xsd:complexType name="InstanceRequestType">
  <xsd:sequence>
    <xsd:element name="abort" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="ufip" type="xsd:string" />
    <xsd:element name="options" type="Options" minOccurs="0" />
    <xsd:element name="properties" type="PropertiesRequestType" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

<!-- properties request type -->
<xsd:complexType name="PropertiesRequestType">
  <xsd:sequence>
    <xsd:element name="property" type="PropertyRequestType" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>

<!-- property request type -->
<xsd:complexType name="PropertyRequestType">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string" />
    <xsd:element name="value" type="ValueRequestType" />
  </xsd:sequence>
</xsd:complexType>

<!-- command request options -->
<xsd:complexType name="Options">
  <xsd:sequence>
    <xsd:element name="option" type="Option" minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>

<!-- command request option -->
<xsd:complexType name="Option">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string" />
    <xsd:element name="value" type="ValueRequestType" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

<!-- value request type -->
<xsd:complexType name="ValueRequestType">
  <xsd:sequence>
    <xsd:element name="val" type="xsd:string" />
  </xsd:sequence>
</xsd:complexType>

<!-- command request include info -->
<xsd:complexType name="Include">
<xsd:sequence>
  <xsd;element name="file" type="xsd:string"/>
  <xsd;element name="abort" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
9.1.2 XML Schema for response file

XML schema of the response XML is as below.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<!-- command response -->
<xs:element name="response">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="command" type="CommandInputline" />
      <xs:element name="cmdstat" type="CommandStatus" />
      <xs:choice minOccurs="0">
        <xs:element name="cd" type="CdResponseType"/>
        <xs:element name="create" type="CreateSetResponseType"/>  
        <xs:element name="delete" type="DeleteResponseType"/>
        <xs:element name="dump" type="DumpLoadResponseType"/>
        <xs:element name="exit" type="ExitResponseType"/>
        <xs:element name="help" type="HelpResponseType"/>
        <xs:element name="load" type="DumpLoadResponseType"/>
        <xs:element name="reset" type="ResetStartStopResponseType"/>
        <xs:element name="set" type="CreateSetResponseType"/>
        <xs:element name="show" type="ShowResponseType"/>
        <xs:element name="start" type="ResetStartStopResponseType"/>
        <xs:element name="stop" type="ResetStartStopResponseType"/>
        <xs:element name="oemverb" type="OemverbResponseType"/>
      </xs:choice>
      <xs:element name="oemdata" type="OemdataCommonType" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- command input line -->
<xs:complexType name="CommandInputline">
  <xs:sequence>
    <xs:element name="inputline" type="xs:string" />
  </xs:sequence>
</xs:complexType>

