

マルチシャーシ マルチリンクPPP (MMP)

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[関連用語](#)

[表記法](#)

[問題の定義](#)

[機能概要](#)

[SGBP](#)

[仮想アクセスインターフェイス](#)

[L2F](#)

[エンドユーザインターフェイス](#)

[SGBP](#)

[MP](#)

[例](#)

[関連情報](#)

概要

このドキュメントでは、Cisco Systems のアクセスサーバプラットフォームにおける、スタックまたはマルチシャーシ環境 (マルチシャーシ マルチリンク PPP (MMP) と呼ばれる場合もある) でのマルチリンク PPP (MP) のサポートについて説明します。

前提条件

要件

このドキュメントに関しては個別の前提条件はありません。

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

このマニュアルの情報は、特定のラボ環境に置かれたデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期 (デフォルト) 設定の状態から起動しています。実稼動中のネットワークで作業をしている場合、実際にコマンドを使用する前に、その潜在的な影響について理解しておく必要があります。

関連用語

このドキュメントで使用する用語は次のとおりです。

- アクセスサーバ - リモート アクセスを提供する ISDN および非同期インターフェイスを含むシスコのアクセスサーバプラットフォーム。
- L2F : レイヤ2(L2)フォワーディングプロトコル (実験的なドラフトRFC)。これはマルチシャシー MP および VPN の両方にとって基礎となるリンクレベルのテクノロジーです。
- リンク - システムが提供する接続ポイント。リンクは専用ハードウェア インターフェイス (非同期インターフェイスなど) またはマルチチャンネル ハードウェア インターフェイス (PRI または BRI) のチャンネルである場合があります。
- MP : マルチリンクPPPプロトコル([RFC 1717](#)を参照してください)。
- マルチシャシーMP:MP + SGBP + L2F + Vtemplate。
- PPP:Point-to-Point Protocol (PPP ; ポイントツーポイントプロトコル) ([RFC 1331](#)を参照)。
- ロータリーグループ - ダイヤルアウトまたはコールの受信用に割り当てられた物理インターフェイスのグループ。このグループは、ダイヤルアウトやコールの受信のために使用できるリンクのプールのような役割を果たします。
- SGBP:Stack Group Bidding Protocol。
- スタックグループ : グループとして動作し、異なるシステム上のリンクを持つMPバンドルをサポートするように設定された2つ以上のシステムの集合。
- VPDN:Virtual Private Dialup Network。Internet Service Provider (ISP; インターネット サービスプロバイダー) から Cisco Home Gateway へ PPP リンクをフォワーディングするもの。
- Vtemplate - 仮想テンプレート インターフェイス。

注 : このドキュメントで参照されているRFCの詳細については、製品速報『[Cisco IOSリリース 11.3-No. 523](#)でサポートされるRFCとその他のSTD』を参照してください。[RFC および標準ドキュメントの入手](#)を、またはInterNICへの直接リンクの[RFCインデックス](#)。

表記法

ドキュメント表記の詳細は、『[シスコテクニカルティップスの表記法](#)』を参照してください。

問題の定義

MP では、要求に応じて新たな帯域幅をユーザに提供します。マルチリンクを形成する論理パイプ (バンドル) のパケットを分割および再結合する機能も備えています。

この機能によって、速度の遅い WAN リンクの伝送の遅延を低減するとともに、受信ユニットでの最大受信量を向上させることができます。

送信側では、MP によって1つのパケットを複数のパケットにフラグメント化し、複数の PPP リンクに送信できるようになります。受信側では、MP によって、複数の PPP リンクから到達したパケットを元のパケットに再構築します。

シスコでは自律エンドシステムへの MP をサポートしています。つまり、同じクライアントからの複数の MP リンクをアクセスサーバで終端できます。ただし、たとえば ISP は、1つのロータリー番号を複数のサーバにまたがる複数の PRI に割り当て、ビジネス ニーズに合わせられるようにサーバ構造をスケーラブルにし、柔軟性を持たせられれば便利だと考えています。

