

Oracle® Solaris 11.3 でのネットワークデータリンクの管理

ORACLE®

Part No: E62571
2016 年 11 月

Part No: E62571

Copyright © 2011, 2016, Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクルまでご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

このソフトウェアまたはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアまたはハードウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用する場合、安全に使用するために、適切な安全装置、バックアップ、冗長性(redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、Oracle Corporationおよびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはオラクル およびその関連会社の登録商標です。その他の社名、商品名等は各社の商標または登録商標である場合があります。

Intel, Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD, Opteron, AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。適用されるお客様とOracle Corporationとの間の契約に別段の定めがある場合を除いて、Oracle Corporationおよびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。適用されるお客様とOracle Corporationとの間の契約に定めがある場合を除いて、Oracle Corporationおよびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

ドキュメントのアクセシビリティについて

オラクルのアクセシビリティについての詳細情報は、Oracle Accessibility ProgramのWeb サイト(<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>)を参照してください。

Oracle Supportへのアクセス

サポートをご契約のお客様には、My Oracle Supportを通して電子支援サービスを提供しています。詳細情報は(<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>)か、聴覚に障害のあるお客様は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>)を参照してください。

目次

このドキュメントの使用法	11
1 ネットワークデータリンク管理の概要	13
Oracle Solaris 11.3 のネットワークデータリンク管理の新機能	13
ネットワークデータリンクの管理に使用される機能およびコンポーネン ト	14
リンクアグリゲーション	14
仮想ローカルエリアネットワーク	15
プライベート仮想ローカルエリアネットワーク	15
ブリッジネットワーク	15
リンク層検出プロトコル	16
データセンターブリッジング	16
2 リンクアグリゲーションを使用した高可用性の構成	17
リンクアグリゲーションの概要	17
リンクアグリゲーションのメリット	18
トランクアグリゲーション	19
スイッチの使用	20
バックツーバックトランクアグリゲーション構成	22
Link Aggregation Control Protocol を指定したスイッチの使用	23
負荷分散のアグリゲーションポリシーの定義	23
データリンクマルチパスアグリゲーション	24
DLMP アグリゲーションのメリット	24
DLMP アグリゲーションの動作	25
DLMP アグリゲーションの障害検出	27
リンクアグリゲーションの要件	29
リンク集約の作成	30
▼ リンクアグリゲーションを作成する方法	30
アグリゲーションへのリンクの追加	33
▼ アグリゲーションにリンクを追加する方法	34

アグリゲーションからのリンクの削除	35
トランクアグリゲーションの変更	35
DLMP アグリゲーションのプローブベースの障害検出の構成	36
▼ DLMP のプローブベースの障害検出の構成方法	37
プローブベースの障害検出のモニタリング	40
リンクアグリゲーションの削除	42
▼ リンクアグリゲーションを削除する方法	43
トランクアグリゲーションと DLMP アグリゲーションの間の切り替え	43
▼ リンクアグリゲーションのタイプを切り替える方法	43
使用事例: リンクアグリゲーションの構成	44
トランクアグリゲーションと DLMP アグリゲーションの比較	47
3 仮想ローカルエリアネットワークを使用した仮想ネットワークの構成	49
VLAN のデプロイの概要	49
どのような場合に VLAN を使用するか	50
VLAN 名の割り当て	50
VLAN トポロジ	51
仮想化での VLAN の使用	54
VLAN 構成の計画	55
VLAN の構成	55
▼ VLAN を構成する方法	56
リンクアグリゲーション上での VLAN の構成	60
▼ リンクアグリゲーション上に VLAN を構成する方法	60
レガシーデバイス上での VLAN の構成	62
▼ レガシーデバイス上で VLAN を構成する方法	62
VLAN 情報の表示	63
VLAN の変更	63
VLAN の VLAN ID の変更	64
別のベースとなるリンクへの VLAN の移行	64
VLAN の削除	66
SR-IOV に対応するポートでの VLAN タグ付けの使用	67
使用事例: リンクアグリゲーションと VLAN 構成を組み合わせる	67
ユースケース: VLAN のフローの構成	70
ユースケース: VLAN VNIC の帯域幅の構成	71
4 プライベート仮想ローカルエリアネットワークの構成	75
プライベート VLAN の概要	75
PVLAN を使用することの利点	77

