



CLI の概要

この章では、CLI について説明します。内容は次のとおりです。

- 「CLI プロンプトの概要」 (P.6-1)
- 「コマンド モード」 (P.6-2)
- 「特殊文字」 (P.6-7)
- 「キーストローク ショートカット」 (P.6-7)
- 「コマンドの省略」 (P.6-9)
- 「コマンドの No 形式の使用法」 (P.6-9)
- 「CLI 変数の使用法」 (P.6-10)
- 「コマンド スクリプトの使用法」 (P.6-12)
- 「ヘルプの使用法」 (P.6-14)

CLI プロンプトの概要

システムに正常にアクセスすると、コンソール ポートのターミナル ウィンドウまたはリモート ワークステーションに、次のような CLI プロンプトが表示されます。

```
switch#
```

このスイッチ プロンプトは別の名前に変更することも、そのままにしておくこともできます。

例:

```
switch# config t
switch(config)# switchname n1000v
n1000v(config)# exit
n1000v#
```

CLI プロンプトから、次の方法を実行できます。

- 機能を設定するための CLI コマンドを使用する
- コマンド履歴にアクセスする
- コマンド解析機能を使用する

コマンドモード

ここでは、次の内容について説明します。

- 「コマンドモードについて」 (P.6-2)
- 「EXEC コマンドモード」 (P.6-3)
- 「グローバル コンフィギュレーション コマンドモード」 (P.6-3)
- 「インターフェイス コンフィギュレーション コマンドモードへのアクセス」 (P.6-3)
- 「コンフィギュレーション モードの終了」 (P.6-4)
- 「コマンドモードの概要」 (P.6-5)

コマンドモードについて

Cisco Nexus 1000V CLI は、いくつかのコマンドモードに分けられます。ユーザが実行できるアクションは、コマンドモードによって定義されます。コマンドモードは「ネスト」されており、順番にアクセスされます。初めてログインしたときは CLI EXEC モードで開始されます。

EXEC モードからグローバル コンフィギュレーション モードにナビゲートしていくと、使用できるコマンド数が増えます。グローバル コンフィギュレーション モードに移るには、次のコマンドを入力します。

config

次の表は、コマンドアクセスがユーザ EXEC モードからグローバル コンフィギュレーション モードへどのように形成されるかを示します。

コマンドモード	プロンプト	説明
EXEC	n1000v#	<ul style="list-style-type: none"> • リモート デバイスに接続します。 • 端末回線の設定を一時的に変更します。 • 基本テストを実行します。 • システム情報を一覧します (show コマンド)。
グローバル コンフィギュレーション	n1000v(config)#	<ul style="list-style-type: none"> • 次のような機能を設定します。 <ul style="list-style-type: none"> – ポート プロファイル – VLAN – インターフェイス • EXEC コマンドへのアクセスを含めます。 <ul style="list-style-type: none"> – リモート デバイスに接続します。 – 端末回線の設定を一時的に変更します。 – 基本テストを実行します。 – システム情報を一覧します (show コマンド)。

EXEC コマンドモードのすべてのコマンドは、グローバル コンフィギュレーション コマンドモードからアクセスできます。たとえば、**show** コマンドは、どのコマンドモードからも使用できます。

EXEC コマンド モード

初めてログインしたときは EXEC モードで開始されます。EXEC モードで使用可能なコマンドには、デバイスの状態および構成情報を表示する **show** コマンド、**clear** コマンド、ユーザがデバイス設定に保存していない処理が実行されるその他のコマンドなどがあります。

グローバル コンフィギュレーション コマンド モード

グローバル コンフィギュレーション モードでは、最も広範囲のコマンドにアクセスできます。たとえば、デバイスによって保存された設定を変更し、保存して、デバイスの再起動時に適用する場合に使用するコマンドなども含まれます。

グローバル コンフィギュレーション モードでコマンドを入力すると、実行中のコンフィギュレーション ファイルがすぐに更新されます。ただし、次のコマンドを使用して、スタートアップ コンフィギュレーション ファイルにコマンドを保存する必要もあります。

copy running-config startup-config

グローバル コンフィギュレーション モードでは、多くのプロトコル固有、プラットフォーム固有、および機能固有コンフィギュレーション モードにアクセスできます。

インターフェイス コンフィギュレーション コマンド モードへのアクセス


インターフェイス コンフィギュレーション コマンドにアクセスして一覧表示するには、次の手順を実行します。

	コマンド	目的
ステップ 1	configure terminal 例: n1000v# configure terminal n1000v(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	interface type number 例: n1000v(config)# interface ethernet 3/2 n1000v(config-if)#	設定するインターフェイスのインターフェイス コンフィギュレーション モードを開始します。