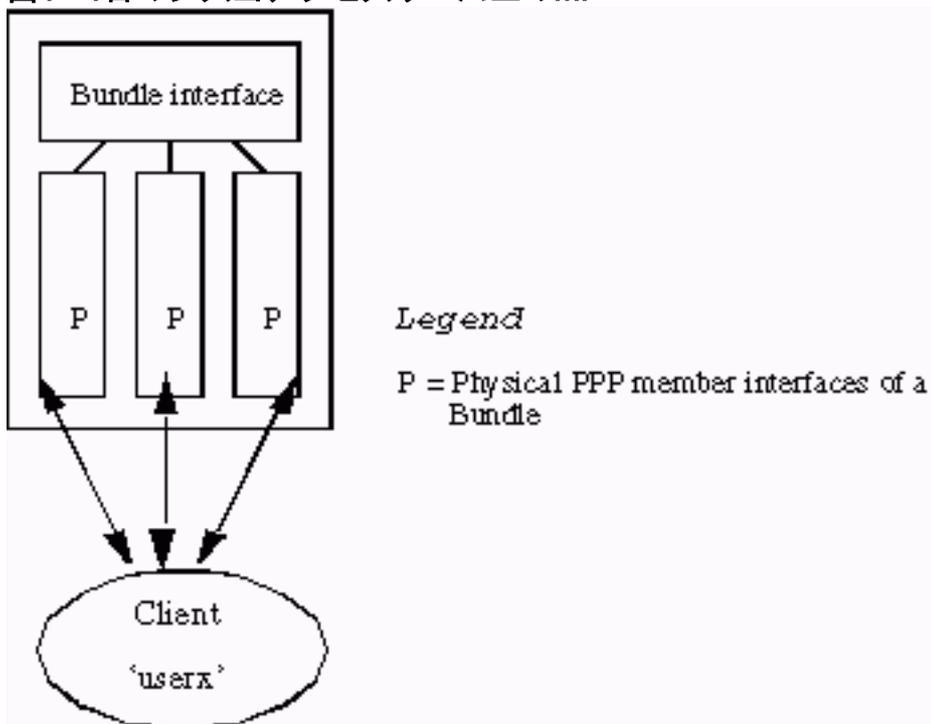
Cisco IOS® ソフトウェア リリース 11.2 では、このような機能を搭載しているため、同じクライアントからの複数の MP リンクを異なるアクセス サーバで終端できるようにしています。同じバンドル内の個々の MP リンクは実際には異なるアクセス サーバで終端しますが、MP クライアントが接続されている限り、これは 1 つのアクセス サーバで終端するのと同様です。

これを実現するために、MP ではマルチシャーシ MP を使用しています。

機能概要

[図 1 は、1 台のシスコアクセス サーバで MP を使用して、この機能をサポートしている仕組みを表しています。](#)

図1 - 1台のシスコアクセスサーバ上のMP



[図 1 では、MP のメンバーであるインターフェイスがバンドル インターフェイスへ接続されている仕組みを示しています。](#)マルチシャーシ MP がイネーブルにされていないスタンドアロン システムでは、メンバー インターフェイスは常に物理インターフェイスになります。

スタック環境をサポートするには、MP の他に、次の 3 つのサブコンポーネントが必要です。

- SGBP
- Vtemplate
- L2F

以降のセクションで、これらのコンポーネントについて詳しく説明します。

SGBP

マルチ アクセス サーバ環境では、ネットワーク管理者がアクセス サーバをスタック グループに属するように指定できます。

スタックグループがシステムA(System A)とシステムB(System B)で構成されているとします。userxというリモートMPクライアントには、システムA(systema)で終端する最初のMPリンクがあ

ります。バンドル *userx* は、*systema* で形成されています。*userx* からの次の MP リンクは、システム B (*systemb*) で終端しています。SGBP は、この *userx* があるバンドルを *systema* から探します。この時点で、別のコンポーネント(L2F)が2番目のMPリンクを *systemb* から *systema* に投影します。投影された MP リンクは、その後 *systema* でバンドルを結合します。

このようにして、SGBP はスタック メンバーのバンドル位置を、定義したスタック グループ内で探します。また、SGBP では、指定したスタック メンバーを調整して、バンドルを作成します。この例では、*systema* で最初の MP リンクが受信されたとき、実際には *systema systemb systema SGBP systema*