PVLAN ポート	77
送信トラフィックのタグ付け	78
PVLAN 構成の要件	78
ゾーンを持つ PVLAN	79
プライベート VLAN の構成	80
プライベート VLAN の変更	81
プライベート VLAN の削除	81
ゾーンへの PVLAN の割り当て	82
▼ ゾーンに PVLAN を割り当てる方法	82
5 ブリッジング機能の管理	85
ブリッジネットワークの概要	85
単純なブリッジネットワーク	86
ブリッジネットワークリング	87
ブリッジネットワークの動作	88
ブリッジングプロトコル	89
STP デーモン	90
TRILL デーモン	91
ブリッジの作成	91
ブリッジの保護タイプの変更	93
既存のブリッジへのリンクの追加	94
ブリッジからのリンクの削除	94
ブリッジのリンクプロパティの設定	94
ブリッジ構成情報の表示	95
構成されているブリッジに関する情報の表示	96
ブリッジリンクに関する構成情報の表示	97
システムからのブリッジの削除	98
▼ システムからブリッジを削除する方法	98
ブリッジネットワーク上の VLAN の管理	98
▼ ブリッジの一部であるデータリンク上に VLAN を構成する方法	99
VLAN と STP プロトコルおよび TRILL プロトコル	100
ブリッジのデバッグ	100
6 リンク層検出プロトコルによるネットワーク接続情報の交換	103
LLDP の概要	103
LLDP 実装のコンポーネント	104
LLDP エージェントの情報源	105
LLDP エージェントモード	105

LLDP エージェントが通知する情報	106
必須の TLV ユニット	106
オプションの TLV ユニット	107
TLV ユニットプロパティ	107
システムでの LLDP の有効化	109
▼ LLDP パッケージのインストール方法	110
▼ LLDP をグローバルに有効にする方法	110
▼ 特定のポートの LLDP を有効にする方法	111
エージェントの LLDP パケットの TLV ユニットと値の指定	113
▼ エージェントの LLDP パケットの TLV ユニットの指定方法	114
▼ TLV ユニットの定義方法	115
LLDP の無効化	117
▼ LLDP を無効にする方法	117
LLDP エージェントのモニタリング	118
通知される情報の表示	119
LLDP 統計情報の表示	121
7 データセンターブリッジングを使用した集中ネットワークの管理	123
データセンターブリッジングの概要	123
DCB を使用する場合の考慮事項	125
優先順位ベースのフロー制御	125
拡張伝送選択	126
DCBX の有効化	127
▼ データセンターブリッジングの交換機能を手動で有効にする方法	127
DCB の操作モードの設定	128
▼ DCB の操作モードを設定する方法	129
DCB の優先順位ベースのフロー制御のカスタマイズ	130
PFC 関連データリンクプロパティの設定	130
PFC TLV ユニットの設定	131
PFC 構成情報の表示	132
データリンクプロパティの表示	132
ローカルホストで PFC 情報を同期する機能の表示	133
ホストとピアの間の PFC マッピング情報の表示	133
優先順位定義の表示	134
アプリケーション優先順位の構成	135
DCB の拡張伝送選択のカスタマイズ	135
ETS 関連データリンクプロパティの設定	136

ETS TLV ユニットの設定	137
ピアに対する ETS 構成の推奨	138
ETS 構成情報の表示	140
A リンクアグリゲーションと IPMP: 機能比較	143
B 推移的プローブのパケット形式	145
索引	147

このドキュメントの使用方法

- **概要** – ネットワークデータリンクを管理してネットワークパフォーマンスを向上させるために使用される拡張機能の概要を示します。トランクアグリゲーションまたは DLMP アグリゲーションを使用してリンクを結合してアグリゲーションにする方法、仮想ローカルエリアネットワークを使用してネットワークをサブネットワークに分割する方法、ブリッジを使用して別々のネットワークセグメントを接続する方法、リンク層検出プロトコルを使用してネットワーク接続情報を交換する方法、およびデータセンターブリッジングを使用して集中ネットワークを管理する方法について説明します。
- **対象読者** – システム管理者。
- **前提知識** – ネットワーク管理の基本的なスキルと一部の高度なスキル。

製品ドキュメントライブラリ

この製品および関連製品のドキュメントとリソースは <http://www.oracle.com/pls/topic/lookup?ctx=E62101-01> で入手可能です。

フィードバック

このドキュメントに関するフィードバックを <http://www.oracle.com/goto/docfeedback> からお聞かせください。

ネットワークデータリンク管理の概要

この章では、このドキュメントの残りの章で説明される、ネットワークデータリンクを管理するために使用される拡張機能について紹介します。このドキュメントで説明されているさまざまなテクノロジーは、具体的な状況に応じて使用します。また、ハードウェア構成によっては、特定のタイプの機能を使用することが必要になる場合があります。したがって、このドキュメントの構成手順をすべて完了する必要があるわけではありません。代わりに、ネットワークの要件に対応するテクノロジーを選択して配備してください。

このドキュメントで説明する構成を実行する前に、基本的なネットワーク構成を完了して基本的なデータリンク構成について理解する必要があります。基本的なネットワーク構成については、『[Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理](#)』を参照してください。データリンク構成の管理の要素については、『[Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理](#)』の第 2 章、「[Oracle Solaris でのデータリンク構成の管理](#)」を参照してください。

