



# CHAPTER 5

## スイッチ スタックの管理

---

この章では、Catalyst 3750-E スイッチ スタックの管理に関する概念と手順について説明します。スイッチ コマンド リファレンスには、コマンド構文と使用に関する情報が記載されています。この章で説明する内容は、次のとおりです。

- 「スイッチ スタックの概要」 (P.5-2)
- 「スイッチ スタックの設定」 (P.5-25)
- 「特定のスタック メンバへの CLI アクセス」 (P.5-31)
- 「スイッチ スタック情報の表示」 (P.5-31)
- 「スタックのトラブルシューティング」 (P.5-32)

StackWise Plus ポートを使用したスイッチの配線方法や LED を使用してスイッチ スタック ステータスを表示する方法など、スイッチ スタックに関するその他の情報については、ハードウェア インストール ガイドを参照してください。



(注)

この章では、Catalyst 3750-E 専用スイッチ スタックの管理方法について説明します。ハードウェア / ソフトウェア スタックの管理方法、およびソフトウェア ライセンスのあるユニバーサル ソフトウェア イメージの使用方法については、Cisco.com の『*Cisco IOS Software Installation*』を参照してください。

---

## スイッチスタックの概要

スイッチスタックは、StackWise Plus または StackWise ポートを通じて接続された最大 9 台のスイッチから構成されます。1 つのスタックに 1 種類のスイッチのみを接続することも、Catalyst 3750-X、Catalyst 3750-E、Catalyst 3750 スイッチを組み合わせて接続することも可能です。Catalyst 3750-X および Catalyst 3750-E スタック メンバは StackWise Plus ポートを備え、Catalyst 3750 メンバは StackWise ポートを備えています。スタックは、次のいずれかの構成にできます。

- 同種スタック : Catalyst 3750-E スイッチのみをスタック メンバとする Catalyst 3750-E 専用スタックや、Catalyst 3750-X スイッチのみをスタック メンバとする Catalyst 3750-X 専用スタック。
- 混合スタック
  - 混合ハードウェアスタック : Catalyst 3750-X スイッチ、Catalyst 3750-E スイッチ、および 3750 スイッチがスタック メンバとして混在するスタック。  
例として、拡張 IP サービス機能をサポートした、Catalyst 3750-E および 3750 スイッチで構成されたスタックがあります。
  - 混合ソフトウェアスタック : スタック メンバとして、Catalyst 3750-X スイッチのみ、Catalyst 3750-E スイッチのみ、または Catalyst 3750 スイッチのみで構成され、各種機能をサポートします。  
例として、一部のメンバでは IP ベース フィーチャセットが稼動し、別の一部のメンバでは IP サービス フィーチャセットが稼動し、残りのメンバでは拡張 IP サービス フィーチャセットが稼動しているような Catalyst 3750-E 専用スタックがあります。
  - 混合ハードウェア/ソフトウェアスタック : 各種機能をサポートしている Catalyst 3750-X、Catalyst 3750-E、および 3750 スイッチをスタック メンバとして構成されています。  
例として、IP サービス フィーチャセットが稼動する Catalyst 3750-E メンバと、IP サービスソフトウェアイメージが稼動する Catalyst 3750 メンバから構成されたスタックがあります。

Catalyst 3750 スイッチの詳細については、『*Catalyst 3750 Switch Software Configuration Guide*』の「Managing Switch Stacks」の章を参照してください。

スイッチのうち 1 台がスタックの動作を制御します。このスイッチをスタック マスターと呼びます。スタック内のスタック マスターと他のスイッチは、すべてスタック メンバです。Catalyst 3750-E スタック メンバは、Cisco StackWise Plus テクノロジーを使用して、1 つの統合システムとして連携し動作します。レイヤ 2 およびレイヤ 3 プロトコルは、ネットワークに対して、スイッチスタック全体を単一のエンティティとして提供します。

Catalyst 3750-X、Catalyst 3750-E、Catalyst 3750 スイッチの混合スタックでは、Catalyst 3750-X スイッチをマスターにし、すべてのスタック メンバで、Cisco IOS Release 12.2(53) SE2 以降を実行することを推奨します。Catalyst 3750 イメージは、スイッチ管理を簡素化するため、Catalyst 3750-X および Catalyst 3750-E スイッチにあります。

スタックをアップグレードするには、**archive download-sw** 特権 EXEC コマンドを使用してマスターにイメージをダウンロードします。たとえば、**archive download-sw /directory tftp://10.1.1.10/c3750-ipservicesk9-tar.122-55.SE1.tar c3750e-universalk9-tar.122-55.SE1.tar** コマンドを使用してディレクトリを指定した後、メンバにダウンロードする tar ファイルのリストを指定します。

- **c3750-ipservicesk9-tar.122-55.SE1.tar** は、Catalyst 3750 メンバ用です。
- **c3750e-universalk9-tar.122-55.SE1.tar** は、Catalyst 3750-X および Catalyst 3750-E メンバ用です。

フラッシュ メモリ内のファイル リストを表示できます。

```
Switch# dir flash: c3750e-universalk9-tar.122-55.SE1
Directory of flash:/c3750e-universalk9-tar.122-55.SE1/

 5  -rwx   14313645   Mar 1 1993 00:13:55 +00:00  C3750e-universalk9-tar.122-55.SE1.tar
 6  drwx   5632      Mar 1 1993 00:15:22 +00:00  html
```

```
443 -rwx 444 Mar 1 1993 00:15:58 +00:00 info
444 -rwx 14643200 Mar 1 1993 00:04:32 +00:00 c3750-ipservicesk9-tar.122-55.SE1.tar
```

スタック マスターは、スタック全体を管理するための単一拠点となります。スタック マスターから、次のものを設定します。

- すべてのスタック メンバに適用されるシステム レベル（グローバル）の機能
- スタック メンバごとのインターフェイス レベルの機能

スイッチスタックは、ブリッジ ID によって、または、レイヤ 3 デバイスとして動作している場合はルータ MAC アドレスによって、ネットワーク内で識別されます。ブリッジ ID とルータ MAC アドレスは、スタック マスターの MAC アドレスによって決まります。各スタック メンバは、専用のスタック メンバ番号によって識別されます。

スタック メンバはすべて、スタック マスターになる条件を満たしています。スタック マスターが使用不能になると、残りのスタック メンバの中から新しいスタック マスターが選択されます。最高のスタック メンバプライオリティ値を持つスイッチが、新しいスタック マスターになります。

スタック マスターでサポートされているシステム レベルの機能は、スイッチスタック全体でサポートされます。IP ベース フィーチャセットまたは IP サービス フィーチャセットと、暗号化（暗号化をサポートする）ユニバーサルソフトウェア イメージとが稼動しているスイッチがスタックに存在する場合は、そのスイッチをスタック マスターにすることを推奨します。IP ベース フィーチャセットまたは IP サービス フィーチャセットと、非暗号化ソフトウェア イメージとが稼動しているスタック マスターでは、暗号化機能は使用できません。



(注)

混合スタックでは、Cisco IOS Release 12.2(53)SE 以前のリリースが動作する Catalyst 3750 または Catalyst 3750-E で、非暗号化イメージを実行することができます。Catalyst 3750-X スイッチ、および 12.2(53)SE より後の Cisco IOS リリースが動作する Catalyst 3750 および 3750-E スイッチでは、暗号化ソフトウェア イメージのみ実行可能です。

スタック マスターには、スイッチスタックの保存済みの実行コンフィギュレーション ファイルが格納されています。コンフィギュレーション ファイルには、スイッチスタックのシステム レベルの設定と、スタック メンバごとのインターフェイス レベルの設定が含まれます。各スタック メンバは、バックアップ目的で、これらのファイルの現在のコピーを保持します。

スイッチスタックは、単一の IP アドレスを使用して管理します。IP アドレスは、システムレベルの設定で、スタック マスターまたは他のすべてのスタック メンバで固有ではありません。スタックからスタック マスターや他のスタック メンバを削除しても、同じ IP アドレスを使用してスタックを管理できます。

次の方法を用いて、スイッチスタックを管理できます。

- Network Assistant (Cisco.com から入手できます)
- スタック メンバのコンソール ポートまたはスタック メンバのイーサネット管理ポートにシリアル接続したうえでの CLI (コマンドライン インターフェイス)
- Simple Network Management Protocol (SNMP; 簡易ネットワーク管理プロトコル) を介したネットワーク管理アプリケーション

SNMP を使用して、サポートされる MIB によって定義されるスイッチスタック全体のネットワーク機能を管理します。スイッチは、スタックのメンバーシップや選択などのスタック構成固有の機能を管理するための MIB をサポートしません。

- CiscoWorks ネットワーク管理ソフトウェア

スイッチ スタックを管理するには、次のことを理解している必要があります。

- スイッチ スタックの形成に関する概念：
  - 「スイッチ スタックのメンバーシップ」 (P.5-4)
  - 「スタック マスターの選択と再選択」 (P.5-6)
- スイッチ スタックとスタック メンバの設定方法に関する概念：
  - 「スイッチ スタック ブリッジ ID とルータ MAC アドレス」 (P.5-7)
  - 「スタック メンバ番号」 (P.5-7)
  - 「スタック メンバのプライオリティ値」 (P.5-8)
  - 「スイッチ スタックのオフライン設定」 (P.5-9)
  - 「スイッチ スタックのハードウェア互換性と SDM 不一致モード」 (P.5-11)
  - 「スイッチ スタックのソフトウェア互換性に関する推奨事項」 (P.5-12)
  - 「スタック プロトコルバージョンの互換性」 (P.5-12)
  - 「スイッチ間のメジャー バージョン番号の非互換性」 (P.5-12)
  - 「スイッチ間のマイナー バージョン番号の非互換性」 (P.5-12)
  - 「互換性のないソフトウェアおよびスタック メンバイメージのアップグレード」 (P.5-16)
  - 「スイッチ スタックのコンフィギュレーション ファイル」 (P.5-16)
  - 「スイッチ スタックのシステム全体の設定に関するその他の考慮事項」 (P.5-17)
  - 「スイッチ スタックの管理接続」 (P.5-18)
  - 「スイッチ スタックの設定のシナリオ」 (P.5-19)
  - 「ローリング スタック アップグレード」 (P.5-21)



(注)

スイッチ スタックはスイッチ クラスタとは異なります。スイッチ クラスタは、10/100/1000 ポートなどの LAN ポートを介して接続されたスイッチのセットです。スイッチ スタックとスイッチ クラスタの違いの詳細については、Cisco.com にある『*Getting Started with Cisco Network Assistant*』の「Planning and Creating Clusters」の章を参照してください。

## スイッチ スタックのメンバーシップ

スイッチ スタックは、StackWise Plus ポートを使用して接続された最大 9 台のスタック メンバから構成されます。スイッチ スタックには、常に 1 台のスタック マスターが存在します。

スタンドアロン スイッチは、スタック マスターとしても動作するスタック メンバを 1 つだけ持つスイッチ スタックです。スタンドアロン スイッチを別のスイッチと接続して (図 5-1 (P.5-5) を参照)、2 つのスタック メンバ (一方がスタック マスター) から構成されたスイッチ スタックを構築できます。スタンドアロン スイッチを既存のスイッチ スタックに接続して (図 5-2 (P.5-6) を参照)、スタック メンバーシップを増やすこともできます。

スタック メンバを同一のモデルと交換した場合、新たなスイッチは交換されたスイッチと同じメンバー番号を使用すると、交換されたスイッチとまったく同じ設定で機能します。スイッチ スタックをプロビジョニングする利点については、「スイッチ スタックのオフライン設定」(P.5-9) を参照してください。障害が発生したスイッチの交換については、ハードウェア インストール ガイドの「Troubleshooting」の章を参照してください。

スタック マスターを削除したり、電源の入ったスタンドアロン スイッチまたはスイッチ スタックを追加したりしないかぎり、メンバーシップの変更中も、スイッチ スタックの動作は中断なく継続されます。



(注) スイッチ スタックに追加または削除するスイッチの電源がオフであることを確認します。

スタック メンバを追加または削除したあとで、スイッチ スタックがすべての帯域幅 (64 Gb/s) で動作していることを確認します。スタック モード LED が点灯するまで、スタック メンバの Mode ボタンを押します。スタック内のすべてのスイッチでは、右側の最後の 2 つのポート LED がグリーンに点灯します。スイッチ モデルに応じて、右側の最後の 2 つのポートは 10 ギガビット イーサネット ポートまたは着脱可能小型フォーム ファクタ (SFP) モジュール ポート (10/100/1000 ポート) になります。スイッチの一方または両方の LED がグリーンでない場合、スタックは全帯域幅で稼動していません。

- 電源が入っているスイッチを追加すると (マージ)、マージ中のスイッチ スタックの各スタック マスターは、自分達の中から 1 台のスタック マスターを選択します。再選択されたスタック マスターは、マスターの役割と設定を保持し、スタック メンバもメンバの役割と設定を保持します。それ以前のスタック マスターを含め残りのすべてのスイッチは、リロードされ、スタック メンバとしてスイッチ スタックに参加します。それらは、スタック メンバ番号を使用可能な最小の番号に変更し、再選択されたスタック マスターのスタック設定を使用します。
- 電源がオンの状態のスタック メンバを取り外すと、スイッチ スタックがそれぞれ同じ設定を持つ複数のスイッチ スタックに分割 (パーティション化) されます。そのため、ネットワーク内で IP アドレス設定が競合してしまうことがあります。スイッチ スタックを分離されたままにしておきたい場合は、新しく作成されたスイッチ スタックの IP アドレス (複数の場合あり) を変更してください。スイッチ スタックを分割しない場合は、次の手順を実行します。
  - a. 新規に作成されたスイッチ スタックのスイッチの電源をオフにします。
  - b. 新しいスイッチ スタックを、StackWise Plus ポートを介して元のスイッチ スタックに再度接続します。
  - c. スイッチの電源を入れます。

