



## コマンドライン インターフェイス

この章では、Catalyst 4500 シリーズ スイッチにおける Cisco IOS のコマンドライン インターフェイス (CLI) について理解し、それらを使用する方法について説明します。この章は、次の内容で構成されています。

- 「ヘルプの表示」 (P.1-1)
- 「コマンド オプションの検索方法」 (P.1-2)
- 「コマンド モードの概要」 (P.1-6)
- 「コマンドの no 形式および default 形式の使用」 (P.1-7)
- 「CLI 文字列検索の使用」 (P.1-7)
- 「変更した設定の保存」 (P.1-12)

Catalyst 4500 シリーズ スイッチ Cisco IOS の設定の概要については、『*Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide*』を参照してください。

### ヘルプの表示

コマンド モードで使用できるコマンドのリストを表示するには、システム プロンプトで疑問符 (?) を入力します。また、この状況依存ヘルプの機能を使用して、コマンドごとにキーワードおよび引数を表示することもできます。

表 1-1 に、コマンド モード、コマンド、キーワード、または引数に依存するヘルプを表示するために入力できるコマンドを示します。

表 1-1 ヘルプの表示

コマンド	目的
コマンドの先頭部分?	特定の文字列で始まるコマンドのリストを表示します (コマンドと疑問符の間にはスペースを入れないでください)。
コマンドの先頭部分<Tab>	途中まで入力したコマンド名の入力補完を行います。
?	コマンド モードのすべてのコマンドを一覧表示します。

表 1-1 ヘルプの表示 (続き)

コマンド	目的
コマンド?	コマンドのすべてのキーワードを一覧表示します。 コマンドと疑問符の間にスペースを入れます。
コマンド キーワード?	キーワードのすべての引数を一覧表示します。キー ワードと疑問符の間にスペースを入れます。

## コマンド オプションの検索方法

ここでは、コマンドの構文を表示する方法の例を示します。構文は任意または必須のキーワードで構成されます。コマンドのキーワードを表示するには、コマンドプロンプトで疑問符 (?) を入力するか、コマンドの一部とスペースを入力したあとに疑問符を入力します。Catalyst 4500 シリーズ スイッチにより使用可能なキーワードと、各キーワードの簡単な説明が表示されます。たとえば、グローバル コンフィギュレーション モードで **arap** コマンドのすべてのキーワードを表示するには、**arap ?** と入力します。

表 1-2 に、コマンドの入力支援を目的とした疑問符 (?) の使用方法の例について示し、次のコマンドの入力方法について順を追って説明します。

- **interface gigabitethernet 1/1**
- **channel-group 1 mode auto**

表 1-2 コマンド オプションの検索方法

コマンド	目的
Switch> <b>enable</b> Password: <password> Switch#	<b>enable</b> コマンドとパスワードを入力して、特権 EXEC コマンドにアクセスします。  特権 EXEC モードが開始されると、プロンプトが Switch# に変わります。
Switch# <b>configure terminal</b> Enter configuration commands, one per line. End with CNTL/Z. Switch(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。  グローバル コンフィギュレーション モードが開始されると、プロンプトが Switch(config)# に変わります。

表 1-2 コマンド オプションの検索方法 (続き)