インターフェイス コマンドおよびコンフィギュレーションの詳細は、『Cisco Nexus 1000V Interface Configuration Guide, Release 4.0(4)SV1(3)』を参照してください。

コンフィギュレーションモードの終了

コンフィギュレーションモードを終了するには、次のいずれかのコマンドを使用します。

コマンド	目的
exit 例: <pre>svs(config-if)# exit svs(config)#</pre>	現在のコンフィギュレーション コマンド モードを終了して、以前のコンフィギュレーション コマンド モードに戻ります。
end 例: <pre>svs(config)# end svs#</pre>	コンフィギュレーション コマンド モードを終了して、EXEC モードに戻ります。
Ctrl-z 例: <pre>svs(config)# ^z svs#</pre>	現在のコンフィギュレーション コマンド モードを終了して、EXEC モードに戻ります。 <div style="border: 1px solid black; padding: 5px; width: fit-content;">  <p>注意 有効なコマンドを入力してから、コマンドラインの最後で Ctrl+Z を使用すると、CLI によってそのコマンドが実行コンフィギュレーション ファイルに追加されます。コンフィギュレーション モードを終了する場合は、exit または end コマンドを使用することを推奨します。</p> </div>

コマンドモードの概要

表 6-1 は、コマンドモードの概要を示します。

表 6-1 コマンドモードの概要

モード	アクセス方式	プロンプト	終了方式
EXEC	ログインプロンプトから、ユーザ名とパスワードを入力します。	n1000v#	終了してログインプロンプトに戻るには、 exit コマンドを使用します。
グローバル コンフィギュレーション	EXEC モードで、 configure terminal コマンドを入力します。	n1000v(config)#	終了して EXEC モードに戻るには、 end または exit コマンドを使用するか、Ctrl+Z キーを押します。
ポート プロファイル設定	グローバル コンフィギュレーション モードで、 port-profile name コマンドを入力します。	n1000v(config-port-prof)#	終了してポート プロファイル コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
インターフェイス コンフィギュレーション	グローバル コンフィギュレーション モードで、インターフェイス veth 2 などの特定のインターフェイスに対して interface コマンドを入力します。	n1000v(config-if)#	終了してインターフェイス コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
VLAN コンフィギュレーション	vlan コマンドを使用します。	n1000v(config-vlan)#	終了して VLAN コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
コンソール コンフィギュレーション	グローバル コンフィギュレーション モードで、 line console コマンドを使用します。	n1000v(config-console)	終了してコンソール コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
仮想端末回線コンフィギュレーション	グローバル コンフィギュレーション モードで、 line vty コマンドを使用します。	n1000v(config-line)#	終了して回線コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。

表 6-1 コマンド モードの概要 (続き)

モード	アクセス方式	プロンプト	終了方式
SVS ドメイン コンフィギュレーション	グローバル コンフィギュレーション モードで、 svs-domain コマンドを使用します。	n1000v(config-svs-domain)#	終了して SVS ドメイン コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
ポリシー マップ QoS コンフィギュレーション	グローバル コンフィギュレーション モードで、 policy-map コマンドを使用します。	n1000v(config-pmap-qos)#	終了してポリシー マップ QoS コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
ポリシー マップ クラス QoS コンフィギュレーション	ポリシー マップ QoS コンフィギュレーション モードで、 class コマンドを使用します。	n1000v(config-pmap-c-qos)#	終了してポリシー マップ クラス QoS コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。
クラス マップ QoS コンフィギュレーション	グローバル コンフィギュレーション モードで、 class-map コマンドを使用します。	n1000v(config-cmap-qos)#	終了してクラス マップ QoS コンフィギュレーション モードに戻るには、 exit コマンドを使用します。 終了して EXEC モードに戻るには、 end コマンドを使用するか、Ctrl+Z キーを押します。