スイッチ スタックの配線方法および電源の投入方法の詳細については、ハードウェア インストール ションガイドの「Switch Installation」の章を参照してください。

図 5-1 2 台のスタンドアロン スイッチからのスイッチ スタックの構築

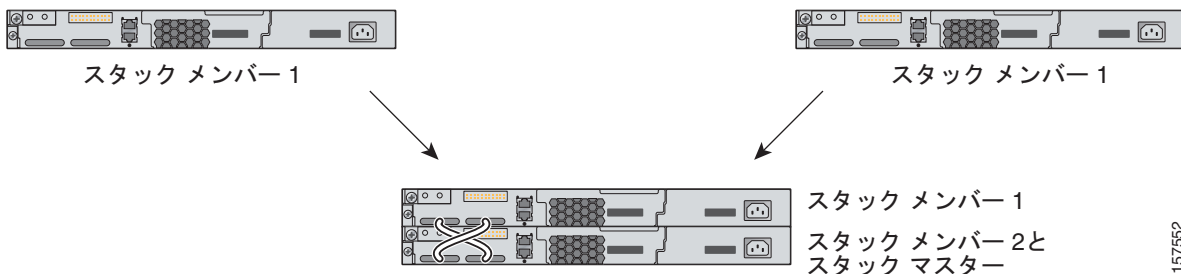
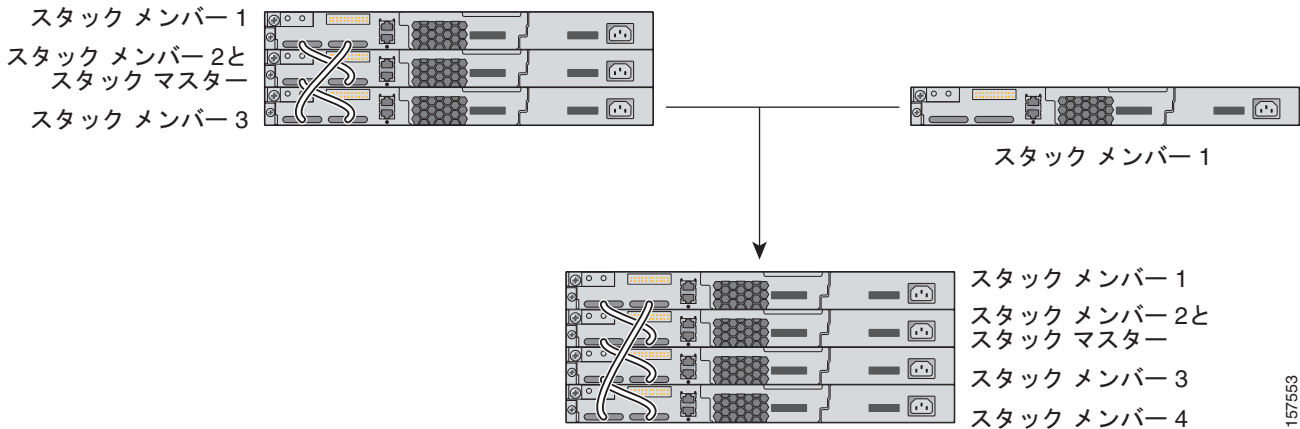


図 5-2 スタンドアロン スイッチのスイッチ スタックへの追加



## スタック マスターの選択と再選択

スタック マスターは、次にリストした順番で、いずれかのファクタに基づいて選択または再選択されます。

1. 現在スタック マスターであるスイッチ
2. 最高のスタック メンバ プライオリティ値を持つスイッチ



(注) スタック マスターにしたいスイッチには、最高のプライオリティ値を割り当てることを推奨します。これによって、再選択の実行時には、必ずそのスイッチがスタック マスターとして選択されます。

3. デフォルトのインターフェイス レベルの設定を使用していないスイッチ
4. 高いプライオリティ フィーチャセットおよびソフトウェア イメージを組み合わせたスイッチ。この組み合わせは、高いプライオリティから低いプライオリティへ順番にリストされています。
  - IP サービス フィーチャセットおよび暗号化ソフトウェアイメージ
  - IP サービス フィーチャセットおよび非暗号化ソフトウェアイメージ
  - IP ベース フィーチャセットおよび暗号化ソフトウェアイメージ
  - IP ベース フィーチャセットおよび非暗号化ソフトウェア イメージ

スタック マスター スイッチ選定においては、フィーチャセットごとの起動時刻の違いによって、スタック マスターが決められます。起動時刻が近い方のスイッチが、スタック マスターになります。

たとえば、IP サービス フィーチャセットが稼働しているスイッチが、IP ベース フィーチャセットが稼働しているスイッチより高いプライオリティを持っている場合でも、起動に 10 秒長くかかった場合は、IP ベース フィーチャセットが稼働しているスイッチの方がスタック マスターになります。この問題を回避するには、IP ベース フィーチャセットを稼働させるスイッチをアップグレードして、他方のスイッチと同じフィーチャセットとソフトウェア イメージにするか、またはマスター スイッチを手動で起動し、最低 8 秒間待機してから、IP ベース フィーチャセットを搭載した新しいメンバスイッチを起動します。

5. MAC アドレスが最小のスイッチ

スタック マスターは、次のイベントのいずれかが発生しないかぎり、役割を維持します。

- スイッチ スタックがリセットされた。\*
- スタック マスターがスイッチ スタックから削除された。
- スタック マスターがリセットされたか、電源が切れた。
- スタック マスターに障害が発生した。
- 電源の入ったスタンドアロン スイッチまたはスイッチ スタックが追加され、スイッチ スタック メンバシップが増えた。\*

アスタリスク (\*) が付いているイベントでは、示されている要素に基づいて現在のスタック マスターが再選択される場合があります。

スイッチ スタック全体に電源を入れるかリセットすると、一部のスタック メンバがスタック マスター選択に参加しない場合があります。同じ 120 秒の間に電源が投入されたスタック メンバは、スタック マスターの選択に参加し、スタック マスターとして選択される可能性があります。120 秒間経過後に電源が投入されたスタック メンバは、この初回の選択には参加しないで、スタック メンバになります。再選択には、すべてのスタック メンバが参加します。スタック マスターの選択に影響を与える電源投入に関する考慮事項については、ハードウェア インストールガイドの「Switch Installation」の章を参照してください。

数秒後、新たなスタック マスターが使用可能になります。その間、スイッチ スタックはメモリ内の転送テーブルを使用してネットワークの中断を最小限に抑えます。新たなスタック マスターが選択され、リセットされている間、他の使用可能なスタック メンバの物理インターフェイスには何も影響はありません。

新たなスタック マスターが選択され、以前のスタック マスターが使用可能になっても、以前のスタック マスターはスタック マスターとしての役割は再開しません。

ハードウェア インストールガイドに記載されているとおり、スイッチの Master LED を使用して、そのスイッチがスタック マスターかどうかを確認できます。

## スイッチ スタック ブリッジ ID とルータ MAC アドレス

ネットワーク内のスイッチ スタックは、ブリッジ ID とルータ MAC アドレスによって識別されます。スイッチ スタックが初期化すると、スタック マスターの MAC アドレスによってブリッジ ID とルータ MAC アドレスが決定します。

スタック マスターが変わると、新たなスタック マスターの MAC アドレスによって、新たなブリッジ ID とルータ MAC アドレスが決定します。ただし、固定 MAC アドレス機能がイネーブルの場合、スタック MAC アドレスは約 4 分で変更されます。この期間、前のスタック マスターがスタックに復帰すると、スイッチがスタック メンバであってスタック マスターではない場合でも、スタックはその MAC アドレスをスタック MAC アドレスとして使用し続けます。前のスタック マスターがこの期間にスタックに復帰しない場合、スイッチ スタックは新しいスタック マスターの MAC アドレスをスタック MAC アドレスとして取得します。詳細については、「[永続的 MAC アドレスのイネーブル化](#)」(P.5-25) を参照してください。

## スタック メンバ番号

スタック メンバ番号 (1 ~ 9) は、スイッチ スタック内の各メンバを識別します。また、メンバ番号によって、スタック メンバが使用するインターフェイス レベルの設定が決定します。show switch ユーザ EXEC コマンドを使用すると、スタック メンバ番号を表示できます。



新しい（つまり、スイッチ スタックに参加していない、またはスタック メンバ番号が手動で割り当てられていない）スイッチには、デフォルト スタック メンバ番号（1）が割り当てられた状態で出荷されます。このスイッチがスイッチ スタックに参加すると、デフォルト スタック メンバ番号は、スタック内で使用可能な、一番小さいメンバ番号に変更されます。

同じスイッチ スタック内のスタック メンバは、同じスタック メンバ番号を持つことはできません。スタンドアロン スイッチを含む各スタック メンバは、番号を手動で変更するか、番号がスタック内の別のメンバによってすでに使用されていないかぎり、自分のメンバ番号を保持します。

- **switch current-stack-member-number renumber new-stack-member-number** 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手動でスタック メンバ番号を変更した場合は、その番号がスタック内の他のメンバに未割り当てなときにだけ、そのスタック メンバのリセット後（または、**reload slot stack-member-number** 特権 EXEC コマンドの使用後）に、新たな番号が有効となります。詳細については、「[スタック メンバ番号の割り当て](#)」(P.5-27) を参照してください。スタック メンバ番号を変更するもう 1 つの方法は、「[環境変数の制御](#)」(P.3-22) に記載されているとおり、**SWITCH\_NUMBER** 環境変数を変更することです。

番号がスタック内の別のメンバによって使用されている場合、スイッチはスタック内で使用可能な最小の番号を選択します。

手動でスタック メンバの番号を変更し、新たなメンバ番号にインターフェイス レベルの設定が関連付けられていない場合は、スタック メンバをデフォルト設定にリセットします。スタック メンバ番号と設定の詳細については、「[スイッチ スタックのコンフィギュレーション ファイル](#)」(P.5-16) を参照してください。

プロビジョニングされたスイッチでは、**switch current-stack-member-number renumber new-stack-member-number** グローバル コンフィギュレーション コマンドを使用できません。使用すると、コマンドは拒否されます。

- スタック メンバを別のスイッチ スタックへ移動した場合、スタック メンバは、自分の番号がスタック内の別のメンバによって使用されていないときにだけ、その番号を保持します。この番号が使用されている場合、スイッチはスタック内で使用可能な最小の番号を選択します。
- スイッチ スタックをマージした場合、新たなスタック マスターのスイッチ スタックに参加したスイッチは、スタック内で使用可能な最小の番号を選択します。スイッチ スタックのマージの詳細については、「[スイッチ スタックのメンバーシップ](#)」(P.5-4) を参照してください。

ハードウェア インストールガイドに記載されているとおり、**Stack** モードのスイッチ ポート LED を使用して、各スタック メンバのスタック メンバ番号を目で見て確認できます。

## スタック メンバのプライオリティ値

スタック メンバのプライオリティ値が高いほど、スタック マスターとして選択され、自分のスタック メンバ番号を保持できる可能性が高くなります。プライオリティ値は 1 ~ 15 の範囲で指定できます。デフォルトのプライオリティ値は 1 です。**show switch** ユーザ EXEC コマンドを使用すると、スタック メンバのプライオリティ値を表示できます。



(注)

スタック マスターにしたいスイッチには、最高のプライオリティ値を割り当てることを推奨します。これによって、そのスイッチがスタック マスターとして必ず選択されます。

**switch stack-member-number priority new-priority-value** 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、スタック メンバのプライオリティ値を変更できます。詳細については、「[スタック メンバ プライオリティ値の設定](#)」(P.5-27) を参照してください。メンバプライオリティ値を変更するもう 1 つの方法は、「[環境変数の制御](#)」(P.3-22) に記載されているとおりに、**SWITCH\_PRIORITY** 環境変数を変更することです。



新しいプライオリティ値はすぐに有効となりますが、現在のスタック マスターには影響しません。新たなプライオリティ値は、現在のスタック マスターまたはスイッチ スタックのリセット時に、どのスタック メンバが新たなスタック マスターとして選択されるかを決定する場合に影響を及ぼします。

## スイッチ スタックのオフライン設定

オフライン設定機能を使用すると、新しいスイッチがスイッチ スタックに参加する前に、スイッチに割り当て（設定を割り当て）できます。現在、スタックに属していないスイッチに関連したスタック メンバ番号、スイッチ タイプ、インターフェイスを事前に設定できます。スイッチ スタックで作成した設定を割り当てられた設定と呼びます。スイッチ スタックに追加され、この設定を受信するスイッチを割り当てられたスイッチと呼びます。

**switch stack-member-number provision type** 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、割り当てられた設定を手動で作成します。スイッチ スタックにスイッチを追加する場合に、割り当てられた設定が存在しないときは、割り当てられる設定が自動的に作成されます。

割り当てられたスイッチと関連する（たとえば、VLAN の一部として）インターフェイスを設定する場合、スイッチ スタックは設定を受け入れ、その情報が実行コンフィギュレーションに表示されます。割り当てられたスイッチと関連するインターフェイスがアクティブでない場合、インターフェイスは管理上のシャットダウンをされたかのように動作し、**no shutdown** インターフェイス コンフィギュレーション コマンドはインターフェイスをアクティブ サービスに戻しません。割り当てられたスイッチと関連するインターフェイスは特定機能のディスプレイに表示されません。たとえば、インターフェイスは **show vlan** ユーザ EXEC コマンドの出力に表示されません。