Oracle Solaris のネットワーク構成機能のサマリーについては、『[Oracle Solaris 11.3 でのネットワーク管理の計画](#)』の第 1 章、「[Oracle Solaris ネットワーク管理のサマリー](#)」を参照してください。

この章の内容は、次のとおりです。

- 13 ページの「[Oracle Solaris 11.3 のネットワークデータリンク管理の新機能](#)」
- 14 ページの「[ネットワークデータリンクの管理に使用される機能およびコンポーネント](#)」

Oracle Solaris 11.3 のネットワークデータリンク管理の新機能

既存の顧客のために、このセクションでは、このリリースの次の主要な変更を強調表示します。

- **プライベート仮想ローカルエリアネットワーク** – Oracle Solaris 11.3 には、RFC 5517 標準の定義に従ってプライベート仮想ローカルエリアネットワーク (PVLAN)

のサポートが含まれています。PVLAN を使用して、通常の VLAN (プライマリ) をサブ VLAN (セカンダリ) に分割できます。詳細については、75 ページの「プライベート VLAN の概要」を参照してください。

- **Converged Enhanced Ethernet (CEE) データセンターブリッジング交換 (DCBX)** – Oracle Solaris 11.3 には、IEEE だけでなく Converged Enhanced Ethernet (CEE) データセンターブリッジング交換 (DCBX) のサポートが含まれているため、データセンターブリッジング (DCB) を使用するときには多様なスイッチを備えた環境で Oracle Solaris Fibre Channel over Ethernet (FCoE) をデプロイできます。Oracle Solaris CEE DCBX は、優先順位に基づくフロー制御 (PFC) およびアプリケーション Type-Length-Value (TLV) をサポートします。詳細については、128 ページの「DCB の操作モードの設定」を参照してください。

ネットワークデータリンクの管理に使用される機能およびコンポーネント

ネットワークデータリンクの管理とは、機能とテクノロジーを使用して、システムでネットワークトラフィックを処理する方法を微調整することです。これらのテクノロジーを使用して構成されたシステムはネットワークトラフィックをより適切に管理でき、それがネットワークの総合パフォーマンスの向上に寄与します。これらの機能はネットワーク操作のさまざまな領域に対処しますが、ネットワーク接続、ネットワーク管理、効率などの一般的なメリットを提供します。

リンクアグリゲーション、仮想ローカルエリアネットワーク、ブリッジネットワークなどの機能を使用してネットワークデータリンクを管理するには、`dladm` コマンドを使用します。データリンクの管理に加えて、取得リンクを管理するために `dladm` コマンドを使用できます。ファイアウォールは取得リンクを使用し、`log` アクションでルールに一致するパケットをログに記録します。取得リンクで傍受されるパケットは、任意の `libpcap` ツール (`tcpdump`、`wireshark` など) で読み取ることができます。詳細については、『Oracle Solaris 11.3 でのネットワークのセキュリティー保護』を参照してください。

リンクアグリゲーション

リンクアグリゲーションを使用すると、複数のデータリンクのリソースをプールして、単一のユニットとして管理できます。複数の物理 NIC を一緒に組み合わせることによって、帯域幅が向上し、アプリケーションの高可用性を提供できます。ネットワークデータリンクのリンクアグリゲーションは、システムがネットワークに継続的にアクセスできるようにします。トランクアグリゲーションと DLMP アグリゲーションは 2 種類のリンクアグリゲーションです。

トランクアグリゲーションはアグリゲーション上に構成されたクライアント用にベースとなるデータリンクの統合された帯域幅を提供します。DLMP アグリゲーションはアグリゲーション上に構成されたクライアント用に複数のスイッチにわたって高い可用性を提供します。DLMP アグリゲーションは、トラフィックを送受信するネットワークの継続的な可用性を確保するためにリンクベースの障害検出とプローブベースの障害検出もサポートします。異なるタイプのリンクアグリゲーションとリンクアグリゲーションを構成および管理する手順の詳細は、[第2章「リンクアグリゲーションを使用した高可用性の構成」](#)を参照してください。

仮想ローカルエリアネットワーク

仮想ローカルエリアネットワーク (VLAN) によって、物理的なネットワーク環境にリソースを追加することなく、ネットワークをサブネットワークに分割できます。したがって、サブネットワークは仮想であり、同じ物理ネットワークリソースを使用します。VLAN は同じ VLAN 内のアプリケーションのみが相互に通信できるように切り離されたサブネットワークをアプリケーションに提供します。VLAN と Oracle Solaris ゾーンを組み合わせることにより、スイッチなどの1つのネットワーク単位内に複数の仮想ネットワークを構成できます。VLAN の構成および管理の詳細については、[第3章「仮想ローカルエリアネットワークを使用した仮想ネットワークの構成」](#)を参照してください。

