

75xx/76xxルータ：分散型機能の設定と確認

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[分散型機能](#)

[分散型 MLPPP](#)

[分散型 LFI](#)

[dMLP と dLFioLL の違い](#)

[分散型 MLFR](#)

[分散型 DDR](#)

[分散型機能の前提条件および制約事項](#)

[バンドルとリンクの数、およびメモリ要件](#)

[7600 SIP ラインカード上のハードウェアおよびソフトウェアの MLPPP または MLFR](#)

[パケットの受信後の流れ](#)

[Rx データパス](#)

[Tx データパス](#)

[リアセンブル](#)

[分散型機能の設定、確認およびデバッグ](#)

[dMFR の設定と確認](#)

[dMLP/dLFioLL の設定と確認](#)

[dLFioFR と dLFioATM の設定と確認](#)

[dDDR の設定と確認](#)

[dMLP と dDDR のデバッグ](#)

[よく寄せられる質問 \(FAQ\)](#)

[デバッグの機能拡張](#)

[関連情報](#)

概要

このドキュメントは、次のような機能の理解、設定、確認に役立ちます。

- 分散型マルチリンク PPP (dMLP)
- 分散型 Link Fragmentation and Interleaving (LFI)
- 専用回線での分散型 LFI (dLFioLL)
- フレームリレーでの分散型 LFI (dLFioFR)
- ATM での分散型 LFI (dLFioATM)
- 分散型 MLP (dMLP) と dLFioLL の違い

- 分散型マルチリンク フレーム リレー (dMLFR)
- 分散型ダイヤル オンデマンド ルーティング (DDR)

前提条件

要件

このドキュメントの読者は、Cisco 7500/7600/6500 での分散型機能について理解している必要があります。

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- すべての Cisco 7500 および 7600 プラットフォーム注：このドキュメントの情報は、6500 プラットフォームにも適用されます。
- 次の表に列挙されている、関連する Cisco IOS® ソフトウェア リリース：

各ブランチおよびプラットフォームでの分散型機能のサポート

機能	サポートされているポートアダプタ (PA) ¹	7500 プラットフォーム		7600 プラットフォーム	
		主要な Cisco IOS ソフトウェア リリース	Cisco IOS リリース (Interim)	主要な Cisco IOS ソフトウェア リリース	Cisco IOS ソフトウェア リリース (Interim)
dMLP	Chan-PA PA-4T+ PA-8T	12.0T 12.0S 12.1 12.1T 12.2 12.2T 12.3 12.3T 12.2S 12.1E ²	12.0(3)T 以降 12.0(9)S 以降	12.2S X 12.1E ²	—
dLFloLL	Chan-PA PA-4T+ PA-8T	12.2T 12.3 12.3T 12.0S	12.2(8)T 以降 12.0(24) S 以降	12.2S X	12.2(17) SXB 以降
dLFloFR	Chan-PA PA-4T+ PA-8T	12.2T 12.3 12.3T	12.2(4)T 3 以降	12.2S X	12.2(17) SXB 以降
dLFloA	PA-A3 PA-	12.2T	12.2(4)T	12.2S	12.2(17)

TM	A6	12.3 12.3T	3 以降	X	SXB 以降
dMLFR	Chan-PA PA-4T+ PA-8T	12.0S 12.3T	12.0(24) S 以降 12.3(4)T 以降	12.2S X	12.2(17) SXB 以降
dMLP 上の QoS	Chan-PA PA-4T+ PA-8T	12.0S 12.2T 12.3 12.3T	12.0(24) S 以降 12.2(8)T 以降	12.2S X	12.2(17) SXB 以降
dMLP上 のMPLS dLFloLL 上の MPLS	Chan-PA PA-4T+ PA-8T	12.2T 12.3	12.2(15) T11 以降 12.3(5a) 以降	12.2S X	12.2(17) SXB 以降
分散型 DDR	PA-MC-xT1 PA-MC-xE1 PA-MC- xTE1 PA- MCX-xTE1	12.3T	12.3 (7) T 以降	—	—

注：次の情報に注意してください。

1. 次の PA は分散型機能をサポートしています。CT3IPPA-MC-T3PA-MC-2T3+PA-MC-E3PA-MC-2E1PA-MC-2T1PA-MC-4T1PA-MC-8T1PA-MC-8E1PA-MC-8TE1+PA-MC-STM-1
2. Cisco IOS ソフトウェア リリース 12.1E は、7500 および 7600 プラットフォームの両方でこれらの機能をサポートしています。

このドキュメントの情報は、特定のラボ環境にある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。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

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。

分散型機能

このドキュメントでは、次の機能を説明します。

- 分散型 MLP
- 分散型 LFI
- 専用回線での分散型 LFI
- フレーム リレーでの分散型 LFI
- ATM での分散型 LFI
- dMLP と dLFloLL の違い
- 分散型 MLFR
- 分散型ダイヤラ
- 分散型機能をサポートするプラットフォームとライン カード

分散型 MLPPP

分散型マルチリンク PPP (dMLP) 機能により、Cisco 7500 または 7600 シリーズ ルータ上のライン カード (VIP、FlexWAN) の全体またはフラクショナル T1/E1 回線をバンドルに結合して、複数リンクの帯域幅を束ねることができます。これを行うには、分散型 MLP バンドルを使用します。ユーザは、ルータでのバンドルの数とバンドルごとのリンク数を選択できます。これにより、T3 回線を購入しなくても、ネットワーク リンクの帯域幅を単一 T1/E1 回線の帯域幅よりも大きくすることができます。非 dMLP では、すべてのパケットがルート プロセッサ (RP) によってスイッチされます。したがって、この実装は RP のパフォーマンスに影響を与え、MLP を実行している少数の T1/E1 回線のみに対して CPU 使用率が上昇します。dMLP により、データパスはライン カードの CPU およびメモリによって処理され制限されるので、ルータで処理できるバンドルの総数が増加します。dMLP により、DS0 (64 Kbps) 以降のフラクショナル T1/E1 をバンドルすることができます。

分散型 LFI

dLFI 機能は、低速のフレーム リレー仮想回線 (VC) や ATM VC 上、および専用回線上で、音声などのリアルタイムトラフィックや、データなどの非リアルタイムトラフィックのトランスポートをサポートします。リアルタイムトラフィックに極端な遅延は発生しません。

この機能はフレーム リレー、ATM、および専用回線を介したマルチリンク PPP (MLP) を使用して実装されます。この機能は、大きなデータ パケットを連続した小さなフラグメントに分割し、遅延の影響を受けやすいリアルタイム パケットと非リアルタイムのパケットが同じリンクを共有できるようにします。その後、フラグメントはリアルタイム パケットとインターリーブされます。リンクの受信側では、フラグメントが再度組み立てられ、パケットが再構築されます。

通常、dLFI 機能が役立つのは、分散型低遅延キューイングを使用して音声などのリアルタイムトラフィックを送信するものの、帯域幅問題が発生するネットワークです。時間に依存しない大きなデータ パケットの転送によって、このリアルタイムトラフィックに遅延が生じます。これらのネットワークで dLFI 機能を使用すると、サイズの大きなデータ パケットを複数のセグメントに分割できます。これらのデータ パケット セグメントの合間に、リアルタイムトラフィック パケットを送信できます。この場合、リアルタイムトラフィックは、低いプライオリティのデータ パケットがネットワークを通過するまで長時間待機しなくて済みます。データ パケットはリンクの受信側で再度組み立てられるので、データは完全な形で配信されます。

リンクのフラグメント サイズは、`ppp multilink fragment-delay n` コマンドで設定されたマルチリンクバンドルでのフラグメントの遅延に基づいて計算されます。

```
fragment size = bandwidth × fragment-delay / 8
```

このフラグメント サイズが示すのは、IP ペイロードだけです。これにはカプセル化バイト (フラグメント サイズ = 重量 - カプセル化バイト) は含まれません。「weight」および「fragment size」は、RP での `show ppp multilink` コマンドの出力に表示される通りです。fragment delay を設定しない場合、デフォルトのフラグメント サイズは fragment-delay の最大値である 30 として計算されます。

注：異なる帯域幅のリンクでは、選択されたフラグメントサイズは最小帯域幅のリンクに基づいています。

専用回線での分散型 LFI

dLFIoLL 機能は分散型 link fragmentation and interleaving (LFI) 機能を専用回線に拡張します。

分散型 LFI は、マルチリンク グループ インターフェイス上で `ppp multilink interleave` コマンドにより設定されます。768 kbps 以下の帯域幅を持つマルチリンク インターフェイス上で分散型 LFI を使用することをお勧めします。これは、768 kbps を超える帯域幅では、1500 バイトのパケットに対するシリアル化遅延が遅延制限の許容範囲内であり、フラグメント化する必要がないためです。

[フレームリレーでの分散型 LFI](#)

dLFIoFR 機能はフレームリレーでのマルチリンク PPP (MLPoFR) 機能の拡張です。MLP はフラグメンテーションに使用されます。この機能は、フラグメンテーションをサポートし、低遅延キューイングによりハイプライオリティパケットをインターリーブできる FRF.12 に似ています。

仮想テンプレート上の `ppp multilink interleave` コマンドは、**関連する仮想アクセス インターフェイスでのインターリーブを有効にするために必要です**。シリアル インターフェイスでの分散型 CEF スイッチングを有効化するだけでなく、このコマンドは分散型 LFI が動作するための前提条件でもあります。

注：フレームリレーから ATM へのインターネットワーキングを使用している場合を除き、dLFIoFR ではなく FRF.12 を使用することをお勧めします。これは、FRF.12 の帯域幅使用率が向上するためです