スイッチ スタックは、割り当てられたスイッチがスタックに属するかどうかに関係なく、実行コンフィギュレーションに割り当てられた設定を保持します。**copy running-config startup-config** 特権 EXEC コマンドを使用すると、スタートアップ コンフィギュレーション ファイルに割り当てられた設定を保存できます。スタートアップ コンフィギュレーション ファイルでは、割り当てられたスイッチがスタックに属するかどうかに関係なく、スイッチ スタックは保存した情報をリロードして使用できます。

## 割り当てられたスイッチのスイッチ スタックへの追加による影響

割り当てられたスイッチをスイッチ スタックに追加すると、スタックは、割り当てられた設定かデフォルト設定のどちらかを適用します。表 5-1 に、スイッチ スタックがプロビジョニングされた設定とプロビジョニングされたスイッチを比較するとき発生するイベントを示します。

表 5-1 プロビジョニングされた設定とプロビジョニングされたスイッチの比較結果

シナリオ	結果
スタック メンバ番号およびスイッチ タイプが一致する	<ol style="list-style-type: none"> <li>1. プロビジョニングされたスイッチのスタック メンバ番号と、スタックのプロビジョニングされた設定のスタック メンバ番号が一致する場合、かつ</li> <li>2. プロビジョニングされたスイッチのスイッチ タイプと、スタックのプロビジョニングされた設定のスイッチ タイプが一致する場合</li> </ol> スイッチ スタックは、プロビジョニングされた設定をプロビジョニングされたスイッチに適用し、スタックに追加します。

表 5-1 プロビジョニングされた設定とプロビジョニングされたスイッチの比較結果（続き）

シナリオ	結果
スタック メンバ番号は一致するが、スイッチ タイプが一致しない	<p>スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。</p> <p>プロビジョニングされた設定は、新しい情報を反映するために変更されます。</p>
プロビジョニングされた設定でスタック メンバ番号が検出されない	<p>スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。</p> <p>プロビジョニングされた設定は、新しい情報を反映するために変更されます。</p>
プロビジョニングされたスイッチのスタック メンバ番号が既存のスタック メンバと競合する	<p>スイッチ スタックは、プロビジョニングされた設定をプロビジョニングされたスイッチに適用し、スタックに追加します。</p> <p>プロビジョニングされた設定は、新しい情報を反映するために変更されます。</p>
<p>スタック マスターは、新しいスタック メンバ番号をプロビジョニングされたスイッチに割り当てます。</p> <p>スタック メンバ番号およびスイッチ タイプが次のように一致します。</p> <ol style="list-style-type: none"> <li>1. プロビジョニングされたスイッチの新しいスタック メンバ番号と、スタックのプロビジョニングされた設定のスタック メンバ番号が一致する場合、かつ</li> <li>2. プロビジョニングされたスイッチのスイッチ タイプと、スタックのプロビジョニングされた設定のスイッチ タイプが一致する場合</li> </ol>	<p>スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。</p> <p>プロビジョニングされた設定は、新しい情報を反映するために変更されます。</p>
<p>スタック メンバ番号は一致しますが、スイッチ タイプが一致しません。</p> <ol style="list-style-type: none"> <li>1. プロビジョニングされたスイッチのスタック メンバ番号と、スタックのプロビジョニングされた設定のスタック メンバ番号が一致する場合、ただし</li> <li>2. プロビジョニングされたスイッチのスイッチ タイプと、スタックのプロビジョニングされた設定のスイッチ タイプが一致しない場合</li> </ol>	<p>スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。</p> <p>プロビジョニングされた設定は、新しい情報を反映するために変更されます。</p>
プロビジョニングされたスイッチのスタック メンバ番号が、プロビジョニングされた設定で検出されない	<p>スイッチ スタックは、デフォルト設定をプロビジョニングされたスイッチに適用し、スタックに追加します。</p>

プロビジョニングされた設定で指定されているタイプとは異なるプロビジョニングされたスイッチを、電源が切られたスイッチ スタックに追加して電力を供給すると、スイッチ スタックはスタートアップ コンフィギュレーション ファイルの（現在は不正な）**switch stack-member-number provision type** グローバル コンフィギュレーション コマンドを拒否します。ただし、スタックの初期化中は、スタート

アップ コンフィギュレーション ファイルのデフォルトでないインターフェイス コンフィギュレーション情報が、(間違っただけの可能性はある) 割り当てられたインターフェイス向けに実行されます。実際のスイッチ タイプと前述の割り当てられたスイッチ タイプの違いによって、拒否されるコマンドと、受け入れられるコマンドがあります。

たとえば、Power over Ethernet (PoE) を装備した 48 ポート スイッチ用にスイッチ スタックが割り当てられる場合、コンフィギュレーションを保存すると、スタックの電源がオフになります。それから、PoE を装備していない 24 ポート スイッチはスイッチ スタックに接続され、スタックの電源がオンになります。この状況では、ポート 25 ~ 48 の設定は拒否され、エラー メッセージが初期化中に表示されます。さらに、PoE 対応インターフェイスで有効な、設定済み PoE 関連コマンドは、ポート 1 ~ 24 に対しても拒否されます。



(注)

スイッチ スタックに新しいスイッチのプロビジョニングされた設定が含まれていない場合、スイッチはデフォルトのインターフェイス設定でスタックに参加します。スイッチ スタックは、新しいスイッチと一致する **switch stack-member-number provision type** グローバル コンフィギュレーション コマンドを実行コンフィギュレーションに追加します。

設定の詳細については、「[スイッチ スタックへの新しいメンバの割り当て](#)」(P.5-28) を参照してください。

## スイッチ スタックの割り当てられたスイッチの交換による影響

スイッチ スタック内のプロビジョニングされたスイッチに障害が発生し、スタックから取り外して別のスイッチと交換する場合、スタックはプロビジョニングされた設定またはデフォルト設定をこのスイッチに適用します。スイッチ スタックがプロビジョニングされた設定とプロビジョニングされたスイッチを比較するときに発生するイベントは、「[割り当てられたスイッチのスイッチ スタックへの追加による影響](#)」(P.5-9) で説明されているイベントと同じです。

## 割り当てられたスイッチのスイッチ スタックからの削除による影響

割り当てられたスイッチをスイッチ スタックから削除すると、削除されたスタック メンバに関連付けられた設定は、割り当てられた情報として実行コンフィギュレーション内に残ります。設定を完全に削除するには、**no switch stack-member-number provision** グローバル コンフィギュレーション コマンドを使用します。

## スイッチ スタックのハードウェア互換性と SDM 不一致モード

Catalyst 3750-E スイッチは、デスクトップ Switch Database Management (SDM; スイッチ データベース管理) テンプレートだけをサポートしています。

スタック メンバはすべて、スタック マスターに設定された SDM テンプレートを使用します。

Version-mismatch (VM; バージョン不一致) モードは、SDM 不一致モードより優先されます。VM モード条件と SDM 不一致モードが存在する場合、スイッチ スタックは先に VM モード条件を解決しようとします。

**show switch** 特権 EXEC コマンドを使用すると、スタック メンバが SDM 不一致モードになっているかどうかを確認できます。

SDM テンプレートと SDM 不一致モードの詳細については、[第 8 章「SDM テンプレートの設定」](#)を参照してください。

混合ハードウェア スタックの詳細については、Cisco.com の『*Cisco Software Activation and Compatibility Document*』を参照してください。

## スイッチ スタックのソフトウェア互換性に関する推奨事項

スタック メンバ間の完全な互換性を確保するには、この項と「[スイッチ スタックのハードウェア互換性と SDM 不一致モード](#)」(P.5-11) に記載された情報を使用します。

スタック メンバ間で互換性を確保するには、すべてのスタック メンバが同じ Cisco IOS ソフトウェア イメージおよびフィーチャ セットを稼動している必要があります。たとえば、すべてのスタック メンバでユニバーサル ソフトウェア イメージを実行し、Cisco IOS Release 12.2(35)SE2 以降に対して、IP サービス フィーチャ セットを有効にする必要があります。

詳細については、「[スタック プロトコル バージョンの互換性](#)」(P.5-12) および Cisco.com の『*Cisco Software Activation and Compatibility Document*』を参照してください。

混合ハードウェア/ソフトウェア スタックの詳細については、Cisco.com の『[Cisco Software Activation and Compatibility Document](#)』を参照してください。

## スタック プロトコル バージョンの互換性

各ソフトウェア イメージには、スタック プロトコル バージョンが含まれます。スタック プロトコル バージョンには、メジャーバージョン番号とマイナーバージョン番号があります（たとえば、1.4 の場合、1 がメジャーバージョン番号、4 がマイナーバージョン番号になります）。両方のバージョン番号によって、スタック メンバ間の互換性レベルが決定します。**show platform stack-manager all** 特権 EXEC コマンドを使用すると、スタック プロトコル バージョンを表示できます。

Cisco IOS ソフトウェア バージョンが同じスイッチは、スタック プロトコル バージョンも同じです。このようなスイッチは完全に互換可能で、すべての機能がスイッチ スタック全体に亘って適切に動作します。スタック マスターと Cisco IOS ソフトウェア バージョンが同じスイッチは、すぐにスイッチ スタックに参加します。

非互換性が混合する場合は、完全な機能を備えたスタック メンバが、特定のスタック メンバとの非互換が生じていることを示すシステム メッセージを生成します。スタック マスターは、すべてのスタック メンバに対してメッセージを送信します。詳細については、「[スイッチ間のメジャーバージョン番号の非互換性](#)」(P.5-12) の手順および「[スイッチ間のマイナーバージョン番号の非互換性](#)」(P.5-12) の手順を参照してください。

## スイッチ間のメジャーバージョン番号の非互換性

多くの場合、異なる Cisco IOS ソフトウェア バージョンのスイッチは、スタック プロトコル バージョンも異なります。メジャーバージョン番号が異なるスイッチは非互換で、同じスイッチ スタック内には存在できません。

## スイッチ間のマイナーバージョン番号の非互換性

メジャーバージョン番号は同じだがマイナーバージョン番号が異なるスイッチは、部分的に互換可能であると見なされます。スイッチ スタックに接続されている場合、部分的に互換可能なスイッチはバージョン不一致 (VM) モードになり、完全な機能を備えたメンバとしてはスタックに参加できません。ソフトウェアは不一致ソフトウェアを検出すると、スイッチ スタック イメージまたはスイッチ スタック フラッシュ メモリの tar ファイル イメージを使用して、VM モードのスイッチをアップグレー

ド (またはダウングレード) しようとしています。ソフトウェアでは、自動的なアップグレード (自動アップグレード) および自動的なアドバイス (自動アドバイス) 機能を使用します。詳細については、「[自動アップグレードおよび自動アドバイスの概要](#)」(P.5-13) を参照してください。

VM モードのスイッチの有無を確認するには、**show switch** ユーザ EXEC コマンドを使用します。VM モードのスイッチ上のポート LED はオフのままです。Mode ボタンを押しても、LED モードは変更されません。

バージョン不一致の場合にマスター スイッチがイメージを取得するために使用する URL パス名を指定するには、**boot auto-download-sw** グローバル コンフィギュレーション コマンドを使用します。

## 自動アップグレードおよび自動アドバイスの概要

ソフトウェアが不一致ソフトウェアを検出し、VM モードのスイッチをアップグレードしようとする場合は、自動アップグレードと、自動アドバイスの 2 つのソフトウェア処理を行います。

- 自動的なアップグレード (自動アップグレード) プロセスには、自動コピー プロセスと自動抽出 プロセスがあります。デフォルトでは、自動アップグレードはイネーブルです (**boot auto-copy-sw** グローバル コンフィギュレーション コマンドがイネーブルです)。自動アップグレードをディセーブルにするには、スタック マスター上で **no boot auto-copy-sw** グローバル コンフィギュレーション コマンドを使用します。**show boot** 特権 EXEC コマンドを使用し、表示された *Auto upgrade* 行を確認することで、自動アップグレードのステータスを確認できます。
  - 自動コピーでは、スタック メンバ上で稼動しているソフトウェア イメージを、VM モードで、アップグレード (自動アップグレード) 対象のスイッチに自動的にコピーします。自動コピーが実行されるのは、自動アップグレードがイネーブルの場合、VM モードのスイッチに十分なフラッシュ メモリがある場合、およびスイッチ スタックで稼動しているソフトウェア イメージが VM モードのスイッチに適している場合です。



(注) VM モードのスイッチでは、すべてのリリース済みのソフトウェアが稼動するとは限りません。たとえば、新しいスイッチ ハードウェアは以前のバージョンのソフトウェアでは認識されません。

- 自動的な抽出 (自動抽出) は、自動アップグレード プロセスが、VM モードでスイッチにコピーするソフトウェアをスタックから見つけられない場合に実行されます。この場合、自動抽出プロセスは、VM モードでスイッチ スタックまたはスイッチをアップグレードするために、VM モードにあるかどうかに関係なく、スタック内のすべてのスイッチを検索して必要な tar ファイルを探します。tar ファイルは、スタック内のどのフラッシュ ファイル システムにあってもかまいません (VM モードのスイッチを含む)。VM モードのスイッチに適した tar ファイルが見つかり、このプロセスはこのファイルを抽出し、スイッチを自動的にアップグレードします。

自動アップグレード (自動コピーおよび自動抽出) プロセスは、不一致ソフトウェアが検出されると数分間待機してから起動します。

自動アップグレード処理が完了すると、VM モードであったスイッチはリロードされ、完全な機能を備えたメンバとしてスタックに参加します。リロード中に両方の StackWise Plus ケーブルが接続されている場合、スイッチ スタックは 2 つのリング上で動作するため、ネットワーク ダウンタイムが発生しません。