CLI 設定変更の保存

ここでは、CLI を使用して加えた変更の保存方法と、次の内容について説明します。

- 「実行コンフィギュレーション」(P.6-6)
- 「スタートアップ コンフィギュレーション」(P.6-7)
- 「実行コンフィギュレーションのスタートアップ コンフィギュレーションへのコピー」(P.6-7)

実行コンフィギュレーション

実行コンフィギュレーションは、デバイス上で現在実行中の設定です。実行コンフィギュレーションには、デバイスが最後に再起動してから、コマンドを入力して加えた設定変更が含まれます。デバイスが再起動した場合、実行コンフィギュレーションは、スタートアップ コンフィギュレーションのコピーに置き換えられます。実行コンフィギュレーションに加えられた変更は、スタートアップ コンフィギュレーションにコピーされていないと、廃棄されます。

スタートアップ コンフィギュレーション

スタートアップ コンフィギュレーションは、保存された設定であり、デバイスを再起動したときに、デバイスによって使用されます。デバイスに設定変更を加えると、設定変更が自動的に実行コンフィギュレーションに保存されます。設定変更を永続的に保存する場合は、設定変更をスタートアップ コンフィギュレーションにコピーする必要があります。そうすると、デバイスをリブートまたは再起動したときに、設定変更が維持されます。

実行コンフィギュレーションのスタートアップ コンフィギュレーションへのコピー

次の手順に従って、実行コンフィギュレーションに加えた変更をスタートアップ コンフィギュレーションにコピーできます。そうすると、変更がリブート、再起動後も永続的に保存されます。

	コマンド	目的
ステップ 1	<pre>copy running-config startup-config</pre> <p>例: n1000v(config)# copy running-config startup-config</p>	(任意) 実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーし、リブートや再起動後も維持されるように永続的に保存します。

特殊文字

表 6-2 に、Cisco Nexus 1000V のテキスト スtring で特別な意味を持つ文字を示します。正規表現あるいはその他の特有用なコンテキストでのみ使用します。

表 6-2 特殊文字

文字	説明
	縦線
<>	より小さい、またはより大きい

キーストローク ショートカット

表 6-3 に、EXEC モードおよびコンフィギュレーション モードの両方で使用されるコマンド キーの組み合わせを示します。

表 6-3 キーストローク ショートカット

キー	説明
Ctrl+A	カーソルを行の先頭に移動します。
Ctrl+B	カーソルを 1 文字分だけ後退させます。複数行にわたってコマンドを入力するときは、←キーまたは Ctrl+B キーを繰り返し押し続けてシステム プロンプトまでスクロールバックして、コマンド エントリの先頭まで移動できます。あるいは Ctrl+A キーを押してコマンド エントリの先頭に移動します。

表 6-3 キーストローク ショートカット (続き)

キー	説明
Ctrl+C	コマンドを取り消して、コマンドプロンプトに戻ります。
Ctrl+D	カーソル位置にある文字を削除します。
Ctrl+E	カーソルを行の末尾に移動します。
Ctrl+F	カーソルを 1 文字分だけ進めます。
Ctrl+G	コマンドストリングを削除せずに、コマンドモードを終了して以前のコマンドモードに戻ります。
Ctrl+K	カーソル位置からコマンドラインの末尾までのすべての文字を削除します。
Ctrl+L	現在のコマンドラインを再表示します。
Ctrl+R	現在のコマンドラインを再表示します。
Ctrl+T	カーソルの左にある文字を、カーソル位置の文字と置き換えます。
Ctrl+U	カーソル位置からコマンドラインの先頭までのすべての文字を削除します。
Ctrl+W	カーソルの左にあるワードを削除します。
Ctrl+X、H	履歴を一覧表示します。 このキーの組み合わせを使用するときは、Ctrl キーと X キーを同時に押して放してから、H キーを押します。
Ctrl+Y	バッファ内の最新のエントリを呼び出します (キーを同時に押します)。
Ctrl+Z	コンフィギュレーションセッションを終了して、EXEC モードに戻ります。 有効なコマンドを入力してから、コマンドラインの最後で Ctrl+Z キーを使用すると、コマンドの結果の設定がまず実行コンフィギュレーションファイルに追加されます。
↑	コマンド履歴の前のコマンドを表示します。
↓	コマンド履歴の次のコマンドを表示します。
→ ←	コマンド履歴内でカーソルを移動して、コマンドストリングを見つけます。
?	使用可能なコマンドのリストを表示します。
Tab	ワードの最初の文字を入力して Tab キーを押すと、ワードが補完されます。文字に一致するすべてのオプションが表示されます。 次の名前を補完する場合に使用します。 <ul style="list-style-type: none"> • コマンド名 • ファイル システム内のスキーム名 • ファイル システム内のサーバ名 • ファイル システム内のファイル名 <p>例</p> <pre>n1000v(config)# xm<Tab> n1000v(config)# xml <Tab> n1000v(config)# xml server</pre>