[ATM での分散型 LFI](#)

dLFIoATM 機能は ATM でのマルチリンク PPP (MLPoATM) 機能の拡張です。MLP はフラグメンテーションに使用されます。

仮想テンプレート上の `ppp multilink interleave` コマンドは、**関連する仮想アクセス インターフェイスでのインターリーブを有効にするために必要です**。シリアル インターフェイスでの分散型 CEF スイッチングを有効化するだけでなく、このコマンドは分散型 LFI が動作するための前提条件でもあります。

dLFIoATM では、ATM セルで不必要なパディングを発生させないように、パケットが ATM セルに適合するようなフラグメントサイズを選択することが重要です。たとえば、選択されたフラグメントサイズが 124 バイトであれば、124 バイトの IP ペイロードが最終的には $124 + 10$ (MLP ヘッダー) + 8 (SNAP ヘッダー) = 142 バイトとなることを意味します。最初のフラグメントは $124 + 10 + 2$ (最初のフラグメントの PID ヘッダー サイズ) + 8 = 144 バイトで送信されることに注意が必要です。これは、このパケットが 3 つの ATM セルを使用してペイロードを転送し、したがって、最も効率的にパケット化されるセルを使用することを意味します。

[dMLP と dLFIoLL の違い](#)

dMLP は送信側でのフラグメンテーションをサポートしませんが、dLFIoLL はサポートします。

注：マルチリンクバンドル内の複数のリンクで音声トラフィック用に使用されるインターリーブおよびフラグメンテーションは、バンドル内の複数のリンクを介して受信される音声トラフィックが順番に受信されることを保証するものではありません。音声の正しい順序付けは、上位層で行われます。

[分散型 MLFR](#)

分散型 MLFR 機能により、フレームリレー フォーラム マルチリンク フレームリレー UNI/NNI 実装合意 (FRF.16) に基づく機能がライン カード対応の Cisco 7500 および 7600 シリーズ ルータに追加されます。分散型 MLFR 機能は、複数のシリアル リンクを単一の帯域幅のバンドルに集約することができるので、特定のアプリケーションの帯域幅を増やすコスト効率の良い方法を提供します。MLFR は、フレーム リレー ネットワークのユーザネットワーク インターフェイス (UNI) およびネットワーク間インターフェイス (NNI) でサポートされます。

バンドルは、バンドル リンクと呼ばれる複数のシリアル リンクから構成されます。バンドル内の各バンドル リンクは物理インターフェイスに対応します。バンドル リンクは、フレームリレー データリンク層からは見えません。従って、フレームリレー機能はこれらのインターフェイス上では設定できません。これらのリンクに適用する正規のフレームリレー機能は、バンドル インターフェイス上で設定する必要があります。バンドル リンクは、ピア デバイスからは見えません。

分散型 DDR

分散型 DDR 機能は、ダイヤラ インターフェイスでの分散型スイッチングを実現できます。この機能がなければ、すべてダイヤルイントラフィックをスイッチングのホストにパントする必要があります。この機能があれば、制御パケットのみが RP に送信され、スイッチングの判断は接続が確立された後に VIP 自身で実行されます。

レガシー ダイヤラ設定およびダイヤラ プロファイル設定はどちらも、PPP カプセル化でのみサポートされています。ダイヤラ インターフェイスでの MLP もサポートされています。QoS はダイヤラ インターフェイスでの分散型スイッチングではサポートされていません。

分散型機能の前提条件および制約事項

前提条件

これらすべての分散型機能の一般的な前提条件を次に示します。

- 分散型シスコ エクスプレス フォワーディング (dCEF) スwitching がグローバルでイネーブルになっている必要があります。
- dCEF スwitching は、MLP バンドルの一部であるメンバー シリアル インターフェイスで有効化する必要があります。
- dCEF スwitching は、dLFloFR と dLFloATM インターフェイスの物理リンクで有効化する必要があります。
- インターリーブ設定は、LFloFR と LFloATM を分散するために必要です。
- dLFloFR と dLFloATM インターフェイス用の仮想テンプレート インターフェイスで、必要な帯域幅を設定します。
- RP で PPP デバッグが有効化されている場合、ルート スイッチ プロセッサ (RSP) 上で、
「MLP:Forwarded to wrong interface」このメッセージは混乱を招き、また不必要なものであるため (特にこのメッセージが Cisco Discovery Protocol (CDP) プロトコルに対するものである場合)、バンドルのメンバー リンクで no cdp enable を設定する必要があります。
- バンドルのすべてのメンバー リンクで、キープアライブを有効化する必要があります。

制約事項

これらすべての分散型機能に対する一般的な制限を次に示します。

- バンドル内には T1 および E1 回線を混在できません。
- サポートされている最大遅延差は 30 ms です。
- バンドルのすべての回線は、同じポート アダプタ (PA) に属している必要があります。
- ハードウェア圧縮はサポートされません。
- VIP または FlexWAN CEF は IP のみに限定されます。他のプロトコルはすべて RSP に送信されます。
- 送信側では、dMLP と dMLFR に対するフラグメンテーションはサポートされません。
- 古いキューイング メカニズムの多くは、dLFI ではサポートされていません。これらのメカニズムには、以下が含まれます。バーチャル テンプレート インターフェイスでの均等化キューイング
バーチャル テンプレート インターフェイスでのランダム検出カスタム キューイング
優先キューイング
- 均等化キューイング、ランダム検出 (dWRED)、プライオリティ キューイングは、Modular QoS CLI によりトラフィック ポリシーで設定できます。
- dLFloFR または dLFloATM を使用している場合、MLP バンドルあたり 1 つのリンクのみがサポートされます。dLFloFR または dLFloATM を使用している際に MLP バンドルで複数のリンクが使用されている場合、dLFI は自動的に無効になります。専用回線での dLFI を使用している場合、MLP バンドルで複数のリンクを dLFI により設定することができます。
- dLFloATM では、aal5snap および aal5mux のみがサポートされます。aal5nlpid と aal5ciscopp のカプセル化はサポートされません。
- Voice over IP のみサポートされます。Voice over Frame Relay と Voice over ATM は、dLFI 機能ではサポートされません。
- 圧縮リアルタイム プロトコル (CRTP) は、次の機能の組み合わせを使用する場合はマルチリンク インターフェイスで設定するべきではありません。LFI が有効化されたマルチリンク インターフェイスマルチリンク バンドルに、複数のメンバ リンクがあるプライオリティ機能付き QoS ポリシーがマルチリンク インターフェイスで有効化されている

dMLP と dLFI 設定では、プライオリティ パケットは MLP ヘッダーとシーケンス番号を含まず、MLP はプライオリティ パケットをすべてのメンバー リンクに分散します。その結果、CRTP で圧縮されたパケットは、受信側ルータに順番どおり届かない可能性があります。その結果、CRTP はパケット ヘッダーの圧縮を解除できなくなり、CRTP にパケットをドロップさせることとなります。

推奨事項

バンドルのメンバー リンクが同じ帯域幅を持っていることが推奨されます。不均等な帯域幅リンクをバンドルに追加すると、多くのパケットの再順序付けが発生し、バンドル全体のスループットが低下します。

これらの分散型機能の使用には、VIP2-50 (8 MB SRAM を持つ) 以上を推奨します。

バンドルとリンクの数、およびメモリ要件

『[Cisco 7500 シリーズ ルータの分散型マルチリンク ポイントツーポイント プロトコル](#)』を参照してください。

7600 SIP ラインカード上のハードウェアおよびソフトウェアの MLPPP または MLFR

MLP および MLFR はソフトウェア ベースまたはハードウェア ベースです。ハードウェア ベース

の MLP または MLFR では、Freedm はマルチリンク機能を提供し、MLP ヘッダーは Freedm チップが追加します。ソフトウェアベースの MLP または MLFR では、SIP ラインカードの CPU はマルチリンク機能 (VIP および FlexWAN の実装と似ています) を提供します。

ハードウェアベースの MLP または MLFR の実行には、制限および条件があります。

- ラインカードあたり最大 336 のバンドルを使用でき、セキュリティ ポスチャ アセスメント (SPA) (Freedm) の場合は 160 バンドルです。
- バンドルごとの最大の DS1/E1 の数は 12 です。
- すべてのリンクは同じ SPA (Freedm) に属している必要があります。
- バンドルのすべてのリンクは、同じ速度で動作する必要があります。
- TXフラグメントサイズは、128、256、または512です。CLIで設定されたフラグメントサイズは、サポートされている最も近いフラグメントサイズにマッピングされます。

```
IF (0 < cli_fragment_size - 6 < 256)
  configured_fragment_size = 128
Else IF (cli_fragment_size < 512)
  configured_fragment_size = 256
Else
  configured_fragment_size = 512
```

- RX フラグメント サイズは 1 ~ 9.6 KB です。
- シスコの独自の形式はサポートされていません (MLFR)。

ハードウェア LFI では、バンドルにリンクが 1 つしかなく、それが DS1/E1 の場合、フラグメンテーションとインターリーブは Freedm によって実施されます。

show ppp multilink の出力は、ハードウェア実装が実行されているかどうかを示します。

```
Multilink1, bundle name is M1
Bundle up for 00:14:51
Bundle is Distributed

0 lost fragments, 0 reordered, 0 unassigned
0 discarded, 0 lost received, 1/255 load
Member links: 1 active, 0 inactive (max not set, min not set)
Se6/1/0/1:0, since 00:14:51, no frags rcvd
Distributed fragmentation on. Fragment size 512. Multilink in Hardware.
```

