CHAPTER 2

Functional Definition

This chapter describes the Cisco ANA shell modes, errors, and output format. It also describes the basic commands, unit node management, and surveillance commands. The chapter includes the following information:

Interface Behavior

The Cisco ANA shell CLI follows similar guidelines to the Cisco IOS interface.

The interface includes the following:

- Command completion: Pressing the Tab key completes the currently partially typed command. If there is more than one valid completion, the Cisco ANA shell will beep.
- Fast help mechanism. Press ? to list all valid completions.
- Completion sound notification: Short alarm notice is given when completion is not available.
- Error messages, see Cisco ANA Shell Errors, page 2-2.
- Case insensitive interface.
- Partial syntax recognition.
- Tests the validity of the input.
- The connection to both the Cisco ANA Manage and the unit machines will use Cisco ANA shell syntax. This means that Telnet to a unit should allow all management functions relevant to a single machine.
- Supports More and terminal length.

Cisco ANA Shell Modes

This section describes the four Cisco ANA shell modes.

Mode Name	Description
exec	Entrance mode—Designed to show general details to the operator user. Activation of this mode is authenticated using a username and password on login to the Cisco ANA shell.
enable	Enhanced user mode—Allows further details and configuration. Activation of this mode is authenticated using a password.
configure	Configuration mode—Used to set different attributes. Activation of this mode is authenticated in and possible only from “enable” mode.
manage	Management mode—Used to perform management operations on the system. This mode requires authentication.

Note Each level includes all previous levels.

Cisco ANA Shell Errors

This section describes error handling and error messages in the Cisco ANA shell.

Errors Style

Upon function termination with error, the printout should be:

ERROR (*error code)*: *error message*

For example:

```
ERROR(10443): IP address already inuse
```

In case of parsing error or type-check error (for example, string instead of integer), the command will be reprinted with an arrow pointing to the erroneous phrase. Otherwise, the command will not be reprinted.

Example: typing `"show ip 192.168.1"` will result with

```
ERROR (203): Invalid Value
show ip 192.168.1
^
```

Command Completion Errors

Error Example	Error Message
`show momomo` `^`	Unknown command (a "^" character will mark the first letter that is unrecognized).
`ip change <cr>`	Incomplete command (when more arguments are needed).
`s <TAB>`	Beep if more then one command starts with "s".

Output Format

This section describes the format of the output that the Cisco ANA shell may return.

Table

- Each table should have a header that lists the names of the columns.
- The header should appear only once at the beginning of the table, and not on each page.
- A vertical ruler should separate the header from the rows of the table.
- All table fields should be left aligned.

Example:

No.	IP	Name	Type	Uptime
1	192.168.2.3	asam1	ASAM1000	29.04.02 13:12
10	192.168.2.4	asam2	ASAM1000	1.05.02 9:43
11	192.168.2.45	RedBack2	SMS500	1.05.02 9:44
100	192.168.2.46	RedBack3	SMS500	1.05.02 9:44

Properties

Text paragraph with the following format:

```
<objectname>:
    <attribute name> = <value>
...
```

Example:

```
192.168.2.3:
    IP address = 192.168.2.3
    Name = asam1
    Type = ASAM1000
    Vendor = Alcatel
    Uptime = 1.05.02 13:13
Status = OK
```

Output Redirection

Output redirection enables sending the output of a command to a file.

There are two options for output redirection:

1. "> <filename>" at the end of the command will create a new file and redirect the command output to it. If the file already exists, the new file will override the old one. In case of an error in creating the output file, the command will not be run. For example: `show device > out.txt`
2. ">> <filename>" at the end of the command will append the output to an existing file. The file should be created if it does not already exist.

Background Processing

Each Cisco ANA shell command may be run at the background using the "&" symbol at the end of the command. For example "`show link > links.txt &`" will run the command "`show link`" at the background, redirecting all output to the file "`links.txt`". By default, unless redirection is specified, the output of a background command is redirected to the Cisco ANA shell terminal

Basic Commands

This section describes the basic commands.

Inline Help ("?")

Name	Inline help
Description	Supplies command completion while typing.
Mode	All modes
Usage	`?`
General	That command executes with out typing <cr>.