(注) 自動アップグレードでは、2 つのフィーチャセットが同じタイプの場合にだけアップグレードを実行します。たとえば、IP サービス フィーチャセットの VM モードのスイッチを、IP ベース フィーチャセット (またはその逆) に自動的にアップグレードすることはありません。

- 自動的なアドバイス（自動アドバイス）は、自動アップグレードプロセスが、VM モードのスイッチにコピーするスタックメンバソフトウェアを見つけれない場合に実行されます。このプロセスによって、VM モードのスイッチスタックまたはスイッチを手動でアップグレードするために必要なコマンド（**archive copy-sw** または **archive download-sw** 特権 EXEC コマンド）およびイメージ名（**tar** ファイル名）が伝えられます。推奨されるイメージは実行中のスイッチスタックイメージ、またはスイッチスタック（VM モードのスイッチを含む）内のいずれかのフラッシュファイルシステムの **tar** ファイルです。スタックのフラッシュファイルシステムで適切なイメージが見つからない場合、自動アドバイスプロセスによって、スイッチスタックに新規ソフトウェアをインストールするように伝えられます。自動アドバイスはディセーブルにできません。また、そのステータスを確認するコマンドはありません。

スイッチスタックソフトウェアおよび VM モードのスイッチのソフトウェアに同じフィアチャーセットが含まれない場合は、自動アドバイスソフトウェアからの指示もありません。たとえば、IP ベースイメージが稼動するスイッチスタックに、IP サービスイメージが稼動するスイッチを追加した場合、自動アドバイスソフトウェアは推奨ソフトウェアを提示しません。

別のソフトウェアイメージのインストールを許可するには、**archive-download-sw /allow-feature-upgrade** 特権 EXEC コマンドを使用します。

## 自動アップグレードおよび自動アドバイスのメッセージ例

マイナーバージョン番号が異なるスイッチをスイッチスタックに追加すると、メッセージが連続して表示されます（スイッチがその他のシステムメッセージを生成しない場合）。

次に、スイッチスタックが、スイッチスタックと異なるマイナーバージョン番号を実行する新しいスイッチを検出した例を示します。自動コピーが起動し、スタックメンバから VM モードのスイッチにコピーするのに適したソフトウェアを検出し、VM モードのスイッチをアップグレードして、リロードします。

```
*Mar 11 20:31:19.247:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to state UP
*Mar 11 20:31:23.232:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack (VERSION_MISMATCH)
*Mar 11 20:31:23.291:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack (VERSION_MISMATCH) (Stack_1-3)
*Mar 11 20:33:23.248:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated for switch number(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Found donor (system #2) for
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:member(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System software to be uploaded:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type:                0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c3750e-universal-mz.122-35.SE2(directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c3750e-universal-mz.122-35.SE2/c3750e-universal-mz.122-35.SE2.bin (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c3750e-universal-mz.122-35.SE2/info (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:examining image...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c3750e-universal-mz.122-35.SE2/info (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Stacking Version Number:1.4
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type:                0x00000000
```

```

*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Ios Image File Size: 0x004BA200
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Total Image File Size:0x00818A00
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Minimum Dram required:0x08000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Suffix:ipservices-122-35.SE2
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image
Directory:c3750e-universal-mz.122-35.SE2
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Name:c3750e-universal-mz.122-35.SE2
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image
Feature:IP|LAYER_3|PLUS|MIN_DRAM_MEG=128
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Old image for switch
1:flash1:c3750e-universal-mz.122-35.SE2
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Old image will be deleted after download.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Extracting images from archive into flash on
switch 1...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:c3750e-universal-mz.122-0.0.313.SE
(directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c3750e-universal-mz.122-0.0.313.SE/c3750e-universal-mz.122-35.SE2 (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c3750e-universal-mz.122-35.SE2/info (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Installing
(renaming):`flash1:update/c3750e-universal-mz.122-0.0.313.SE2' ->
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
`flash1:c3750e-universal-mz.122-35.SE2'
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:New software image installed in
flash1:c3750e-i5-mz.122-35.SE2
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Removing old
image:flash1:c3750e-universal-mz.122-35.SE2
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:All software images installed.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Requested system reload in progress...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Software successfully copied to
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:system(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Done copying software
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Reloading system(s) 1

```

次に、スイッチスタックが、スイッチスタックと異なるマイナーバージョン番号を実行する新しいスイッチを検出した例を示します。自動コピーは起動しますが、スイッチスタックと互換可能にするための、VMモードのスイッチにコピーするソフトウェアをスイッチスタック内で検出できません。自動アドバースプロセスが起動し、ネットワークからVMモードのスイッチにtarファイルをダウンロードするように推奨されます。

```

*Mar 1 00:01:11.319:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to
state UP
*Mar 1 00:01:15.547:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH)
stack_2#
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process
initiated for switch number(s) 1
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Software was not copied
*Mar 1 00:03:15.562:%IMAGEMGR-6-AUTO_ADVISE_SW_INITIATED:Auto-advise-software process
initiated for switch number(s) 1
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:

```



```
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:Systems with incompatible software
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:have been added to the stack. The
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:storage devices on all of the stack
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:members have been scanned, and it has
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:been determined that the stack can be
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:repaired by issuing the following
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:command(s):
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW: archive download-sw /force-reload
/overwrite /dest 1 flash1:c3750e-universal-mz.122-35.SE2.tar
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
```

**archive download-sw** 特権 EXEC コマンドの使用の詳細については、「ソフトウェア イメージの操作」(P.A-27) を参照してください。



(注)

自動アドバイスおよび自動コピーでは、**info** ファイルの調査およびスイッチ スタック上の ディレクトリ構造の検索により、実行中のイメージを識別します。**archive download-sw** 特権 EXEC コマンドではなく、**copy ftp:** ブートローダ コマンドを使用してイメージをダウンロードすると、正しいディレクトリ構造が作成されません。**info** ファイルの詳細については、「サーバまたは Cisco.com 上のイメージのファイル形式」(P.A-28) を参照してください。

## 互換性のないソフトウェアおよびスタック メンバ イメージのアップグレード

**archive copy-sw** 特権 EXEC コマンドを使用すると、互換性のないユニバーサル ソフトウェア イメージのあるスイッチをアップグレードできます。コマンドを使用すると、既存のスタック メンバのソフトウェア イメージを、互換性のないソフトウェアを移動しているメンバにコピーできます。その場合、スイッチは自動的にリロードされ、完全に機能しているメンバとしてスタックに加入します。詳細については、「あるスタック メンバから別のスタック メンバへのイメージファイルのコピー」(P.A-44) を参照してください。

## スイッチ スタックのコンフィギュレーション ファイル

コンフィギュレーション ファイルには、次の設定情報が格納されています。

- すべてのスタック メンバに適用されるシステム レベル (グローバル) コンフィギュレーション設定: IP、STP (スパンニングツリー プロトコル)、VLAN、SNMP 設定など
- スタック メンバのインターフェイス固有のコンフィギュレーション設定: 各スタック メンバに固有

スタック マスターには、スイッチ スタックの保存済みの実行コンフィギュレーション ファイルが格納されています。すべてのスタック メンバは、定期的に、スタック マスターからコンフィギュレーション ファイルの同期化されたコピーを受け取ります。スタック マスターが使用不能になると、スタック マスターの役割を引き受けたスタック メンバが最新のコンフィギュレーション ファイルを保持します。



(注)

スタック メンバでは Cisco IOS Release 12.2(35)SE2 以降を稼働させることを推奨します。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保存しないでスタック マスターが交換されても、スタック マスターのインターフェイス固有設定は保存されます。

新規のスイッチがスイッチ スタックに参加した場合、そのスイッチはスイッチ スタックのシステム レベルの設定を使用します。スイッチが別のスイッチ スタックに移動された場合、そのスイッチは保存済みのコンフィギュレーション ファイルを失い、新たなスイッチ スタックのシステム レベルの設定を使用します。

各スタック メンバのインターフェイス固有のコンフィギュレーションには、スタック メンバ番号が関連付けられます。「[スタック メンバ番号](#)」(P.5-7)に記載されているとおり、スタック メンバは、番号が手動で変更されるか、同じスイッチ スタック内の別のメンバによってすでに使用されているかしないかぎり、自分の番号を保持します。

- そのメンバ番号に対応するインターフェイス固有のコンフィギュレーションが存在しない場合は、スタック メンバはデフォルトのインターフェイス固有のコンフィギュレーションを使用します。
- そのメンバ番号に対応するインターフェイス固有のコンフィギュレーションが存在する場合は、スタック メンバはそのメンバ番号に関連付けられたインターフェイス固有のコンフィギュレーションを使用します。

スタック メンバに障害が生じ、それを同一のモデルと交換した場合、交換後のスイッチは自動的に、障害の生じたスイッチと同じインターフェイス固有のコンフィギュレーションを使用します。そのため、インターフェイス設定を再設定する必要はありません。交換後のスイッチは、障害の生じたスイッチと同じスタック メンバ番号を持つ必要があります。スイッチ スタックをプロビジョニングする利点については、「[スイッチ スタックのオフライン設定](#)」(P.5-9)を参照してください。

スタンドアロン スイッチのコンフィギュレーションの場合と同じ方法で、スタック コンフィギュレーションをバックアップし復元します。ファイル システムとコンフィギュレーション ファイルの詳細については、[付録 A 「Cisco IOS ファイル システム、コンフィギュレーション ファイル、およびソフトウェア イメージの操作」](#)を参照してください。

## スイッチ スタックのシステム全体の設定に関するその他の考慮事項

ここでは、スイッチ スタックにシステム全体の機能を設定する場合にさらに考慮すべき事項について説明します。

- Cisco.com から入手できる『*Getting Started with Cisco Network Assistant*』の「Planning and Creating Clusters」の章
- 「[MAC アドレスとスイッチ スタック](#)」(P.7-15)
- 「[SDM テンプレートの設定](#)」(P.8-5)
- 「[802.1x 認証とスイッチ スタック](#)」(P.10-11)
- 「[VTP とスイッチ スタック](#)」(P.14-8)
- 「[プライベート VLAN とスイッチ スタック](#)」(P.16-6)
- 「[スパニング ツリーとスイッチ スタック](#)」(P.18-13)
- 「[MSTP とスイッチ スタック](#)」(P.19-8)
- 「[DHCP スヌーピングとスイッチ スタック](#)」(P.22-8)
- 「[IGMP スヌーピングとスイッチ スタック](#)」(P.24-7)
- 「[ポート セキュリティとスイッチ スタック](#)」(P.26-20)
- 「[CDP とスイッチ スタック](#)」(P.27-2)
- 「[SPAN と RSPAN とスイッチ スタック](#)」(P.30-11)
- 「[ACL とスイッチ スタック](#)」(P.35-7)
- 「[EtherChannel とスイッチ スタック](#)」(P.38-10)

- 「IP ルーティングおよびスイッチ スタック」 (P.40-3)
- 「IPv6 とスイッチ スタック」 (P.41-11)
- 「HSRP およびスイッチ スタック」 (P.42-5)
- 「マルチキャスト ルーティングおよびスイッチ スタック」 (P.46-11)
- 「フォールバック ブリッジングおよびスイッチ スタック」 (P.48-3)

## スイッチ スタックの管理接続

スイッチ スタックおよびスタック メンバ インターフェイスは、スタック マスターを経由して管理します。CLI、SNMP、Network Assistant、および CiscoWorks ネットワーク管理アプリケーションを使用できます。スイッチごとにスタック メンバを管理することはできません。

ここでは、スイッチ スタック接続情報について説明します。

- 「IP アドレスによるスイッチ スタックへの接続」 (P.5-18)
- 「SSH セッションによるスイッチ スタックへの接続」 (P.5-18)
- 「コンソール ポートまたはイーサネット管理ポートによるスイッチ スタックへの接続」 (P.5-19)
- 「特定のスタック メンバへの接続」 (P.5-19)

## IP アドレスによるスイッチ スタックへの接続

スイッチ スタックは、単一 IP アドレスを介して管理されます。IP アドレスは、システムレベルの設定で、スタック マスターまたは他のすべてのスタック メンバで固有ではありません。IP 接続が確保されている前提で、スタックからスタック マスターまたは他のすべてのスタック メンバを削除した場合でも、同じ IP アドレスを介してスタックを管理できます。



(注)

スイッチ スタックからスタック メンバを削除した場合、各スタック メンバは自身の IP アドレスを保持します。したがって、ネットワーク内で同じ IP アドレスを持つ 2 つのデバイスが競合するのを避けるため、スイッチ スタックから削除したスイッチの IP アドレスを変更しておきます。

スイッチ スタックの設定に関連する情報については、「[スイッチ スタックのコンフィギュレーション ファイル](#)」 (P.5-16) を参照してください。

## SSH セッションによるスイッチ スタックへの接続

暗号化ソフトウェア イメージと IP ベース フィーチャ セットまたは IP サービス フィーチャ セットとが稼動するスタック マスターに障害が発生する場合、またはこのスタック マスターが交換されて、非暗号化ソフトウェア イメージと IP ベース フィーチャ セットまたは IP サービス フィーチャ セットとが稼動するスイッチになった場合、スイッチ スタックへの Secure Shell (SSH; セキュア シェル) 接続が失われることがあります。暗号化ソフトウェア イメージと、IP ベースまたは IP サービス フィーチャ セットとが稼動しているスイッチをスタック マスターにすることを推奨します。スタック マスターで非暗号化ソフトウェア イメージが稼動していると、暗号化機能は利用できません。



(注)

非暗号化イメージは、Cisco IOS Release 12.2(53)SE 以前のリリースが動作する Catalyst 3750 または Catalyst 3750-E でのみ使用できます。Catalyst 3750-X スイッチは、暗号化ソフトウェア イメージのみを実行します。