プライベート仮想ローカルエリアネットワーク

RFC 5517 で定義されたプライベート仮想ローカルエリアネットワーク (PVLAN) 標準では、通常の VLAN (プライマリ) をサブ VLAN (セカンダリ) に分割できます。通常の VLAN は、単一のブロードキャストドメインです。PVLAN は、1つのブロードキャストドメインを複数のサブドメインにパーティション化します。PVLAN の構成および管理の詳細については、[第4章「プライベート仮想ローカルエリアネットワークの構成」](#)を参照してください。

ブリッジネットワーク

ブリッジは別々のネットワークセグメントを接続し、2つのノードの間のパスとなります。ブリッジで接続すると、接続されたネットワークセグメントは単一のネットワークセグメントであるかのように通信を行います。ブリッジはパケット転送メカニズムを使用してサブネットワークを相互に接続し、最短の接続ルートを使用してシステムがパケットを宛先に転送できるようにします。ブリッジネットワークとブリッジを管理する手順の詳細は、[第5章「ブリッジング機能の管理」](#)を参照してください。

リンク層検出プロトコル

リンク層検出プロトコル (LLDP) は、トポロジの検出の目的でネットワーク上のシステム間の接続や管理の情報を交換できます。この情報には、システムの機能、管理アドレス、およびネットワーク操作に関連するそのほかの情報を含めることができます。ネットワーク診断サービスは、ネットワーク接続が制限されるか機能低下する可能性のある問題を検出するために LLDP を使用します。LLDP と LLDP を構成する手順の詳細は、[第6章「リンク層検出プロトコルによるネットワーク接続情報の交換」](#)を参照してください。

データセンターブリッジング

データセンターブリッジング (DCB) は、ネットワークングとストレージのプロトコルの間でデータリンクを共有する場合など、同じネットワークリンクを共有する際に複数のトラフィックタイプの帯域幅、関連する優先順位、およびフロー制御を管理するために使用されます。DCB によって、LLDP を使用して集中ネットワークをサポートする機能についてピアとの情報交換を行うことが可能になります。情報は、データセンターなどトラフィックの重い環境における、ネットワークパケットの整合性に影響する構成に関連します。DCB は、ストレージエリアネットワーク (SAN) やローカルエリアネットワーク (LAN) を統合して大規模な設備の運用および管理コストを削減することにより、効率的なネットワークインフラストラクチャーを可能にします。

パケットロスを防止する優先順位ベースのフロー制御 (PFC) や、サービスクラス (CoS) 優先順位に基づいてパケット間で帯域幅を共有する拡張伝送選択 (ETS) など、DCB 機能を構成できます。詳細は、[第7章「データセンターブリッジングを使用した集中ネットワークの管理」](#)を参照してください。

◆◆◆ 第 2 章

リンクアグリゲーションを使用した高可用性の構成

この章では、リンクアグリゲーションの概要と、リンクアグリゲーションを構成および管理するための手順について説明します。

この章の内容は、次のとおりです。

- 17 ページの「リンクアグリゲーションの概要」
- 19 ページの「トランクアグリゲーション」
- 24 ページの「データリンクマルチパスアグリゲーション」
- 29 ページの「リンクアグリゲーションの要件」
- 30 ページの「リンク集約の作成」
- 33 ページの「アグリゲーションへのリンクの追加」
- 35 ページの「アグリゲーションからのリンクの削除」
- 35 ページの「トランクアグリゲーションの変更」
- 36 ページの「DLMP アグリゲーションのプロープベースの障害検出の構成」
- 40 ページの「プロープベースの障害検出のモニタリング」
- 42 ページの「リンクアグリゲーションの削除」
- 43 ページの「トランクアグリゲーションと DLMP アグリゲーションの間の切り替え」
- 44 ページの「使用事例: リンクアグリゲーションの構成」
- 47 ページの「トランクアグリゲーションと DLMP アグリゲーションの比較」

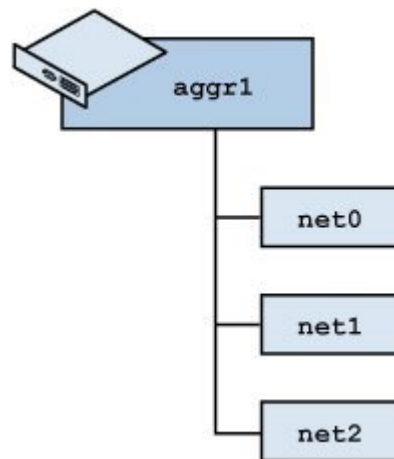
リンクアグリゲーションの概要

リンクアグリゲーションはシステム上の複数の物理データリンクを組み合わせてアプリケーションの帯域幅を改善する方法です。これらの物理データリンクを単一の論理ユニットとして一緒に構成することにより、ネットワークトラフィックのスループットが向上し、データリンク層で高可用性を実現できます。

