



Cisco ASR 9000 シリーズ ルータでのフレームリレーの設定

このモジュールでは、フレームリレーカプセル化を設定した Packet-over-SONET/SDH (POS)、マルチリンク、およびシリアルインターフェイスで使用できる、任意設定のフレームリレーパラメータについて説明します。

Cisco IOS XR ソフトウェアのフレームリレーインターフェイス設定の機能履歴

リリース	変更内容
リリース 4.0.0	フレームリレーのサポートは、次の SPA に対して追加されました。 <ul style="list-style-type: none">• Cisco 2 ポート チャネライズド OC-12c/DS0 SPA• Cisco 1 ポート チャネライズド OC-48/STM-16 SPA• Cisco 8 ポート OC-12c/STM-4 POS SPA• Cisco 2 ポート OC-48c/STM-16 POS/RPR SPA• Cisco 1 ポート OC-192c/STM-64 POS/RPR XFP SPA 次のフレームリレー機能のサポートが Cisco 2 ポート チャネライズド OC-12c/DS0 SPA に対して追加されました。 <ul style="list-style-type: none">• Multilink Frame Relay (FRF.16)• エンドツーエンドフラグメンテーション (FRF.12)
リリース 4.0.1	フレームリレーのサポートは、次の SPA に対して追加されました。 <ul style="list-style-type: none">• Cisco 1 ポート チャネライズド OC-3/STM-1 SPA• Cisco 2 ポートおよび 4 ポート クリア チャネル T3/E3 SPA• Cisco 4 ポート OC-3c/STM-1 POS SPA• Cisco 8 ポート OC-3c/STM-1 POS SPA
リリース 4.1.0	フレームリレーのサポートは、次の SPA に対して追加されました。 <ul style="list-style-type: none">• Cisco 4 ポート チャネライズド T3/DS0 SPA• Cisco 8 ポート チャネライズド T1/E1 SPA

内容

- 「フレームリレー設定の前提条件」 (P.592)
- 「フレームリレー インターフェイスに関する情報」 (P.592)
- 「フレーム リレーの設定」 (P.600)
- 「フレーム リレーの設定例」 (P.617)
- 「その他の関連資料」 (P.621)

フレームリレー設定の前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

フレームリレーを設定する前に、次の条件を満たしていることを確認します。

- 使用しているハードウェアが POS インターフェイスまたはシリアル インターフェイスをサポートしている必要があります。
- 対応するモジュールの説明に従って、**encapsulation frame relay** コマンドを使用し、インターフェイスでフレームリレーのカプセル化をイネーブルにしました。
 - マルチリンク バンドル インターフェイスでフレーム リレー カプセル化をイネーブルにするには、「[マルチリンク フレーム リレー バンドル インターフェイスの設定](#)」 (P.606) を参照してください。
 - POS インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco ASR 9000 シリーズ ルータでの POS インターフェイスの設定](#)」モジュールを参照してください。
 - シリアル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco ASR 9000 シリーズ ルータでのシリアル インターフェイスの設定](#)」モジュールを参照してください。

フレームリレー インターフェイスに関する情報

ここでは、フレームリレー インターフェイスを設定する際のさまざまな側面について説明します。

- 「[フレーム リレー カプセル化](#)」 (P.592)
- 「[Multilink Frame Relay \(FRF.16\)](#)」 (P.596)
- 「[エンドツーエンド フラグメンテーション \(FRF.12\)](#)」 (P.600)

フレーム リレー カプセル化

Cisco ASR 9000 シリーズ ルータでは、フレーム リレーは、POS インターフェイス、シリアル メイン インターフェイス、およびそれらのインターフェイスで設定された PVC でサポートされます。フレームリレーのカプセル化をインターフェイスでイネーブルにするには、インターフェイス コンフィギュレーション モードで **encapsulation frame-relay** コマンドを使用します。

フレーム リレー インターフェイスは、次の 2 つのタイプのカプセル化フレームをサポートします。

- Cisco (これがデフォルト値です)
- IETF

PVC に Cisco または IETF カプセル化を設定するには、インターフェイス コンフィギュレーション モードで **encapsulation frame-relay** コマンドを使用します。



(注) **encapsulation** コマンドで PVC のカプセル化タイプが明示的に設定されていない場合、その PVC はメイン インターフェイスからカプセル化タイプを継承します。

encapsulation frame relay コマンドおよび **encap (PVC)** コマンドについては、次のモジュールを参照してください。

- POS インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco ASR 9000 シリーズ ルータでの POS インターフェイスの設定](#)」モジュールを参照してください。
- シリアル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「[Cisco ASR 9000 シリーズ ルータでのシリアル インターフェイスの設定](#)」モジュールを参照してください。

フレームリレーのカプセル化でインターフェイスを設定し、その他の追加コンフィギュレーション コマンドを適用しない場合、表 23 のデフォルト インターフェイス設定が示されます。これらのデフォルト設定は、このモジュールの説明に従って設定で変更できます。

表 23 フレームリレーのカプセル化のデフォルト設定

パラメータ	設定ファイルのエントリ	デフォルト設定	コマンド モード
PVC Encapsulation	encap {cisco ietf}	cisco (注) encap コマンドを設定しない場合、PVC のカプセル化タイプはフレームリレーのメイン インターフェイスから継承されます。	PVC コンフィギュレーション
Type of support provided by the interface	frame-relay intf-type {dce dte}	dte	インターフェイス コンフィギュレーション

表 23 フレームリレーのカプセル化のデフォルト設定 (続き)

パラメータ	設定ファイルのエントリ	デフォルト設定	コマンド モード
LMI type supported on the interface	frame-relay lmi-type [ansi cisco q933a]	DCE の場合、デフォルト設定は cisco です。 DTE の場合、デフォルト設定は DCE でサポートされる LMI タイプに合わせて同期されます。 (注) インターフェイスをデフォルトの LMI タイプに戻すには、 no frame-relay lmi-type [ansi cisco q933a] コマンドを使用します。	インターフェイス コンフィギュレーション
Disable or enable LMI	frame-relay lmi disable	デフォルトでフレームリレー インターフェイスの LMI はイネーブルです。 LMI をディセーブルにした後で改めてイネーブルにするには、 no frame-relay lmi disable コマンドを使用します。	インターフェイス コンフィギュレーション



(注) LMI ポーリング関連のコマンドのデフォルト設定については、表 24 (P.595) と表 25 (P.595) を参照してください。

LMI

ローカル管理インターフェイス (LMI) プロトコルは、PVC の追加、削除、およびステータスをモニタリングします。また、フレームリレーのユーザネットワーク インターフェイス (UNI) を構成するリンクの完全性も検証します。

フレームリレー インターフェイスは、UNI で次のタイプの LMI をサポートします。

- ANSI-ANSI T1.617 Annex D
- Q.933-ITU-T Q.933 Annex A
- シスコ

インターフェイスで使用する LMI タイプを設定するには、**frame-relay lmi-type** コマンドを使用します。



(注) 使用する LMI タイプは、メイン インターフェイスに設定されている PVC と対応している必要があります。フレームリレー接続の両エンドの LMI タイプは一致する必要があります。

使用しているルータが別の非フレームリレー ルータに接続するスイッチとして機能する場合、**frame-relay intf-type dce** コマンドを使用して、データ通信機器 (DCE) をサポートする LMI タイプを設定します。

ルータがフレームリレー ネットワークに接続している場合、**frame-relay intf-type dte** コマンドを使用して、データ端末装置 (DTE) をサポートする LMI タイプを設定します。



(注) DTE インターフェイスでは、デフォルトで LMI タイプの自動検知がサポートされています。

システムのフレーム リレー インターフェイスの情報と統計情報を表示するには、**show frame-relay lmi** および **show frame-relay lmi-info** コマンドを EXEC モードで使用します。(type および *interface-path-id* 引数を指定するとき、メイン インターフェイスの情報を指定する必要があります)。エラーしきい値、イベント数とポーリング検証タイマーを変更し、フレーム リレー インターフェイスのモニタおよびトラブルシューティングを実行する際に役立つ情報を収集する **show frame-relay lmi** コマンドを使用できます。