## コンソール ポートまたはイーサネット管理ポートによるスイッチ スタックへの接続

スタック マスターに接続するには、次のいずれかの方法を使用します。

- 1 つまたは複数のスタック メンバのコンソール ポートを経由して、ターミナルまたは PC をスタック マスターに接続できます。
- 1 つまたは複数の Catalyst 3750-E スタック メンバのイーサネット管理ポートを経由して、PC をスタック マスターに接続できます。イーサネット管理ポートを経由してスイッチ スタックに接続する方法については、「イーサネット管理ポートの使用」(P.12-19) を参照してください。

複数の CLI セッションをスタック マスターに使用する場合は注意が必要です。1 つのセッションで入力したコマンドは、別のセッションには表示されません。そのため、コマンドを入力したセッションを識別できなくなることがあります。

スイッチ スタックを管理する場合は、1 つの CLI セッションだけを使用することを推奨します。

## 特定のスタック メンバへの接続

特定のスタック メンバ ポートを設定する場合は、CLI コマンド インターフェイス表記にスタック メンバ番号を含めてください。詳細については、「インターフェイス コンフィギュレーション モードの使用方法」(P.12-13) を参照してください。

特定のスタック メンバをデバッグする場合は、**session stack-member-number** 特権 EXEC コマンドでスタック マスターからアクセスできます。スタック メンバ番号は、システム プロンプトに追加されます。たとえば、Switch-2# はスタック メンバ 2 の特権 EXEC モードのプロンプトであり、スタック マスターのシステム プロンプトは Switch です。特定のスタック メンバへの CLI セッションで使用できるのは、**show** コマンドと **debug** コマンドに限ります。

## スイッチ スタックの設定のシナリオ

表 5-2 では、スイッチ スタック コンフィギュレーションのシナリオを示します。大半のシナリオは、少なくとも 2 台のスイッチが StackWise Plus ポートを使用して接続されていることを前提にしています。

表 5-2 スイッチ スタックの設定のシナリオ

シナリオ	結果	
既存のスタック マスターによって、明確に決定されるスタック マスター選択	StackWise Plus ポートを使用して 2 つの電源の入ったスイッチ スタックを接続します。	2 つのスタック マスターの一方だけが、新たなスタック マスターになります。その他のスタック メンバはどれも、スタック マスターにはなりません。
スタック メンバのプライオリティ値によって、明確に決定されるスタック マスター選択	<ol style="list-style-type: none"> <li>1. StackWise Plus ポートを使用して、2 台のスイッチを接続します。</li> <li>2. <b>switch stack-member-number priority new-priority-number</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、1 つのスタック メンバに、より高いメンバプライオリティ値を設定します。</li> <li>3. 両方のスタック メンバを同時に再起動します。</li> </ol>	より高いプライオリティ値を持つスタック メンバがスタック マスターとして選択されます。

表 5-2 スイッチ スタックの設定のシナリオ (続き)

シナリオ	結果	
<p>コンフィギュレーションファイルによって、明確に決定されるスタック マスター選択</p>	<p>両方のスタック メンバが同じプライオリティ値を持つものと仮定します。</p> <ol style="list-style-type: none"> <li>一方つのスタック メンバがデフォルトのコンフィギュレーションを持ち、他方のスタック メンバが保存済み（デフォルトでない）のコンフィギュレーションファイルを持つことを確認します。</li> <li>両方のスタック メンバを同時に再起動します。</li> </ol>	<p>保存済みのコンフィギュレーション ファイルを持つスタック メンバがスタック マスターとして選択されます。</p>
<p>暗号化ソフトウェア イメージと IP サービス フィーチャ セットによって明確に決定されるスタック マスターの選択</p>	<p>すべてのスタック メンバが同じプライオリティ値を持つものと仮定します。</p> <ol style="list-style-type: none"> <li>一方のスタック メンバに暗号化イメージがインストールされて IP サービス フィーチャ セットがイネーブルになっており、他方のスタック メンバに非暗号化イメージがインストールされて IP サービス フィーチャ セットがイネーブルになっていることを確認します。</li> <li>両方のスタック メンバを同時に再起動します。</li> </ol>	<p><b>(注)</b> 暗号化ソフトウェア イメージと IP サービス フィーチャ セットを備えたスタック メンバがスタック マスターとして選択されます。</p>
<p>暗号化ソフトウェア イメージと IP ベース フィーチャ セットによって明確に決定されるスタック マスター選択</p>	<p>すべてのスタック メンバが同じプライオリティ値を持つものと仮定します。</p> <ol style="list-style-type: none"> <li>一方のスタック メンバに暗号化イメージがインストールされて IP ベース フィーチャ セットがイネーブルになっており、他方のスタック メンバに非暗号化イメージがインストールされて IP ベース フィーチャ セットがイネーブルになっていることを確認します。</li> <li>両方のスタック メンバを同時に再起動します。</li> </ol>	<p><b>(注)</b> 暗号化イメージと IP ベース フィーチャ セットを備えたスタック メンバがスタック マスターとして選択されます。</p>
<p>MAC アドレスによって、明確に決定されるスタック マスター選択</p>	<p>両方のスタック メンバが同じプライオリティ値、コンフィギュレーション ファイル、フィーチャ セットを持つものと仮定し、両方のスタック メンバを同時に再起動します。</p>	<p>より小さい MAC アドレスを持つスタック メンバがスタック マスターとして選択されます。</p>

表 5-2 スイッチ スタックの設定のシナリオ (続き)

シナリオ		結果
スタック メンバ番号の競合	<p>一方のスタック メンバが他方のスタック メンバより高いプライオリティ値を持つものと仮定します。</p> <ol style="list-style-type: none"> <li>両方のスタック メンバが同じスタック メンバ番号を持つように確認します。必要に応じて、<b>switch current-stack-member-number renumber new-stack-member-number</b> グローバル コンフィギュレーション コマンドを使用します。</li> <li>両方のスタック メンバを同時に再起動します。</li> </ol>	より高いプライオリティ値を持つスタック メンバが、自分のスタック メンバ番号を保持します。もう一方のスタック メンバは、新たなスタック メンバ番号を持ちます。
スタック メンバの追加	<ol style="list-style-type: none"> <li>新しいスイッチの電源を切ります。</li> <li>StackWise Plus ポートを使用して、新たなスイッチを電源の入ったスイッチ スタックに接続します。</li> <li>新しいスイッチの電源を入れます。</li> </ol>	スタック マスターはそのままです。新たなスイッチがスイッチ スタックに追加されます。
スタック マスターの障害	スタック マスターを削除します (または、電源を切ります)。	「スタック マスターの選択と再選択」(P.5-6) に記載されたファクタに基づき、残りのスタック メンバのいずれかが新たなスタック マスターになります。スタック内の他のすべてのスタック メンバは、スタック メンバのまま、再起動はされません。
9 台を超えるスタック メンバの追加	<ol style="list-style-type: none"> <li>StackWise Plus ポートを使用して、10 台のスイッチを接続します。</li> <li>すべてのスイッチの電源を入れます。</li> </ol>	<p>2 台のスイッチがスタック マスターになります。一方のスタック マスターが 9 つのスタック メンバを制御します。もう一方のスタック マスターは、スタンドアロン スイッチとして維持されます。</p> <p>Mode ボタンとスイッチのポート LED を使用して、どのスイッチがスタック マスターで、各スイッチ マスターにはどのスイッチが属しているかを識別できます。Mode ボタンと LED の使用方法については、ハードウェア インストール ガイドを参照してください。</p>

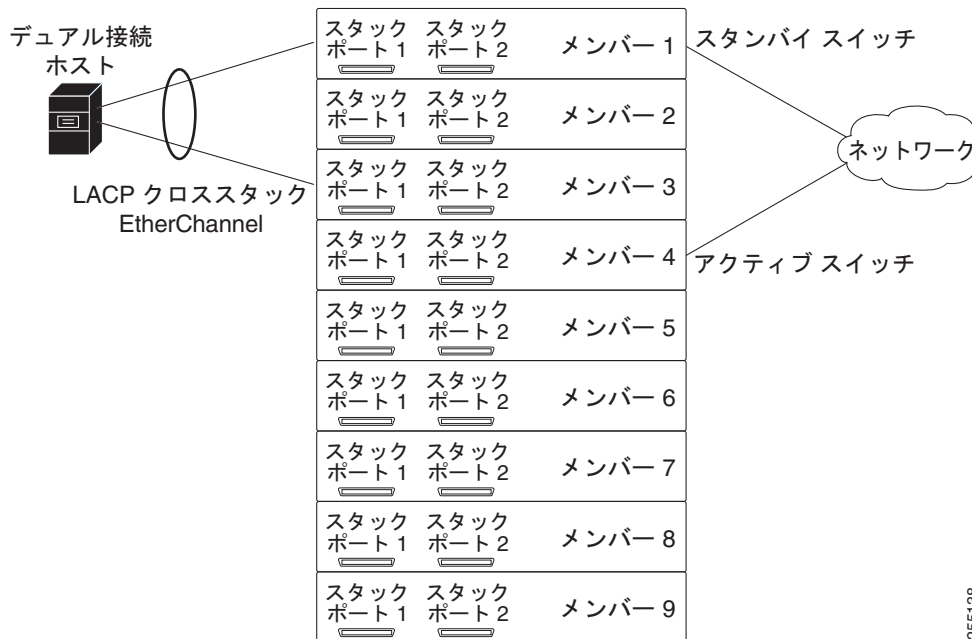
## ローリング スタック アップグレード

スタックをアップグレードまたはダウングレードすると、その後にスタックがリロードし、接続されているホストはネットワーク接続を失います。スタックに冗長リンクがある場合に限り、ローリング スタック アップグレード機能を使用してネットワークの中断を最小限に抑えられます。メンバは 1 つずつアップグレードされます。

## スタック設定

次の設定のスタック上でローリング スタック アップグレードを実行できます。

図 5-3 冗長リンクのあるスタック



- デュアル接続ホスト（サーバやアクセス ポイントなど）が、Link Aggregation Control Protocol (LACP) クロススタック EtherChannel 経由で 2 つのメンバに接続されています。
- スタックが次のように設定されています。
  - 冗長接続を持つスタック リング内でメンバがそれぞれの StackWise Plus ポートを介して接続されています。設定例については、スイッチのハードウェア インストール ガイドを参照してください。
  - アップグレード中、スイッチが固定 MAC アドレスをイネーブルにします。
- 少なくとも 1 つの冗長アップリンクがネットワークに接続されています。このアップリンクには、アクティブ スイッチとスタンバイ スイッチがあります。
  - アクティブ ロールのインターフェイスを持つメンバはアクティブ スイッチです。
  - スタンバイ ロールのインターフェイスを持つもう 1 つのメンバはスタンバイ スイッチです。

## アップグレード プロセス

このプロセスを開始する前に、ネットワークへの冗長アップリンクを設定して、スタックのネットワーク接続を確保します。また、スタック起動時間も設定できます。

**archive download-sw /rolling-stack-upgrade** 特権 EXEC コマンドを入力して、スタック アップグレードを開始します。アップグレード中、メンバ アップグレード シーケンスやアップグレード中ではないメンバのアップグレード ステータスを表示できます。



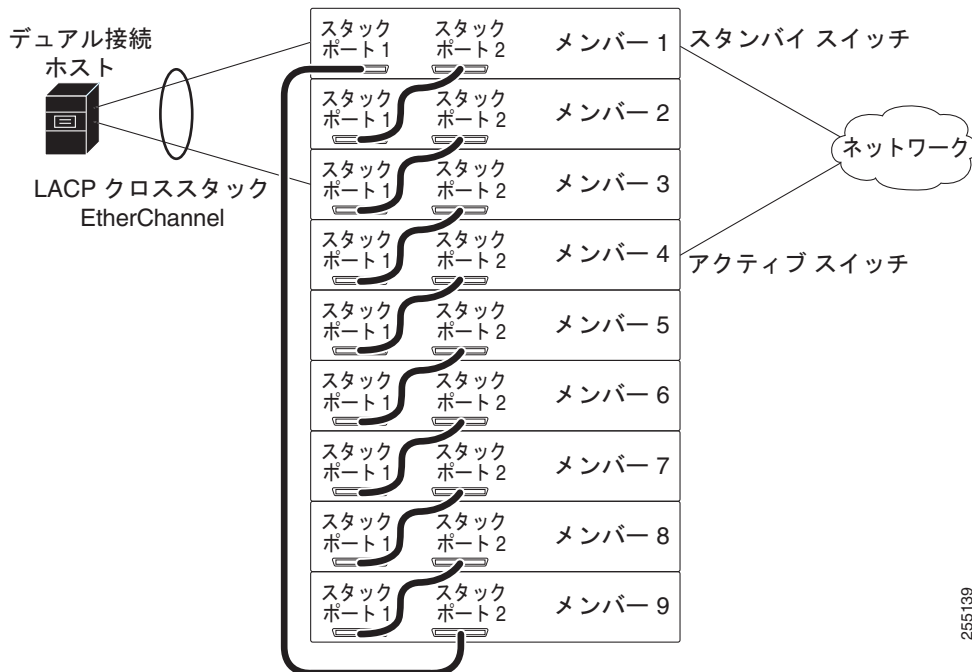
このプロセスは、ソフトウェア内で進行します。

1. スタックは、アップグレードされていないスタックとアップグレード済みのスタックに分割されます。これらのスタックは、トラフィックを交換しない別々のスタックとして動作します。アップグレードされていないスタック内のレイヤ3 インターフェイスは非アクティブです。
2. ソフトウェアにより、アップグレード シーケンスが決定されます。
  - a. 最初に、スタンバイ メンバがアップグレードされます。
  - b. 最初のスタンバイ メンバ上のスタック ポート 1 から到達可能なスタンバイ メンバが、アクティブ メンバに到達するまで順番にアップグレードされます。
  - c. その後、最初のスタンバイ メンバ上のスタック ポート 2 から到達可能なスタンバイ メンバが、アクティブ メンバに到達するまで順番にアップグレードされます。