表 6-3 キーストローク ショートカット (続き)

キー	説明
	<p>例</p> <pre>n1000v(config)# c<Tab> class-map class-map clock cts cdp cli control-plane</pre> <p>n1000v(config)# cl<Tab> class-map cli clock n1000v(config)# cla<Tab> n1000v(config)# class-map</p>
	<p>例</p> <pre>n1000v# cd bootflash:<Tab> bootflash: bootflash://sup-1/ bootflash://sup-remote/ bootflash:/// bootflash://sup-2/ bootflash://sup-standby/ bootflash://module-5/ bootflash://sup-active/ bootflash://module-6/ bootflash://sup-local/</pre> <p>例</p> <pre>n1000v# cd bootflash://mo<Tab> bootflash://module-5/ bootflash://module-6/ n1000v# cd bootflash://module-</pre>

コマンドの省略

コマンドの最初の数文字を入力することで、コマンドおよびキーワードを省略できます。省略形には、コマンドまたはキーワードを一意に識別でき得る文字数を含める必要があります。コマンド入力で問題に遭遇した場合は、システムプロンプトを確認するとともに、疑問符(?)を入力して使用できるコマンドのリストを表示してください。コマンドモードが間違っているか、間違った構文を使用している可能性があります。

表 6-4 に、コマンド省略形の例を示します。

表 6-4 コマンド省略形の例

コマンド	省略形
configure terminal	conf t
copy running-config startup-config	copy run start
interface ethernet 1/2	int e 1/2
show running-config	sho run

コマンドの No 形式の使用法

ほとんどすべてのコンフィギュレーションコマンドには **no** 形式があり、これを使用して機能や関数を無効にできます。たとえば、VLAN を削除するには、**no vlan** コマンドを使用します。もう一度有効にするには、**vlan** プレーンコマンド形式を使用します。『Cisco Nexus 1000V Command Reference, Release 4.0(4)SV1(3)』では、コマンドの **no** 形式を使用できる場合とその動作について説明しています。

たとえば、グローバル コンフィギュレーション モードで **boot** コマンドを使用している場合は、**no boot** コマンドを使用して結果を元に戻すことができます。

```
n1000v(config)# boot system bootflash: svs1.bin
n1000v(config)# no boot system bootflash: svs1.bin
```

CLI 変数の使用方法

Cisco Nexus 1000V は、CLI コマンドでの変数の定義および使用をサポートします。CLI 変数は、次のようにして使用できます。

- コマンドラインで直接入力します。
- **run-script** コマンドを使用して開始された子スクリプトに渡します。親シェルで定義した変数は子 **run-script** コマンドプロセスで使用できます（「スクリプトの実行」(P.6-12) を参照）。
- コマンドライン引数として **run-script** コマンドに渡します（「スクリプトの実行」(P.6-12) を参照）。

CLI 変数には、次の特性があります。

- 入れ子状態の参照を使用して、別の変数から変数を参照することはできません。
- スイッチのリロード後も存続します。
- 現在のセッションの間だけ存在できます。

Cisco Nexus 1000V ソフトウェアには、事前定義のシステム変数である **TIMESTAMP** 変数があります。

ユーザ定義の CLI セッション変数

EXEC モードで **cli var name** コマンドを使用して、CLI セッション変数の存続期間を自分の CLI セッション中に限るように定義できます。CLI セッション変数は周期的に実行するスクリプトに役立ちます。

次の例は、ユーザ定義の CLI セッション変数の作成方法を示します。

```
svs# cli var name testinterface ethernet 3/2
```