LMI のタイプが **cisco** (デフォルトの LMI タイプ) である場合、1 つのインターフェイスでサポートできる PVC の最大数は、メイン インターフェイスの MTU サイズに関連しています。カードまたは SPA でサポートされる PVC の最大数を計算するには、次の公式を使用します。

$$(MTU - 13) / 8 = \text{PVC の最大数}$$

cisco LMI で設定した POS PVC でサポートされる PVC のデフォルトの数は 557 です。また、**cisco** LMI で設定したシリアル PVC でサポートされる PVC のデフォルトの数は 186 です。

シスコ製ではない LMI タイプの場合、単一のメイン インターフェイスで最大 992 PVC がサポートされます。



(注) 特定の LMI タイプをインターフェイスに設定する場合、**no frame-relay lmi-type [ansi | cisco | q933a]** コマンドを使用して、インターフェイスをデフォルトの LMI タイプに戻します。

表 24 は、DCE 用に設定した PVC で LMI ポーリング オプションを変更するときに使用できるコマンドです。

表 24 DCE の LMI ポーリング コンフィギュレーション コマンド

パラメータ	設定ファイルのエントリ	デフォルト設定
Sets the error threshold on a DCE interface.	lmi-n392dce threshold	3
Sets the monitored event count.	lmi-n393dce events	4
Sets the polling verification timer on the DCE end.	lmi-t392dce seconds	15

表 25 は、DTE 用に設定した PVC で LMI ポーリング オプションを変更するときに使用できるコマンドです。

表 25 DTE の LMI ポーリング コンフィギュレーション コマンド

パラメータ	設定ファイルのエントリ	デフォルト設定
Set the number of Line Integrity Verification (LIV) exchanges performed before requesting a full status message.	lmi-n391dte polling-cycles	6
Sets the error threshold.	lmi-n392dte threshold	3
Sets the monitored event count.	lmi-n393dte events	4
Sets the polling interval (in seconds) between each status inquiry from the DTE end.	frame-relay lmi-t391dte seconds	10

Multilink Frame Relay (FRF.16)

マルチリンク フレーム リレー (MFR) は、次の共有ポート アダプタ (SPA) でのみサポートされません。

- Cisco 1 ポート チャネライズド STM-1/OC-3 SPA
- Cisco 2 ポート チャネライズド OC-12c/DSO SPA

マルチリンク フレーム リレー ハイ アベイラビリティ

MFR は、次のレベルのハイ アベイラビリティをサポートします。

- MFR は、プロセス再起動をサポートしていますが、一部の統計情報は、特定のプロセスの再起動時にリセットされます。
- MFR メンバー リンクは、ルート スイッチ プロセッサ (RSP) スイッチオーバー中も動作可能でず。

マルチリンク フレーム リレーの設定の概要

マルチリンク フレームリレー インターフェイスは、インターフェイスでフレームリレーのカプセル化を可能にするマルチリンク バンドルの一部です。マルチリンク フレームリレー インターフェイスを作成するには、次のコンポーネントを設定します。

- MgmtMultilink コントローラ
- フレームリレーのカプセル化を可能にするマルチリンク バンドル インターフェイス
- バンドル ID 名
- マルチリンク フレームリレー サブインターフェイス
- バンドル インターフェイスの帯域幅クラス
- シリアル インターフェイス

MgmtMultilink コントローラ

次のコマンドを使用して、コントローラのマルチリンク バンドルを設定します。

```
controller MgmtMultilink rack/slot/bay/controller-id  
bundle bundleId
```

この設定で、汎用マルチリンク バンドルのコントローラが作成されます。コントローラ ID 番号はコントローラ チップのゼロベース インデックスです。現在、マルチリンク フレームリレーをサポートする SPA には、1 ペイごとに 1 コントローラしかないため、コントローラの ID 番号は常にゼロ (0) です。

マルチリンク バンドル インターフェイス

マルチリンク バンドルを作成した後は、次のコマンドを使用して、フレームリレーのカプセル化を可能にするマルチリンク バンドル インターフェイスを作成します。

```
interface multilink interface-path-id  
encapsulation frame-relay
```

この設定で、マルチリンク バンドル インターフェイスにマルチリンク フレームリレー サブインターフェイスを作成できます。



(注) マルチリンク バンドル インターフェイス上のカプセル化をフレームリレーに設定した後は、マルチリンク バンドルに関連付けられたメンバー リンクがインターフェイスにある場合、カプセル化は変更できません。

バンドル ID 名



(注) バンドル ID 名は、フレームリレー フォーラム 16.1 (FRF 16.1) でのみ設定できます。

バンドル ID (**bid**) 名は、インターフェイスの両エンドポイントのバンドル インターフェイスを識別します。バンドル ID 名は、一貫したリンクの割り当てを確保するために情報要素で交換されます。

デフォルトで、インターフェイス名 (たとえば Multilink 0/4/1/0/1) がバンドル ID 名として使用されます。ただし、オプションで **frame-relay multilink bid** コマンドを使用して名前を作成することもできます。



(注) デフォルトの名前を使用するか、**frame-relay multilink bid** コマンドを使用して名前を作成するかにかかわらず、各バンドルに固有の名前を指定することを推奨します。

バンドル ID 名の長さは、ヌルの終端文字を含めて 50 文字までです。バンドル ID 名はバンドル インターフェイス レベルで設定され、各メンバー リンクに適用されます。

バンドル ID 名を設定するには、次のコマンドを使用します。

```
interface multilink interface-path-id  
frame-relay multilink bid bundle-id-name
```

マルチリンク フレームリレー サブインターフェイス

マルチリンク フレームリレー サブインターフェイスを設定するには、次のコマンドを使用します。

```
interface multilink interface-path-id[.subinterface {12transport | point-to-point}]
```

1 つのマルチリンク バンドル インターフェイスには最大 992 サブインターフェイスを設定できます。



(注) サブインターフェイス レベルで特定のフレームリレー インターフェイス機能を設定します。

マルチリンク フレームリレー サブインターフェイス機能

次のコマンドは、マルチリンク フレームリレー バンドル サブインターフェイスで特定の機能を設定するために使用できます。

- **mtu** *MTU size*
- **description**
- **shutdown**
- **bandwidth** *bandwidth*
- **service-policy** {**input** | **output**} *policy-map-name*



(注) **service-policy** コマンドを入力すると、ポリシー マップをマルチリンク フレームリレー バンドル サブ インターフェイスに付加できるようになりますが、フレームリレー PVC コンフィギュレーション モードで次の操作を実行する必要があります。詳細については、「[マルチリンク フレーム リレー バンドル インターフェイスの設定](#)」(P.606) を参照してください。

バンドル インターフェイスの帯域幅クラス



(注) 帯域幅クラスは、マルチリンク バンドル インターフェイスでのみ設定できます。

マルチリンク フレームリレー インターフェイスでは、3 タイプの帯域幅クラスのいずれかを設定できます。

- a — 帯域幅クラス A
- b — 帯域幅クラス B
- c — 帯域幅クラス C

帯域幅クラス A を設定し、1 つまたは複数のメンバー リンクがアップ (PH_ACTIVE) の場合、バンドル インターフェイスもアップで、BL_ACTIVATE がフレームリレー接続にシグナリングされます。すべてのメンバー リンクがダウンの場合、バンドル インターフェイスはダウンで、BL_DEACTIVATE がフレームリレー接続にシグナリングされます。

帯域幅クラス B を設定し、すべてのメンバー リンクがアップ (PH_ACTIVE) の場合、バンドル インターフェイスはアップで、BL_ACTIVATE がフレームリレー接続にシグナリングされます。いずれかのメンバー リンクがダウンの場合、バンドル インターフェイスはダウンで、BL_ACTIVATE がフレームリレー接続にシグナリングされます。

帯域幅クラス C を設定する場合、バンドルのリンクのしきい値を 1 ~ 255 に設定する必要があります。このしきい値は、バンドル インターフェイスをアップにするため、およびフレームリレー接続に BL_ACTIVATE をシグナリングするために必要な、アップ (PH_ACTIVE) にするリンクの最小数です。アップ状態のリンク数がこのしきい値未満の場合、バンドル インターフェイスがダウンになり、BL_DEACTIVATE がフレームリレー接続にシグナリングされます。しきい値に 1 を入力した場合の動作は、帯域幅クラス A と同じです。アップ状態のメンバー リンク数よりも多いしきい値を入力すると、バンドルはダウンのままです。