Example	`show ?`
	`path  find path from source to destination`
	`blabla a command that generates a very long description that exceeds a single line and therefore should  properly wraparound to the next line`

Output Format	List of valid options with a short description for each option. If the description exceeds a single line, the lines after the first one should be indented with the first description line.
Remarks	
Priority	High

Enable

Name	Enable
Description	Enter enable mode.
Mode	`exec`
Usage	`enable`
General	Demands authentication. After entering enable mode the interface prompt is suffixed with a hash (#) sign.
Example	`enable`
Output Format	None
Remarks	
Priority	High

Configure

Name	Configure
Description	Enter configure mode.
Mode	`enable`
Usage	`configure <cr>`
General	After entering configure mode, the interface prompt is suffixed with a “conf”.
Example	`configure <cr>`
Output Format	None
Remarks	
Priority	High

Exit Cisco ANA Shell

Name	Exit Cisco ANA shell
Description	Exit Cisco ANA shell interface.
Mode	`exec`
Usage	`exit <cr>`
General	
Example	`exit`
Output Format	None
Remarks	If the Cisco ANA shell serves as the shell for the machine, it should return to the login window.
Priority	High

Exit Current Mode

Name	Exit current mode
Description	Exit current Cisco ANA shell mode and return to the previous mode.
Mode	`enable, config, manage`
Usage	`exit <cr>`
General	Running exit in enable mode returns to exec mode. Running exit in config mode returns to enable mode. Running exit in manage mode will return to the mode from which the user entered it (could be enable or configure).
Example	`exit`
Output Format	None
Remarks	
Priority	

Help

Name	Help
Description	Prints general help. Lists all the commands with a short description line for each command.
Mode	All modes
Usage	`help <cr>`
General	
Example	`help`
Output Format	A fixed help message.
Remarks	
Priority	Medium

Terminal Length

Name	Terminal length
Description	Set terminal length.
Mode	`all modes`
Usage	`terminal length <integer> <cr>`
General	Use length 0 for no pausing.
Example	`terminal length 40`
Output Format	None

Remarks	
Priority	Low

Show History

Name	Show history		
Description	Show previously run command		
Mode	`all modes`		
Usage	`history <cr>`		
General	The history should contain the last 100 commands . This is not configurable.		
Example	`history`		
Output Format	Type	Table	
	Columns	Column	Description
		Index	The index of the command. Index 1 refers to the previous command.
		Command	The string of the command.
Remarks			
Priority			

Clear History

Name	Clear history
Description	Clear the command list stored in the history buffer.
Mode	`all modes`
Usage	`history clear<cr>`
General	
Example	`history clear`
Output Format	None
Remarks	
Priority	

Access History

Name	Access history
Description	Runs against a command in the history buffer.
Mode	`all modes`
Usage	`history <integer> <cr>`

General	1 in the index is the last command (not including the current history command), the command before is 2 and so on.
Example	`history 3`
Output Format	None
Remarks	
Priority	

Execute Script

Name	Execute script
Description	Run a script file of Cisco ANA shell commands.
Mode	All modes
Usage	`run <filename> [async] [silent] <cr>`
General	The scripts must reside on the UNIX machine running the Cisco ANA shell on the directory /Main. This directory resides under the directory where the system was installed. The files should be transferred to this directory or its subdirectories using FTP. The scripts may reside in subdirectories of the base directory /Main, in which case the name of the script should include the relative path of the script. `"silent"` indicates suppressing any output that the command sends to the terminal. **Note** The script can be run at the background using "&".
Example	`run provision.cmd` `run scripts/provision.cmd`
Output Format	None
Remarks	
Priority	

Node Management

This section describes the commands needed to manage a unit node. Unit node management includes management of the AVM processes and VNEs within the AVMs.