マルチリンクがソフトウェアベースの場合、**show ppp multilink** の出力には Multilink in Hardware

パケットの受信後の流れ

Rx データ パス

1. パケットがドライバに受信されます。
2. カプセル化が次のように確認されます。基本的なカプセル化：dMLP では、入インターフェイスのカプセル化タイプは ET_PPP です。dMLFR では、入インターフェイスのカプセル化タイプは ET_FRAME_RELAY です。dLFloLL では、入インターフェイスのカプセル化タイプは ET_PPP です。dLFloFR では、入インターフェイスのカプセル化タイプは ET_FRAME_RELAY です。dLFloATM では、入インターフェイスのカプセル化タイプは ET_ATM です。dDialer では、カプセル化タイプは ET_PPP です。追加のカプセル化処理：ET_PPP では、NLPID が探し出されます。dMLP では、NLPID は MULTILINK です。

dLFloLL については、2 つのことを考慮します。VoIP パケット：MLP ヘッダーはなく、IP を示す NLPID があります。データ パケット：NLPID は MULTILINK です。dDialer では、パケットには MLP ヘッダーはなく、IP を示す NLPID があります。注：この場合、dCRTP(Distributed Compressed Real-Time Protocol)を設定できます。そのような場合、ヘッダーは次の処理の前に復元されます。

3. ET_FRAME_RELAY では、パケットが受信されたリンクが dMLFR に設定されている場合、パケットは dMLFR に対して処理されます。
4. dLFloFR と dLFloATM では、カプセル化タイプはそれぞれ ET_FRAME_RELAY と ET_ATM ですが、その中には PPP ヘッダーがあります。dLFloLL と同様、PPP ヘッダーはパケットが音声パケットであるかデータ パケットであるかを示します。dCRTP が設定されている場合、ヘッダーは次の処理の前に復元されます。音声パケットはただちにスイッチングされます。フラグメント化されたデータ パケットはスイッチングの前にリアセンブルする必要があります。ET_PPP では、PPP リンク パケットが含まれることがあります。ET_FRAME_RELAY では、MLFR 制御パケットが含まれることがあります。これら制御パケットはすべて、処理のために RP にパントされます。
5. 前述の復号化によっては、パケットに必要なスイッチングのタイプが確認されることがあります。リンク タイプは、パケットが IP スイッチドか MPLS スイッチドかを判定します。パケットは、それぞれのスイッチ機能に渡されます。
6. 分散型機能を伴うバンドルにより、IP ターボ ファスト スイッチング ベクトルが失われます。これは、パケットがメンバー リンクで受信されるために起こります。ただし、バンドルで受信されるように扱う必要があります。また、ホストにパントされる制御パケットをチェックする必要があります。主に dMLFR では、MLFR 制御パケットではないローカル管理インターフェイス (LMI) パケットがあります。これらについては、dLCI の番号スペースの異なる部分が使用されます。dLCI が復号されこのスペースに含まれる場合は必ず、LMI パケットとして認識されるので、パケットはホストにパントされます。VoIP パケット (低遅延キューにキューイングされている) は、MLP ヘッダーを追加することなくスイッチングされます。分散型機能は、フラグメント化されたデータ パケットが受信されると、パケットを受信してリアセンブルすることができます。リアセンブル プロセスは、後のセクションで説明します。パケットにタグ スイッチングが必要な場合、dMLP のタグ スイッチング ルーチンに渡されます。IP スイッチングが必要な場合、IP スイッチング ルーチンに渡されます。注：すべての非IPパケットは、dMLFRでホストにパントされます。
7. IP:IP スイッチング機能はすべてのパケットに共通です。これは主に次の3つのことをします。任意の機能が設定されると、パケットに必要な処理を行います。また、分散型ダイヤラが使用される場合、「インタレスティング パケット」が受信される際にアイドル タイマーの更新を行います。『[dialer idle-timeout \(interface\)](#)』、『[dialer fast-idle \(interface\)](#)』、および『[idle timer 設定パラメータの詳細に対するダイヤラ プロファイルの設定](#)』を参照してください。75xx ルータでは、隣接関係は出カインターフェイスの tx_acc_ptr 出カインターフェイスが仮想アクセス インターフェイスの場合、tx_acc_ptr NULL です。この場合、カプセル化を修正し、fib hwidb tx_acc_ptr を取得します。このルックアップとカプセル化の修正は、dLFloFR と dLFloATM では必須です。dLFloLL では、リンクはマルチリンク バンドルの一部として扱われます。注：パケットのTTLはここで調整され、IPフラグメンテーションのチェックが行われます。mci_status RXTYPE_DODIP
8. スイッチングの決定がなされると、パケットはインターフェイスから出力される準備が整います。インターフェイスは、ローカル スイッチングをサポートしているかどうかを確認されます。サポートしている場合、fastsend を介して直接送信されます。そうでなければ、ルート キャッシュ スイッチを試行します。QoS がインターフェイスに設定されている場合、ローカル スイッチング ベクトルは QoS によって失われることに注意してください。HQF はパケットをキューイングしてパケットをさらに処理し、最終的にインターフェイスから送

信されます。これは dLFI の場合です。dLFI では、フラグメンテーションおよびインターリーブが設定されます。QoS はフラグメンテーション ルーチンの起動を処理し、プライオリティ キューにキューイングされる音声パケットを持つフラグメント化されたパケットをインターリーブします (LLQ が設定されている場合)。これにより、VoIP パケットは巨大なデータ パケットをリンク経由で送出するために必要な遅延の影響を受けません。

Tx データ パス

`vip_dtq_consumer idb` を取得します。 `idb fastsend`

i) Fastsend

1. dMFR では、`fr_info` 構造は、`fr_info` への `if_index` に一致するテーブルから取得されます。制御パケットは、そのまま送信されます。フレーム ヘッダーには dLCI が含まれており、これによりこのパケットが LMI パケットであるかデータ パケットであるかを判定することができます。フレーム ヘッダーの `dlci` フィールドは、`dmfr` シーケンス番号によって上書きされます。LMI とデータ パケットに対して、異なるシーケンス番号が使用されます。注：個々の dLCI には、個別のシーケンス番号が使用されます。
2. dMLP では、制御パケットには高いプライオリティが設定されて送信されます。データ パケットでは、`dCRTP` が設定される場合、ヘッダーは圧縮されます。シーケンシング情報を含む VIP MLP ヘッダーが追加され、メンバー リンクから送信されます。
3. dLFI では、HQF はインターフェイスを介して送信されるパケットを傍受します。これが音声パケットの場合、音声パケットはプライオリティ キューに置かれ (LLQ が設定されている場合)、MLP カプセル化なしでインターフェイスから送信されます。データ パケットの場合、dLFI フラグメンテーション コードを呼び出し、フラグメントを QoS コードに返し、次いで音声トラフィックの遅延要求が満たされるように、プライオリティトラフィックによりインターリーブされます。また、`dCRTP` が設定されている場合は、音声パケットのヘッダーのみが圧縮されます。データ パケット ヘッダーはそのまま残されます。
4. dDialer では、パケットが送信される前に出力リンクのアイドル タイマーをリセットするために、パケットが分類されます。これは、複数のリンクが同じダイヤラに結合されている場合、出力リンクが選択された後に実行されます。ダイヤラ パケットにはヘッダーは追加されません。したがって、パケットのシーケンシングおよびリアセンブルはダイヤラ インターフェイスではサポートされません。

注：複数のリンクを持つ dMLP、dDialer、dMLFR、および dLFI では、トラフィックが転送される物理リンクはリンクの輻輳によって異なります。リンクで輻輳が発生した場合、次のリンクに移動し、それを繰り返します。(dMLFR、QoS のない dMLP、および dDialer 機能は、リンクに配分されるバイト数に基づいてリンクが選択されます。現在のリンクがすでにバイトのクォータをラウンドロビンごとに送信している場合、次のリンクを選択します。このクォータはリンクの `frag_bytes` によって決定されます。ダイヤラのメンバー インターフェイスでは、`frag_bytes` はインターフェイス帯域幅のデフォルト値に設定されます。)

注：出力VIPのインターフェイスのHQF設定では、HQFは`dtq_consumer`ベクトルを取得します。出力VIPにDMAされたパケットにはまず、HQFチェックが実行されます。出力インターフェイス上でQoSが設定されている場合、パケットがインターフェイスから`fastsent`される前に、パケットを処理するためにHQFが有効になります。

リアセンブル

通常の dDialer インターフェイスはリアセンブルとシーケンシングをサポートしていません。こ

これをダイヤラ インターフェイスで有効にするには、ダイヤラ インターフェイスでの MLP を設定する必要があります。これを実行すると、Rx および Tx パスは dMLP パスと同一になります。パケットが受信されると、シーケンス番号が、予期されたシーケンス番号と照合されます。

- シーケンス番号が一致する場合：パケットがフラグメント化されていないパケットであれば、リアセンブルは必要ありません。スイッチングの手順に進みます。パケットがフラグメントである場合、先頭ビットとエンドビットを確認し、フラグメントが受信されたときにパケットを構築します。
- シーケンス番号が一致しない場合：シーケンス番号が予期されたシーケンス番号のウィンドウ内にある場合、ソートされた「未割当フラグメント リスト」に配置します。のちに、予期されたシーケンス番号が受信されなかった場合、パケットがここに保存された場合はこのリストがチェックされます。シーケンス番号がウィンドウ内にはない場合、それを廃棄し、「損失フラグメントが受信されました」と報告します。のちにこのパケットの待機中にタイムアウトが発生すると、受信側が再同期され、次に受信されたパケットから再開します。

これらすべての場合に、正しく順序付けされたストリームがこのインターフェイスから送信されます。フラグメントが受信されると、完全なパケットが形成され、送信されます。

分散型機能の設定、確認およびデバッグ

このセクションでは、各分散型機能を確認しデバッグするために利用できる `show` および `debug` コマンドについて説明します。

dMFR の設定と確認

MFR の設定例

```
interface MFR1
  no ip address

interface MFR1.1 point-to-point
  ip address 181.0.0.2 255.255.0.0
  frame-relay interface-dlci 16
```