コマンド	目的
<pre>Switch(config)# interface gigabitethernet ? &lt;1-9&gt; GigabitEthernet interface number Switch(config)# interface gigabitethernet 1/1 Switch(config-if)#</pre>	<p>インターフェイス コンフィギュレーション モードを開始するには、<b>interface gigabitethernet</b> グローバル コンフィギュレーション コマンドを使用して、設定するギガビットイーサネット インターフェイスを指定します。</p> <p><b>?</b> を入力し、コマンドラインで次に入力が必要な項目を表示します。この例では、<b>module-number/port-number</b> という形式で <b>1~9</b> のインターフェイス番号を入力する必要があります。</p> <p>インターフェイス コンフィギュレーション モードが開始されると、プロンプトが <code>Switch(config-if)#</code> に変わります。</p>
<pre>Switch(config-if)#? Interface configuration commands:   access-expression  Build a bridge boolean access expression   apollo              Apollo interface subcommands   appletalk           Appletalk interface subcommands   arp                 Set arp type (arpa, probe, snap) or timeout   backup              Modify backup parameters   bandwidth           Set bandwidth informational parameter   bgp-policy          Apply policy propagated by bgp community string   bridge-group        Transparent bridging interface parameters   carrier-delay       Specify delay for interface transitions   cdp                 CDP interface subcommands   channel-group       Etherchannel/port bundling configuration   clns                CLNS interface subcommands   cmns                OSI CMNS   custom-queue-list  Assign a custom queue list to an interface   decnet              Interface DECnet config commands   default             Set a command to its defaults   delay               Specify interface throughput delay   description         Interface specific description   dlsw                DLSw interface subcommands   dspu                Down Stream PU   exit                Exit from interface configuration mode   fair-queue          Enable Fair Queuing on an Interface   flowcontrol         Configure flow operation.   fras                DLC Switch Interface Command   help                Description of the interactive help system   hold-queue          Set hold queue depth   ip                  Interface Internet Protocol config commands   ipx                 Novell/IPX interface subcommands   isis                IS-IS commands   iso-igrp            ISO-IGRP interface subcommands . . . Switch(config-if)#</pre>	<p><b>?</b> を入力し、ギガビットイーサネット インターフェイスに対して使用できるすべてのインターフェイス コンフィギュレーション コマンドのリストを表示します。</p>

表 1-2 コマンド オプションの検索方法 (続き)

コマンド	目的
<pre>Switch(config-if)# <b>channel-group</b> ?   group channel-group of the interface  Switch(config-if)#channel-group</pre>	<p>コントローラに対して設定するコマンドを入力します。この例では <b>channel-group</b> コマンドを使用しています。</p> <p>? を入力し、コマンドラインで次に入力が必要な項目を表示します。この例では、<b>group</b> キーワードを入力する必要があります。</p> <p>&lt;cr&gt; が表示されていないため、コマンドを完了するにはさらに情報を入力する必要があることがわかります。</p>
<pre>Switch(config-if)# <b>channel-group</b> ?   &lt;1-256&gt; Channel group number  Switch(config-if)#channel-group</pre>	<p><b>group</b> キーワードを入力したら、? を入力し、コマンドラインで次に入力が必要な項目を表示します。この例では、チャンネルグループ番号 (1 ~ 256) を入力する必要があります。</p> <p>&lt;cr&gt; が表示されていないため、コマンドを完了するにはさらに情報を入力する必要があることがわかります。</p>
<pre>Switch(config-if)# <b>channel-group 1</b> ?   mode Etherchannel Mode of the interface  Switch(config-if)#</pre>	<p>チャンネルグループ番号を入力したら、? を入力し、コマンドラインで次に入力が必要な項目を表示します。この例では、<b>mode</b> キーワードを入力する必要があります。</p> <p>&lt;cr&gt; が表示されていないため、コマンドを完了するにはさらに情報を入力する必要があることがわかります。</p>
<pre>Switch(config-if)# <b>channel-group 1 mode</b> ?   auto      Enable PAgP only if a PAgP device is detected   desirable Enable PAgP unconditionally   on        Enable Etherchannel only  Switch(config-if)#</pre>	<p><b>mode</b> キーワードを入力したら、? を入力し、コマンドラインで次に入力が必要な項目を表示します。この例では、<b>auto</b>、<b>desirable</b>、または <b>on</b> キーワードを入力する必要があります。</p> <p>&lt;cr&gt; が表示されていないため、コマンドを完了するにはさらに情報を入力する必要があることがわかります。</p>