SGBP の受信権要求プロセスに関するこの説明は多少単純化したものになっています。実際には、スタック メンバーからの SGBP 送信権要求はローカルの機能であり、ユーザが設定可能な重み付けされたメトリック、CPU タイプ、MP バンドルの数などです。この送信権要求プロセスを使用すると、指定したシステム上でバンドルを作成できます。アクセス インターフェイスがないシステム上でも可能です。たとえば、スタック環境は、10台のアクセスサーバシステムと2台の4500で構成され、12台のスタックメンバーで構成されます。

注：2つの4500の間など、送信権要求が等しい場合、SGBPはランダムに1つを送信権要求の勝者に指定します。常に他のスタック メンバーに競り勝つように 4500 を設定できます。そのため、4500は、MPパケットのフラグメント化と再構成を専門とするマルチシャーシMPサーバの負荷を軽減します。これは、アクセスサーバに比べて高いCPUパワーを必要とする場合に適したタスクです。

簡潔に言うと、SGBP は、マルチシャーシ MP の配置および調整を行うメカニズムです。

[仮想アクセスインターフェイス](#)

バーチャル アクセス インターフェイスは、バンドル インターフェイス ([図 1 および 図 2 を参照](#)) と、[投影された PPP リンク \(図 2 を参照\)](#) の両方として機能します。これらのインターフェイスは要求に応じて動的に作成され、システムに戻されます。

バーチャル テンプレート インターフェイスは、バーチャル アクセス インターフェイスがクローン化されるときの設定情報のレポジトリとなります。ダイヤラ インターフェイスの設定は、設定情報の別のソースとなります。仮想アクセス インターフェイスのクローンを作成する設定のソースを選択する方法は「[マルチシャーシ マルチリンク PPP \(MMP\) \(パート2\)](#)」で説明します。

[L2F](#)

L2F は、実際の PPP リンクを指定されたエンド システムに投影します。

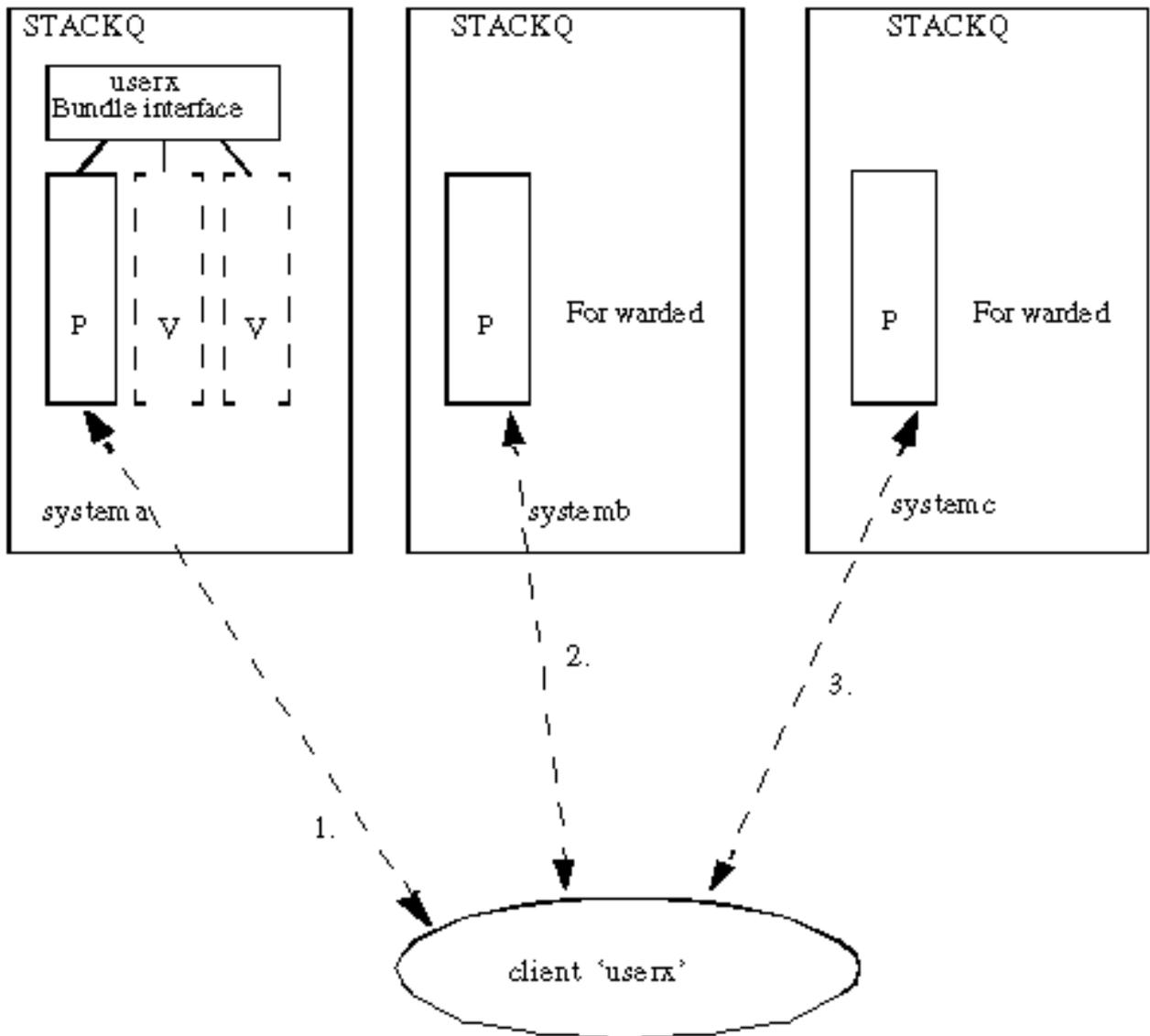
L2F では、標準の PPP の操作を、リモート クライアントを認識する認証段階まで実行します。この認証段階は、ローカルでは完了しません。L2F には SGBP から対象とするスタック メンバーが通知され、PPP リンクをその対象とするスタック メンバーに投影します。ここで認証段階が再開され、投影された PPP リンクで完了されます。したがって、最終的な認証の成功または失敗は、対象とされるスタック メンバー上で決まります。