注：MFR インターフェイスは別の FR インターフェイスと同様であるため、ほとんどの FR 設定をサポートしています。

```
interface Serial5/0/0/1:0
  no ip address
  encapsulation frame-relay MFR1
  tx-queue-limit 26

interface Serial5/0/0/2:0
  no ip address
  encapsulation frame-relay MFR1
  tx-queue-limit 26

interface Serial5/0/0/3:0
  no ip address
  encapsulation frame-relay MFR1
```

RP の MFR バンドル状態の確認

show frame-relay multilink

```
Bundle: MFR1, State = up, class = A, fragmentation disabled
BID = MFR1
Bundle links:
  Serial5/0/0/3:0, HW state = up, link state = Add_sent, LID = Serial5/0/0/3:0
  Serial5/0/0/2:0, HW state = up, link state = Up, LID = Serial5/0/0/2:0
  Serial5/0/0/1:0, HW state = up, link state = Up, LID = Serial5/0/0/1:0
```

これは、2つのインターフェイスが正しく追加され、1つのインターフェイスはまだ MLFR LIP メッセージをネゴシエートしていないことを示します。

MFR バンドルおよびメンバー リンクについての詳細を取得するには、このコマンドを実行します。

show frame-relay multilink mfr1 detailed

```
Bundle: MFR1, State = up, class = A, fragmentation disabled
BID = MFR1
No. of bundle links = 3, Peer's bundle-id = MFR1
Rx buffer size = 36144, Lost frag timeout = 1000
Bundle links:
  Serial5/0/0/3:0, HW state = up, link state = Add_sent, LID = Serial5/0/0/3:0
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = , RTT = 0 ms
    Statistics:
      Add_link sent = 35, Add_link rcv'd = 0,
      Add_link ack sent = 0, Add_link ack rcv'd = 0,
      Add_link rej sent = 0, Add_link rej rcv'd = 0,
      Remove_link sent = 0, Remove_link rcv'd = 0,
      Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
      Hello sent = 0, Hello rcv'd = 0,
      Hello_ack sent = 0, Hello_ack rcv'd = 0,
      outgoing pak dropped = 0, incoming pak dropped = 0
  Serial5/0/0/2:0, HW state = up, link state = Up, LID = Serial5/0/0/2:0
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = Serial6/1/0/2:0, RTT = 32 ms
    Statistics:
      Add_link sent = 0, Add_link rcv'd = 0,
      Add_link ack sent = 0, Add_link ack rcv'd = 0,
      Add_link rej sent = 0, Add_link rej rcv'd = 0,
      Remove_link sent = 0, Remove_link rcv'd = 0,
      Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
      Hello sent = 7851, Hello rcv'd = 7856,
      Hello_ack sent = 7856, Hello_ack rcv'd = 7851,
      outgoing pak dropped = 0, incoming pak dropped = 0
  Serial5/0/0/1:0, HW state = up, link state = Up, LID = Serial5/0/0/1:0
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = Serial6/1/0/1:0, RTT = 32 ms
    Statistics:
      Add_link sent = 0, Add_link rcv'd = 0,
      Add_link ack sent = 0, Add_link ack rcv'd = 0,
      Add_link rej sent = 0, Add_link rej rcv'd = 0,
      Remove_link sent = 0, Remove_link rcv'd = 0,
      Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
      Hello sent = 7851, Hello rcv'd = 7856,
```

```
Hello_ack sent = 7856, Hello_ack rcv'd = 7851,  
outgoing pak dropped = 0, incoming pak dropped = 0
```

MFR デバッグ コマンド

これらのデバッグは、リンクがバンドルに追加されない問題のトラブルシューティングに役立ちます。

```
debug frame-relay multilink control
```

注：特定のMFRインターフェイスまたはシリアルインターフェイスが指定されていない場合、すべてのMFRリンクのデバッグが有効になります。ルータに多数のMFRリンクがある場合、膨大な数になることがあります。

このデバッグは、RPで受信されたMFRパケットのデバッグやMFR制御アクティビティのデバッグに役立ちます。

```
debug frame-relay multilink
```

注：トラフィックが多い場合、CPUに負荷がかかることがあります。

LC の dMFR バンドル状態の確認

```
show frame-relay multilink
```

注：現在は、LCでは使用できませんが、まもなく追加されます。それまでは、show ppp multilinkを使用します。

```
Bundle MFR1, 2 members  
bundle 0x62DBDD20, frag_mode 0  
tag vectors 0x604E8004 0x604C3628  
Bundle hwidb vector 0x6019271C  
idb MFR1, vc 0, RSP vc 0  
QoS disabled, fastsend (mlp_fastsend), visible_bandwidth 3072  
board_encap 0x60577554, hw_if_index 0, pak_to_host 0x0  
max_particles 400, mrru 1524, seq_window_size 0x200  
working_pak 0x0, working_pak_cache 0x0  
una_frag_list 0x0, una_frag_end 0x0, null_link 0  
rcved_end_bit 1, is_lost_frag 0, resync_count 0  
timeout 0, timer_start 0, timer_running 0, timer_count 0  
next_xmit_link Serial0/0:1, member 0x3, congestion 0x3  
dmlp_orig_pak_to_host 0x603E7030  
dmlp_orig_fastsend 0x6035DBC0  
bundle_idb->lc_ip_turbo_fs 0x604A7750  
0 lost fragments, 0 reordered, 0 unassigned  
0 discarded, 0 lost received  
0x0 received sequence, 0x58E sent sequence  
DLCI: 16  
769719 lost fragments, 22338227 reordered,  
0 unassigned  
27664 discarded, 27664 lost received  
0xF58 received sequence, 0x8DE sent sequence  
timer count 767176
```

Member Link: 2 active

Serial0/0:0, id 0x1, fastsend 0x60191E34, lc_turbo 0x601913AC, PTH 0x603E7030, OOF 0

Serial0/0:1, id 0x2, fastsend 0x60191E34, lc_turbo 0x601913AC, PTH 0x603E7030, OOF 0

[dMLP/dLFloLL の設定と確認](#)

[マルチリンク PPP の設定](#)

```
interface Multilink1
 ip address 151.0.0.2 255.255.0.0
 no cdp enable
 ppp chap hostname M1
 ppp multilink
!
```

シリアル インターフェイスでの設定例 :

```
interface Serial5/0/0/4:0
 no ip address
 encapsulation ppp
 tx-queue-limit 26
 no cdp enable
 ppp chap hostname M1
 ppp multilink group 1
!
interface Serial5/0/0/5:0
 no ip address
 encapsulation ppp
 tx-queue-limit 26
 no cdp enable
 ppp chap hostname M1
 ppp multilink group 1
!
```

注 : ppp chap hostname M1 コマンドは、CHAP 認証が有効であることを意味するものではありません。このコマンドでの文字列 M1 は、エンドポイント識別子として作用し、同じ 2 つのルータ間に複数のマルチリンクバンドルが存在しうる場合にのみ必要となります。このような場合、バンドルに属するすべてのリンクは同じエンドポイント識別子を持ち、異なるバンドルに属する 2 つのリンクはどれも同じエンドポイント識別子を持ちません。

[オプションの設定パラメータ](#)

[no] ppp multilink interleave

これにより、マルチリンクバンドルのインターリーブが有効化されます。これは、Modular QoS CLI と連携して動作します。ハイプライオリティパケットは、MLP シーケンスとヘッダーを追加することなく送信され、他のパケットはフラグメント化され MLP シーケンスとヘッダーと共に送信されます。

注 : インターリーブが複数のリンクで有効になっている場合は、優先順位の高いトラフィックが再順序付けされる可能性があります。インターリーブが有効化または無効化されるとき、バンドルをラインカードで有効化するためにバンドルのリセットが必要です。

```
ppp multilink mrru local value
```

これは、マルチリンクでの最大受信ユニットを指定します。最大でこのサイズのパケットがマルチリンク インターフェイスで許容されます。このサイズには MLP ヘッダーは含まれません。

```
ppp multilink mrru remote value
```

これは、リモート エンドがサポートする最小 MRRU を指定します。リモート エンド MRRU がこの値よりも小さいと、バンドルのネゴシエーションは失敗します。

```
ppp multilink fragment delay seconds
```

これはデータ フラグメントが原因の許容された遅延をミリ秒 (ms) で指定します。つまり、遅延の値は、許容される最大のフラグメント サイズを計算するために使用されます。この分散型実装は、次の 3 つの点で Cisco IOS の実装と異なります。

1. フラグメンテーションは、インターリーブが有効化されていない限り行われません。
2. さまざまな帯域幅のリンクがあるので、フラグメント サイズは最小帯域幅のインターフェイスに基づいて選択されます。

```
ppp multilink fragment disable
```

このコマンドは、分散型実装に何の機能も追加しません。フラグメンテーションは、インターリーブが有効化されている場合のみ発生します。また、インターリーブが有効化されている場合、`ppp multilink fragment disable` コマンドは無視されます。

[RP の dMFR バンドル状態の確認](#)

```
show ppp multilink
```

```
Multilink1, bundle name is M1
Endpoint discriminator is M1
Bundle up for 00:09:09, 1/255 load
Receive buffer limit 24000 bytes, frag timeout 1000 ms
Bundle is Distributed
  0/0 fragments/bytes in reassembly list
  0 lost fragments, 0 reordered
  0/0 discarded fragments/bytes, 0 lost received
  0x9 received sequence, 0x0 sent sequence
dLFI statistics:
      DLFI Packets   Pkts In   Chars In   Pkts Out   Chars Out
      Fragmented      0         0         0         0
      UnFragmented    9        3150      0         0
      Reassembled     9        3150      0         0
      Reassembly Drops 0
      Fragmentation Drops 0
      Out of Seq Frags 0
Member links: 2 active, 0 inactive (max not set, min not set)
  Se5/0/0/4:0, since 00:09:09, 768 weight, 760 frag size
  Se5/0/0/5:0, since 00:09:09, 768 weight, 760 frag size
```