表 1-2 コマンド オプションの検索方法 (続き)

コマンド	目的
<pre>Switch(config-if) # channel-group 1 mode auto ? &lt;cr&gt;  Switch(config-if) #</pre>	<p>この例では、<b>auto</b> キーワードを入力しています。<b>auto</b> キーワードを入力したら、<b>?</b>を入力し、コマンドラインで次に入力が必要な項目を表示します。</p> <p>&lt;cr&gt; が表示されているので、<b>Return</b> を押してコマンドを完了できます。他のキーワードが一欄に表示されている場合、キーワードを追加入力するか、<b>Return</b> を押してコマンドを終了します。</p>
<pre>Switch(config-if) # channel-group 1 mode auto Switch(config-if) #</pre>	<p>この例では、<b>Return</b> を押してコマンドを終了しています。</p>

## コマンド モードの概要

Catalyst 4500 シリーズ スイッチの Cisco IOS ユーザ インターフェイスにはさまざまなモードがあります。使用できるコマンドの種類は、現在のモードによって異なります。システム プロンプトで疑問符 (?) を入力すると、コマンド モードごとに利用できるコマンドのリストを表示できます。

Catalyst 4500 シリーズ スイッチでセッションを開始するときは、まずユーザ モード (EXEC モードとも呼ばれます) を開始します。EXEC モードでは、一部のコマンドしか使用できません。すべてのコマンドを使用できるようにするには、特権 EXEC モードを開始する必要があります。特権 EXEC モードを開始するには、通常、パスワードが必要です。特権 EXEC モードでは、すべての EXEC コマンドが使用でき、またグローバル コンフィギュレーション モードを開始することもできます。大部分の EXEC コマンドは一時的なものです。例として、特定のアイテムの現在の状態を表示する **show** コマンドや、カウンタやインターフェイスを消去する **clear** コマンドがあげられます。Catalyst 4500 シリーズ スイッチの再起動時に EXEC コマンドは保存されません。

コンフィギュレーション モードを使用すると、実行コンフィギュレーションに変更を加えることができます。コンフィギュレーションに対する変更を保存すると、Catalyst 4500 シリーズ スイッチを再起動してもその変更内容が保持されます。グローバル コンフィギュレーション モードでは、インターフェイス コンフィギュレーション モード、サブインターフェイス コンフィギュレーション モード、およびその他のプロトコル固有のモードを開始できます。

ROM モニタ モードとは、Catalyst 4500 シリーズ スイッチを正常に起動できない場合に使用する個別のモードです。Catalyst 4500 シリーズ スイッチやアクセス サーバが起動時に有効なシステム イメージを検出できない場合、または起動時にコンフィギュレーション ファイルが破損していた場合、システムは ROM モニタ モードを開始する場合があります。

表 1-3 に、主なコマンド モードの概要を示します。

表 1-3 主なコマンド モードの概要

コマンド モード	アクセス方法	プロンプト	終了方法
ユーザ EXEC モード	ログインします。	Switch>	<b>logout</b> コマンドを使用します。
特権 EXEC モード	ユーザ EXEC モードから、 <b>enable</b> EXEC コマンドを入力します。	Switch#	ユーザ EXEC モードに戻る場合は、 <b>disable</b> コマンドを入力します。  グローバル コンフィギュレーション モードから、 <b>configure terminal</b> 特権 EXEC コマンドを入力します。
グローバル コンフィギュレーション モード	特権 EXEC モードから、 <b>configure terminal</b> 特権 EXEC コマンドを入力します。	Switch(config)#	特権 EXEC モードに戻る場合は、 <b>exit</b> または <b>end</b> コマンドを入力するか、Ctrl+Z を押します。  インターフェイス コンフィギュレーション モードを開始するには、 <b>interface</b> コンフィギュレーション コマンドを入力します。
インターフェイス コンフィギュレーション モード	グローバル コンフィギュレーション モードで、 <b>interface</b> コマンドでインターフェイスを指定して開始します。	Switch(config-if)#	グローバル コンフィギュレーション モードに戻る場合は、 <b>exit</b> コマンドを入力します。  特権 EXEC モードに戻る場合は、 <b>exit</b> コマンドを入力するか、Ctrl+Z を押します。  サブインターフェイス コンフィギュレーション モードを開始するには、 <b>interface</b> コマンドを使用してサブインターフェイスを指定します。