着信コールを受信した元の物理インターフェイスは、*L2F forwarded* であると考えられます。PPP の認証が成功した場合に L2F が動的に作成した、これに対応するインターフェイスは、投影されたバーチャル アクセス インターフェイスです。

注：投影された仮想アクセスインターフェイスも、仮想テンプレートインターフェイス (定義されている場合) から複製されます。

図2では、systema、systemb、および systemc で構成されているスタックグループ stackq について説明しています。

図2：スタックにコール中のクライアント



Legend

- Client PPP MP links across stack members STACKQ
- L2F projected links to the stack member containing bundle interface 'userx'
- Bundle Interface Bundle Interface for client 'userx' (Virtual Access interface)
- P Physical interface
- V Projected PPP link (Virtual Access Interface)

1. クライアント userx がコールします。systema SGBP が userx についてバンドルを既存のスタックグループメンバーの中から探そうとします。見つからない場合は、MP が PPP でネゴシエートされているため、バンドルインターフェイスが systema 上に作成されます。
2. systemb userx 2 SGBP は、systema L2F は、このリンクを systemb systema に転送するの

に役立ちます。投影された PPP リンクが `systema` に作成されます。その後、投影された MP リンクは、バンドル インターフェイスと一緒にになります。

3. `systemc userx 3` 再度、SGBP は、`systema L2F` は、このリンクを `systemc systema` へ転送するために使用されます。投影された PPP リンクが `systema` に作成されます。その後、投影された MP リンクは、バンドル インターフェイスと一緒にになります。

注：バンドルインターフェイスは、`systema` のバンドルを表します。個々の発信者ごとに、同じ発信者からの MP メンバー インターフェイスは、あるバンドル インターフェイスで終端するか、そこから発信されます。

エンド ユーザ インターフェイス

Vtemplate ユーザ インターフェイスは、通常はここで指定されます。詳細は、『[仮想テンプレートの機能仕様](#)』を参照してください。

SGBP

1. `sgbp group <name>` このグローバル コマンドでは、スタック グループを定義して、そのグループに名前を付け、このシステムをそのスタック グループのメンバーにします。注：グローバルに定義できるスタックグループは1つだけです。 `stackq` というスタック グループを定義します。

```
systema(config)#sgbp group stackq
```

注：PPP CHAP チャレンジまたは `systema` からの PPP PAP 要求には、`stackq` という名前が付けられます。アクセス サーバに対してスタック グループ名を定義すると、通常は同じシステムに定義されたホスト名がこの名前です置き換えられます。