1. バンドルが分散型モードの場合、**show ppp multilink** の出力に次が表示されます。Bundle is Distributed そうでなければ、バンドルは何らかの理由で分散されていません。
2. ラインカードで **ppp multilink interleave** が設定され有効化されている場合、**show ppp multilink dLFI** インターリーブが有効でないとき、出力は次のようになります。

```

Multilink1, bundle name is M1
Endpoint discriminator is M1
Bundle up for 00:00:00, 0/255 load
Receive buffer limit 24000 bytes, frag timeout 1000 ms
Bundle is Distributed
  0/0 fragments/bytes in reassembly list
  0 lost fragments, 0 reordered
  0/0 discarded fragments/bytes, 0 lost received
  0x0 received sequence, 0x2 sent sequence
Member links: 2 active, 0 inactive (max not set, min not set)
  Se5/0/0/5:0, since 00:00:00, 768 weight, 760 frag size
  Se5/0/0/4:0, since 00:00:03, 768 weight, 760 frag size

```

前の例のフラグメント サイズは 760 バイトです。

[LC の dMFR バンドル状態の確認](#)

show ppp multilink

```

dmlp_ipc_config_count 24
dmlp_bundle_count 2
dmlp_ipc_fault_count 1
dmlp_il_inst 0x60EE4340, item count 0
0, store 0, hwidb 0x615960E0, bundle 0x622AA060, 0x60579290, 0x6057A29C
1, store 0, hwidb 0x615985C0, bundle 0x622AA060, 0x60579290, 0x6057A29C
2, store 0, hwidb 0x0, bundle 0x0,
3, store 0, hwidb 0x0, bundle 0x0,
4, store 0, hwidb 0x0, bundle 0x0,
5, store 0, hwidb 0x0, bundle 0x0,
6, store 0, hwidb 0x0, bundle 0x0,
7, store 0, hwidb 0x0, bundle 0x0,
8, store 0, hwidb 0x0, bundle 0x0,
9, store 0, hwidb 0x0, bundle 0x0,

Bundle Multilink1, 2 members
  bundle 0x622AA060, frag_mode 0
  tag vectors 0x604E8004 0x604C3628
  Bundle hwidb vector 0x6057B198
  idb Multilink1, vc 4, RSP vc 4
  QoS disabled, fastsend (qos_fastsend), visible_bandwidth 3072
  board_encap 0x60577554, hw_if_index 0, pak_to_host 0x0
  max_particles 400, mrru 1524, seq_window_size 0x8000
  working_pak 0x0, working_pak_cache 0x0
  una_frag_list 0x0, una_frag_end 0x0, null_link 0
  rcved_end_bit 1, is_lost_frag 1, resync_count 0
  timeout 0, timer_start 0, timer_running 0, timer_count 1
  next_xmit_link Serial0/0:3, member 0x3, congestion 0x3
dmlp_orig_pak_to_host 0x603E7030
dmlp_orig_fastsend 0x6035DBC0
bundle_idb->lc_ip_turbo_fs 0x604A7750
  0 lost fragments, 0 reordered, 0 unassigned
  0 discarded, 0 lost received
  0xC3 received sequence, 0x0 sent sequence
Member Link: 2 active

```

Serial0/0:4, id 0x1, fastsend 0x60579290, lc_turbo 0x6057A29C, PTH 0x60579A18, OOF 0

Serial0/0:3, id 0x2, fastsend 0x60579290, lc_turbo 0x6057A29C, PTH 0x60579A18, OOF 0

dMFR では、シーケンス番号は LMI dLCI に使用されるバンドルのシーケンス番号と共に dLCI ごとに維持されます。

フィールド	説明
dmlp_ipc_config_count	マルチリンクまたは MLFR 設定に対する、LC に受信される IPC メッセージの数
dmlp_bundle_count	LC での MLP および MLFR バンドルの数
dmlp_ipc_fault_count	LC での失敗した設定メッセージの数。0 のはずです。ゼロでない場合は、何らかの問題があります。
tag_vectors	タグ スイッチングに使用される idb to tag_optimum_fs および idb to ip2tag_optimum_fs ベクトルを示します。
board_encap	7500 プラットフォームにチャネライズド リンクがある場合、ボード カプセル化の 2 バイトを追加するために使用される board_encap ベクトルを示します。リンクに非チャネライズド インターフェイスが含まれている場合、NULL のはずです。
max_particles	リアセンブル バッファに保持できるパーティクルの最大数
mrru	MLP のカプセル化を考慮に入れずに許容できるパケットの最大サイズ。MLFR インターフェイスには適用されません。
seq_window_size	シーケンス番号の最大ウィンドウ サイズ
working_pak	リアセンブルされている現在の PAK を示します。NULL です。
working_pak_cache	リアセンブルに使用される静的 PAK へのポインタです。最初の非完全パケットがバンドルで受信されたときに割り当てられます。
una_frag_list	リアセンブル キューでの最初のエントリです。エントリが NULL
una_frag_end	リアセンブル キューの最後のエントリです。
rcved_end_bit	バンドルがエンドビットを受信したため、先頭ビットにハントしていることを示します。
is_lost_frag	True の場合、フラグメントは損

	失されたと宣言されます。予期されたシーケンスを持つフラグメントが受信されるとクリアされます。
resync_count	受信側が送信側と同期されず、最後に受信したシーケンシング済みのフラグメントから再同期する必要があった回数を示します。
timeout	リアセンプル タイムアウトが発生し、パケットがリアセンプルキューから処理されていることを示します。
timer_start	リアセンプル タイマーが開始された回数
timer_running	リアセンプル タイマーが実行されているかどうかを示します。
timer_count	リアセンプル タイマーが期限切れになった回数を示します。
next_xmit_link	次のパケットが送信されるリンク
Member	存在するメンバーを示すビットフィールド。
Congestion	すべてのブランチで利用されていないフィールド。どのメンバーリンクが輻輳していないかを示します。
dmlp_orig_pak_to_host	パケットを RP にパントするために使用されるベクトル。
dmlp_orig_fastsend	MLP または MLFR がドライバの fastsend を変更する前の元のドライバの fastsend。
lost fragments	損失されたフラグメントの数 (受信側はこれらのフラグメントを受信していません)。更新がホストに送信される際に、定期的にクリアされます。
Reordered	予期された順序から外れて受信されたフラグメントの数。更新がホストに送信される際に、定期的にクリアされます。
Discarded	完全なパケットが作成されなかったために廃棄されたフラグメントの数
lost received	損失されたと見なされた、受信したフラグメントの数。リンク間遅延が 30 ms の dMLP リアセンプル タイムアウトよりも大きいことを示します。

dLFloFR と dLFloATM の設定と確認

```
class-map voip
  match ip precedence 3

policy-map llq
  class voip
    priority

int virtual-templatel
  service-policy output llq
  bandwidth 78
  ppp multilink
  ppp multilink interleave
  ppp multilink fragment-delay 8

int serial5/0/0/6:0
encapsulation frame-relay
frame-relay interface-dlci 16 ppp virtual-templatel
!--- Or

int ATM4/0/0
  no ip address
int ATM4/0/0.1 point-to-point
  pvc 5/100
  protocol ppp virtual-template 1
```

RP の dLFloFR/ATM バンドル状態の確認

show ppp multilink

```
Virtual-Access3, bundle name is dLFI
Endpoint discriminator is dLFI
Bundle up for 00:01:11, 1/255 load
Receive buffer limit 12192 bytes, frag timeout 1524 ms
Bundle is Distributed
  0/0 fragments/bytes in reassembly list
  0 lost fragments, 0 reordered
  0/0 discarded fragments/bytes, 0 lost received
  0x0 received sequence, 0x0 sent sequence
dLFI statistics:
      DLFI Packets   Pkts In   Chars In   Pkts Out   Chars Out
      Fragmented           0           0           0           0
      UnFragmented         0           0           0           0
      Reassembled          0           0           0           0
      Reassembly Drops           0
      Fragmentation Drops           0
      Out of Seq Frags           0
Member links: 1 (max not set, min not set)
Vi2, since 00:01:11, 240 weight, 230 frag size
```

注：バンドルは、仮想テンプレートで ppp multilink interleave が設定されている場合にのみ配布されます。このコマンドが設定されていない場合、バンドルは分散されません。