構文 **\$(variable)** を使用して変数を参照できます。

次の例は、ユーザ定義の CLI セッション変数の参照方法を示します。

```
n1000v# show interface $(testinterface)
Ethernet3/2 is up
  Hardware is Ethernet, address is 0050.565a.2341 (bia 0050.565a.2341)
  MTU 1500 bytes, BW -332641784 Kbit, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  Port mode is trunk
  full-duplex, 1000 Mb/s
  Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  Rx
  222045 Input Packets 24263 Unicast Packets
  89347 Multicast Packets 108435 Broadcast Packets
  22529316 Bytes
  Tx
  33710 Output Packets 31393 Unicast Packets
  1898 Multicast Packets 419 Broadcast Packets 461 Flood Packets
  5221175 Bytes
  91323 Input Packet Drops 0 Output Packet Drops
```

```
n1000v#
```

Use the `show cli variables` command to display user-defined CLI session variables. The following example displays user-defined CLI session variables.

```
n1000v# show cli variables
VSH Variable List
-----
TIMESTAMP="2008-07-02-13.45.15"
testinterface="ethernet 3/1"
n1000v#
```

ユーザ定義の CLI セッション変数を削除するには、`cli no var name` コマンドを使用します。

次の例は、ユーザ定義の CLI セッション変数を削除します。

```
n1000v# cli no var name testinterface
```

ユーザ定義の CLI 固定変数

コンフィギュレーション モードで `cli var name` コマンドを使用して、CLI 変数が複数の CLI セッションを通じて存続し、スイッチのリロード後も存続するように定義できます。このような CLI 固定変数はコンフィギュレーション モードで定義され、実行中のコンフィギュレーション ファイルに保存されます。

次の例は、ユーザ定義の CLI 固定変数の作成方法を示します。

```
n1000v# config t
n1000v(config)# cli var name mgmtport mgmt 0
n1000v(config)# exit
n1000v#
```

構文 `$(variable)` を使用して変数を参照できます。

次の例は、ユーザ定義の CLI 固定変数の参照方法を示します。

```
n1000v# show interface $(mgmtport)
mgmt0 is up
  Hardware is GigabitEthernet, address is 0000.0000.0000 (bia 0050.5681.5578)
  Internet Address is 10.78.1.63/24
  MTU 1500 bytes, BW 0 Kbit, DLY 0 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA
  full-duplex, 1000 Mb/s
  Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  321949 packets input, 67199373 bytes
  0 multicast frames, 0 compressed
  0 input errors, 0 frame, 0 overrun, 0 fifo
  30178 packets output, 7071526 bytes
  0 underrun, 0 output errors, 0 collisions
  0 fifo, 0 carrier errors
```

```
n1000v#
```

ユーザ定義の CLI 固定変数を表示するには、`show cli variables` コマンドを使用します。

次の例では、ユーザ定義の CLI 固定変数を表示します。

```
n1000v# show cli variables
VSH Variable List
-----
TIMESTAMP="2005-10-24-21.37.13"
mgmtport="mgmt 0"
```

ユーザ定義の CLI 固定変数を削除するには、コンフィギュレーション モードで **no cli var name** コマンドを使用します。

次の例では、ユーザ定義の CLI 固定変数を削除します。

```
n1000v# config t
n1000v(config)# cli no var name mgmtport
```

システム定義の変数

Cisco Nexus 1000V は、TIMESTAMP という事前定義の変数をサポートします。この変数は YYYY-MM-DD-HH.MM.SS フォーマットのコマンドの実行時間を参照します。



(注) TIMESTAMP 変数名は大文字と小文字を区別します。文字はすべて大文字です。

次の例では、**show** コマンド出力をファイルにリダイレクトするときに \$(TIMESTAMP) を使用します。

例:

```
n1000v# show running-config > rcfg.$(TIMESTAMP)
n1000v# dir
      5718      Jul 02 14:09:58 2008  rcfg.2008-07-02-14.09.58
```

```
Usage for volatile://
      8192 bytes used
 20963328 bytes free
 20971520 bytes total
n1000v#
```

コマンドスクリプトの使用方法

ここで説明する内容は、次のとおりです。

- 「スクリプトの実行」(P.6-12)
- 「スクリプトでの CLI 変数の使用」(P.6-13)
- 「コマンド処理の遅延」(P.6-14)

スクリプトの実行

run-script コマンドは、ファイルで指定されているコマンドを実行します。このコマンドを使用するには、必ずファイルを作成して、必要な順序でコマンドを指定してください。



(注) スイッチ プロンプトでは、スクリプト ファイルを作成できません。スクリプト ファイルを外部マシンで作成し、**bootflash:** ディレクトリにコピーすることができます。ここでは、スクリプト ファイルが **bootflash:** ディレクトリにあると想定しています。