AVM Management

Show AVM List

Name	Show AVM list
Description	Show a list of AVMs with their minimum set of properties.
Mode	`manage`
Usage	`show [unit [<IPAddress>]] avm <cr>`

General	If an IP address is given, shows only AVMs on the specified machine. Otherwise, shows AVMs on all machines. If no unit is given, then the command refers to the current machine.		
Example	`show unit avm <cr>`		
Output Format	Type	Table	
	Columns	Column	Description
		Machine	IP address of the machine where the AVM resides.
		ID	AVM ID
		PID	Process ID
		Port	Management port
		Uptime	Process uptime (date format)
		Version	AVM version
Remarks			
Priority	High		

Show AVM VNEs

Name	Show AVM VNEs
Description	List all the VNEs of a specific AVM.
Mode	`manage`
Usage	`show [unit <IPAddress>] avm <integer> | all agent [detailed] <cr>`
General	Lists all the VNEs in the AVM. If no unit is given, then the command refers to the current machine. If detailed is not given, only DAs are displayed. Otherwise, all VNE types (DA, CA, IA) are displayed. all refers to all AVMs in the current machine. The command should also display configured VNEs, which are configured in the XML but are not loaded. In this case, all the non-relevant fields should be empty.
Example	`show unit 192.168.2.10 avm 32 agent`

Output Format	Type	Table	
	Columns	Column	Description
		IP address	VNE leading IP.
		Type	VNE Type (DA, IA, CA).
		State	VNE state (idle, wait, block, running, configured).
		Runtime	The total time spent by the VNE processing messages.
		Wait time	The total time spent by the VNE waiting to process messages.
		Last run	The last time the VNE visited the scheduler. Units are milliseconds relative to now.
		Transport address	The VNE's transport address in hexadecimal format.
		Parent	Parent VNE. Transport address of the parent VNE.
Remarks			
Priority			

VNE Management

Show All VNEs

Name	Show all VNEs in the unit
Description	Show the basic information about all the VNEs in the system (all AVMs).
Mode	`manage`
Usage	`show agent [detailed] <cr>`
General	If detailed is not given, only DAs are displayed. Otherwise, all VNE types (DA, CA, IA) are displayed.
Example	`show agent <cr>`
Output Format	See Show AVM VNEs, page 2-9.
Remarks	
Priority	

Show VNE Information (-)

Name	Show VNE information
Description	Show the information for a specific VNE.
Mode	`manage`
Usage	`show agent <IPAddress | name> <cr>`
General	The parameter can be the leading IP of the VNE, or the device name.

Example	`show agent 192.168.2.2 <cr>`		
Output Format	Type	Properties	
	Fields	Field	Description
		IP Address	VNE leading IP
		Type	VNE Type (DA, IA, CA)
		Machine	IP address of the machine where the VNE is installed.
		AVM	AVM number where the VNE is installed.
		Transport address	The VNE's transport address in hexadecimal format.
		State	VNE state (idle, wait, block, running, configured).
		Runtime	The total time spent by the VNE processing messages.
		Wait time	The total time spent by the VNE waiting to process messages.
		Last run	The last time the VNE visited the scheduler. Units are milliseconds relative to now.
		Parent	Parent VNE. Transport address of the parent VNE.
Remarks			
Priority			

Add AVM

Name	Add AVM
Description	Add a new AVM to a unit.
Mode	`manage`
Usage	`unit <IPAddress> avm <integer> add <cr>`
General	The parameter represents the ID of the AVM that is to be added.
Example	`unit 192.168.2.10 avm 32 add`
Output Format	None
Remarks	
Priority	

Remove AVM

Name	Remove AVM
Description	Remove an AVM from a unit machine.
Mode	`manage`
Usage	`unit <IPAddress> avm <integer> remove<cr>`

General	The parameter represents the ID of the AVM to be deleted.
Example	`unit 192.168.2.10 avm 32 remove`
Output Format	None
Remarks	
Priority	

Load AVM

Name	Load AVM
Description	Add a configured AVM to the unit bootstrap list.
Mode	`manage`
Usage	`unit <IPAddress> avm <integer> load <cr>`
General	The parameter represents the ID of the AVM to be loaded. The newly loaded AVM starts immediately, and in addition, it will be started in all consecutive restarts of the system.
Example	`unit 192.168.2.10 avm 32 load`
Output Format	None
Remarks	To immediately start the new AVM, refer to .
Priority	

Unload AVM

Name	Unload AVM
Description	Remove an AVM from the bootstrap list.
Mode	`manage`
Usage	`unit <IPAddress> avm <integer> unload <cr>`
General	The AVM will be automatically stopped, if currently executing. The parameter represents the ID of the AVM to be unloaded.
Example	`unit 192.168.2.10 avm 32 unload`
Output Format	None
Remarks	
Priority	