LC の dLFloFR/ATM バンドル状態の確認

LC で dLFI が実際に正常に動作していることを確認するには、次のコマンドを実行します。

show hqf interface

Interface Number 6 (type 22) Serial0/0:5

```
blt (0x62D622E8, index 0, hwidb->fast_if_number=35) layer PHYSICAL
scheduling policy: FIFO
classification policy: NONE
drop policy: TAIL
blt flags: 0x0
```

```
qsize 0 txcount 3 drops 0 qdrops 0 nobuffers 0
aggregate limit 16 individual limit 4 availbuffers 16
weight 1 perc 0.00 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 64 allocated_bw 64 qlimit_tuned 0 vc_encap 2
quantum 1500 credit 0 backpressure_policy 0 nothingoncalQ 1
```

next layer HQFLAYER_FRAMEDLCI_IFC (max entries 1024)

```
blt (0x62D620E8, index 0, hwidb->fast_if_number=35) layer FRAMEDLCI_IFC
scheduling policy: FIFO
classification policy: NONE
drop policy: TAIL
blt flags: 0x0
```

```
qsize 0 txcount 1 drops 0 qdrops 0 nobuffers 0
aggregate limit 16 individual limit 4 availbuffers 16
weight 1 perc 0.00 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 64 allocated_bw 64 qlimit_tuned 0 vc_encap 2
quantum 1500 credit 0 backpressure_policy 0 nothingoncalQ 1
```

```
blt (0x62D621E8, index 16, hwidb->fast_if_number=35) layer FRAMEDLCI_IFC
scheduling policy: WFQ
classification policy: PRIORITY_BASED
drop policy: TAIL
frag policy: root
blt flags: 0x0
```

```
qsize 0 txcount 2 drops 0 qdrops 0 nobuffers 0
aggregate limit 16 individual limit 4 availbuffers 16
weight 1 perc 0.00 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 64 allocated_bw 64 qlimit_tuned 0 vc_encap 2
quantum 240 credit 0 backpressure_policy 0 nothingoncalQ 1
```

next layer HQFLAYER_PRIORITY (max entries 256)

```
blt (0x62D61FE8, index 0, hwidb->fast_if_number=35) layer PRIORITY
scheduling policy: FIFO
classification policy: NONE
drop policy: TAIL
frag policy: leaf
blt flags: 0x0
```

```
qsize 0 txcount 0 drops 0 qdrops 0 nobuffers 0
aggregate limit 8 individual limit 2 availbuffers 8
weight 0 perc 0.99 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 32 allocated_bw 32 qlimit_tuned 0 vc_encap 2
quantum 240 credit 0 backpressure_policy 0 nothingoncalQ 1
```

```
blt (0x62D61CE8, index 1, hwidb->fast_if_number=35) layer PRIORITY
```

```

scheduling policy: FIFO
classification policy: NONE
drop policy: TAIL
blt flags: 0x0
Priority Conditioning enabled
qsize 0 txcount 0 drops 0 qdrops 0 nobuffers 0
aggregate limit 0 individual limit 0 availbuffers 0
weight 1 perc 0.00 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 0 allocated_bw 0 qlimit_tuned 0 vc_encap 2
quantum 240 credit 0 backpressure_policy 0 nothingoncalQ 1

PRIORITY: bandwidth 32 (50%)
        last 0 tokens 1500 token_limit 1500

blt (0x62D61EE8, index 255, hwidb->fast_if_number=35) layer PRIORITY
scheduling policy: WFQ
classification policy: CLASS_BASED
drop policy: TAIL
frag policy: MLPPP (1)
    frag size: 240, vc encaps: 0, handle: 0x612E1320
blt flags: 0x0

qsize 0 txcount 2 drops 0 qdrops 0 nobuffers 0
aggregate limit 8 individual limit 2 availbuffers 8
weight 1 perc 0.01 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 32 allocated_bw 32 qlimit_tuned 0 vc_encap 2
quantum 1 credit 0 backpressure_policy 0 nothingoncalQ 1

    next layer HQFLAYER_CLASS_HIER0 (max entries 256)

blt (0x62D61DE8, index 0, hwidb->fast_if_number=35) layer CLASS_HIER0
scheduling policy: FIFO
classification policy: NONE
drop policy: TAIL
frag policy: leaf
blt flags: 0x0

qsize 0 txcount 2 drops 0 qdrops 0 nobuffers 0
aggregate limit 8 individual limit 2 availbuffers 8
weight 1 perc 50.00 ready 1 shape_ready 1 wfq_clitype 0
visible_bw 32 allocated_bw 32 qlimit_tuned 0 vc_encap 2
quantum 240 credit 0 backpressure_policy 0 nothingoncalQ 1

```

プライオリティ層と WFQ 層があるはずですが、フラグメンテーションは、WFQ リーフ層の blt で実行されます。

[dDDR の設定と確認](#)

分散型 DDR は、グローバル設定で ip cef distributed を有効化し、ダイヤラ インターフェイスで ip route-cache distributed を有効化した場合にアクティブ化されます。

```

!--- Global configuration that enables distributed CEF switching. ip cef distributed --- Enable
distributed switching on the dialer interface (on the D-channel interface). int serial 3/1/0:23
ip route-cache distributed !--- Or, enable it on the dialer interface. int Dialer1 ip route-
cache distributed

```

分散型 DDR には他の特別な設定はありません。その他の設定は、通常の DDR 設定に従います。

[分散型ダイヤラ オンデマンド ルーティングの確認](#)

```
BOX2002# show isdn status
```

```
Global ISDN Switchtype = primary-net5
ISDN Serial3/1/0:23 interface
--- Network side configuration. dsl 0, interface ISDN Switchtype = primary-net5 Layer 1 Status:
ACTIVE Layer 2 Status: TEI = 0, Ces = 1, SAPI = 0, State = MULTIPLE_FRAME_ESTABLISHED
```

```
The ISDN status should be MULTIPLE_FRAME_ESTABLISHED. This means that the physical layer is
ready for ISDN connectivity. Layer 3 Status: 0 Active Layer 3 Call(s) Active dsl 0 CCBs = 0 The
Free Channel Mask: 0x807FFFFFF Number of L2 Discards = 0, L2 Session ID = 6 EDGE# show dialer
```

```
Serial6/0:0 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (15 secs)
Dialer state is data link layer up
Time until disconnect 119 secs
Current call connected never
Connected to 54321
```

```
Serial6/0:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (15 secs)
Dialer state is idle
```

dialer type ISDN PROFILE dialer state 接続されていないダイヤラ インターフェイスの状態は idle です。Idle timer このタイマーが期限切れになると、インターフェイスは即座に切断します。Idle timer 詳細については、『[ペアツーピア DDR をダイヤラ プロファイルで設定する](#)』を参照してください。

```
show ppp multilink
```

```
!--- From LC for dialer profile. dmlp_ipc_config_count 2 dmlp_bundle_count 1 dmlp_il_inst
0x60EE4340, item count 0 0, store 0, hwidb 0x0, bundle 0x0, 1, store 0, hwidb 0x0, bundle 0x0,
2, store 0, hwidb 0x0, bundle 0x0, 3, store 0, hwidb 0x0, bundle 0x0, 4, store 0, hwidb 0x0,
bundle 0x0, 5, store 0, hwidb 0x0, bundle 0x0, 6, store 0, hwidb 0x0, bundle 0x0, 7, store 0,
hwidb 0x0, bundle 0x0, 8, store 0, hwidb 0x0, bundle 0x0, 9, store 0, hwidb 0x0, bundle 0x0,
Bundle Dialer1, 1 member bundle 0x62677220, frag_mode 0 tag vectors 0x604E8004 0x604C3628 Bundle
hwidb vector 0x0 idb Dialer1, vc 22, RSP vc 22 QoS disabled, fastsend (mlp_fastsend),
visible_bandwidth 56 board_encap 0x60577554, hw_if_index 0, pak_to_host 0x0 max_particles 200,
mrru 1524, seq_window_size 0x8000 working_pak 0x0, working_pak_cache 0x0 una_frag_list 0x0,
una_frag_end 0x0, null_link 0 rcved_end_bit 1, is_lost_frag 0, resync_count 0 timeout 0,
timer_start 0, timer_running 0, timer_count 0 next_xmit_link Serial1/0:22, member 0x1,
congestion 0x1 dmlp_orig_pak_to_host 0x603E7030 dmlp_orig_fastsend 0x60381298 bundle_idb-
>lc_ip_turbo_fs 0x604A7750 0 lost fragments, 0 reordered, 0 unassigned 0 discarded, 0 lost
received 0x0 received sequence, 0x0 sent sequence Member Link: 1 active Serial1/0:22, id 0x1,
fastsend 0x60579290, lc_turbo 0x6057A29C, PTH 0x60579A18, OOF 0
```

表示されている変数は、dMLP のものと同じです。

dMLP と dDDR のデバッグ

RP で利用可能なデバッグ

dDDR

```
debug dialer [events | packets | forwarding | map]
```

コール セットアップなどの制御パス機能をデバッグするには、次のコマンドを実行します。詳細については、『[ダイヤラ イベントのデバッグ](#)』を参照してください。

```
debug ip cef dialer
```

CEF 関連のダイヤラ イベントをデバッグするには、次のコマンドを実行します。詳細については、『[ダイヤラ CEF](#)』を参照してください。

LC で利用可能なデバッグ

dMLP

制御パスのデバッグ : `debug multilink event`

データ パスのデバッグ : `debug multilink fragments`

データ パスと制御パス エラーのデバッグ : `debug multilink error`

SIP ラインカードでの dMLP のデバッグ

CI に基づいてパケットをダンプします。データ パケットと制御パケットは、制御 CI とシーケンス CI に基づいてラインカードにダンプできます。

```
test hw-module subslot_num dump ci CI-NUM [rx|tx] num_packets_to_dump
```

CI は次のように取得できます。

```
!--- Issue show controller serial interface for CTE1.
```

```
SIP-200-6# show controller serial 6/0/0:0
```

```
SPA 6/0 base address 0xB8000000 efc 1
```

```
Interface Serial6/0/0:0 is administratively down
Type 0xD Map 0x7FFFFFFF, Subrate 0xFF, mapped 0x1, maxmtu 0x5DC
Mtu 1500, max_buffer_size 1524, max_pak_size 1608 enc 84
ROM rev: 0, FW OS rev: 0x00000000 Firmware rev: 0x00000000
  idb=0x42663A30, pa=0x427BF6E0, vip_fci_type=0, port_per_spa=0
  SPA port type is set
  Host SPI4 in sync
  SPA=0x427BF6E0 status=00010407, host=00000101, fpga=0x427EDF98
  cmd_head=113, cmd_tail=113, ev_head=184, ev_tail=184
  ev_dropped=0, cmd_dropped=0
```

```
!--- Start Link Record Information. tag 0, id 0, anyphy 0, anyphy_flags 3, state 0
crc 0, idle 0, subrate 0, invert 0, priority 0
encap hdlc
corrupt_ci 65535, transparent_ci 1
```

```
!--- End Link Record Information. Interface Serial6/0/0:0 is administratively down Channel
Stats: in_throttle=0, throttled=0, unthrottled=0, started=1 rx_packets=0, rx_bytes=0,
rx_frame_aborts=0, rx_crc_errors=0 rx_giants=0, rx_non_aligned_frames=0, rx_runts=0,
rx_overruns=0 tx_packets=0, tx_bytes=0, tx_frame_aborts=0 is_congested=0, mapped=1, is_isdn_d=0,
tx_limited=1 fast_if_number=15, fastsend=0x403339E4 map=0x7FFFFFFF, turbo_vector_name=Copperhead
```

to Draco switching lc_ip_turbo_fs=403A9EEC, lc_ip_mdifs=403A9EEC

CT3 には、**show interface serial CT3_interface_name** の出力から取得できる `vc num` を取得する必要があります。