2. `sgbp member <peer-name> <peer-IP-address>` このグローバル コマンドでは、スタック グループ内のピアを指定します。このコマンドでは、`<peer-name>` がホスト名、`<peer-IP-address>` がリモート スタック メンバーの IP アドレスです。したがって、スタック内の自分自身を除くすべてのスタック グループ メンバのエントリを定義する必要があります。ドメイン ネーム サーバ (DNS) はピア名を解決できます。DNS を使用している場合は、IP アドレスを入力する必要はありません。

```
systema(config)#sgbp member systemb 1.1.1.2
```

```
systema(config)#sgbp member systemc 1.1.1.3
```

3. `sgbp seed-bid {default | オフロード | 転送専用 | <0-9999>}` スタック メンバーがバンドルに対して送信権要求に使用する、設定変更が可能な重みです。すべてのスタック メンバーに対して `default userx` 同じユーザから他のスタック メンバーへのこの後のすべてのコールは、このスタック メンバーに投影されます。 `sgbp seed-bid` を定義しない場合は、`default offload` が定義されると、事前に調整されたプラットフォームごとの送信権要求を送信し、バンドル負荷を差し引いた CPU パワーを概算します。 `<0-9999>` を設定すると、送信される送信権要求はバンドルの負荷を差し引いたユーザ設定の値になります。バンドルの負荷は、スタック メンバー内のアクティブなバンドルの数として定義されます。複数の PRI にまたがるロータリー グループ内に、コールを受信するためにスタックされた同一のスタック メンバーがある場合は、`sgbp seed-bid default across all stack members` コマンドを発行します。同一のスタック メンバーの例としては、4 台の AS5200 で構成されたスタック グループが考えられます。ユーザ `userx` に対する最初のコールを受信したスタック メンバーは、常に送信権要求に勝ち、マスター バンドル インターフェイスをホストします。同じユーザから他のスタック メンバーへのこの後のすべてのコールは、このスタック メンバーに投

影されます。複数のコールが同時に複数のスタックメンバー宛てに着信した場合は、SGBPのタイブレーキングメカニズムによって、平衡状態が破られます。他のスタックメンバーよりも高いパワーを持つCPUをスタックメンバーとして利用できる場合は、そのスタックメンバーの比較的高いパワーを残りのメンバーに活用できます(たとえば、他の同様なスタックメンバーよりも高いパワーのCPUをスタックメンバーとして利用できる場合、たとえば、4500 1台とAS5200 4台など)。指定した高パワーのスタックメンバーを **sgbp seed-bid offload** コマンドを使用してオフロードサーバとして設定できます。この場合、オフロードサーバはマスターバンドルをホストします。他のスタックメンバーからのすべてのコールは、このスタックメンバーに投影されます。実際には、1つ以上のオフロードサーバを定義できます。プラットフォームが同じ(同等)の場合は、送信権は等しくなります。SGBPのタイブレーキングメカニズムによって、均衡状態が破られ、いずれかのプラットフォームが勝者として指定されます。注: 2つの異なるプラットフォームをオフロードサーバとして指定した場合は、CPUパワーが高い方が送信権要求を勝ち取ります。自分で分類したプラットフォーム、またはまったく同じプラットフォームを使用しており、1台以上のプラットフォームをオフロードサーバとして指定する場合は、**sgbp seed-bid 9999** コマンドを使用して送信要求値を残りのプラットフォームよりも大幅に高い値に設定します。たとえば、4700が1台(最も高いseed-bidで指定)、4000が2台、7000が1台などです。特定のプラットフォームに関連付けた最初の送信要求値を特定するには、**show sgbp** を使用します。フロントエンドのスタックメンバーが常に1台または複数台のオフロードサーバにオフロードするマルチシャーシ環境では、マルチリンクのバンドルがローカルで形成されているような場合には、このフロントエンドのスタックメンバーが実際にはオフロードできない場合があります。このような状況は、すべてのオフロードサーバがダウンしている場合などに発生します。ネットワーク管理者が着信コールを切断する場合は、切断する代わりに **seed-bid forward-only** コマンドを使用します。