フレームリレー マルチリンク バンドル インターフェイスの帯域幅クラスを設定するには、次のコマンドを使用します。

```
interface multilink interface-path-id
    frame-relay multilink bandwidth-class {a | b | c [threshold]}
```

デフォルトは a (帯域幅クラス A) です。

シリアル インターフェイス

T3 コントローラと T1 コントローラを設定した後は、マルチリンク フレームリレー バンドル サブ インターフェイスにシリアル インターフェイスを追加できます。この場合、シリアル インターフェイスを設定し、マルチリンク フレームリレー (mfr) としてカプセル化し、それをバンドル インターフェイス (マルチリンク グループ番号によって指定されます) に割り当て、リンクの名前を設定します。MFR 確認応答タイムアウト値、再送信の再試行回数および hello 間隔もバンドル リンクに対して設定できます。

マルチリンク フレームリレー シリアル インターフェイスを設定するには、次のコマンドを使用します。


```

interface serial rack/slot/module/port/t1-num:channel-group-number
encapsulation mfr
multilink group group number
frame-relay multilink lid link-id name
frame-relay multilink ack ack-timeout
frame-relay multilink hello hello-interval
frame-relay multilink retry retry-count

```



(注)

MFR バンドルのすべてのシリアル リンクは、マルチリンク インターフェイスの **mtu** コマンドの値を継承します。そのため、MFR バンドルのメンバーとして設定する前に、シリアル インターフェイスで **mtu** コマンドを設定しないでください。デフォルト以外の MTU 値を設定している場合、MFR バンドルのメンバーとしてシリアル インターフェイスを設定しようとすると、Cisco IOS XR ソフトウェアによってブロックされます。また、MFR バンドルのメンバーとして設定されているシリアル インターフェイスの **mtu** コマンド値を変更しようとした場合もブロックされます。

show コマンド

マルチリンク フレームリレー シリアル インターフェイスの設定を検証するには、次の **show** コマンドを使用します。

```
show frame-relay multilink location node id
```

```
show frame-relay multilink interface serial interface-path-id [detail | verbose]
```

次の例は、**show frame-relay multilink location** コマンドの表示出力です。

```

RP/0/RSP0/CPU0:router# show frame-relay multilink location 0/4/cpu0
Member interface: Serial0/4/2/0/9:0, ifhandle 0x05007b00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800

Bundle interface: Multilink0/4/2/0/2, ifhandle 0x05007800
  Member Links: 4 active, 0 inactive
  State = Up,   BW Class = C (threshold  3)
  Member Links:
    Serial0/4/2/0/12:0, HW state = Up, link state = Up
    Serial0/4/2/0/11:0, HW state = Up, link state = Up
    Serial0/4/2/0/10:0, HW state = Up, link state = Up
    Serial0/4/2/0/9:0,  HW state = Up, link state = Up

Member interface: Serial0/4/2/0/10:0, ifhandle 0x05007c00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800

Member interface: Serial0/4/2/0/11:0, ifhandle 0x05007d00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800

Member interface: Serial0/4/2/0/12:0, ifhandle 0x05007e00
HW state = Up, link state = Up
Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800

```

次の例は、コマンドの表示出力です。

```

RP/0/RSP0/CPU0:router# show frame-relay multilink interface serial 0/4/2/0/10:0

Member interface: Serial0/4/2/0/10:0, ifhandle 0x05007c00
HW state = Up, link state = Up

```

Member of bundle interface Multilink0/4/2/0/2 with ifhandle 0x05007800

エンドツーエンド フラグメンテーション (FRF.12)

データリンク接続識別子 (DLCI) を使用して、FRF.12 エンドツーエンド フラグメンテーション接続を設定することができます。ただし、チャネライズドフレームリレー シリアルインターフェイスで設定する必要があります。



(注) **fragment end-to-end** コマンドは、POS インターフェイス、またはマルチリンク フレームリレー バンドルインターフェイスの DLCI では使用できません。

FRF.12 エンドツーエンド フラグメンテーションを DLCI 接続で設定するには、次のコマンドを使用します。

fragment end-to-end fragment-size

fragment-size 引数には、シリアルインターフェイスのフラグメント サイズをバイト単位で定義します。



(注) DLCI 接続では、優先順位の高いフラグメントと低いフラグメントのインターリーブが発生するように、パケットを優先順位の高低で分類する出力サービス ポリシーを設定することを強くお勧めします。

フレーム リレーの設定

次の項では、フレームリレー インターフェイスの設定方法について説明します。

- 「インターフェイスでのデフォルト フレームリレー設定の変更」 (P.600)
- 「フレームリレーのカプセル化を設定したインターフェイスでの LMI のディセーブル」 (P.603)
- 「マルチリンク フレーム リレー バンドル インターフェイスの設定」 (P.606)
- 「チャネライズドフレームリレー シリアルインターフェイスでの FRF.12 エンドツーエンドフラグメンテーションの設定」 (P.612)

インターフェイスでのデフォルト フレームリレー設定の変更

このタスクは、Packet-over-SONET/SDH (POS)、マルチリンク、またはシリアルインターフェイスでフレームリレー カプセル化が設定されている場合にデフォルトのフレームリレー パラメータを変更するために実行します。

前提条件

デフォルトのフレームリレー設定を変更する前に、次のモジュールの説明に従ってインターフェイスでフレームリレーをイネーブルにする必要があります。

- POS インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「Cisco ASR 9000 シリーズ ルータでの POS インターフェイスの設定」モジュールを参照してください。

- シリアル インターフェイスでフレームリレーのカプセル化をイネーブルにするには、このマニュアルの「Cisco ASR 9000 シリーズ ルータでのシリアル インターフェイスの設定」モジュールを参照してください。



(注)

POS インターフェイスまたはシリアル インターフェイスでフレーム リレー カプセル化をイネーブルにする前に、そのインターフェイスに割り当て済みの IP アドレスがないことを確認します。IP アドレスが割り当て済みの場合、フレームリレーのカプセル化をイネーブルにすることはできません。フレームリレーの場合、IP アドレスとサブネット マスクはサブインターフェイスで設定します。

制約事項

- LMI タイプは、アクティブにする接続の両エンドで一致する必要があります。
- インターフェイスでフレームリレーのカプセル化を削除し、そのインターフェイスを PPP または HDLC のカプセル化で設定し直す前に、すべてのインターフェイス、サブインターフェイス、LMI、およびそのインターフェイスのフレームリレー設定を削除する必要があります。

手順の概要

- configure**
- interface type interface-path-id**
- frame-relay intf-type {dce | dte}**
- frame-relay lmi-type [ansi | cisco | q933a]**
- encap {cisco | ietf}**
- end**
または
commit
- show interfaces [summary | [type interface-path-id] [brief | description | detail | accounting [rates]]] [location node-id]**

手順の詳細

	コマンドまたはアクション	目的
ステップ1	configure 例： RP/0/RSP0/CPU0:router# configure	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。