このコマンドの構文は **run-script filename** です。

この例では、**bootflash** 内にあるテストファイルで指定された CLI コマンドを表示します。

```
n1000v# show file bootflash:testfile
conf t
show interface mgmt 0
```

次のファイル出力は、テストファイル内の内容を実行する **run-script** コマンドに応じたものです。

```
pvk-s33# run-script bootflash:testfile
`conf t`
`show interface mgmt 0`
mgmt0 is up
Hardware: Ethernet, address: 0050.5682.4ace (bia 0050.5682.4ace)
Internet Address is 10.78.1.99/24
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA
full-duplex, 1000 Mb/s
Auto-Negotiation is turned on
25427 packets input, 2602757 bytes
0 multicast frames, 0 compressed
0 input errors, 0 frame, 0 overrun, 0 fifo
9077 packets output, 2433391 bytes
0 underrun, 0 output errors, 0 collisions
0 fifo, 0 carrier errors
...
```

スクリプトでの CLI 変数の使用

cli var コマンドで定義された CLI 変数、または引数として **run-script** コマンドに渡された CLI 変数を使用できます。**cli var** コマンドの詳細については、「[CLI 変数の使用方法](#)」(P.6-10) を参照してください。

次の例は、**run-script** コマンドで使用されるスクリプト ファイル内での CLI セッション変数の使用方法を示します。

```
n1000v# cli var name testinterface e 3/1

n1000v# show file bootflash:test1.vsh
show interface $(testvar)

n1000v# run-script bootflash:test1.vsh
`show interface $(testvar)`
Ethernet3/1 is down (Administratively down)
Hardware is 10/100/1000 Ethernet, address is 0000.0000.0000 (bia 0019.076c.4da
c)
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA
auto-duplex, auto-speed
Beacon is turned off
Auto-Negotiation is turned on
Input flow-control is off, output flow-control is off
Auto-mdix is turned on
Switchport monitor is off
Last clearing of "show interface" counters never
5 minute input rate 0 bytes/sec, 0 packets/sec
5 minute output rate 0 bytes/sec, 0 packets/sec
L3 in Switched:
ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
L3 out Switched:
ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
```

```

Rx
 0 input packets 0 unicast packets 0 multicast packets
 0 broadcast packets 0 jumbo packets 0 storm suppression packets
 0 bytes
Tx
 0 output packets 0 multicast packets
 0 broadcast packets 0 jumbo packets
 0 bytes
 0 input error 0 short frame 0 watchdog
 0 no buffer 0 runt 0 CRC 0 ecc
 0 overrun 0 underrun 0 ignored 0 bad etype drop
 0 bad proto drop 0 if down drop 0 input with dribble
 0 input discard
 0 output error 0 collision 0 deferred
 0 late collision 0 lost carrier 0 no carrier
 0 babble
 0 Rx pause 0 Tx pause 0 reset

```

次の例は、CLI セッション変数を引数として **run-script** コマンド プロセスに渡す方法を示します。

```

n1000v# show file bootflash:test1.vsh
show interface $(var1) $(var2)

n1000v# run bootflash:test2.vsh var1="e3/1" var2="brief"
`show interface $(var1) $(var2)`
-----
Ethernet      VLAN   Type Mode   Status Reason          Speed   Port
Interface
-----
Eth2/45       --    eth  routed down  Administratively down  auto(D) --

```

コマンド処理の遅延

sleep コマンドは、指定された秒数だけ処理を遅延させるもので、スクリプト内で特に役立ちます。

このコマンドの構文は **sleep seconds** です。

```
n1000v# sleep 30
```

スイッチ プロンプトは 30 秒後に返されます。

ヘルプの使用方法

CLI には次のヘルプ機能があります。

機能	説明
?	疑問符 (?) を入力すると、有効な入力オプションを一覧できます。
^	CLI はキャレット記号 (^) を構文行の下に印刷して、コマンドストリングのキーワードまたは引数の入力エラーを示します。
↑	↑ を使用して、直前に入力したコマンドを CLI に表示し、エラーを修正することができます。