Oracle Enterprise Manager Ops Center を使用して、個々のインターフェースとして使用可能な NIC にリストされた任意のリンクアグリゲーションを含めることができます。また、Oracle Enterprise Manager Ops Center を使用して、トランクアグリゲーションおよび DLMP アグリゲーションの詳細を表示することもできます。Oracle Enterprise Manager Ops Center の詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=oc122&id=OPCCM> を参照してください。

次の図は、システム上に構成された単純なリンクアグリゲーションの例を示しています。

図 1 リンクアグリゲーション構成



この図は、ベースとなる 3 つのデータリンク (net0、net1、および net2) で構成されたアグリゲーション aggr1 を示します。これらのデータリンクは、このアグリゲーションを介してシステムを通過するトラフィック専用で利用されます。ベースとなるリンクは外部アプリケーションには隠されています。代わりに、データリンク aggr1 がアクセス可能になります。

リンクアグリゲーションのメリット

リンクアグリゲーションには次のメリットがあります。

- **帯域幅の増加** – 複数のリンクの伝送容量が 1 つの論理的なリンクに統合されます。
- **自動フェイルオーバーおよびフェイルバック** – 失敗したリンクのトラフィックは、アグリゲーション内のほかの動作しているリンクに自動的に切り替えられるため、高可用性が実現されます。

- **管理の向上** – ベースとなるすべてのリンクが単一のユニットとして管理されます。
- **ネットワークアドレスプールのアドレスの節約** – 集約全体に 1 つの IP アドレスを割り当てることができます。

リンクアグリゲーションが複数のリンクを単一の論理データリンクにグループ化するため、リンク保護やリソース管理などのデータリンクの機能はリンクアグリゲーションと適切に連携します。リンク保護については、『[Oracle Solaris 11.3 でのネットワークのセキュリティ保護](#)』の第 1 章、「[仮想化環境でのリンク保護の使用](#)」を参照してください。リソース管理については、『[Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理](#)』の第 7 章、「[ネットワークリソースの管理](#)」を参照してください。

リンクアグリゲーションは、ネットワークの仮想化を備えた高可用性機能 (IPMP など) を使用する際に発生する問題の一部を克服します。リンクアグリゲーションを使用する場合、まず大域ゾーン内でアグリゲーションを作成してから、非大域ゾーンを構成する際にベースとなるリンクとして指定します。ゾーンがブートされると、リンクアグリゲーションの仮想ネットワークインタフェースカード (VNIC) に割り当てられます。各非大域ゾーンに構成する必要がある IPMP とは異なり、リンクアグリゲーションは大域ゾーンにのみ構成されており、高可用性 VNIC を非大域ゾーンに提供します。また、大域ゾーンは、ゾーンに割り当てられた VNIC の帯域幅などのプロパティも構成できます。

注記 - リンクアグリゲーションは IP マルチパス (IPMP) と同様の機能を実行して、データリンク層でネットワークのパフォーマンスと可用性を向上させます。これら 2 つのテクノロジーの比較については、[付録A リンクアグリゲーションと IPMP: 機能比較](#)を参照してください。

次のタイプのリンクアグリゲーションがサポートされています。

- トランクアグリゲーション
- データリンクマルチパス (DLMP) アグリゲーション

これら 2 種類のリンクアグリゲーションの違いについては、[47 ページの「トランクアグリゲーションと DLMP アグリゲーションの比較」](#)を参照してください。

トランクアグリゲーション

トランクアグリゲーションは IEEE 802.3ad 標準に基づいており、集約されたポートのセット全体に分散するトラフィックの複数のフローを有効にすることにより動作します。IEEE 802.3ad にはスイッチ構成が必要であり、複数のスイッチにわたって動作するためにスイッチベンダー独自の拡張機能が必要です。トランクアグリゲーション

では、各ネットワークポートがアグリゲーション上に構成されている各データリンクに関連付けられているため、アグリゲーション上に構成されたクライアントはベースとなるリンクの統合された帯域幅を取得します。リンクアグリゲーションを作成すると、デフォルトではアグリゲーションがトランクモードで作成されます。次の状況でトランクアグリゲーションを使用できます。