■ フレーム リレーの設定

コマンドまたはアクション	目的
<p>ステップ2 <code>interface type interface-path-id</code></p> <p>例: RP/0/RSP0/CPU0:router(config)# interface pos 0/4/0/1</p>	<p>インターフェイス コンフィギュレーション モードを開始します。</p>
<p>ステップ3 <code>frame-relay intf-type {dce dte}</code></p> <p>例: RP/0/RSP0/CPU0:router(config-if)# frame-relay intf-type dce</p>	<p>インターフェイスで提供するサポートのタイプを設定します。</p> <ul style="list-style-type: none"> • 使用しているルータが別のルータに接続するスイッチとして機能する場合、frame-relay intf-type dce コマンドを使用して、DCE をサポートする LMI タイプを設定します。 • ルータがフレームリレー ネットワークに接続している場合、frame-relay intf-type dte コマンドを使用して、DTE をサポートする LMI タイプを設定します。 <p>(注) デフォルトのインターフェイス タイプは DTE です。</p>
<p>ステップ4 <code>frame-relay lmi-type [ansi q933a cisco]</code></p> <p>例: RP/0/RSP0/CPU0:router(config-if)# frame-relay lmi-type ansi</p>	<p>インターフェイスでサポートする LMI タイプを選択します。</p> <ul style="list-style-type: none"> • ANSI T1.617a-1994 Annex D の定義に従って LMI を使用するには、frame-relay lmi-type ansi コマンドを入力します。 • Cisco の定義（標準ではありません）に従って LMI を使用するには、frame-relay lmi-type cisco コマンドを使用します。 • ITU-T Q.933 (02/2003) Annex A の定義に従って LMI を使用するには、frame-relay lmi-type q933a コマンドを使用します。 <p>(注) デフォルトの LMI タイプは Cisco です。</p>
<p>ステップ5 <code>encap {cisco ietf}</code></p> <p>例: RP/0/RSP0/CPU0:router (config-fr-vc)# encap ietf</p>	<p>フレームリレー PVC のカプセル化を設定します。</p> <p>(注) PVC のカプセル化タイプを明示的に設定しない場合、その PVC はメイン インターフェイスからカプセル化タイプを継承します。</p>

	コマンドまたはアクション	目的
ステップ6	<pre>end または commit</pre> <p>例:</p> <pre>RP/0/RSP0/CPU0:router(config-if)# end または RP/0/RSP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> end コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ7	<pre>show interfaces [summary [type interface-path-id] [brief description detail accounting [rates]]] [location node-id]</pre> <p>例:</p> <pre>RP/0/RSP0/CPU0:router# show interface pos 0/4/0/1</pre>	<p>(任意) 指定したインターフェイスの設定を検証します。</p>

フレームリレーのカプセル化を設定したインターフェイスでの LMI のディセーブル

フレームリレーのカプセル化が設定されたインターフェイスで LMI をディセーブルにするには、次のタスクを実行します。



(注)

フレームリレーのカプセル化がイネーブルなインターフェイスでは、デフォルトで LMI がイネーブルです。インターフェイスの LMI をディセーブルにした後で改めてイネーブルにするには、インターフェイス コンフィギュレーション モードで **no frame-relay lmi disable** コマンドを使用します。

手順の概要

1. **configure**
2. **interface type interface-path-id**
3. **frame-relay lmi disable**

■ フレーム リレーの設定

4. **end**
または
commit
5. **show interfaces** [**summary** | [*type interface-path-id*] [**brief** | **description** | **detail** | **accounting** [**rates**]]] [**location node-id**]

手順の詳細

	コマンドまたはアクション	目的
ステップ1	configure 例 : RP/0/RSP0/CPU0:router# <code>configure</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	interface <i>type interface-path-id</i> 例 : RP/0/RSP0/CPU0:router(config)# <code>interface POS 0/4/0/1</code>	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3	frame-relay lmi disable 例 : RP/0/RSP0/CPU0:router(config-if)# <code>frame-relay lmi disable</code>	指定したインターフェイスで LMI をディセーブルにします。

	コマンドまたはアクション	目的
<p>ステップ4</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RSP0/CPU0:router(config-if)# end または RP/0/RSP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを実行すると、変更をコミットするように要求されます。 <p>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</p> <ul style="list-style-type: none"> - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 - cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。 <ul style="list-style-type: none"> • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。 	
<p>ステップ5</p> <pre>show interfaces [summary [type interface-path-id] [brief description detail accounting [rates]]] [location node-id]</pre> <p>例:</p> <pre>RP/0/RSP0/CPU0:router# show interfaces POS 0/1/0/0</pre>	<p>(任意) 指定したインターフェイスで LMI がディセーブルになっていることを確認します。</p>	

マルチリンク フレーム リレー バンドル インターフェイスの設定

次に示す手順は、マルチリンク フレームリレー (MFR) バンドル インターフェイスとそのサブインターフェイスを設定するときに実行します。

前提条件

MFR バンドルを設定する前に、次の SPA がインストールされていることを確認してください。

- 1 ポート チャネライズド STM-1/OC-3 SPA
- 2 ポート チャネライズド OC-12c/DS0 SPA

制約事項

- マルチリンク フレーム リレー バンドル インターフェイスのすべてのメンバリンクは、同じタイプにする必要があります (たとえば、T1 または E1)。メンバー リンクは、ポイントツーポイントなど、同じフレーム構成タイプにし、同じ帯域幅クラスにする必要があります。
- すべてのメンバリンクがフル T1 または E1 である必要があります。DS0 など、フラクショナルリンクはサポートされません。
- すべてのメンバリンクは、同じ SPA 上にある必要があります。そうでなければ、関連しないバンドルと見なされます。
- すべてのメンバリンクは、遠端でも同じラインカードまたは SPA に接続している必要があります。
- サポートされる DLCI の範囲 16 ~ 1007 に基づいて、最大 992 の MFR サブインターフェイスが各メインインターフェイスでサポートされます。
- Cisco 1 ポート チャネライズド OC-3/STM-1 SPA および 2 ポート チャネライズド OC-12c/DS0 SPA には次の追加の注意事項があります。
 - ラインカードごとに最大 700 の MFR バンドルがサポートされます。
 - システムごとに最大 2600 の MFR バンドルがサポートされます。
 - ラインカードごとに最大 4000 のフレーム リレー レイヤ 3 サブインターフェイスがサポートされます。
 - システムごとに最大 8000 のフレーム リレー レイヤ 3 サブインターフェイスがサポートされます。
- MLFR バンドルの一部であるフレーム リレー サブインターフェイスでのフラグメンテーションはサポートされません。
- MFR バンドルのすべてのシリアル リンクは、マルチリンク インターフェイスの **mtu** コマンドの値を継承します。そのため、MFR バンドルのメンバーとして設定する前に、シリアルインターフェイスで **mtu** コマンドを設定しないでください。Cisco IOS XR ソフトウェアは次の処理をブロックします。
 - インターフェイスがデフォルト以外の MTU 値で設定されている場合、MFR バンドルのメンバーとしてシリアル インターフェイスを設定しようとする処理。
 - MFR バンドルのメンバーとして設定されているシリアル インターフェイスの **mtu** コマンド値を変更しようとする処理。

手順の概要

1. configure

2. **controller MgmtMultilink** *rack/slot/bay/controller-id*
3. **exit**
4. **controller t3** *interface-path-id*
5. **mode** *type*
6. **clock source** {**internal** | **line**}
7. **exit**
8. **controller {t1 | e1}** *interface-path-id*
9. **channel-group** *channel-group-number*
10. **timeslots** *range*
11. **exit**
12. **exit**
13. **interface multilink** *interface-path-id*[*.subinterface* {**l2transport** | **point-to-point**}]
14. **encapsulation frame-relay**
15. **frame-relay multilink bid** *bundle-id-name*
16. **frame-relay multilink bandwidth-class** {**a** | **b** | **c** [*threshold*]}
17. **exit**
18. **interface multilink** *interface-path-id*[*.subinterface* {**l2transport** | **point-to-point**}]
19. **ipv4 address** *ip-address*
20. **pvc** *dldi*
21. **service-policy** {**input** | **output**} *policy-map*
22. **exit**
23. **exit**
24. **interface serial** *interface-path-id*
25. **encapsulation mfr**
26. **multilink group** *group-id*
27. **frame-relay multilink lid** *link-id name*
28. **frame-relay multilink ack** *ack-timeout*
29. **frame-relay multilink hello** *hello-interval*
30. **frame-relay multilink retry** *retry-count*
31. **exit**
32. **end**
または
commit
33. **exit**
34. **show frame-relay multilink interface** *type interface-path-id* [**detail** | **verbose**]