CI 情報は、SPA コンソールから取得できます。まず、SPA コンソール コマンドの出力を、`spa_redirect rp ct3_freedm336` コマンドにより RP にリダイレクトします。

`spa_ct3_test freedm show linkrec vc` コマンドは、必要な CI 情報を表示します。

dMFR

制御パスのデバッグ : `debug dmfr event`

データ パスのデバッグ : `debug dmfr packets`

データ パスと制御パス エラーのデバッグ : `debug dmfr error`

CI に基づいてパケットをダンプします。 [dMLP](#) を参照してください。

dLFI

制御パスのデバッグ : `debug dlfi event`

データ パスのデバッグ : `debug dlfi fragments`

データ パスと制御パス エラーのデバッグ : `debug dlfi error`

dDDR

特別なデバッグ コマンドはありません。 [dMLP debugs](#) を使用する必要があります。

dLFIoLL の場合、状況によっては dMLP および dLFI 両方のデバッグを使用する必要があります。これらのデバッグは条件に依存せず、したがってすべてのバンドルに対してトリガーします。

[よく寄せられる質問 \(FAQ\)](#)

1. **dMLP について**dMLPは、分散マルチリンクPPP([RFC1990](#)に記載)に対して短い [形式](#) です。この機能は、Cisco 7500 シリーズおよび 7600 シリーズなどの分散型プラットフォームでサポートされています。dMLPを使用すると、Cisco 7500シリーズルータのVIPまたは7600シリーズルータのFlexWANのT1/E1回線を、複数のT1/E1回線の合計帯域幅を持つバンドルに結合できます。これにより、顧客は T3/E3 回線を購入しなくても、T1/E1 よりも大きい帯域幅を増やすことができます。
2. **dMLP で「分散される」もの「分散される」という語は、**パケットスイッチングが RSP ではなく VIP によって実行されることを意味しています。これは、なぜですか。RSP スイッチング機能はある程度限定的であり、ほかにも多くの重要な働きをしています。パケットスイッチングに対応している VIP は RSP からこのアクティビティをオフロードします。RSP ベースの Cisco IOS は依然としてリンクを管理します。バンドルの作成およびティアダウンは RSP によって実行されます。また、PPP コントロールプレーンの処理は、すべての PPP 制御パケット (LCP、認証および NCP) の処理を含めて、依然として RSP によって実行されます。ただし、バンドルが確立されると、MLP パケットの処理はオンボード

CPU によるスイッチングのために VIP に引き継がれます。dMLP エンジン (VIP 上の) は、フラグメンテーション、インターリーブ、カプセル化、複数リンク間でのロード バランシングおよびインバウンドフラグメントのソートとリアセンブルを含め、すべての MLP の手順を処理します。7500 システムの VIP に実行される機能は、7600 ベースのシステムでは Flexwan/Enhanced-FlexWAN によって実行されます。

3. **バンドルが分散されたかどうかを確認するにはどうすればよいですか。ルータ コンソールで、show ppp multilink コマンドを実行します。**

```
Router# show ppp multilink
```

```
Multilink1, bundle name is udho2
Bundle up for 00:22:46
Bundle is Distributed
174466 lost fragments, 95613607 reordered, 129 unassigned
37803885 discarded, 37803879 lost received, 208/255 load
0x4D987C received sequence, 0x9A7504 sent sequence
Member links: 28 active, 0 inactive (max not set, min not set)
  Sell1/1/0/27:0, since 00:22:46, no frags rcvd
  Sell1/1/0/25:0, since 00:22:46, no frags rcvd
!--- Output suppressed.
```

4. **RSP16 または SUP720 にアップグレードした場合、dMLP のパフォーマンスは向上しますか。** いいえ。dMLP (または任意の分散機能) のスイッチングパフォーマンスは、対象の VIP または FlexWAN によって異なります。たとえば、VIP6-80 のパフォーマンスは VIP2-50 のパフォーマンスよりも優れています。
5. **どの PA をこの機能で使用できますか。** PA-MC-T3PA-MC-2T3+PA-MC-E3PA-MC-2E1PA-MC-2T1PA-MC-4T1PA-MC-8T1PA-MC-8E1PA-MC-STM-1PA-MC-8TE1+PA-4T+PA-8TCT3IP-50 (7500 のみ)
6. **いくつかのリンクを単一のバンドルで設定できますか。** この回答には多くの側面が含まれます。主なボトルネックはラインカード (VIP/FlexWAN/Enhanced-FlexWAN2) の CPU の性能です。ハードリミットは、バンドルごとに 56 のリンクですが、CPU の性能または限られたバッファが原因で、多くの場合それほど多くを設定する (およびそれほど多くのトラフィックスイッチングを持つ) ことはできません。これらの数値は、次のガイドラインに基づきます (VIP/FlexWAN/Enhanced-FlexWAN2 の CPU およびメモリに基づく)。VIP2-50 (w/ 4 MB SRAM) 最大で T1 = 12VIP2-50 (w/ 8 MB SRAM) 最大で T1 = 16VIP4-80 最大で T1 = 40VIP6-80 最大で T1 = 40FlexWAN の T1 の最大数はまもなく更新されます Enhanced-FlexWAN 最大で E1 = 1 日あたり 21 の E1 (ラインカードあたり 42 の E1)
7. **3 つの T1 それぞれに 3 つのバンドルを設定するか、9 つの T1 に 1 つのバンドルを設定する場合、パフォーマンスに違いはありますか。** パフォーマンスに違いはないことが、ラボテストで証明されています。ただし、単一のバンドルに多くの T1 が含まれる場合 (例えば単一のバンドルに 24 または 28 の T1)、バッファが不足するという問題が生じます。単一のバンドルに 8 よりも多いメンバーリンク (T1/E1) を使用しないことを推奨します。
8. **バンドルの帯域幅はどのように決定されますか。** バンドルの帯域幅を設定する必要はありません。すべてのメンバーリンクの集約帯域幅です。バンドルに 4 つの T1 がある場合、バンドルの帯域幅は 6.144Mbps になります。
9. **どちらが優れていますか。CEF ロード バランシング、もしくは dMLP でしょうか。** 簡単な答えはありません。必要により、どちらが適切かが変わってきます。MLP の利点: CEF ロード バランシングは IP トラフィックだけに適用されます。MLP はバンドルを介して送信されるすべてのトラフィックのバランシングを実行します。MLP はパケットの順序を維持します。IP そのものは再順序付けに耐性があるため、これは問題とはならないでしょう。実際、シーケンシングを維持することに関連する余分のコストは、MLP を避ける理由となり得ます。IP は、データグラムをばらばらの順序で配送するネットワークを意図しており、IP を使用するものはすべて、再順序付けをすることが想定されています。とはいえ、この事実

にかかわらず、実際には再順序付けにより様々な問題が生じます。MLP はピアシステムに単一の論理接続を提供します。QoS は、マルチリンクバンドルでサポートされています。MLP は、ユーザが現在のニーズに基づいてメンバーリンクを追加または削除できるように、動的帯域幅の機能を提供します。MLP は、CEF ロードバランシングが 6 つの平行 IP パスに限定される、大量の数のリンクをバンドルできます。フロー単位の CEF ロードバランシングでは、1 つの T1 に対する任意のフローの最大帯域幅が制限されます。たとえば、音声ゲートウェイを使用しているお客様は、同じ送信元と宛先を持つコールを大量に使用できるため、パスを 1 つだけ使用できます。MLP の欠点：MLP は各パケットやフレームに余分なオーバーヘッドを追加します。MLP は CPU に負荷をかけます。dMLP はラインカード CPU に負荷をかけます。