4. **sgbp ppp-forwardsgbp ppp-forward** を定義すると、PPPとMPのコールの両方が、SGBP送信権要求の勝者に投影されます。デフォルトでは、MPコールだけが転送されます。
5. **show sgbp** このコマンドでは、スタックグループメンバーの状態が表示されます。状態は、ACTIVE、CONNECTING、WAITINFO、IDLEのいずれかです。各スタックのグループメンバーがACTIVEになっている状態が最適です。CONNECTINGとWAITINFOは、過渡期にある状態で、ACTIVEに移行する途中にだけ見られる状態です。IDLEは、スタックグループ `systema` `systemd` を検出できないことを示します。たとえば、メンテナンスのために `systemd` それ以外の場合は、このスタックメンバーと `systemd` 間のあるルーティングの問題やその他の問題を確認します。

```
systema#show sgbp
Group Name: stack Ref: 0xC38A529
Seed bid: default, 50, default seed bid setting

Member Name: systemb State: ACTIVE Id: 1
Ref: 0xC14256F
Address: 1.1.1.2

Member Name: systemc State: ACTIVE Id: 2
Ref: 0xA24256D
Address: 1.1.1.3 Tcb: 0x60B34439

Member Name: systemd State: IDLE Id: 3
Ref: 0x0
Address: 1.1.1.4
```

6. **show sgbp queries** 現在シードされている送信権要求の値を表示します。

```
systema# show sgbp queries
Seed bid: default, 50
```

```

systema# debug sgbp queries
%SGBPQ-7-MQ: Bundle: userX State: Query_to_peers OurBid: 050
%SGBPQ-7-PB: 1.1.1.2 State: Open_to_peer Bid: 000 Retry: 0
%SGBPQ-7-PB: 1.1.1.3 State: Open_to_peer Bid: 000 Retry: 0
%SGBPQ-7-PB: 1.1.1.4 State: Open_to_peer Bid: 000 Retry: 0
%SGBPQ-7-MQ: Bundle: userX State: Query_to_peers OurBid: 050
%SGBPQ-7-PB: 1.1.1.2State: Rcvd Bid: 000 Retry: 0
%SGBPQ-7-PB: 1.1.1.3State: Rcvd Bid: 000 Retry: 0
%SGBPQ-7-PB: 1.1.1.4State: Rcvd Bid: 000 Retry: 0
%SGBPQ-7-DONE: Query #9 for bundle userX, count 1, master is local

```

MP

1. **multilink virtual-template <1-9>**これは、MP バンドル インターフェイスが、自身のインターフェイス パラメータのクローンを作成するときに使用するバーチャル テンプレート番号です。MP が仮想テンプレートにどのように関連付けられるかの例を次に示します。仮想テンプレート インターフェイスも、次のように定義する必要があります。

```

systema(config)#multilink virtual-template 1
systema(config)#int virtual-template 1
systema(config-i)#ip unnum e0
systema(config-i)#encap ppp
systema(config-i)#ppp multilink
systema(config-i)#ppp authen chap

```

2. **show ppp multilink**このコマンドでは、MP バンドルのバンドル情報を表示します。

```

systema#show ppp multilink
Bundle userx 2 members, Master link is Virtual-Access4
0 lost fragments, 0 reordered, 0 unassigned, 100/255 load
0 discarded, 0 lost received, sequence 40/66 rcvd/sent
members 2
Serial0:4
systemb:Virtual-Access6 (1.1.1.2)

```

次の例は、スタックグループ `stackq` のスタックグループメンバー `systema` で、バンドル `userx` のバンドルインターフェイスが `Virtual-Access4` として設定されていることを示しています。2つのメンバーインターフェイスがこのバンドルインターフェイスに結合されています。最初のもは、ローカルの PRI チャネルであり、2つ目のものはスタックグループメンバー `systemb` から投影されたインターフェイスです。

例

これらの例については、[「マルチシャーシ マルチリンク PPP \(MMP\) \(パート 2\)」](#) を参照してください。

- [スタック内の AS5200 \(ダイヤラが設定された場合\)](#)
- [オフロードサーバの使用](#)
- [物理インターフェイスを使用するオフロードサーバ](#)
- [非同期、シリアルおよび他の非ダイヤラインターフェイス](#)
- [マルチシャーシからのダイヤルアウト](#)
- [マルチシャーシへのダイヤル](#)

また、次に関するセクションも参照してください。

- [設定および制限](#)
- [トラブルシューティング](#)

[関連情報](#)

- [ダイヤルとアクセス テクノロジーのサポート ページ](#)
- [テクニカル サポートとドキュメント – Cisco Systems](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。