- ネットワーク内のあるシステムが、分散された多くのトラフィックを処理するアプリケーションを実行している場合、トランクアグリゲーションをそのアプリケーションのトラフィック専用で使用すると、より大きい帯域幅を利用できます。
- IP アドレス空間が制限されいながら大容量の帯域幅が必要なサイトの場合、データリンクのトランクアグリゲーションに対して 1 つの IP アドレスのみで済みます。
- すべての内部データリンクを隠す必要があるサイトの場合、トランクアグリゲーションの IP アドレスによって、これらのデータリンクを外部アプリケーションから隠します。
- 信頼性の高いネットワーク接続が必要なアプリケーションの場合、トランクアグリゲーションはネットワーク接続をリンク障害から保護します。

トランクアグリゲーションは次の機能をサポートします。

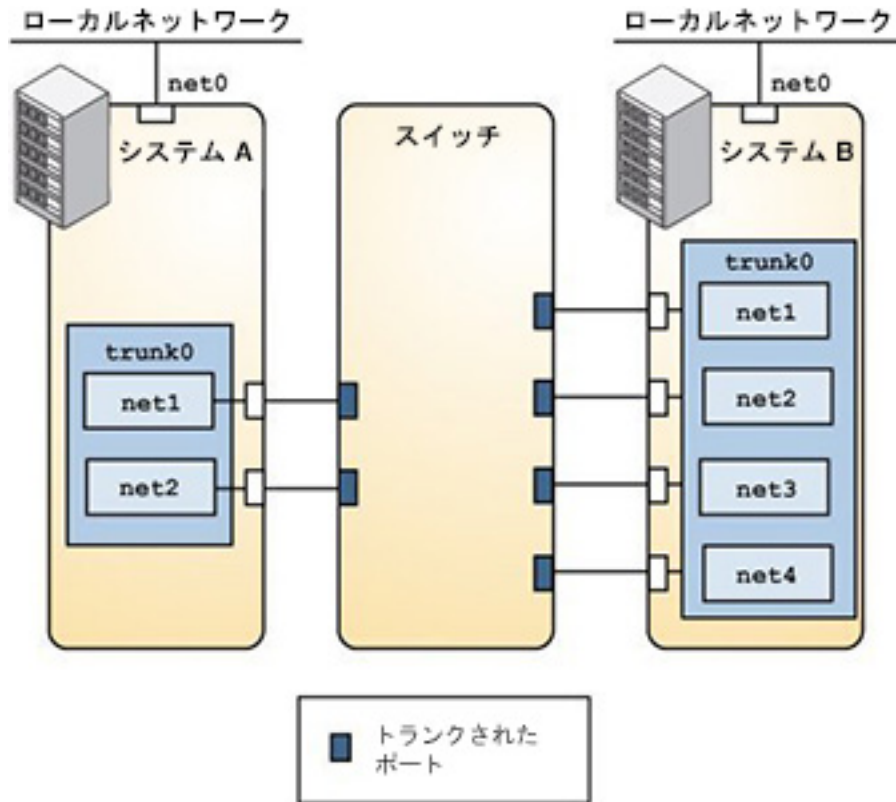
- スイッチの使用
- Link Aggregation Control Protocol (LACP) を指定したスイッチの使用
- バックツーバックトランクアグリゲーション構成
- アグリゲーションポリシーと負荷分散

次のセクションでは、トランクアグリゲーションの機能について説明します。

スイッチの使用

トランクアグリゲーションが構成されているシステムは、ほかのシステムへの接続に外部スイッチを使用する場合があります。次の図は、各システムにトランクアグリゲーションが構成されている、2 つのシステムを備えたローカルネットワークを示しています。