表 1-3 主なコマンド モードの概要 (続き)

コマンド モード	アクセス方法	プロンプト	終了方法
サブインターフェイス コンフィギュレーション	インターフェイス コンフィギュレーション モードで、 <b>interface</b> コマンドを使用してサブインターフェイスを指定します。	Switch(config-subif)#	グローバル コンフィギュレーション モードに戻る場合は、 <b>exit</b> コマンドを入力します。  特権 EXEC モードを開始するには、 <b>end</b> コマンドを使用するか、 <b>Ctrl+Z</b> を押します。
ROM モニタ	特権 EXEC モードから、 <b>reload EXEC</b> コマンドを入力します。システムの起動時、最初の 60 秒以内に <b>Break</b> キーを押します。	Rommon>	ROM モニタ モードを終了する場合は、 <b>boot</b> コマンドを入力してイメージをリロードする必要があります。ファイル名または他の起動命令を指定せずに <b>boot</b> コマンドを使用すると、システムはデフォルトのフラッシュ イメージ (オンボードフラッシュ メモリ上の最初のイメージ) から起動します。また、特定のフラッシュ イメージから起動するように指定することもできます ( <b>boot system flash filename</b> コマンドを使用)。

コマンド モードの詳細については、『*Configuration Fundamentals Configuration Guide*』の「Using the Command Line Interface」の章を参照してください。

## コマンドの no 形式および default 形式の使用

ほぼすべてのコンフィギュレーション コマンドに **no** 形式があります。通常、コマンドの機能をディセーブルにする場合に **no** 形式のコマンドを指定します。キーワード **no** を指定せずにコマンドを使用すると、ディセーブルにした機能が再びイネーブルになり、また、デフォルトでディセーブルに設定されている機能がイネーブルになります。たとえば、IP ルーティングはデフォルトでイネーブルに設定されています。IP ルーティングをディセーブルにするには、**no ip routing** コマンドを指定します。これを再度イネーブルにするには、**ip routing** を指定します。このマニュアルでは、コンフィギュレーション コマンドのすべての構文を紹介し、コマンドの **no** 形式の役割について説明します。

一部のコンフィギュレーション コマンドには、**default** 形式があります。コマンドの **default** 形式は、コマンド設定をそのデフォルト設定に戻します。大部分のコマンドはデフォルトでディセーブルに設定されているので、**default** 形式は **no** 形式と同じになります。ただし、一部のコマンドはデフォルトでイネーブルになっており、変数が特定のデフォルト値に設定されています。このような場合は、コマンドの **default** 形式を使用すると、コマンドをイネーブルにし、その変数をデフォルト値に戻すことができます。

## CLI 文字列検索の使用

コマンド出力内のパターンを文字列と呼びます。CLI 文字列検索機能を使用すると、**show** または **more** コマンドの出力を検索またはフィルタリングを行うことができ、また **--More--** プロンプトで検索やフィルタリングを行うことができます。この機能は、大量の出力をソートする場合や、出力から不要な情報を除外する場合に役立ちます。

検索機能を使用すると、指定した正規表現を含む最初の行からフィルタリングされていない出力を開始できます。コマンド 1 つあたり最大で 1 つのフィルタを指定するか、**--More--** プロンプトで新しい検索を開始できます。