■ フレーム リレーの設定

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>configure</code> 例: RP/0/RSP0/CPU0:router# config	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>controller MgmtMultilink rack/slot/bay/controller-id</code> 例: RP/0/RSP0/CPU0:router(config)# controller MgmtMultilink 0/1/0/0	<code>rack/slot/bay/controller-id</code> 表記で汎用マルチリンクバンドルのコントローラを作成し、マルチリンク管理コンフィギュレーション モードを開始します。コントローラ ID 番号はコントローラ チップのゼロベース インデックスです。現在、マルチリンクフレームリレーをサポートする SPA には、1 ベイごとに 1 コントローラしかないため、コントローラの ID 番号は常にゼロ (0) です。
ステップ 3	<code>exit</code> 例: RP/0/RSP0/CPU0:router(config-mgmtmultilink)# exit	マルチリンク管理コンフィギュレーション モードを終了します。
ステップ 4	<code>controller t3 interface-path-id</code> 例: RP/0/RSP0/CPU0:router(config)# controller t3 0/1/0/0	<code>rack/slot/module/port</code> 表記で T3 コントローラ名を指定し、T3 コンフィギュレーション モードを開始します。
ステップ 5	<code>mode type</code> 例: RP/0/RSP0/CPU0:router(config-t3)# mode t1	チャネライズするマルチリンクのタイプを設定します (たとえば、28 T1)。
ステップ 6	<code>clock source {internal line}</code> 例: RP/0/RSP0/CPU0:router(config-t3)# clock source internal	(任意) 個々の E3 リンクのクロッキングを設定します。 (注) デフォルトのクロック ソースは internal です。 (注) シリアル リンクでクロッキングを設定する場合、一方のエン드를 internal にし、もう一方を line にする必要があります。接続の両エン드에 internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エン드에 line クロッキングを設定すると、ラインはアップ状態になりません。
ステップ 7	<code>exit</code> 例: RP/0/RSP0/CPU0:router(config-t3)# exit	T3/E3 コントローラ コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 8	<p><code>controller {t1 e1} interface-path-id</code></p> <p>例： RP/0/RSP0/CPU0:router(config)# controller t1 0/1/0/0/0</p>	T1 または E1 コンフィギュレーション モードを開始します。
ステップ 9	<p><code>channel-group channel-group-number</code></p> <p>例： RP/0/RSP0/CPU0:router(config-t1)# channel-group 0</p>	T1 チャンネル グループを作成し、そのチャンネル グループのチャンネル グループ コンフィギュレーション モードを開始します。
ステップ 10	<p><code>timeslots range</code></p> <p>例： RP/0/RSP0/CPU0:router(config-t1-channel_group)# timeslots 1-24</p>	<p>1 つまたは複数の DS0 タイムスロットをチャンネル グループに関連付け、関連付けたシリアル サブインターフェイスをそのチャンネル グループに作成します。</p> <ul style="list-style-type: none"> • T1 コントローラの場合、範囲は 1 ~ 24 タイムスロットです。 • E1 コントローラの場合、範囲は 1 ~ 31 タイムスロットです。 • すべてのタイムスロットを単一のチャンネル グループに割り当てることも、タイムスロットを複数のチャンネル グループに分割することもできます。
ステップ 11	<p><code>exit</code></p> <p>例： RP/0/RSP0/CPU0:router(config-t1-channel_group)# exit</p>	チャンネル グループ コンフィギュレーション モードを終了します。
ステップ 12	<p><code>exit</code></p> <p>例： RP/0/RSP0/CPU0:router(config-t1)# exit</p>	T1 コンフィギュレーション モードを終了します。
ステップ 13	<p><code>interface multilink interface-path-id[.subinterface {l2transport point-to-point}]</code></p> <p>例： RP/0/RSP0/CPU0:router(config)# interface Multilink 0/1/0/0/100</p>	バンドルのフレームリレーのカプセル化を指定するマルチリンク バンドル インターフェイスを作成します。マルチリンク バンドル インターフェイスにマルチリンク フレームリレー サブインターフェイスを作成します。
ステップ 14	<p><code>encapsulation frame-relay</code></p> <p>例： Router(config-if)# encapsulation frame-relay</p>	フレームリレーのカプセル化タイプを指定します。
ステップ 15	<p><code>frame-relay multilink bid bundle-id-name</code></p> <p>例： Router(config-if)# frame-relay multilink bid MFRBundle</p>	(注) (任意) デフォルトで、インターフェイス名 (たとえば Multilink 0/4/1/0/1) がバンドル ID 名として使用されます。ただし、オプションで frame-relay multilink bid コマンドを使用して名前を作成することもできます。

■ フレーム リレーの設定

コマンドまたはアクション	目的
<p>ステップ 16 <code>frame-relay multilink bandwidth-class {a b c [threshold]}</code></p> <p>例 : Router(config-if)# frame-relay multilink bandwidth-class a</p>	<p>マルチリンク フレームリレー インターフェイスでは、3 タイプの帯域幅クラスのいずれかを設定します。</p> <ul style="list-style-type: none"> • a — 帯域幅クラス A • b — 帯域幅クラス B • c — 帯域幅クラス C <p>デフォルトは a (帯域幅クラス A) です。</p>
<p>ステップ 17 <code>exit</code></p> <p>例 : RP/0/RSP0/CPU0:router(config-if)# exit</p>	<p>インターフェイス コンフィギュレーション モードを終了します。</p>
<p>ステップ 18 <code>interface multilink interface-path-id[.subinterface {l2transport point-to-point}]</code></p> <p>例 : RP/0/RSP0/CPU0:router(config)# interface Multilink 0/1/0/0/100.16 point-to-point</p>	<p><code>rack/slot/bay/controller-id bundleid.subinterace [point-to-point l2transport]</code> 表記でマルチリンク サブインターフェイスを作成し、サブインターフェイス コンフィギュレーション モードを開始します。</p> <ul style="list-style-type: none"> • l2transport - 接続回線として扱います。 • point-to-point - ポイントツーポイントリンクとして扱います。 <p>1 つのマルチリンク バンドル インターフェイスには最大 992 サブインターフェイスを設定できます。DLCI は 16 ~ 1007 です。</p>
<p>ステップ 19 <code>ipv4 address ip-address</code></p> <p>例 : RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 3.1.100.16 255.255.255.0</p>	<p>次の形式でインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。</p> <p><code>A.B.C.D/prefix</code> または <code>A.B.C.D/mask</code></p>
<p>ステップ 20 <code>pvc dlci</code></p> <p>例 : RP/0/RSP0/CPU0:router (config-subif)# pvc 16</p>	<p>POS 相手先固定接続 (PVC) を作成し、フレームリレー PVC コンフィギュレーション サブモードを開始します。</p> <p><code>dlci</code> を 16 から 1007 の範囲の PVC ID に置き換えます。</p> <p>(注) 各サブインターフェイスに設定できる PVC は 1 つだけです。</p>
<p>ステップ 21 <code>service-policy {input output} policy-map</code></p> <p>例 : RP/0/RSP0/CPU0:router(config-fr-vc)# service-policy output policy-mapA</p>	<p>ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。</p> <p>(注) ポリシー マップの作成と設定については、『Cisco IOS XR Modular Quality of Service Configuration Guide』を参照してください。</p>
<p>ステップ 22 <code>exit</code></p> <p>例 : RP/0/RSP0/CPU0:router(config-fr-vc)# exit</p>	<p>フレームリレー仮想回線モードを終了します。</p>

	コマンドまたはアクション	目的
ステップ 23	<code>exit</code> 例： RP/0/RSP0/CPU0:router(config-subif)# exit	サブインターフェイス コンフィギュレーション モードを終了します。
ステップ 24	<code>interface serial interface-path-id</code> 例： RP/0/RSP0/CPU0:router(config)# interface serial 0/1/0/0/0/0:0	完全なインターフェイス番号を <code>rack/slot/module/port/T3Num/T1num:instance</code> 表記で指定します。
ステップ 25	<code>encapsulation mfr</code> 例： RP/0/RSP0/CPU0:router(config)# encapsulation mfr	シリアル インターフェイスでマルチリンク フレーム リレーをイネーブルにします。
ステップ 26	<code>multilink group group-id</code> 例： RP/0/RSP0/CPU0:router(config-if)# multilink group 100	このインターフェイスのマルチリンク グループ ID を指定します。
ステップ 27	<code>frame-relay multilink lid link-id name</code> 例： RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink lid sj1	(注) フレームリレー マルチリンク バンドル リンクの名前を設定します。
ステップ 28	<code>frame-relay multilink ack ack-timeout</code> 例： RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink ack 5	フレーム リレー マルチリンク バンドル リンクの確認応答タイムアウト値を設定します。
ステップ 29	<code>frame-relay multilink hello hello-interval</code> 例： RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink hello 60	フレーム リレー マルチリンク バンドル リンクの hello 間隔を設定します。
ステップ 30	<code>frame-relay multilink retry retry-count</code> 例： RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink retry 2	フレーム リレー マルチリンク バンドル リンクの再送信の再試行回数を設定します。
ステップ 31	<code>exit</code> 例： RP/0/RSP0/CPU0:router(config-if)# exit	インターフェイス コンフィギュレーション モードを終了します。