スタックのアップグレード完了後、スタック設定をコンフィギュレーションファイルに保存します。スタックで元のマスターを維持し、新規のマスターを選択しないようにする場合は、スタックをリロードします。

## アップグレード シーケンスの例

図 5-4 メンバ 1 上のスタック ポート 1 がメンバ 9 に接続されている場合

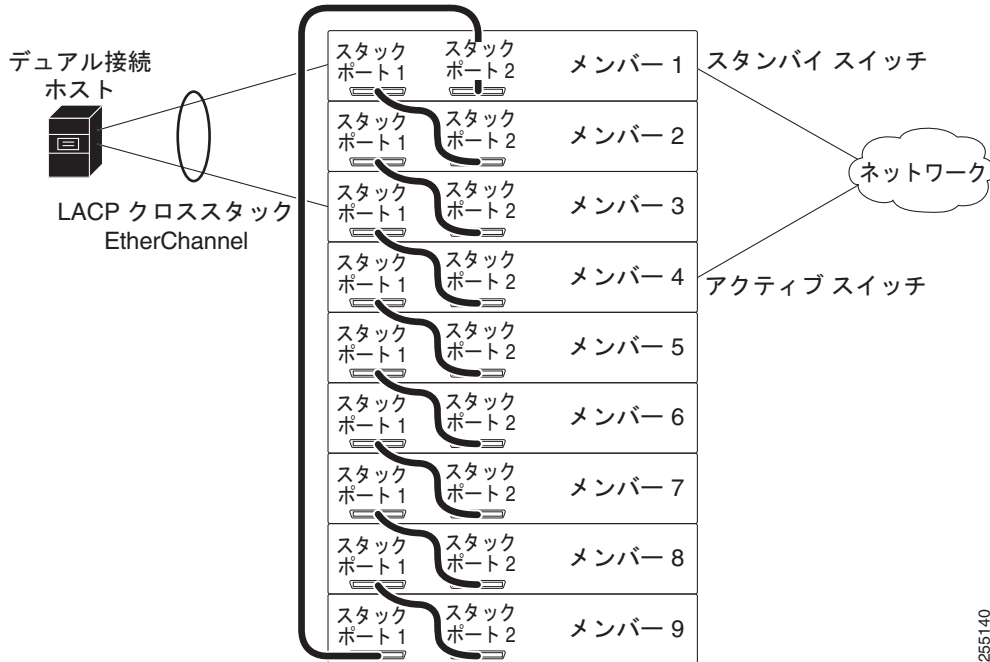


メンバ 1（最初のスタンバイ スイッチ）上のスタック ポート 1 がメンバ 9 上のスタック ポート 2 に接続されている場合のアップグレード シーケンスは次のとおりです。

1. メンバ 1
2. メンバ 9
3. メンバ 8
4. メンバ 7
5. メンバ 6

6. メンバ 5
7. メンバ 2
8. メンバ 3
9. メンバ 4

図 5-5 メンバ 1 上のスタック ポート 1 がメンバ 2 に接続されている場合



メンバ 1 (最初のスタンバイ スイッチ) 上のスタック ポート 1 がメンバ 2 上のスタック ポート 2 に接続されている場合のアップグレード シーケンスは次のとおりです。

1. メンバ 1
2. メンバ 2
3. メンバ 3
4. メンバ 9
5. メンバ 8
6. メンバ 7
7. メンバ 6
8. メンバ 5
9. メンバ 4

255140

## スイッチ スタックの設定

ここでは、次の設定について説明します。

- 「デフォルトのスイッチ スタック設定」 (P.5-25)
- 「永続的 MAC アドレスのイネーブル化」 (P.5-25)
- 「スタック メンバ情報の割り当て」 (P.5-27)
- 「ローリング スタック アップデートの実行」 (P.5-29)

### デフォルトのスイッチ スタック設定

表 5-3 に、デフォルトのスイッチ スタック設定を示します。

表 5-3 デフォルトのスイッチ スタック設定

機能	デフォルト設定
スタック MAC アドレス タイマー	ディセーブル
スタック メンバ番号	1
スタック メンバのプライオリティ値	1
オフライン設定	スイッチ スタックはプロビジョニングされていません。

### 永続的 MAC アドレスのイネーブル化

スイッチ スタック MAC アドレスは、スタック マスターの MAC アドレスによって決まります。スタック マスターがスタックから削除されて新しいスタック マスターに引き継がれた場合、デフォルトでは新しいスタック マスターの MAC アドレスがただちに新しいスタック MAC ルータ アドレスになります。ただし、スタック MAC アドレスを変更する前の時間遅延を可能にする、固定 MAC アドレス機能をイネーブルにできます。この期間、前のスタック マスターがスタックに復帰すると、スイッチがスタック メンバであってスタック マスターではない場合でも、スタックはその MAC アドレスをスタック MAC アドレスとして使用し続けます。前のスタック マスターがこの期間にスタックに復帰しない場合、スイッチ スタックは新しいスタック マスターの MAC アドレスをスタック MAC アドレスとして取得します。また、スタックが新しいスタック マスターの MAC アドレスに切り替わらないように、スタック MAC の持続性を設定することもできます。



(注) この機能を設定するためにコマンドを入力すると、設定の結果を記述した警告メッセージが表示されません。この機能は慎重に使用してください。前のスタック マスター MAC アドレスを同じドメインで使用すると、トラフィックが失われることがあります。

時間は 0 ～ 60 分の範囲で指定できます。

- このコマンドに値を入力しない場合、デフォルトの遅延は 4 分です。必ず値を入力することを推奨します。値を指定しないでコマンドを入力すると、実行コンフィギュレーション ファイルには、遅延時間は明示タイマー値 4 分として書き込まれます。

- **0** を入力すると、スタック MAC アドレスを現在のスタック マスターの MAC アドレスにただちに変更するための **no stack-mac persistent timer** コマンドを入力するまで、前のスタック マスターのスタック MAC アドレスが使用されます。**no stack-mac persistent timer** コマンドを入力しない場合は、スタック MAC アドレスは変更されません。
- 遅延時間に 1 ~ 60 分を入力すると、設定された時間が過ぎるまで、または **no stack-mac persistent timer** コマンドを入力するまで、以前のスタック マスターのスタック MAC アドレスが使用されます。



(注) スタック全体をリロードする場合、スタック マスターの MAC アドレスがスタック MAC アドレスとして使用されます。

永続的 MAC アドレスをイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

	コマンド	目的
ステップ1	<b>configure terminal</b>	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	<b>stack-mac persistent timer [0   time-value]</b>	<p>スタック マスターが変更された後、スタック MAC アドレスが新しいスタック マスターの MAC アドレスに変更されるまでの遅延時間をイネーブルにします。この間に以前のスタック マスターがスタックに再加入した場合、スタックはその MAC アドレスをスタック MAC アドレスとして使用します。</p> <ul style="list-style-type: none"> <li>• 約 4 分というデフォルトの遅延を設定するには、値を指定しないでコマンドを入力します。必ず値を指定することを推奨します。</li> <li>• 現在のスタック マスターの MAC アドレスを無期限に使用し続けるには、<b>0</b> を入力します。</li> <li>• スタック MAC アドレスが新しいスタック マスターの MAC アドレスに変更されるまでの時間を設定するには、<i>time-value</i> に 1 ~ 60 分の範囲内の値を入力します。</li> </ul> <p>(注) このコマンドを入力すると、古いスタック マスターの MAC アドレスがネットワーク ドメイン内にあるとトラフィックが失われる可能性があることを示す警告が表示されます。</p> <p>新しいスタック マスターが引き継いでから有効期間が切れる前に、<b>no stack-mac persistent timer</b> コマンドを入力すると、スイッチ スタックは現在のスタック マスター MAC アドレスに移ります。</p>
ステップ3	<b>end</b>	特権 EXEC モードに戻ります。
ステップ4	<b>show running-config</b> または	スタック MAC アドレス タイマーがイネーブルであることを確認します。イネーブルの場合、stack-mac persistent timer と時間が分単位で表示されます。
ステップ5	<b>show switch</b>	イネーブルの場合、次のように表示されます。  Mac persistency wait time、設定済みの分単位の時間、現在のスタック MAC アドレス
ステップ6	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

永続的 MAC アドレス機能をディセーブルにするには、**no stack-mac persistent timer** グローバル コンフィギュレーション コマンドを使用します。

次に、永続的 MAC アドレス機能に 7 分の遅延時間を設定し、設定を確認する例を示します。

```
Switch(config)# stack-mac persistent timer 7
WARNING: The stack continues to use the base MAC of the old Master
WARNING: as the stack MAC after a master switchover until the MAC
WARNING: persistency timer expires. During this time the Network
WARNING: Administrators must make sure that the old stack-mac does
WARNING: not appear elsewhere in this network domain. If it does,
WARNING: user traffic may be blackholed.
Switch(config)# end
Switch# show switch
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins

                H/W   Current
Switch#  Role   Mac Address      Priority Version  State
-----
*1      Master 0016.4727.a900    1         0         Ready
```

## スタック メンバ情報の割り当て

ここでは、スタック メンバ情報を割り当てる方法について説明します。

- 「スタック メンバ番号の割り当て」(P.5-27) (任意)
- 「スタック メンバプライオリティ値の設定」(P.5-27) (任意)
- 「スイッチ スタックへの新しいメンバの割り当て」(P.5-28) (任意)

## スタック メンバ番号の割り当て



(注) この作業は、スタック マスターからだけ実行できます。

メンバ番号をスタック メンバに割り当てるには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

	コマンド	目的
ステップ1	<b>configure terminal</b>	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	<b>switch current-stack-member-number renumber new-stack-member-number</b>	スタック メンバの現在のスタック メンバ番号と新たなスタック メンバ番号を指定します。指定できる範囲は 1 ~ 9 です。 <b>show switch</b> ユーザ EXEC コマンドを使用すると、現在のスタック メンバ番号を表示できます。
ステップ3	<b>end</b>	特権 EXEC モードに戻ります。
ステップ4	<b>reload slot stack-member-number</b>	スタック メンバをリセットします。
ステップ5	<b>show switch</b>	スタック メンバ番号を確認します。
ステップ6	<b>copy running-config startup-config</b>	コンフィギュレーション ファイルに設定を保存します。

## スタック メンバ プライオリティ値の設定



(注) この作業は、スタック マスターからだけ実行できます。

## ■ スイッチ スタックの設定

プライオリティ値をスタック メンバに割り当てるには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

	コマンド	目的
ステップ 1	<b>configure terminal</b>	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	<b>switch stack-member-number priority new-priority-number</b>	スタック メンバのスタック メンバ番号と、新しいプライオリティを指定します。スタック メンバ番号の有効範囲は 1 ~ 9 です。プライオリティ値の範囲は 1 ~ 15 です。  <b>show switch</b> ユーザ EXEC コマンドを使用すると、現在のプライオリティ値を表示できます。  新しいプライオリティ値はすぐに有効となりますが、現在のスタック マスターには影響しません。新たなプライオリティ値は、現在のスタック マスターまたはスイッチ スタックのリセット時に、どのスタック メンバが新たなスタック マスターとして選択されるかを決定する場合に影響を及ぼします。
ステップ 3	<b>end</b>	特権 EXEC モードに戻ります。
ステップ 4	<b>reload slot stack-member-number</b>	スタック メンバをリセットし、このコンフィギュレーションに変更を適用します。
ステップ 5	<b>show switch stack-member-number</b>	スタック メンバプライオリティ値を確認します。
ステップ 6	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

## スイッチ スタックへの新しいメンバの割り当て



(注) この作業は、スタック マスターからだけ実行できます。

新しいメンバをスイッチ スタックに割り当てるには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

	コマンド	目的
ステップ 1	<b>show switch</b>	スイッチ スタックのサマリー情報を表示します。
ステップ 2	<b>configure terminal</b>	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 3	<b>switch stack-member-number provision type</b>	スタック メンバ番号を事前に設定されたスイッチに指定します。デフォルトでは、スイッチはプロビジョニングされません。  <i>stack-member-number</i> の範囲は 1 ~ 9 です。スイッチ スタック内でまだ使用されていないスタック メンバ番号を指定します。ステップ 1 を参照してください。  <i>type</i> には、コマンドライン ヘルプ ストリングに示されたサポート対象のスイッチのモデル番号を入力します。
ステップ 4	<b>end</b>	特権 EXEC モードに戻ります。
ステップ 5	<b>show running-config</b>	実行コンフィギュレーション ファイルでインターフェイスの番号付けが正しいか確認します。

	コマンド	目的
ステップ6	<code>show switch stack-member-number</code>	プロビジョニングされたスイッチのステータスを確認します。 <code>stack-member-number</code> については、ステップ 1 と同じ番号を入力します。
ステップ7	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

プロビジョニングされた情報を削除し、エラー メッセージを受信しないようにするには、このコマンドの **no** 形式を使用する前に、指定されたスイッチをスタックから取り外します。