正規表現とは、ソフトウェアが **show** または **more** コマンドの出力とのマッチングを行うために使用するパターン（語句、数値、またはより複雑なパターン）のことです。正規表現では大文字と小文字が区別され、また複雑な一致要件を設定することができます。単純な正規表現の例としては、「Serial」、「misses」、「138」などがあります。複雑な正規表現の例としては、「00210...」、「( is )」、「[Oo]utput」などがあります。

実行できるフィルタリングは 3 種類あります。

- **begin** キーワードを使用すると、出力は指定した正規表現を含む行から始まります。
- **include** キーワードを使用すると、出力には指定した正規表現を含む行が抽出されます。
- **exclude** キーワードを使用すると、出力では指定した正規表現を含む行が除外されます。

その後、このフィルタリングした出力を **--More--** プロンプトで検索できます。



(注)

CLI 文字列検索機能では、前の出力にさかのぼって検索やフィルタリングを行うことはできません。また、CLI に HTTP アクセスしている場合は、フィルタリングを指定できません。

## 正規表現

正規表現は、コマンド出力内の 1 文字と一致する 1 文字にすることも、コマンド出力内の複数文字と一致する複数文字にすることもできます。ここでは、単一文字パターンおよび複数文字パターンを作成する方法、および繰り返し指定、選択、位置指定、およびカッコを用いたより複雑な正規表現を作成する方法について説明します。

## 単一文字パターン

最も単純な正規表現は、コマンドの出力内の同じ 1 つの文字と一致する 1 文字のパターンです。任意の文字 (A ~ Z, a ~ z) または数字 (0 ~ 9) を 1 文字のパターンとして使用できます。また、その他のキーボード文字 (「!」や「~」など) も 1 文字のパターンとして使用できますが、一部のキーボード文字は正規表現では特別な意味を持ちます。表 1-4 に、特別な意味を持つキーボード文字の一覧を示します。

表 1-4 特別な意味を持つ文字

文字	特殊な意味
.	任意の 1 文字 (スペースを含む) と一致します。
*	0 個以上のパターンのシーケンスと一致します。
+	1 個以上のパターンのシーケンスと一致します。
?	0 または 1 個のパターンと一致します。
^	文字列の最初と一致します。
\$	文字列の最後と一致します。
_ (アンダースコア)	カンマ (,), 左波カッコ ({}, 右波カッコ (}), 左カッコ ((), 右カッコ ()), 文字列の先頭、文字列の末尾、またはスペースと一致します。

これらの特殊文字を 1 文字のパターンとして入力するには、各文字の前にバックスラッシュ (\) を付けて特別な意味を持たないようにします。次の例は、ドル記号、アンダースコア、+ 記号と一致する 1 文字のパターンです。

```
\$ \_ \+
```



単一文字パターンの範囲を指定して、コマンド出力とのマッチングを行うことができます。たとえば、文字 `a`、`e`、`i`、`o`、`u` のいずれかを含む文字列に一致する正規表現を作成できます。パターン マッチングが成功するには、これらの文字のうち 1 文字が文字列に必要です。1 文字のパターンの範囲を指定するには、1 文字のパターンを角カッコ (`[]`) で囲みます。次に例を示します。