■ フレーム リレーの設定

	コマンドまたはアクション	目的
ステップ 32	<pre>end または commit 例 : RP/0/RSP0/CPU0:router(config-if)# end または RP/0/RSP0/CPU0:router(config-if)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> • end コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: - yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、ルータが EXEC モードに戻ります。 - no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 - cancel と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、commit コマンドを使用します。
ステップ 33	<pre>exit 例 : RP/0/RSP0/CPU0:router(config)# exit</pre>	<p>グローバル コンフィギュレーション モードを終了します。</p>
ステップ 34	<pre>show frame-relay multilink interface type interface-path-id [detail verbose] 例 : RP/0/RSP0/CPU0:router# show frame-relay multilink interface Multilink 0/5/1/0/1</pre>	<p>バンドル固有の情報やフレームリレー情報など、インターフェイス記述ブロック (IDB) から取得した情報を表示します。</p>

チャネライズド フレームリレー シリアル インターフェイスでの FRF.12 エンドツーエンド フラグメンテーションの設定

チャネライズド フレームリレー シリアル インターフェイスで FRF.12 エンドツーエンド フラグメンテーションを設定するには、次の手順で操作します。

手順の概要

1. **config**
2. **controller t3 interface-path-id**
3. **mode type**
4. **clock source {internal | line}**

5. **exit**
6. **controller t1** *interface-path-id*
7. **channel-group** *channel-group-number*
8. **timeslots** *range*
9. **exit**
10. **exit**
11. **interface serial** *interface-path-id*
12. **encapsulation** **frame-relay**
13. **exit**
14. **interface serial** *interface-path-id*
15. **ipv4 address** *ip-address*
16. **pvc** *dldi*
17. **service-policy** {**input** | **output**} *policy-map*
18. **fragment end-to-end** *fragment-size*
19. **exit**
20. **exit**
21. **exit**
22. **end**
または
commit
23. **exit**
24. **show frame-relay pvc** [**dldi** | **interface** | **location**]

手順の詳細

	コマンドまたはアクション	目的
ステップ1	config 例： RP/0/RSP0/CPU0:router# config	グローバル コンフィギュレーション モードを開始します。
ステップ2	controller t3 <i>interface-path-id</i> 例： RP/0/RSP0/CPU0:router(config)# controller t3 0/1/0/0	<i>rack/slot/module/port</i> 表記で T3 コントローラ名を指定し、T3 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	mode type 例： RP/0/RSP0/CPU0:router(config-t3)# mode t1	チャネライズするマルチリンクのタイプを設定します (たとえば、28 T1)。

■ フレーム リレーの設定

コマンドまたはアクション	目的
<p>ステップ4 <code>clock source {internal line}</code></p> <p>例: RP/0/RSP0/CPU0:router(config-t3)# clock source internal</p>	<p>(任意) 個々の E3 リンクのクロッキングを設定します。</p> <p>(注) デフォルトのクロック ソースは internal です。</p> <p>(注) シリアル リンクでクロッキングを設定する場合、一方のエンドを internal にし、もう一方を line にする必要があります。接続の両エンドに internal クロッキングを設定すると、フレーム同期のずれが生じます。接続の両エンドに line クロッキングを設定すると、ラインはアップ状態になりません。</p>
<p>ステップ5 <code>exit</code></p> <p>例: RP/0/RSP0/CPU0:router(config-t3)# exit</p>	<p>T3/E3 または T1/E1 コントローラ コンフィギュレーション モードを終了します。</p>
<p>ステップ6 <code>controller t1 interface-path-id</code></p> <p>例: RP/0/RSP0/CPU0:router(config)# controller t1 0/1/0/0/0</p>	<p>T1 コンフィギュレーション モードを開始します。</p>
<p>ステップ7 <code>channel-group channel-group-number</code></p> <p>例: RP/0/RSP0/CPU0:router(config-t1)# channel-group 0</p>	<p>T1 チャンネル グループを作成し、そのチャンネル グループのチャンネルグループ コンフィギュレーション モードを開始します。</p>
<p>ステップ8 <code>timeslots range</code></p> <p>例: RP/0/RSP0/CPU0:router(config-t1-channel_group)# timeslots 1-24</p>	<p>1 つまたは複数の DS0 タイムスロットをチャンネル グループに関連付け、関連付けたシリアル サブインターフェイスをそのチャンネル グループに作成します。</p> <ul style="list-style-type: none"> • 範囲は 1 ~ 24 タイムスロットです。 • 24 タイムスロットすべてを単一のチャンネル グループに割り当てることも、タイムスロットを複数のチャンネル グループに分割することもできます。 <p>(注) 個々の T1 コントローラは、24 DS0 タイムスロットの合計をサポートします。</p>
<p>ステップ9 <code>exit</code></p> <p>例: RP/0/RSP0/CPU0:router(config-t1-channel_group)# exit</p>	<p>チャンネル グループ コンフィギュレーション モードを終了します。</p>
<p>ステップ10 <code>exit</code></p> <p>例: RP/0/RSP0/CPU0:router(config-t1)# exit</p>	<p>T1 コンフィギュレーション モードを終了します。</p>

	コマンドまたはアクション	目的
ステップ 11	<pre>interface serial interface-path-id</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config)# interface serial 0/1/0/0/0:0</pre>	<p>完全なインターフェイス番号を <i>rack/slot/module/port/T3Num/T1num:instance</i> 表記で指定します。</p>
ステップ 12	<pre>encapsulation frame-relay</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:Router(config-if)# encapsulation frame-relay</pre>	<p>フレームリレーのカプセル化タイプを指定します。</p>
ステップ 13	<pre>exit</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-if)# exit</pre>	<p>インターフェイス コンフィギュレーション モードを終了します。</p>
ステップ 14	<pre>interface serial interface-path-id</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config)# interface serial 1/0/0/0/0:0.1</pre>	<p><i>rack/slot/module/port[/channel-num:channel-group-number].subinterface</i> 表記で、完全なサブインターフェイス番号を指定します。</p>
ステップ 15	<pre>ipv4 address ip-address</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 3.1.100.16 255.255.255.0</pre>	<p>次の形式でインターフェイスに IP アドレスおよびサブネット マスクを割り当てます。</p> <p><i>A.B.C.D/prefix</i> または <i>A.B.C.D/mask</i></p>
ステップ 16	<pre>pvc dlci</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router (config-subif)# pvc 100</pre>	<p>POS 相手先固定接続 (PVC) を作成し、フレームリレー PVC コンフィギュレーション サブモードを開始します。</p> <p><i>dlci</i> を 16 から 1007 の範囲の PVC ID に置き換えます。</p> <p>(注) 各サブインターフェイスに設定できる PVC は 1 つだけです。</p>
ステップ 17	<pre>service-policy {input output} policy-map</pre> <p>例 :</p> <pre>RP/0/RSP0/CPU0:router(config-fr-vc)# service-policy output policy-mapA</pre>	<p>ポリシー マップを入力サブインターフェイスまたは出力サブインターフェイスに付加します。付加すると、そのサブインターフェイスのサービス ポリシーとしてポリシー マップが使用されます。</p> <p>(注) 効率的な FRF.12 機能 (具体的にはインターリーブ) のためには、出力サービス ポリシーに優先順位を設定します。</p> <p>(注) ポリシー マップの作成と設定については、『Cisco IOS XR Modular Quality of Service Configuration Guide』を参照してください。</p>