図 2 スイッチを使用するトランクアグリゲーション

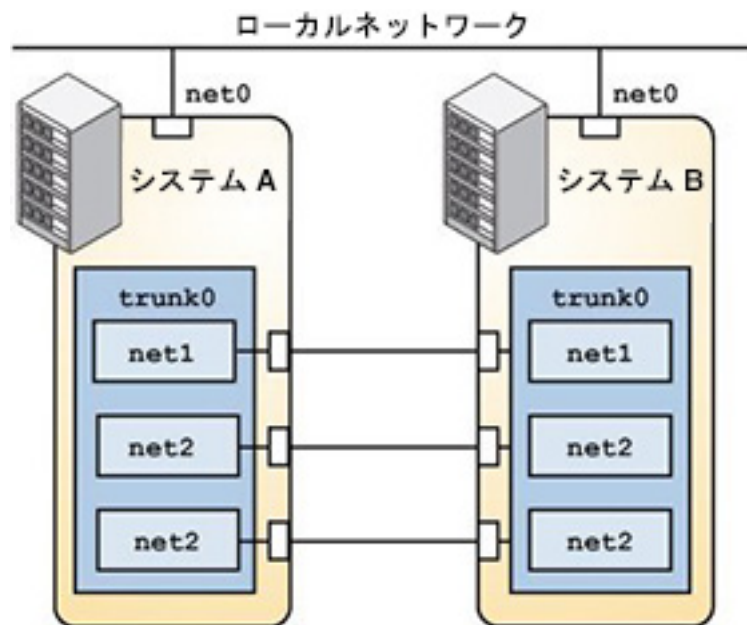


2つのシステムはスイッチによって接続されています。システム A には、2つのデータリンク (net1 と net2) で構成されるトランクアグリゲーションがあります。これらのデータリンクは、集約されたポートを介してスイッチに接続されています。システム B には、4つのデータリンク (net1 から net4) のトランクアグリゲーションがあります。これらのデータリンクもスイッチの集約されたポートに接続されています。このトランクアグリゲーショントポロジでは、スイッチが IEEE 802.3ad 標準をサポートし、スイッチポートがアグリゲーション用に構成されている必要があります。スイッチを構成するには、スイッチの製造元のドキュメントを参照してください。

バックツーバックトランクアグリゲーション構成

トランクアグリゲーションはバックツーバック構成をサポートします。次の図に示すように、スイッチを使用する代わりに、2つのシステムが直接接続され、並列アグリゲーションを実行します。

図 3 バックツーバックトランクアグリゲーション構成



この図では、システム A のトランクアグリゲーション trunk0 とシステム B のトランクアグリゲーション trunk0 が、それぞれのベースとなるデータリンク間の対応するリンクを使用して直接接続されています。この設定により、システム A とシステム B は冗長性と高可用性を提供し、両方のシステム間での高速通信を提供できます。各システムではさらに、ローカルネットワーク内のトラフィックフロー用の net0 も構成されています。

バックツーバックトランクアグリゲーションのもっとも一般的な用途は、データセンターなどの大規模な配備でのミラー化されたデータベースサーバーの構成です。両方のサーバーを同時に更新する必要があるため、大きな帯域幅、高速のトラフィックフロー、および信頼性が必要になります。

Link Aggregation Control Protocol を指定したスイッチの使用

トランクアグリゲーションの設定にスイッチが含まれ、スイッチが LACP をサポートしている場合、スイッチおよびシステムに対して LACP を有効にできます。Oracle Switch ES1-24 は LACP をサポートします。LACP の構成の詳細については、[Sun Ethernet Fabric Operating System, LA Administration Guide](#) を参照してください。その他のスイッチを使用している場合は、スイッチ製造元のドキュメントを参照してスイッチを構成してください。

LACP により、データリンク障害の信頼性の高い検出方法が有効になります。LACP を使用しない場合、リンクアグリゲーションはデバイスドライバによって報告されたリンク状態にのみ依存して、集約されたデータリンクの障害を検出します。LACP を使用した場合、集約されたデータリンクがトラフィックを送受信できるように LACPDU は定期的に交換されます。また、LACP は、データリンクのグループ化が 2 つのピアの間で一致しない場合など、構成ミスの一部のケースを検出します。

LACP は、LACP がシステム上で有効な場合、Link Aggregation Control Protocol Data Units (LACPDU) という特別なフレームをアグリゲーションとスイッチの間で交換します。LACP は、集約されたデータリンクの状態を保持するためにこれらの LACPDU を使用します。

アグリゲーションの LACP を次の 3 つのモードのいずれかに構成するには、`dladm create-aggr` コマンドを使用します。

- `off` – アグリゲーションのデフォルトのモード。システムは LACPDU を生成しません。
- `active` – システムは指定された間隔で LACPDU を生成します。
- `passive` – システムは、スイッチから LACPDU を受け取った場合のみ LACPDU を生成します。アグリゲーションとスイッチの両方が `passive` モードで構成されている場合、それらの間で LACPDU を交換しません。

LACP の構成方法については、[30 ページの「リンクアグリゲーションを作成する方法」](#) を参照してください。

負荷分散のアグリゲーションポリシーの定義

使用可能なアグリゲーションのリンク全体に負荷を分散する方法を指定する、送信トラフィック用のポリシーを定義して、負荷分散を確立できます。次の負荷指示子を使用してさまざまな負荷分散ポリシーを強制できます。

- `L2` – 各パケットの MAC (L2) ヘッダーを使用することで送信リンクを決定します

- L3 – 各パケットの IP (L3) ヘッダーを使用することで送信リンクを決定します
- L4 – 各パケットの TCP、UDP、またはほかの ULP (L4) ヘッダーを使用することで送信リンクを決定します