**[aeiou]**

5 つの母音を示す小文字のアルファベットのいずれかと一致します。

**[abcdABCD]**

アルファベットの最初の 4 文字の小文字または大文字と一致します。

範囲の両端をダッシュ (`-`) でつなぐと、範囲指定を簡単にできます。上の範囲は次のように単純化されます。

**[a-dA-D]**

ダッシュを範囲内の単一文字パターンとして追加するには、ダッシュをもう 1 つ追加し、その前にバックスラッシュを入力します。

**[a-dA-D\-]**

また、右角カッコ (`]`) も 1 文字のパターンとして範囲に入れることができます。そのためには、次のように入力します。

**[a-dA-D\-]**

上の例は、大文字または小文字のアルファベットの最初の 4 文字、ダッシュ、右角カッコのいずれかに一致します。

範囲の先頭にキャレット (`^`) を追加することで、範囲の一致を反転させることができます。次の例では、範囲指定された文字以外のすべての文字と一致します。

**[^a-dqsv]**

次の例では、右角カッコ (`]`) と `d` を除くすべての文字と一致します。

**[^]d]**

## 複数文字のパターン

正規表現を作成するとき、複数の文字を含むパターンを指定することもできます。複数文字の正規表現は、文字、数字、特別な意味のないキーボード文字を組み合わせて作成します。たとえば、`a4%` は複数文字の正規表現です。特殊な意味があるキーボード文字の前にバックスラッシュを入力すると、特殊な意味がなくなります。

複数文字のパターンでは、順序が大切です。`a4%` という正規表現は、`a` という文字のあとに `4` が続き、そのあとに `%` 記号が続く文字と一致します。文字列の中に `a4%` という文字がその順序で含まれていないと、パターン マッチングは失敗します。次のような複数文字の正規表現があるとします。

**a.**

ピリオド文字の特別な意味を使用しており、`a` という文字の後に任意の文字が 1 つ来る文字列と一致します。この例では、`ab`、`a!`、または `a2` という文字列がすべてこの正規表現と一致します。

ピリオドの前にバックスラッシュを入力すると、ピリオドの特別な意味はなくなります。次のような正規表現があるとします。

**a\.**

「a.」という文字列だけが、この正規表現に一致します。

すべての文字、すべての数字、すべてのキーボード文字、文字と数字とその他のキーボード文字の組み合わせを含む複数文字の正規表現を作成できます。次の例はいずれも有効な正規表現です。

telebit 3107 v32bis

## 繰り返し指定

特殊な文字を 1 文字のパターンや複数文字のパターンと組み合わせることにより、指定された正規表現の繰り返しと一致する複雑な正規表現を作成できます。表 1-5 に、正規表現の「複数回の出現」を示す特殊文字の一覧を示します。

表 1-5 量指定子として使用される特殊文字

文字	説明
*	1 文字または複数文字のパターンが 0 回以上続くことを表す。
+	1 文字または複数文字のパターンが 1 回以上続くことを表す。
?	1 文字または複数文字のパターンが 0 回または 1 回発生することを表す。

次の例では、a が任意の回数 (0 回を含む) 続いている文字列と一致します。

**a\***

次のパターンは、a が 1 回以上続いている文字列と一致します。

**a+**

次のパターンは、文字列 bb または bab と一致します。

**ba?b**

次のパターンは、アスタリスク (\*) が任意の回数続いている文字列と一致します。

**\\*\***

複数文字のパターンとともに量指定子を使用するには、パターンをカッコで囲みます。次の例で、パターンは複数文字列 ab の任意の回数の出現と一致します。

**(ab)\***

次の例はもっと複雑で、任意のアルファベットと数字の組み合わせが 1 回以上連続している文字列と一致します (0 回、つまり空白は対象とはなりません)。

**([A-Za-z][0-9])+**

量指定子 (\*、+、または ?) を使用した一致の順序は、最長構造優先です。ネストした構造は、外側から内側に一致します。連結された構造は、構造の左側から一致します。そのため、この正規表現は A9b3 に一致しますが、9Ab3 には一致しません。これは、英字が数字の前に指定されているためです。

## 選択

選択記号を使用して、文字列に対して一致する選択肢を指定できます。選択肢は垂直線 (|) で区切ります。選択肢の 1 つだけを文字列と一致させることができます。例として、次の正規表現を見てみましょう。

**codex | telebit**

この場合、「codex」と「telebit」のうち一方の文字列とは一致しますが、「codex」と「telebit」の両方とは一致しません。

## 位置指定

正規表現パターンを文字列の先頭または末尾と一致させることができます。つまり、先頭または末尾に特定のパターンがある文字列を指定できます。表 1-6 に示す特殊文字を使用して、文字列の一部に対して、これらの正規表現の「位置指定」を行います。