10. どのように 2 つのルータ間の複数のバンドルを設定できますか。マルチリンクは、どのバンドルにリンクが追加されるかを、ピアの名前とエンドポイント識別子により判定します。2 つのシステム間の複数の個別のバンドルを作成する標準の方法は、いくつかのリンクに自身を異なるものとして認識させることです。推奨される方法は、`ppp chap hostname name` コマンドを使用することです。
11. 別の PA からのメンバーリンクを使用することはできますか。いいえ。dMLP を実行する場合は、サポートされていません。ただし、別の PA からメンバーリンクが追加されると、制御は RSP に与えられ、dMLP ではなくなります。MLP は依然として機能しますが、dMLP の利点はなくなります。
12. 両方のベイからのメンバーリンクを混在させることはできますか。いいえ。dMLP を実行する場合は、サポートされていません。ただし、別の PA からメンバーリンクが追加されると、制御は RSP に与えられ、dMLP ではなくなります。MLP は依然として機能しますが、dMLP の利点はなくなります。
13. 異なる VIP または FlexWAN をまたがってメンバーリンクを設定できますか。いいえ。dMLP を実行する場合は、サポートされていません。ただし、別の PA からメンバーリンクが追加されると、制御は RSP に与えられ、dMLP ではなくなります。MLP は依然として機能しますが、dMLP の利点はなくなります。
14. 単一の PA からの異なるポートをまたがってメンバーリンクを設定できますか。(たとえば、PA-MC-2T3+ の各 CT3 ポートから 1 つのメンバーリンク)。はい。同じ PA からである限り、問題はありません。
15. T3 または E3 ポートをバンドルできますか。いいえ。7500/VIP、7600/FlexWAN、および 7600/FlexWAN2 では、dMLP を使用できるのは DS0、n*DS0、T1、および E1 の速度だけです。注：分散 MLPPP は、T1/E1 またはサブレート T1/E1 速度で設定されたメンバーリンクでのみサポートされます。チャネライズド STM-1/T3/T1 インターフェイスも、T1/E1 またはサブレート T1/E1 の速度の dMLPPP をサポートしています。分散型 MLPPP は、クリアチャンネル T3/E3 以上のインターフェイス速度で設定されているメンバーリンクではサポートされません。
16. 「再順序付け」フラグメントとは何ですか。受信されたフラグメントまたはパケットが予期されるシーケンス番号に一致しない場合、パケットサイズが様々である場合、これは必ず発生します。固定サイズのパケットの場合でも、PA ドライバは 1 つのリンクで受信されたパケットを処理し、ラウンドロビンごとには実行されないため、これが発生することがあります(パケット送信中に dMLP で発生するのと同じです)。再順序付けは、パケット損失を意味しません。
17. 「損失」フラグメントとは何ですか。フラグメントまたはパケットがばらばらの順序で受信され、ばらばらのフラグメントまたはパケットがすべてのリンクで受信された場合はいつでも、もう 1 つのケースは、順序がばらばらのフラグメントがリストに保存され、限界(VIP の SRAM に基づいて決定され、バンドルに割り当てられたもの)に達する場合で、損失フラグメントカウンタはインクリメントされ、リンクの次のシーケンス番号が処理さ

れます。

18. **dMLP はどのように損失フラグメントを検出しますか。**シーケンス番号：シーケンス番号 N のフラグメントが届くことを予期していたのに、全てのリンクがシーケンス番号 N よりも大きいフラグメントを受け取る場合、フラグメント N の損失に気づきます。同じリンクで数字の大きな番号のフラグメントの後にそれが正当な方法で届くことはないからです。
タイムアウト：かなりの時間待機してもフラグメントが届かない場合、最終的には損失とみなし、次に進みます。リアセンブルバッファ オーバーフロー：フラグメント N の到着を待っており、その間に他のフラグメント (N よりも大きいシーケンス番号を持つ) がいくつかのリンクに到着する場合、フラグメント N が届くまでそれらのフラグメントをリアセンブルバッファにパーク保留する必要があります。バッファできる量には限りがあります。バッファがオーバーフローすると、フラグメント N の損失を宣言し、バッファにあるものに対する処理を再開します。
19. **「受信損失」とは何ですか。**受信損失フラグメントまたはパケットには 2 つの原因が考えられます。受信されたフラグメントまたはパケットが、予期されたシーケンス範囲ウィンドウから外れている場合、パケットに受信損失とマークすることで、ドロップされます。受信されたフラグメントまたはパケットが予期されたシーケンス範囲ウィンドウ内にあるが、このパケットをリペアメントするためにパケットヘッダーを割り当てることができない場合、パケットはドロップされ、受信損失としてマークされます。
20. **暗号化は dMLP でサポートされていますか。**No.
21. **PFC ヘッダー圧縮はサポートしていますか。**いいえ、分散型パスではサポートされていません。遠端ルータは、PFC ヘッダー圧縮を設定するように推奨されていません。圧縮されたヘッダー フレームまたはパケットを受信すると、非分散型モードにフォールバックされるからです。dMLP を引き続き実行したい場合、PFC ヘッダー圧縮を両端で無効にする必要があります。
22. **ソフトウェア圧縮は dMLP でサポートされていますか。**いいえ。ソフトウェア圧縮は分散型パスでは動作しないからです。
23. **フラグメンテーションは送信側でサポートされていますか。**Vanilla dMLP ではサポートされていません。Vanilla dMLP でフラグメントを受信するには問題はありませんが、送信側では、フラグメンテーションは発生しません。送信側のフラグメンテーションは、dMLP インターフェイスで `ppp multilink interleave` が設定されている場合にサポートされます。
24. **MLP バンドルのメンバー リンクに ping を実行することはできますか。**いいえ、メンバーリンクに IP アドレスを設定することはできません。
25. **リンクの MTU および MLP フラグメント サイズには何らかの依存関係がありますか。**いいえ。MTU サイズは MLP フラグメント サイズとは関係ありません。他のフレームと同様に、MLP フラグメントがシリアルリンクの MTU サイズを超えることはできません。
26. **ルータの単一ペア間で 2 つの MLP バンドルを設定することは可能ですか。**はい、できます。ただし、これは不十分なロード バランシングにつながる可能性があります。2 つのルータのみを使用して 3 つ以上のルータをシミュレートするというテストベッドには役立ちますが、現実的な価値は何もありません。共通のピアに向かうすべてのリンクは同じバンドルに配置する必要があります。定義からして、バンドルは特定のピアに向かうリンクのセットです。「ピア」は、LCP および認証フェーズで提供されるユーザ名とエンドポイント識別子により識別されます。2 つのルータ間で複数のバンドルを作成しようとすることは、各ルータを相手に対して複数に見せるようにマスカレードすることを意味します。これらは自身を適切に特定する必要があります。
27. **メンバー リンクに異なるキューイング アルゴリズムを設定できますか。**バンドルに関連するすべてのキューイング メカニズムは、メンバー リンク レベルではなくバンドル レベルで適用される必要があります。ただし、キュー アルゴリズムを設定しても、パケットがバンドルからスイッチングされる方法には影響を及ぼしません。

28. Cisco 7500 で dMLP が有効化されている場合、マルチリンクバンドルのメンバーリンクに対して tx-queue-limit がデフォルトで 26 に設定されているのはなぜですか。帯域幅 T1/E1 のシリアルインターフェイスの場合、tx-queue-limit は 4 または 5 です。マルチリンクで T1/E1 をバンドルすると、バンドルの帯域幅が増加します。スイッチングは、MLP インターフェイスの帯域幅に基づいて実行されるため、メンバーリンクの tx-queue-limit を増やす必要があります。プライマリリンクと呼ばれる 1 つのメンバーリンクのみがスイッチングに使用されるので、その tx-queue-limit を増やす必要があります。また、この値は、テストとその後の調整を実施した後に選択された実証された値です。一般の導入では、バンドルは 4 つから 6 つよりも多い T1/E1 リンクを含むことはありません。26 という値は、6 つから 8 つの T1/E1 リンクに完全に対応しているため、この値が選択されました。
29. dMLP 実装での遅延差およびその値は何ですか。dMLP は 30 ミリ秒の遅延差をサポートします。これは、フラグメントが時間 T で受信され、順序がばらばらであるかどうかを意味します (シーケンス番号 100 を予期していたが、101 を受信した)。シーケンス番号 100 が T+30 ミリ秒まで受信されない場合は損失と宣言され、101 から処理を開始できる場合はそうします。101 から開始できない場合 (それが中間フラグメントの場合)、開始フラグメントを持つフラグメント (例えば 104) を探し、そこから開始します。
30. 7500 で、パケットがマルチリンクを持つ IP レベルでフラグメント化される場合はどうなりますか。パケットが IP レベルでフラグメント化される場合、中間ホップでリアセンブルされずに送信されますが、宛先ルータでリアセンブルされます。
31. 7500 で、パケットが MLP レベルでフラグメント化されるとどうなりますか。パケットが MLP レベルでフラグメント化され、リアセンブルされたパケットが MRRU よりも大きい場合、パケットはマルチリンクでドロップされます。送信側のフラグメンテーションは、dLFI でのみ dMLP でサポートされます。packet_size が frag_size よりも大きく、MRRU よりも小さい場合にのみ、パケットは MLP レベルでフラグメント化されます。MRRU よりも多くのパケットが送信され、IP レベルでフラグメント化されていない場合、パケットは MRRU よりも多いため、MLP レベルでフラグメント化されていないすべてのパケットサイズをドロップします。
32. MRRU はどのように計算されますか。MRRU は次の設定に従って計算されます。入ってくる新しいメンバーリンクに対しては、MRRU は、メンバーリンクで設定された MRRU に従って LCP レベルで再びネゴシエートされます。ppp multilink mrru interface コマンドにより、リンクインターフェイスで設定された値。設定されていない場合、親インターフェイスでの ppp multilink mrru コマンドの設定から継承された値。どちらの値も設定されている場合、リンクインターフェイスの値が優先されます。デフォルトの MRRU は 1524 です。

デバッグの機能拡張

次の機能拡張が今後実施されます。計画はまだ完了していません。

- LC での debug frame-relay multilink コマンドを有効化する。
- インターフェイスごと、およびパケットの特定の数ごとに現在のデバッグ CLI を拡張する。
- dDDR では、QoS 機能はまだサポートされていません。これは適切なビジネスケースでのみ実施されます。

関連情報

- [ダイヤラ CEF](#)

- [ダイヤラ プロファイルを使用したピアツーピア DDR の設定](#)
- [MPLS : マルチリンクPPPサポート](#)
- [Cisco 7500 シリーズ ルータの分散型マルチリンク ポイントツーポイント プロトコル](#)
- [分散型マルチリンク フレームリレー \(FRF.16 \)](#)
- [専用線上での分散リンク フラグメンテーションとインターリーピング](#)
- [QoS を実装した PPP リンク上での VoIP \(LLQ / IP RTP プライオリティ、LFI、cRTP \)](#)
- [トラブルシューティング テクニカル ノート : Cisco 7500 シリーズ ルータ](#)
- [ルータ製品のサポート ページ : Cisco Systems](#)
- [テクニカル サポートとドキュメント - Cisco Systems](#)