たとえば、次のように設定されたスタック内の割り当てられたスイッチを削除します。

- スタックは 4 つのメンバを持つ
- スタック メンバ 1 がマスターである
- スタック メンバ 3 が割り当てられたスイッチである

さらに、割り当てられた情報を削除し、エラー メッセージを受信しないようにするには、スタック メンバ 3 の電源を切り、スタック メンバ 3 とそれが接続されているスイッチとの間の **StackWise Plus** ケーブルを抜き、ケーブルを他のメンバ間で再接続して、**no switch stack-member-number provision** グローバル コンフィギュレーション コマンドを入力します。

次に、スタック メンバ番号 2 が設定されたスイッチをスイッチ スタックに割り当てる例を示します。**show running-config** コマンドの出力は、プロビジョニングされたスイッチに関連付けられたインターフェイスを示します。

```
Switch(config)# switch 2 provision switch_PID
Switch(config)# end
Switch# show running-config | include switch 2
!
interface GigabitEthernet2/0/1
!
interface GigabitEthernet2/0/2
!
interface GigabitEthernet2/0/3
<output truncated>
```

## ローリング スタック アップデートの実行

特権 EXEC モードで次の手順を実行します。

	コマンド	目的
ステップ1	<code>configure terminal</code>	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	<code>boot time minutes</code>	(任意) スタック 起動時間を分単位で設定します。 指定できる範囲は 7 ~ 30 です。 デフォルトは 7 です。
ステップ3	<code>interface interface-id</code>	ポートを指定し、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。



## ■ スイッチスタックの設定

	コマンド	目的
ステップ4	<code>rsu {active   standby}</code>	1つのメンバ上で、ネットワークへの冗長アップリンクの半分を設定し、次のロールのいずれかをメンバインターフェイスに割り当てます。 <ul style="list-style-type: none"> <li>• <b>active</b> : インターフェイスをアクティブに設定します。</li> <li>• <b>standby</b> : インターフェイスをスタンバイに設定します。</li> </ul> <b>(注)</b> Spanning Tree Protocol (STP; スパニング ツリー プロトコル) がイネーブルの場合、ブロック インターフェイスにスタンバイ ロールを設定します。  デフォルトでは、ロールは設定されません。
ステップ5	<code>interface interface-id</code>	別のメンバ上のポートを指定し、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。
ステップ6	<code>rsu {active   standby}</code>	別のメンバ上で、冗長アップリンクの残り半分を設定し、残りのロールをメンバ インターフェイスに割り当てます。 <ul style="list-style-type: none"> <li>• <b>active</b> : インターフェイスをアクティブに設定します。</li> <li>• <b>standby</b> : インターフェイスをスタンバイに設定します。</li> </ul> <b>(注)</b> Spanning Tree Protocol (STP; スパニング ツリー プロトコル) がイネーブルの場合、ブロック インターフェイスにスタンバイ ロールを設定します。  デフォルトでは、ロールは設定されません。  追加のペアを設定するには、ステップ 3 ~ 6 を繰り返します。
ステップ7	<code>end</code>	特権 EXEC モードに戻ります。
ステップ8	<code>archive download-sw /rolling-stack-upgrade</code>	ローリング ステート アップグレード プロセスを開始して、メンバを1つずつアップグレードします。
ステップ9	<code>show switch stack-upgrade sequence</code>	(任意) アップグレード プロセス中に、メンバアップグレード シーケンスを表示します。
ステップ10	<code>show switch stack-upgrade status</code>	(任意) アップグレード プロセス中に、ローリング スタック アップグレード ステータスを表示します。
ステップ11	<code>show running-config</code>	(任意) メンバ上で稼動しているソフトウェア イメージを確認します。
ステップ12	<code>copy running-config startup-config</code>	スタック設定をコンフィギュレーション ファイルに保存します。
ステップ13	<code>archive download-sw /force-reload</code>	(任意) ローリング スタック アップグレード後に、システムを強制的にリロードします。  スタックで元のマスターを維持し、新規のマスターを選択しないようにする場合に限り、このコマンドを入力します。

次に、ローリング スタック アップグレードを実行する例を示します。

```
Switch# configure terminal
Switch(config)# boot time 15
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# rsu active
Switch(config-if)# exit
Switch(config)# interface gigabitethernet 4/0/1
Switch(config-if)# rsu standby
Switch(config-if)# end
Switch# archive download-sw /rolling-stack-upgrade
Switch# show switch stack-upgrade status
Upgrade Time Remaining: 21 minutes
```

```

Unupgraded Stack:
Switch#                Status
  1                    RSU in Progress
  2                    RSU in Progress
  3                    RSU in Progress

Upgraded Stack:
Switch#                Status
Switch#
...
Switch# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
.....
Switch# archive download-sw /force-reload
...
Switch# show switch stack-upgrade status
Upgrade Time Remaining: 21 minutes
Unupgraded Stack:
Switch#                Status
  1                    Reload In Progress
  2                    RSU in Progress
  3                    RSU in Progress

Upgraded Stack:
Switch#                Status
Switch#

```

## 特定のスタック メンバへの CLI アクセス



(注)

この作業はデバッグだけを目的とし、実行できるのはスタック マスターからだけです。

**remote command {all | stack-member-number}** 特権 EXEC コマンドを使用して、すべてまたは特定のスタック メンバにアクセスできます。スタック メンバ番号の有効範囲は 1～9 です。

**session stack-member-number** 特権 EXEC コマンドを使用して、特定のスタック メンバにアクセスできます。スタック メンバ番号は、システム プロンプトに追加されます。たとえば、スタック メンバ 2 のプロンプトは Switch-2#、スタック マスターのプロンプトは Switch# です。スタック マスターの CLI セッションに戻るには、**exit** と入力します。特定のスタック メンバ上では、**show** コマンドと **debug** コマンドだけが使用できます。

## スイッチ スタック情報の表示

特定のスタック メンバまたはスタックをリセットした後で保存済みの設定変更を表示するには、次の特権 EXEC コマンドを使用します。

表 5-4 スタック情報を表示するコマンド

コマンド	説明
<b>show platform stack-manager all</b>	スタック プロトコルバージョンなど、すべてのスイッチ スタック情報を表示します。
<b>show platform stack ports {buffer   history}</b>	スタックのポート イベントおよび履歴を表示します。

表 5-4 スタック情報を表示するコマンド (続き)

コマンド	説明
<code>show switch</code>	プロビジョニングされたスイッチおよびバージョン mismatches モードのスイッチのステータスなど、スタックに関するサマリー情報を表示します。
<code>show switch stack-member-number</code>	特定のスタック メンバに関する情報を表示します。
<code>show switch detail</code>	スタック リングに関する詳細情報を表示します。
<code>show switch neighbors</code>	スタックのネイバーを表示します。
<code>show switch stack-ports [summary]</code>	スタックのポート情報を表示します。スタックのケーブル長、スタックのリンク ステータス、およびループバック ステータスを表示するには、 <b>summary</b> キーワードを使用します。
<code>show switch stack-ring activity [detail]</code>	スタック メンバ単位でスタック リングに送信されるフレーム数を表示します。スタック メンバ単位でスタック リング、受信キュー、および ASIC に送信されるフレーム数を表示するには、 <b>detail</b> キーワードを使用します。

## スタックのトラブルシューティング

- 「スタック ポートの手動ディセーブル」 (P.5-32)
- 「別のメンバがスタート中のスタック ポートの再イネーブル化」 (P.5-33)
- 「`show switch stack-ports summary` コマンドの出力の概要」 (P.5-33)
- 「ループバックの問題について」 (P.5-34)
- 「切断されたスタック ケーブルの特定」 (P.5-40)
- 「スタック ポート間の不安定な接続の修正」 (P.5-41)

## スタック ポートの手動ディセーブル

スタック ポートがフラッピングしていることが原因で、スタック リングが不安定になるためにポートをディセーブルにするには、`switch stack-member-number stack port port-number disable` 特権 EXEC コマンドを入力します。ポートを再びイネーブルにするには、`switch stack-member-number stack port port-number enable` コマンドを入力します。



(注)

`switch stack-member-number stack port port-number disable` コマンドの使用には注意が必要です。スタック ポートをディセーブルにすると、スタックは半分の帯域幅で稼働します。

- スタック ポートを通じてすべてのメンバが接続されており、準備完了状態であれば、スタックはフルリング状態です。
- スタックが *partial-ring* ステートになるのは次のような場合です。
  - すべてのメンバがスタック ポートを通じて接続されているが、一部またはすべてのメンバが準備完了状態ではない。
  - スタック ポートを通じて接続されていないメンバがある。

**switch stack-member-number stack port port-number disable** 特権 EXEC コマンドを入力した場合、次のようになります。

- スタックがフルリング状態であれば、ディセーブルできるスタック ポートは 1 つだけです。次のメッセージが表示されます。

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

- スタックが **partial-ring** ステートのときは、ポートをディセーブルにできません。次のメッセージが表示されます。

```
Disabling stack port not allowed with current stack configuration.
```

## 別のメンバがスタート中のスタック ポートの再イネーブル化

スイッチ 1 のポート 1 がスイッチ 4 のポート 2 に接続されています。ポート 1 でフラッピングが発生した場合、**switch 1 stack port 1 disable** 特権 EXEC コマンドを使用してポート 1 をディセーブルにします。

スイッチ 1 のポート 1 がディセーブルで、スイッチ 1 の電源が入ったままのときに、次の手順を実行します。

- スイッチ 1 のポート 1 とスイッチ 4 のポート 2 の間のスタック ケーブルを取り外します。
- スタックからスイッチ 4 を取り外します。
- スイッチを追加してスイッチ 4 を交換し、スイッチ番号 4 を割り当てます。
- スイッチ 1 のポート 1 とスイッチ 4 (交換後のスイッチ) のポート 2 の間のケーブルを再接続します。
- スイッチ間のリンクを再びイネーブルにします。**switch 1 stack port 1 enable** 特権 EXEC コマンドを入力して、スイッチ 1 のポート 1 をイネーブルにします。
- スイッチ 4 の電源を入れます。



### 注意

スイッチ 1 のポート 1 をイネーブルにする前にスイッチ 4 の電源を入れると、スイッチのいずれかがリロードされる場合があります。

最初にスイッチ 4 の電源を入れると、リンクを起動するために **switch 1 stack port 1 enable** および **switch 4 stack port 2 enable** 特権 EXEC コマンドを入力する必要がある場合があります。

## show switch stack-ports summary コマンドの出力の概要

スタック メンバ 2 のポート 1 だけがディセーブルです。

```
Switch# show switch stack-ports summary
Switch#/  Stack  Neighbor  Cable  Link  Link  Sync  #  In
Port#    Port   Status   Length OK   Active OK  Changes  Loopback
                                     To LinkOK
-----
1/1      OK     3         50 cm  Yes  Yes  Yes  1      No
1/2      Down  None     3 m    Yes  No   Yes  1      No
2/1      Down  None     3 m    Yes  No   Yes  1      No
2/2      OK     3         50 cm  Yes  Yes  Yes  1      No
3/1      OK     2         50 cm  Yes  Yes  Yes  1      No
3/2      OK     1         50 cm  Yes  Yes  Yes  1      No
```

表 5-5 show switch stack-ports summary コマンドの出力

フィールド	説明
Switch#/Port#	メンバ番号と、そのスタック ポート番号
Stack Port Status	<ul style="list-style-type: none"> <li>• <b>Absent</b> : スタック ポートにケーブルが検出されません。</li> <li>• <b>Down</b> : ケーブルは検出されましたが、接続されたネイバーがアップになっていないか、スタック ポートがディセーブルになっています。</li> <li>• <b>OK</b> : ケーブルが検出され、接続済みのネイバーが起動しています。</li> </ul>
Neighbor	スタック ケーブルの接続先の、アクティブなメンバのスイッチの数。
Cable Length	有効な長さは 50 cm、1 m、または 3 m です。 スイッチがケーブルの長さを検出できない場合は、値は <i>no cable</i> になります。ケーブルが接続されていないか、リンクが信頼できない可能性があります。
Link OK	これは、リンクが安定しているかどうかを示します。 リンク パートナーは、ネイバー スイッチ上のスタック ポートのことです。 <ul style="list-style-type: none"> <li>• <b>No</b> : リンクの相手側は、ポートから無効なプロトコル メッセージを受信します。</li> <li>• <b>Yes</b> : リンクの相手側は、ポートから有効なプロトコル メッセージを受信します。</li> </ul>
Link Active	スタック ポートが、リンク パートナーと同じ状態であることを意味します。 <ul style="list-style-type: none"> <li>• <b>No</b> : ポートはリンクの相手側にトラフィックを送信できません。</li> <li>• <b>Yes</b> : ポートはリンクの相手側にトラフィックを送信できます。</li> </ul>
Sync OK	<ul style="list-style-type: none"> <li>• <b>No</b> : リンク パートナーからスタック ポートに有効なプロトコル メッセージが送信されません。</li> <li>• <b>Yes</b> : リンクの相手側は、ポートに有効なプロトコル メッセージを送信します。</li> </ul>
# Changes to LinkOK	これは、リンクの相対的安定性を示します。 短時間で多数の変更が行われた場合は、リンクのフラップが発生することがあります。
In Loopback	<ul style="list-style-type: none"> <li>• <b>No</b> : メンバの 1 つ以上のスタック ポートに、スタック ケーブルが接続されています。</li> <li>• <b>Yes</b> : メンバのどのスタック ポートにも、スタック ケーブルが接続されていません。</li> </ul>

## ループバックの問題について

- 「ソフトウェア ループバック」 (P.5-36)
- 「ソフトウェア ループバックの例：スタック ケーブルが接続されていない」 (P.5-37)
- 「ソフトウェア ループバックの例：スタック ケーブルが接続されている」 (P.5-37)
- 「ハードウェア ループバック」 (P.5-38)