次の例は、構文エラーの分離および状況依存ヘルプの使用方法を説明します。

コマンド	目的
<p>ステップ1 show interface virtual ?</p> <p>例:</p> <pre>n1000v# show interface virtual ? <CR> > Redirect it to a file module Limit display to interfaces on module vm Show interfaces owned by a Virtual Machine vmk Show interfaces owned by the Virtual Machine Kernel vswif Show interfaces owned by the Virtual Service Console Pipe command output to filter n1000v# show interface virtual</pre>	<p>EXEC モードで、show interface virtual コマンドと一緒に使用されるオプションのパラメータを表示します。</p>
<p>ステップ2 show interface module ?</p> <p>例:</p> <pre>n1000v# show interface module ? ^ % invalid command detected at '^' marker. n1000v#</pre>	<p>無効なコマンド エラーのメッセージを表示して、構文エラーをポイント (^) します。</p>
<p>ステップ3 Ctrl+P キーまたは↑キー</p> <p>例:</p> <pre>n1000v# <Ctrl-P> n1000v# show interface virtual ?</pre>	<p>直前に入力したコマンドを表示して、エラーを修正できます。</p>
<p>ステップ4 show interface virtual module ?</p> <p>例:</p> <pre>n1000v# show interface virtual module ? <1-256> Enter module number n1000v# show interface virtual module</pre>	<p>仮想インターフェイス モジュールを表示するための構文を表示します。</p>
<p>ステップ5 show interface virtual module 3</p> <p>例:</p> <pre>n1000v# show interface virtual module 3 ----- Port Adapter Owner Mod Host ----- n1000v#</pre>	<p>仮想インターフェイス モジュール 3 を表示します。</p>

コマンド	目的
ステップ 6 show module ? 例: <pre>n1000v# show module ? <CR> <1-66> Enter module number > Redirect it to a file internal Show line card manager related info uptime Show how long the module has been up and running vem Show Virtual Ethernet Module information Pipe command output to filter</pre>	show module コマンドのオプションのパラメータを表示します。
ステップ 7 show module 例: 例 6-1 (P.6-16)	モジュール情報を表示します。

例 6-1 ヘルプの使用方法

```
n1000v# show interface virtual ?
<CR>
>           Redirect it to a file
module      Limit display to interfaces on module
port-mapping Show hypervisor port mapping
vm          Show interfaces owned by a Virtual Machine
vmk         Show interfaces owned by the Virtual Machine Kernel
vswif       Show interfaces owned by the Virtual Service Console
|           Pipe command output to filter
n1000v# show interface module ?
^
% invalid command detected at '^' marker.
n1000v# <Ctrl-P>
n1000v# show interface virtual ?
n1000v# show interface virtual module ?
<1-256> Enter module number

n1000v# show interface virtual module ?
<1-256> Enter module number

n1000v# show interface virtual module 3

-----
Port          Adapter      Owner          Mod Host
-----
n1000v# show module ?
<CR>
<1-32> Enter module number
>       Redirect it to a file
internal Show line card manager related info
uptime  Show how long the module has been up and running
|       Pipe command output to filter

n1000v# show module
show module
Mod  Ports  Module-Type          Model          Status
```



```

-----
1    0    Virtual Supervisor Module    Nexus1000V    ha-standby
2    0    Virtual Supervisor Module    Nexus1000V    active *
3    248  Virtual Ethernet Module        NA            ok
4    248  Virtual Ethernet Module        NA            ok

Mod  Sw                Hw
---  -
1    4.0(4)SV1(0.33)  0.0
2    4.0(4)SV1(0.33)  0.0
3    4.0(4)SV1(0.33)  0.4
4    4.0(4)SV1(0.33)  0.4

Mod  MAC-Address(es)                Serial-Num
---  -
1    00-19-07-6c-5a-a8 to 00-19-07-6c-62-a8  NA
2    00-19-07-6c-5a-a8 to 00-19-07-6c-62-a8  NA
3    02-00-0c-00-03-00 to 02-00-0c-00-03-80  NA
4    02-00-0c-00-04-00 to 02-00-0c-00-04-80  NA

Mod  Server-IP                Server-UUID                Server-Name
---  -
1    10.78.1.99                NA                            NA
2    10.78.1.99                NA                            NA
3    10.78.1.92                8aca99de-16b7-300b-b572-730ea83c3de7  10.78.1.92
4    10.78.1.93                44454c4c-4800-104e-804d-b7c04f563153  10.78.1.93

```

* this terminal session