■ フレーム リレーの設定

コマンドまたはアクション	目的
ステップ 18 <code>fragment end-to-end fragment-size</code> 例 : RP/0/RSP0/CPU0:router(config-fr-vc)# fragment end-to-end 100	(任意) インターフェイスでのフレーム リレー フレームのフラグメンテーションをイネーブルにし、元のフレームから各フラグメントに入れるペイロードのサイズ (バイト単位) を指定します。この数値には、元のフレームのフレームリレー ヘッダーは含まれません。 使用するハードウェアによって、有効値は 64 ~ 512 です。
ステップ 19 <code>exit</code> 例 : RP/0/RSP0/CPU0:router(config-fr-vc)# exit	フレームリレー仮想回線モードを終了します。
ステップ 20 <code>exit</code> 例 : RP/0/RSP0/CPU0:router(config-subif)# exit	サブインターフェイス コンフィギュレーション モードを終了します。
ステップ 21 <code>exit</code> 例 : RP/0/RSP0/CPU0:router(config-if)# exit	インターフェイス コンフィギュレーション モードを終了します。
ステップ 22 <code>end</code> または <code>commit</code> 例 : RP/0/RSP0/CPU0:router(config-if)# end または RP/0/RSP0/CPU0:router(config-if)# commit	設定変更を保存します。 <ul style="list-style-type: none"> • end コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> – yes と入力すると、実行コンフィギュレーション ファイルに設定変更が保存され、コンフィギュレーションセッションが終了し、ルータが EXEC モードに戻ります。 – no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。 – cancel と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。 • 実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーションセッションを継続するには、commit コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 23	<code>exit</code> 例： RP/0/RSP0/CPU0:router(config)# exit	グローバル コンフィギュレーション モードを終了します。
ステップ 24	<code>show frame-relay pvc [dlci interface location]</code> 例： RP/0/RSP0/CPU0:router# show frame-relay pvc 100	指定した PVC DLCI、インターフェイス、または場所の情報を表示します。

フレーム リレーの設定例

ここでは、次の設定例について説明します。

- 「オプションのフレームリレー パラメータ : 例」 (P.617)
- 「マルチリンク フレームリレー : 例」 (P.620)
- 「エンドツーエンド フラグメンテーション : 例」 (P.620)

オプションのフレームリレー パラメータ : 例

次の例は、フレームリレーのカプセル化を設定した POS インターフェイスを始動および設定する方法です。この例では、インターフェイスが DCE で ANSI T1.617a-1994 Annex D LMI をサポートするように、デフォルトのフレームリレー設定を変更します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface POS 0/3/0/0
RP/0/RSP0/CPU0:router(config-if)# encapsulation frame-relay IETF
RP/0/RSP0/CPU0:router(config-if)# frame-relay intf-type dce
RP/0/RSP0/CPU0:router(config-if)# frame-relay lmi-type ansi
RP/0/RSP0/CPU0:router(config-if)# no shutdown
RP/0/RSP0/CPU0:router(config-if)# end
```

Uncommitted changes found, commit them? [yes]: **yes**

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router (config)# interface pos 0/3/0/0.10 point-to-point
RP/0/RSP0/CPU0:router (config-subif)#ipv4 address 10.46.8.6/24
RP/0/RSP0/CPU0:router (config-subif)# pvc 20
RP/0/RSP0/CPU0:router (config-fr-vc)# encaps ietf
RP/0/RSP0/CPU0:router (config-subif)# commit
```

次の例では、フレーム リレー カプセル化を設定した POS インターフェイスで LMI をディisableにする方法を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface
RP/0/RSP0/CPU0:router(config)# interface pos 0/3/0/0
RP/0/RSP0/CPU0:router(config-if)# frame-relay lmi disable
RP/0/RSP0/CPU0:router(config-if)# end
```

Uncommitted changes found, commit them? [yes]: **yes**

■ フレーム リレーの設定例

次の例は、シリアルインターフェイスで LMI を再度イネーブルにする方法です。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface
RP/0/RSP0/CPU0:router(config)# interface serial 0/3/0/0
RP/0/RSP0/CPU0:router(config-if)# no frame-relay lmi disable
RP/0/RSP0/CPU0:router(config-if)# end
```

Uncommitted changes found, commit them? [yes]: **yes**

次の例では、すべてのインターフェイスの LMI のフレーム リレー統計情報を表示する方法を示します。

```
RP/0/RSP0/CPU0:router# show frame-relay lmi
```

```
LMI Statistics for interface POS0/1/0/0/ (Frame Relay DCE) LMI TYPE = ANSI
Invalid Unnumbered Info 0          Invalid Prot Disc 0
Invalid Dummy Call Ref 0          Invalid Msg Type 0
Invalid Status Message 0          Invalid Lock Shift 9
Invalid Information ID 0           Invalid Report IE Len 0
Invalid Report Request 0          Invalid Keep IE Len 0
Num Status Enq. Rcvd 9444         Num Status Msgs Sent 9444
Num Full Status Sent 1578         Num St Enq. Timeouts 41
Num Link Timeouts 7
```

```
LMI Statistics for interface POS0/1/0/1/ (Frame Relay DCE) LMI TYPE = CISCO
Invalid Unnumbered Info 0          Invalid Prot Disc 0
Invalid Dummy Call Ref 0          Invalid Msg Type 0
Invalid Status Message 0          Invalid Lock Shift 0
Invalid Information ID 0           Invalid Report IE Len 0
Invalid Report Request 0          Invalid Keep IE Len 0
Num Status Enq. Rcvd 9481         Num Status Msgs Sent 9481
Num Full Status Sent 1588         Num St Enq. Timeouts 16
Num Link Timeouts 4
```

次の例は、メインシリアルインターフェイス上の PVC でシリアルサブインターフェイスを作成する方法です。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface serial 0/3/0/0/0:0.10 point-to-point
RP/0/RSP0/CPU0:router(config-subif)#ipv4 address 10.46.8.6/24
RP/0/RSP0/CPU0:router(config-subif)# pvc 20
RP/0/RSP0/CPU0:router(config-fr-vc)# encapsulation ietf
RP/0/RSP0/CPU0:router(config-subif)# commit
```

次の例は、システムに設定されているすべての PVC に関する情報を表示する方法です。

```
RP/0/RSP0/CPU0router# show frame-relay pvc
```

```
PVC Statistics for interface Serial0/3/2/0 (Frame Relay DCE)
```

	Active	Inactive	Deleted	Static
Local	4	0	0	0
Switched	0	0	0	0
Dynamic	0	0	0	0

```
DLCI = 612, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTI
VE, INTERFACE = Serial0/3/2/0.1
```

```
input pkts 0          output pkts 0          in bytes 0
out bytes 0           dropped pkts 0         in FECN packets 0
in BECN pkts 0       out FECN pkts 0       out BECN pkts 0
in DE pkts 0         out DE pkts 0
out bcast pkts 0     out bcast bytes 0
pvc create time 00:00:00      last time pvc status changed 00:00:00
```

```

DLCI = 613, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTI
VE, INTERFACE = Serial0/3/2/0.2
  input pkts 0          output pkts 0          in bytes 0
  out bytes 0          dropped pkts 0        in FECN packets 0
  in BECN pkts 0      out FECN pkts 0      out BECN pkts 0
  in DE pkts 0        out DE pkts 0
  out bcast pkts 0    out bcast bytes 0
  pvc create time 00:00:00    last time pvc status changed 00:00:00

DLCI = 614, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTI
VE, INTERFACE = Serial0/3/2/0.3
  input pkts 0          output pkts 0          in bytes 0
  out bytes 0          dropped pkts 0        in FECN packets 0
  in BECN pkts 0      out FECN pkts 0      out BECN pkts 0
  in DE pkts 0        out DE pkts 0
  out bcast pkts 0    out bcast bytes 0
  pvc create time 00:00:00    last time pvc status changed 00:00:00