- 「ハードウェア ループバックの例 : LINK OK イベント」 (P.5-38)
- 「ハードウェア ループバックの例 : LINK NOT OK イベント」 (P.5-39)

## ソフトウェア ループバック

メンバが 3 台あるスタックでは、スタック ケーブルですべてのメンバが接続されます。

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 OK 3 50 cm Yes Yes Yes 1 No
1/2 OK 2 3 m Yes Yes Yes 1 No
2/1 OK 1 3 m Yes Yes Yes 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 OK 1 50 cm Yes Yes Yes 1 No
```

スイッチ 1 のポート 1 からスタック ケーブルを切断すると、次のメッセージが表示されます。

```
01:09:55: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 3 has changed to state DOWN
01:09:56: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 1 has changed to state DOWN
```

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 Absent None No cable No No No 1 No
1/2 OK 2 3 m Yes Yes Yes 1 No
2/1 OK 1 3 m Yes Yes Yes 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 Down None 50 cm No No No 1 No
```

スイッチ 1 のポート 2 からスタック ケーブルを切断すると、スタックが分割されます。

スイッチ 2 とスイッチ 3 が、スタック ケーブルで接続された 2 メンバスタックのメンバになります。

```
Switch# show sw stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
2/1 Down None 3 m No No No 1 No
2/2 OK 3 50 cm Yes Yes Yes 1 No
3/1 OK 2 50 cm Yes Yes Yes 1 No
3/2 Down None 50 cm No No No 1 No
```

スイッチ 1 はスタンドアロン スイッチです。

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes Loopback
Status To LinkOK
-----
1/1 Absent None No cable No No No 1 Yes
1/2 Absent None No cable No No No 1 Yes
```

## ソフトウェア ループバックの例：スタック ケーブルが接続されていない

Catalyst 3750 スイッチ ポートのステータス：

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes # In
Status To LinkOK Loopback
-----
1/1 Absent None No cable Yes No Yes 1 Yes
1/2 Absent None No cable Yes No Yes 1 Yes
```

Catalyst 3750-E または 3750-X スイッチ ポートのステータス：

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes # In
Status To LinkOK Loopback
-----
1/1 Absent None No cable No No No 1 Yes
1/2 Absent None No cable No No No 1 Yes
```

## ソフトウェア ループバックの例：スタック ケーブルが接続されている

- スイッチ 1 のポート 1 のポート ステータスが *Down* で、ケーブルが接続されています。スイッチ 1 のポート 2 のポート ステータスが *Absent* で、ケーブルが接続されていません。

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes # In
Status To LinkOK Loopback
-----
1/1 Down None 50 Cm No No No 1 No
1/2 Absent None No cable No No No 1 No
```

- 物理ループバックでは、ケーブルはスタック ポートとスイッチの両方に接続されています。この設定を使用して、次のテストを行えます。
  - 正常に稼動しているスイッチのケーブル
  - 正常なケーブルを使用したスタック ポート

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Length OK Active OK Changes # In
Status To LinkOK Loopback
-----
2/1 OK 2 50 cm Yes Yes Yes 1 No
2/2 OK 2 50 cm Yes Yes Yes 1 No
```

ポート ステータスを見ると、次のことがわかります。

- スイッチ 2 はスタンドアロン スイッチである。
- ポートはトラフィックを送受信できる。



## ハードウェア ループバック

**show platform stack ports buffer** 特権 EXEC コマンドの出力は、ハードウェア ループバックの値を示します。

```
Switch# show platform stack ports buffer
Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====
Event      Stack      Stack PCS Info      Ctrl-Status  Loopback  Cable
Count     Port                                     IOS / HW     length
=====
Event type: LINK OK Stack Port 1
0000000011  1  FF08FF00 860302A5 AA55FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
0000000011  2  FF08FF00 86031805 55AAFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
Event type: LINK OK Stack Port 2
0000000012  1  FF08FF00 860302A5 AA55FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
0000000012  2  FF08FF00 86031805 55AAFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
Event type: RAC
0000000013  1  FF08FF00 860302A5 AA55FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
0000000013  2  FF08FF00 86031805 55AAFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
```

Catalyst 3750 メンバの場合：

- 少なくとも1つのポートにスタック ケーブルが接続されている場合は、両方のスタック ポートの *Loopback HW* 値は *No* になります。
- どちらのスタック ポートにもスタック ケーブルが接続されていない場合は、両方のスタック ポートの *Loopback HW* 値は *Yes* になります。

Catalyst 3750-E または Catalyst 3750-X メンバの場合：

- スタック ポートにスタック ケーブルが接続されている場合は、スタック ポートの *Loopback HW* 値は *No* になります。
- スタック ポートにスタック ケーブルが接続されていない場合は、スタック ポートの *Loopback HW* 値は *Yes* になります。

## ハードウェア ループバックの例：LINK OK イベント

Catalyst 3750 スイッチの場合：

```
Switch# show platform stack ports buffer
Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====
Event      Stack      Stack PCS Info      Ctrl-Status  Loopback  Cable
Count     Port                                     IOS / HW     length
=====
Event type: LINK OK Stack Port 1
0000000008  1  FF08FF00 8603F083 55AAFFFF FFFFFFFF  0CE60C10  No /No   50 cm
0000000008  2  FF08FF00 0001DBDF 01000B00 FFFFFFFF  0CE60C10  No /No   No cable
Event type: RAC
0000000009  1  FF08FF00 8603F083 55AAFFFF FFFFFFFF  0CE60C10  No /No   50 cm
0000000009  2  FF08FF00 0001DC1F 02000100 FFFFFFFF  0CE60C10  No /No   No cable
```

Catalyst 3750-E または 3750-X スイッチの場合 :

Switch# **show platform stack ports buffer**

```

Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====
Event   Stack      Stack PCS Info          Ctrl-Status  Loopback  Cable
Count  Port                                     IOS / HW     length
=====  =====
Event type: LINK OK Stack Port 1
0000000153  1  FF01FF00 860351A5 55A5FFFF FFFFFFFF  0CE60C10  No /No  50 cm
0000000153  2  FF01FF00 00017C07 00000000 0000FFFF  0CE60C10  No /No  3 m
Event type: RAC
0000000154  1  FF01FF00 860351A5 55A5FFFF FFFFFFFF  0CE60C10  No /No  50 cm
0000000154  2  FF01FF00 00017C85 00000000 0000FFFF  0CE60C10  No /No  3 m

```

## ハードウェア ループバックの例 : LINK NOT OK イベント

Catalyst 3750 スイッチの場合 :

Switch# **show platform stack ports buffer**

```

Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====
Event   Stack      Stack PCS Info          Ctrl-Status  Loopback  Cable
Count  Port                                     IOS / HW     length
=====  =====
Event type: LINK OK Stack Port 2
0000000005  1  FF08FF00 0001FBD3 0801080B EFFFFFFF  0C100CE6  No /No  No cable
0000000005  2  FF08FF00 8603E4A9 5555FFFF FFFFFFFF  0C100CE6  No /No  50 cm
Event type: RAC
0000000006  1  FF08FF00 0001FC14 08050204 EFFFFFFF  0C100CE6  No /No  No cable
0000000006  2  FF08FF00 8603E4A9 5555FFFF FFFFFFFF  0C100CE6  No /No  50 cm
Event type: LINK NOT OK Stack Port 2
00000000939  1  FF08FF00 00016879 00010000 EFFFFFFF  0C100C14  No /No  No cable
00000000939  2  FF08FF00 0001901F 00000000 FFFFFFFF  0C100C14  No /No  No cable
Event type: RAC
00000000940  1  FF08FF00 000168BA 00010001 EFFFFFFF  0C100C14  No /No  No cable
00000000940  2  FF08FF00 0001905F 00000000 FFFFFFFF  0C100C14  No /No  No cable
Event type: LINK OK Stack Port 1
00000000956  1  FF08FF00 86034DAC 5555FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
00000000956  2  FF08FF00 86033431 55AAFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
Event type: LINK OK Stack Port 2
00000000957  1  FF08FF00 86034DAC 5555FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
00000000957  2  FF08FF00 86033431 55AAFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
Event type: RAC
00000000958  1  FF08FF00 86034DAC 5555FFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable
00000000958  2  FF08FF00 86033431 55AAFFFF FFFFFFFF  1CE61CE6  Yes/Yes  No cable

```

Catalyst 3750-E または 3750-X スイッチの場合 :

```
Switch# show platform stack ports buffer
Stack Debug Event Data Trace
=====
Event type LINK: Link status change
Event type RAC: RAC changes to Not OK
Event type SYNC: Sync changes to Not OK
=====
Event      Stack      Stack PCS Info      Ctrl-Status  Loopback  Cable
Count      Port                                     IOS / HW      length
=====
Event type: LINK OK Stack Port 1
0000000014  1  FF01FF00 860204A7 5555FFFF 00000000  0CE60CA6  No /No  50 cm
0000000014  2  FF01FF00 85020823 AAAAFFFF 00000000  0CE60CA6  No /No   3 m
Event type: RAC
0000000015  1  FF01FF00 860204A7 5555FFFF 00000000  0CE60CA6  No /No  50 cm
0000000015  2  FF01FF00 85020823 AAAAFFFF 00000000  0CE60CA6  No /No   3 m
Event type: LINK OK Stack Port 2
0000000029  1  FF01FF00 860204A7 5555FFFF 00000000  1CE61CE6  No /No  50 cm
0000000029  2  FF01FF00 86020823 AAAAFFFF 00000000  1CE61CE6  No /No   3 m
Event type: RAC
0000000030  1  FF01FF00 860204A7 5555FFFF 00000000  1CE61CE6  No /No  50 cm
0000000030  2  FF01FF00 86020823 AAAAFFFF 00000000  1CE61CE6  No /No   3 m
Event type: LINK NOT OK Stack Port 1
0000009732  1  FF01FF00 00015B12 5555FFFF A49CFFFF  0C140CE4  No /No  50 cm
0000009732  2  FF01FF00 86020823 AAAAFFFF 00000000  0C140CE4  No /No   3 m
Event type: RAC
0000009733  1  FF01FF00 00015B4A 5555FFFF A49CFFFF  0C140CE4  No /No  50 cm
0000009733  2  FF01FF00 86020823 AAAAFFFF 00000000  0C140CE4  No /No   3 m
Event type: LINK NOT OK Stack Port 2
0000010119  1  FF01FF00 00010E69 25953FFF FFFFFFFF  0C140C14  No /Yes  No cable
0000010119  2  FF01FF00 0001D98C 81AAC7FF 0300FFFF  0C140C14  No /No   3 m
Event type: RAC
0000010120  1  FF01FF00 00010EEA 25953FFF FFFFFFFF  0C140C14  No /Yes  No cable
0000010120  2  FF01FF00 0001DA0C 81AAC7FF 0300FFFF  0C140C14  No /No   3 m
```

## 切断されたスタック ケーブルの特定

すべてのスタック メンバは、スタック ケーブルで接続されます。スイッチ 1 のポート 2 と、スイッチ 2 のポート 1 が接続されます。

次に、メンバのポート ステータスを示します。

```
Switch# show switch stack-ports summary
Switch#/  Stack  Neighbor  Cable  Link  Link  Sync  #  In
Port#    Port  Status    Length OK   Active OK  Changes  Loopback
-----  -
1/1      OK    2         50 cm  Yes  Yes  Yes  0      No
1/2      OK    2         50 cm  Yes  Yes  Yes  0      No
2/1      OK    1         50 cm  Yes  Yes  Yes  0      No
2/2      OK    1         50 cm  Yes  Yes  Yes  0      No
```

スイッチ 1 のポート 2 からケーブルを切断すると、次のメッセージが表示されます。

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 2 has changed to state DOWN
```

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 1 has changed to state DOWN
```

ポート ステータスは次のようになります。

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Status Length OK Active OK Changes Loopback
To LinkOK
-----
1/1 OK 2 50 cm Yes Yes Yes 1 No
1/2 Absent None No cable No No No 2 No
2/1 Down None 50 cm No No No 2 No
2/2 OK 1 50 cm Yes Yes Yes 1 No
```

ケーブルの片方だけが、スタック ポート（スイッチ 2 のポート 1）に接続されます。

- スイッチ 1 のポート 2 の *Stack Port Status* 値は *Absent* で、スイッチ 2 のポート 1 の値は *Down* です。
- *Cable Length* 値は *No cable* です。

問題の診断

- スイッチ 1 のポート 2 のケーブル接続を確認します。
- スイッチ 1 のポート 2 が次の状態であれば、ポートまたはケーブルに問題があります。
  - *In Loopback* 値が *Yes* である。  
または
  - *Link OK*、*Link Active*、または *Sync OK* 値が *No* である。

## スタック ポート間の不安定な接続の修正

すべてのメンバは、スタック ケーブルで接続されます。スイッチ 1 のポート 2 と、スイッチ 2 のポート 1 が接続されます。

ポート ステータスは次のとおりです。

```
Switch# show switch stack-ports summary
Switch#/ Stack Neighbor Cable Link Link Sync # In
Port# Port Status Length OK Active OK Changes Loopback
To LinkOK
-----
1/1 OK 2 50 cm Yes Yes Yes 1 No
1/2 Down None 50 cm No No No 2 No
2/1 Down None 50 cm No No No 2 No
2/2 OK 1 50 cm Yes Yes Yes 1 No
```

問題の診断

- *Stack Port Status* 値が *Down* になっています。
- *Link OK*、*Link Active*、および *Sync OK* 値が *No* です。
- *Cable Length* 値が *50 cm* です。スイッチがケーブルを検出し、正しく識別しています。

スイッチ 1 のポート 2 と、スイッチ 2 のポート 1 との接続は、少なくとも 1 つのコネクタ ピンで不安定になっています。