<!-- command status -->
<xs:complexType name="CommandStatus">
  <xs:sequence>
    <xs:element name="status" type="xs:int" />
    <xs:element name="status_tag" type="xs:string" />
    <xs:sequence minOccurs="0">
      <xs:element name="error" type="xs:int" />
      <xs:element name="error_tag" type="xs:string" />
    </xs:sequence>
  </xs:sequence>
</xs:complexType>
```
<!-- cd command response type -->
<xs:complexType name="CdResponseType">
  <xs:choice minOccurs="0">
    <xs:element name="ufip" type="xs:string" />
    <xs:element name="help" type="Help" />
    <xs:element name="examine" type="Examine" />
  </xs:choice>
</xs:complexType>

<!-- create, set command response type -->
<xs:complexType name="CreateSetResponseType">
  <xs:choice minOccurs="0">
    <xs:element name="instance" type="InstancePropertyType" minOccurs="0" />
    <xs:element name="help" type="Help" />
    <xs:element name="examine" type="Examine" />
  </xs:choice>
</xs:complexType>

<!-- delete command response type -->
<xs:complexType name="DeleteResponseType">
  <xs:choice minOccurs="0">
    <xs:element name="target" type="TargetReferenceType" minOccurs="0" maxOccurs="unbounded" />
    <xs:element name="help" type="Help" />
    <xs:element name="examine" type="Examine" />
  </xs:choice>
</xs:complexType>

<!-- dump, load command response type -->
<xs:complexType name="DumpLoadResponseType">
  <xs:choice minOccurs="0">
    <xs:sequence>
      <xs:element name="source" type="Path" minOccurs="0" /> 
      <xs:element name="destination" type="Path" minOccurs="0" />
    </xs:sequence>
    <xs:element name="help" type="Help" />
    <xs:element name="examine" type="Examine" />
  </xs:choice>
</xs:complexType>

<!-- exit command response type -->
<xs:complexType name="ExitResponseType">
  <xs:choice minOccurs="0">
    <xs:element name="help" type="Help" />
    <xs:element name="examine" type="Examine" />
  </xs:choice>
</xs:complexType>

<!-- help command response type -->

<xsd:complexType name="HelpResponseType">
  <xsd:choice minOccurs="0">
    <xsd:element name="text" type="xsd:string" />
    <xsd:element name="help" type="Help" />
    <xsd:element name="examine" type="Examine" />
  </xsd:choice>
</xsd:complexType>

<!-- reset, start, stop command response type -->
<xsd:complexType name="ResetStartStopResponseType">
  <xsd:choice minOccurs="0">
    <xsd:element name="instance" type="InstanceReferenceType" />
    <xsd:element name="help" type="Help" />
    <xsd:element name="examine" type="Examine" />
  </xsd:choice>
</xsd:complexType>

<!-- show command response type -->
<xsd:complexType name="ShowResponseType">
  <xsd:choice minOccurs="0">
    <xsd:element name="target" type="TargetFullType" />
    <xsd:element name="help" type="Help" />
    <xsd:element name="examine" type="Examine" />
  </xsd:choice>
</xsd:complexType>

<!-- oem verb command response type -->
<xsd:complexType name="OemverbResponseType">
  <xsd:sequence>
    <xsd:element name="verbnname" type="xsd:string" />
    <xsd:choice minOccurs="0">
      <xsd:element name="oemdata" type="OemdataOemverbType" />
      <xsd:element name="help" type="Help" />
      <xsd:element name="examine" type="Examine" />
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>

<!-- target full type -->
<xsd:complexType name="TargetFullType">
  <xsd:sequence>
    <xsd:element name="instance" type="InstanceFullType" />
    <xsd:element name="target" type="TargetFullType" minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>

<!-- target reference type -->
<xsd:complexType name="TargetReferenceType">
  <xsd:sequence>
    <xsd:element name="instance" type="InstanceReferenceType" />
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="target" type="TargetReferenceType" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>

<!-- instance full type -->
<xsd:complexType name="InstanceFullType">
  <xsd:sequence>
    <xsd:element name="ufit" type="Ufit" />
    <xsd:element name="ufip" type="xsd:string" />
    <xsd:element name="properties" type="PropertiesResponseType" minOccurs="0" />
    <xsd:element name="verbs" type="Verbs" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

<!-- instance property type -->
<xsd:complexType name="InstancePropertyType">
  <xsd:sequence>
    <xsd:element name="ufit" type="Ufit" />
    <xsd:element name="ufip" type="xsd:string" />
    <xsd:element name="properties" type="PropertiesResponseType" />
  </xsd:sequence>
</xsd:complexType>

<!-- instance reference type -->
<xsd:complexType name="InstanceReferenceType">
  <xsd:sequence>
    <xsd:element name="ufit" type="Ufit" />
    <xsd:element name="ufip" type="xsd:string" />
  </xsd:sequence>
</xsd:complexType>

<!-- properties response type -->
<xsd:complexType name="PropertiesResponseType">
  <xsd:sequence>
    <xsd:element name="property" type="PropertyResponseType" minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>

<!-- property response type -->
<xsd:complexType name="PropertyResponseType">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string" />
    <xsd:element name="value" type="ValueResponseType" />
  </xsd:sequence>
</xsd:complexType>

<!-- value response type -->
<xsd:complexType name="ValueResponseType">
  <xsd:sequence>
    <xsd:element name="val" type="xsd:string"/>
    <xsd:element name="valstring" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<comment>command response help</comment>
<xsd:complexType name="Help">
  <xsd:sequence>
    <xsd:element name="text" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>

<comment>command response examine</comment>
<xsd:complexType name="Examine">
  <xsd:sequence>
    <xsd:element name="text" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>

<comment>command response source or destination path</comment>
<xsd:complexType name="Path">
  <xsd:choice>
    <xsd:element name="ufip" type="xsd:string"/>
    <xsd:element name="uri" type="xsd:string"/>
  </xsd:choice>
</xsd:complexType>

<comment>command response support commands</comment>
<xsd:complexType name="Verbs">
  <xsd:sequence>
    <xsd:element name="standardverbs" type="xsd:string" minOccurs="0"/>
    <xsd:element name="oemverbs" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<comment>OemData common type</comment>
<xsd:complexType name="OemdataCommonType">
  <xsd:sequence>
    <xsd:element name="exitcode" type="xsd:int" minOccurs="0"/>
    <xsd:element name="result" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<comment>OemData oemverb type</comment>
<xsd:complexType name="OemdataOemverbType">
  <xsd:sequence>
    <xsd:element name="exitcode" type="xsd:int" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="result" type="xsd:string" minOccurs="0"/>
<xsd:element name="instance" type="InstancePropertyType" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<!-- command response UFiT -->
<xsd:complexType name="Ufit">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="ufct" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:pattern value="([a-z])+/"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="instance" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:pattern value="([0-9])+"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
</xsd:schema>
## Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>2011/05/11</td>
<td>Initial edition.</td>
</tr>
<tr>
<td>1.01</td>
<td>2012/07/20</td>
<td>Added request/response XML function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added descriptions for a management function of update packages that does not support automatic update.</td>
</tr>
<tr>
<td>1.02</td>
<td>2013/02/13</td>
<td>Added &quot;FreeSpace&quot; property to &quot;Repository settings&quot;.</td>
</tr>
<tr>
<td>1.03</td>
<td>2014/01/06</td>
<td>Fixed a cross-reference link error.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added the Windows OS.</td>
</tr>
<tr>
<td>1.04</td>
<td>2014/08/04</td>
<td>Fixed NEC ESM PRO Manager's version description.</td>
</tr>
<tr>
<td>1.05</td>
<td>2015/06/04</td>
<td>Fixed 'show' command examples of repository settings.</td>
</tr>
<tr>
<td>1.06</td>
<td>2015/12/03</td>
<td>Fixed Cover</td>
</tr>
<tr>
<td>E6.12-01-STD</td>
<td>2016/06/10</td>
<td>Fixed revision format.</td>
</tr>
<tr>
<td>E6.16-01-STD</td>
<td>2016/12/19</td>
<td>Revised edition(6.16)</td>
</tr>
</tbody>
</table>