DLCI = 615, DLCI USAGE = LOCAL, ENCAP = CISCO, INHERIT = TRUE, PVC STATUS = ACTI
VE, INTERFACE = Serial0/3/2/0.4
  input pkts 0          output pkts 0          in bytes 0
  out bytes 0          dropped pkts 0        in FECN packets 0
  in BECN pkts 0      out FECN pkts 0      out BECN pkts 0
  in DE pkts 0        out DE pkts 0
  out bcast pkts 0    out bcast bytes 0
  pvc create time 00:00:00    last time pvc status changed 00:00:00

```

次の例では、DTE 用に設定されている PVC の LMI ポーリング オプションを変更してから、**show frame-relay lmi** コマンドと **show frame-relay lmi-info** コマンドを使用してインターフェイスのモニタリングとトラブルシューティングの情報を表示する方法を示します。

```

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface pos 0/3/0/0
RP/0/RSP0/CPU0:router(config-if)# frame-relay lmi-n391dte 10
RP/0/RSP0/CPU0:router(config-if)# frame-relay lmi-n391dte 5
RP/0/RSP0/CPU0:router(config-if)# frame-relay lmi-t391dte 15
RP/0/RSP0/CPU0:router(config-subif)# commit

RP/0/RSP0/CPU0:router# show frame-relay lmi interface pos 0/3/0/0

LMI Statistics for interface pos 0/3/0/0 (Frame Relay DTE) LMI TYPE = ANSI
Invalid Unnumbered Info 0          Invalid Prot Disc 0
Invalid Dummy Call Ref 0          Invalid Msg Type 0
Invalid Status Message 0          Invalid Lock Shift 9
Invalid Information ID 0          Invalid Report IE Len 0
Invalid Report Request 0          Invalid Keep IE Len 0
Num Status Enq. Rcvd 9444          Num Status Msgs Sent 9444
Num Full Status Sent 1578          Num St Enq. Timeouts 41
Num Link Timeouts 7

RP/0/RSP0/CPU0:router# show frame-relay lmi-info interface pos 0/3/0/0

LMI IDB Info for interface POS0/3/0/0
  ifhandle:          0x6176840
  Interface type:    DTE
  Interface state:   UP
  Line Protocol:     UP
  LMI type (cnf/oper):  AUTO/CISCO
  LMI type autosense:  OFF
  Interface MTU:     1504
  ----- DTE -----
  T391:              15s
  N391: (cnf/oper):  5/5

```

■ フレーム リレーの設定例

```

N392: (cnf/oper):      3/0
N393:                  4
My seq#:               83
My seq# seen:          83
Your seq# seen:        82
----- DCE -----
T392:                  15s
N392: (cnf/oper):      3/0
N393:                  4
My seq#:               0
My seq# seen:          0
Your seq# seen:        0

```

マルチリンク フレームリレー : 例

次の例は、シリアル インターフェイスでマルチリンク フレームリレーを設定する方法です。

```

RP/0/RSP0/CPU0:router# config
RP/0/RSP0/CPU0:router(config)# controller MgmtMultilink 0/3/1/0
RP/0/RSP0/CPU0:router(config-mgmtmultilink)# bundle 100
RP/0/RSP0/CPU0:router(config-mgmtmultilink)# exit

RP/0/RSP0/CPU0:router(config)# controller T3 0/3/1/0
RP/0/RSP0/CPU0:router(config-t3)# mode t1
RP/0/RSP0/CPU0:router(config-t3)# clock source internal
RP/0/RSP0/CPU0:router(config-t3)# exit

RP/0/RSP0/CPU0:router(config)# controller T1 0/3/1/0/0
RP/0/RSP0/CPU0:router(config-t1)# channel-group 0
RP/0/RSP0/CPU0:router(config-t1-channel_group)# timeslots 1-24
RP/0/RSP0/CPU0:router(config-t1-channel_group)# exit
RP/0/RSP0/CPU0:router(config-t1-channel_group)# exit
RP/0/RSP0/CPU0:router(config-t1)# exit

RP/0/RSP0/CPU0:router(config)# interface Multilink 0/3/1/0/100
RP/0/RSP0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/RSP0/CPU0:router(config-if)# exit

RP/0/RSP0/CPU0:router(config)# interface Multilink 0/3/1/0/100.16 point-to-point
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 3.1.100.16 255.255.255.0
RP/0/RSP0/CPU0:router(config-subif)# pvc 16
RP/0/RSP0/CPU0:router(config-fr-vc)# service-policy output policy-mapA
RP/0/RSP0/CPU0:router(config-fr-vc)# exit
RP/0/RSP0/CPU0:router(config-subif)# exit

RP/0/RSP0/CPU0:router(config)# interface Serial 0/3/1/0/0:0
RP/0/RSP0/CPU0:router(config-if)# encapsulation mfr
RP/0/RSP0/CPU0:router(config-if)# multilink group 100
RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink lid sj1
RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink ack 5
RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink hello 60
RP/0/RSP0/CPU0:router(config-if)# frame-relay multilink retry 2
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config)#

```

エンドツーエンド フラグメンテーション : 例

次の例は、チャネライズド フレームリレー シリアル インターフェイスで FRF.12 エンドツーエンド フラグメンテーションを設定する方法です。

```
RP/0/RSP0/CPU0:router# config
RP/0/RSP0/CPU0:router(config)# controller T30/3/1/0
RP/0/RSP0/CPU0:router(config-t3)# mode t1
RP/0/RSP0/CPU0:router(config-t3)# clock source internal
RP/0/RSP0/CPU0:router(config-t3)# exit
RP/0/RSP0/CPU0:router(config-t3)# controller T10/3/1/0/0
RP/0/RSP0/CPU0:router(config-t1)# channel-group 0
RP/0/RSP0/CPU0:router(config-t1-channel_group)# timeslots 1-24
RP/0/RSP0/CPU0:router(config-t1-channel_group)# exit
RP/0/RSP0/CPU0:router(config-t1-channel_group)# interface Serial 0/3/1/0/0:0
RP/0/RSP0/CPU0:router(config-if)# encapsulation frame-relay
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config-if)# interface Serial 0/3/1/0/0:0.100 point-to-point
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 3.1.1.1 255.255.255.0
RP/0/RSP0/CPU0:router(config-subif)# pvc 100
RP/0/RSP0/CPU0:router(config-fr-vc)# service-policy output LFI
RP/0/RSP0/CPU0:router(config-fr-vc)# fragment end-to-end 256
```

その他の関連資料

ここでは、フレーム リレーに関する関連資料について説明します。

関連資料

関連項目	参照先
Cisco IOS XR マスター コマンド リファレンス	『Cisco IOS XR Master Commands List』
Cisco IOS XR インターフェイス コンフィギュレーション コマンド	『Cisco IOS XR Interface and Hardware Component Command Reference』
Cisco IOS XR ソフトウェアを使用した初期システム ブートアップとルータの設定情報	『Cisco IOS XR Getting Started Guide』
Cisco IOS XR AAA サービス構成情報	『Cisco IOS XR System Security Configuration Guide』 および 『Cisco IOS XR System Security Command Reference』

標準

標準	タイトル
FRF.12	フレームリレー フォーラム .12
FRF.16	フレームリレー フォーラム .16
ANSI T1.617 Annex D	American National Standards Institute T1.617 Annex D
ITU Q.933 Annex A	International Telecommunication Union Q.933 Annex A

MIB

MIB	MIB のリンク
FRF.16 MIB Cisco Frame Relay MIB IF-MIB 『Management Information Base for Frame Relay DTEs』 Management Information Base for Frame Relay DTEs Using SMIv2	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

RFC

RFC	タイトル
RFC 1294	『Multiprotocol Interconnect Over Frame Relay』
RFC 1315	『Management Information Base for Frame Relay DTEs』
RFC 1490	『Multiprotocol Interconnect Over Frame Relay』
RFC 1586	『Guidelines for Running OSPF Over Frame Relay Networks』
RFC 1604	『Definitions of Managed Objects for Frame Relay Service』
RFC 2115	『Management Information Base for Frame Relay DTEs Using SMIv2』
RFC 2390	『Inverse Address Resolution Protocol』

RFC	タイトル
RFC 2427	『Multiprotocol Interconnect Over Frame Relay』
RFC 2954	『Definitions of Managed Objects for Frame Relay Service』
RFC 3020	『RFC for FRF.16 MIB』

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html