Add VNE

Name	Add VNE
Description	Add a VNE to the configuration database of a given AVM.
Mode	`manage`

Usage	`agent <IPAddress> add unit <IPAddress> avm <integer> name <name> vendor <string> type <string> <cr>`
General	If no name is given, the IP address will be used as the device name.
Example	`agent 192.168.2.3 add unit 192.168.2.10 avm 32 vendor alcatel type asam1000`
Output Format	None
Remarks	
Priority	

Remove VNE

Name	Remove VNE
Description	Remove a VNE from a given AVM.
Mode	`manage`
Usage	`agent <IPAddress> remove <cr>`
General	
Example	`agent 192.168.2.3 remove`
Output Format	None
Remarks	If the VNE is currently running, it will stop. If the VNE is loaded, it should be unloaded from the bootstrap list.
Priority	

Load VNE

Name	Load VNE
Description	The newly loaded VNE starts immediately. It will be loaded every time the system restarts.
Mode	`manage`
Usage	`agent <IPAddress> load <cr>`
General	
Example	`agent 192.168.2.3 load`
Output Format	None
Remarks	
Priority	

Unload VNE

Name	Unload VNE
Description	Unload an VNE from the AVM bootstrap list. If the VNE is currently running, it is stopped before unloading from the bootstrap list.

Mode	`manage`
Usage	`agent <IPAddress> unload <cr>`
General	
Example	`agent 192.168.2.3 unload`
Output Format	None
Remarks	
Priority	

Add Static Topology Link

Name	Add static topology link
Description	Add a static link between two devices in the network.
Mode	`manage`
Usage	`topology link source <IPAddress> [shelf <integer>] module <integer> [submodule <integer>] port <integer> destination <IPAddress> [shelf <integer>] module <integer> [submodule <integer>] port <integer> add [unidirectional] <cr>`
General	By default, the link is bidirectional and enabled automatically. Unidirectional represents a unidirectional link.
Example	`topology link source 192.168.2.3 module 1 port 1 destination 192.168.2.4 module 2 port 1 add`
Output Format	None
Remarks	
Priority	

Remove Static Topology Link

Name	Remove static topology link
Description	Remove an existing static topology link.
Mode	`manage`
Usage	`topology link source <IPAddress> [shelf <integer>] module <integer> [submodule <integer>] port <integer> destination <IPAddress> [shelf <integer>] module <integer> [submodule <integer>] port <integer> remove [unidirectional] <cr>`
General	
Example	`topology link source 192.168.2.3 module 1 port 1 destination 192.168.2.4 module 2 port 1 remove`
Output Format	None
Remarks	
Priority	

Surveillance

This section describes the surveillance commands that should be supported by the Cisco ANA shell interface.

Show Links

Name	Show links		
Description	Show the topological links managed by the unit.		
Mode	`enable`		
Usage	`show link <cr>`		
General			
Example	`show link`		
Output Format	Type	Table	
	Columns	Column	Description
		Index	Unique running index.
		From	A-side location.
		To	Z-side location.
		State	Automatic, Static, Configured.
Priority			

Drools Rules Management

For more information about the Drools Rules Engine, refer to the *Cisco Active Network Abstraction Administrator* Guide.

Show Rules

Name	Show rules
Description	Show all rules.
Mode	`enable`
Usage	`show rule`
General	
Example	`show rule`
Output Format	contextID, ruleName, isValid
Remarks	
Priority	

Show Rules

Name	Show rules
Description	Show rules of a specific context.
Mode	`enable`
Usage	`show rule <contextID>`
General	
Example	`show rule aaa`
Output Format	contextID, ruleName, isValid
Remarks	
Priority	

Reload Rules

Name	Reload rules
Description	Reloads all rules of a specific context.
Mode	`config`
Usage	`rule <contextID> reload`
General	
Example	`rule aaa reload`
Output Format	
Remarks	
Priority	

Validate Rule

Name	Validate rule
Description	Validate a specific rule.
Mode	
Usage	`rule <contextID> <ruleName> validate`
General	
Example	`rule aaa bbb validate`
Output Format	
Remarks	
Priority	