**表 1-6 位置指定に使用する特殊文字**

文字	説明
^	文字列の最初と一致します。
\$	文字列の最後と一致します。

次の正規表現は、文字列が「abcd」で始まる場合のみ一致します。

**^abcd**

一方、次の正規表現の範囲指定は、a、b、c、d を除く任意の 1 文字と一致します。

**[^abcd]**

次の例の正規表現は、「.12」で終わる文字列と一致します。

**\\.12\$**

これらの位置指定文字は、特殊文字アンダースコア ( \_ ) とともに使用します。アンダースコアは文字列の先頭 (^)、文字列の末尾 (\$)、カッコ (( )), スペース ( ), 中カッコ ( { } ), コンマ ( , ), アンダースコア ( \_ ) と一致します。アンダースコア文字を使用すると、パターンが文字列中の任意の場所に存在することを指定できます。

次に例を示します。

**\_1300\_**

これは文字列内の任意の位置にある「1300」と一致します。文字列の 1300 の前後にスペース、波カッコ、カンマ、またはアンダースコアを置くことができます。次に例を示します。

**{1300\_**

この文字列はこの正規表現と一致しますが、21300 や 13000 は一致しません。

アンダースコアを使用すると、次のような多数の正規表現の組を置き換えることができます。

**^1300\$ ^1300( )1300 {1300, ,1300, {1300}, ,1300, (1300**

これは次のようにまとめられます。

**\_1300\_**

## 再帰呼び出しのカッコ

「繰り返し指定」(P.1-10) で説明したように、複数文字の正規表現とともにカッコを使用して、パターンを繰り返すことができます。この他に、カッコと 1 文字または複数文字のパターンを組み合わせることにより、パターンを正規表現内の別の場所で再使用することができます。

前のパターンを再使用する正規表現を作成するには、再使用する特定のパターンをカッコで示し、バックスラッシュ (\) の後に整数を入力してそのパターンを再使用します。この整数は、正規表現パターン内でのカッコの順序を示しています。正規表現内で複数のパターンを再使用する場合、\1 は再使用する最初のパターン、\2 は 2 番目のパターンとなります。これ以降の整数についても同様です。

次の正規表現では、再帰呼び出しのカッコを使用しています。

```
a(.)bc(.)\1\2
```

この正規表現は、先頭から順に a、任意の文字（文字 1 と呼ぶ）、bc、任意の文字（文字 2 と呼ぶ）と続いたあとに、再度、文字 1、文字 2 と続く文字列と一致します。そのため、この正規表現を aZbcTZT と一致させることができます。ソフトウェアによって、文字 1 が Z で文字 2 が T であることが記憶され、そのあとの正規表現の中で再び Z および T が使用されます。

## 変更した設定の保存

システムのリロードや停電時に設定の変更内容が失われないように設定の変更内容をスタートアップ コンフィギュレーションに保存するには、次のコマンドを入力します。

```
Switch# copy system:running-config nvram:start-up-config
Building configuration...
```

設定の保存には 1 ～ 2 分かかります。設定が保存されると、次の出力が表示されます。

```
[OK]
Switch#
```

ほとんどのプラットフォームでは、この手順によって設定が NVRAM に保存されます。クラス A フラッシュ ファイル システム プラットフォームでは、この手順によって CONFIG\_FILE 環境変数で指定されている場所に設定が保存されます。CONFIG\_FILE 環境変数のデフォルト値は NVRAM です。

## show platform コマンド

これらのコマンドは、問題解決のためにテクニカル サポート担当者と直接作業している場合にだけ使用してください。これらのコマンドは、テクニカル サポート担当者がこのコマンドの使用を推奨した場合以外には使用しないでください。



(注)

---

このマニュアルでは、**show platform** コマンドについては説明しません。

---