これらのポリシーを任意に組み合わせて使用することもできます。デフォルトのポリシーは L4 です。

データリンクマルチパスアグリゲーション

データリンクマルチパス (DLMP) アグリゲーションは、スイッチ構成を必要とせずに複数のスイッチにわたって高可用性を提供する一種のリンクアグリゲーションです。DLMP アグリゲーションは、トラフィックを送受信するネットワークの継続的な可用性を確保するためにリンクベースの障害検出とプローブベースの障害検出をサポートします。

DLMP アグリゲーションのメリット

DLMP アグリゲーションには次のメリットがあります。

- トランクアグリゲーションによって実装された IEEE 802.3ad 標準には、複数のスイッチにまたがるプロビジョニングがありません。トランクモードで複数のスイッチ間のフェイルオーバーを可能にするには、ベンダー間で互換性のないスイッチにベンダー独自の拡張機能が必要です。DLMP アグリゲーションを使用すると、ベンダー独自の拡張機能を必要とせずに複数のスイッチ間のフェイルオーバーが可能になります。
- ネットワーク仮想化での高可用性のために IPMP を使用すると非常に複雑になります。IPMP グループをゾーンに直接割り当てることはできません。ネットワークインタフェースカード (NIC) が複数のゾーンの間で共有される必要がある場合、各ゾーンが各物理 NIC から 1 つの VNIC を取得するように VNIC を構成する必要があります。各ゾーンは VNIC を IPMP グループにグループ化して、高可用性を実現する必要があります。たとえば、多数のシステム、ゾーン、NIC、仮想 NIC (VNIC)、および IPMP グループを含むシナリオなど、構成を拡大するほど複雑さが増します。DLMP アグリゲーションを使用する場合、アグリゲーション上に VNIC を作成するか、ゾーンの anet リソースを構成すると、ゾーンに高可用性 VNIC が表示されます。
- DLMP アグリゲーションにより、リンク保護、ユーザー定義フロー、リンクプロパティ (帯域幅など) をカスタマイズする機能など、リンク層の機能を使用できます。

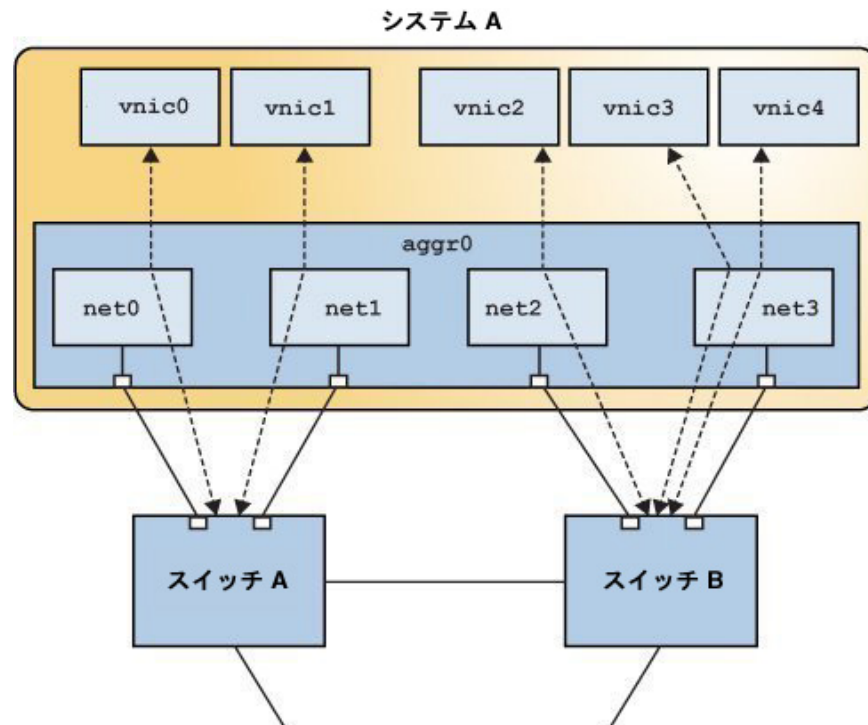
注記 - DLMP アグリゲーション上に IPMP グループを構成してはいけません。ただし、トランクアグリゲーション上に IPMP グループを構成することはできます。

DLMP アグリゲーションの動作

トランクアグリゲーションでは、アグリゲーション上に構成されている各データリンクに各ポートが関連付けられます。DLMP トランクアグリゲーションでは、アグリゲーションに構成されているいずれかのデータリンクにポートが関連付けられます。

次の図は、DLMP アグリゲーションの動作を示しています。

図 4 DLMP アグリゲーション



この図は、リンクアグリゲーション aggr0 を含むシステム A を示しています。このアグリゲーションは、ベースとなる 4 つのリンク net0 から net3 で構成されています。

す。vnic0 から vnic4 までの VNIC もアグリゲーション上に構成されています。アグリゲーションはスイッチ A およびスイッチ B に接続されており、これらのスイッチはより広いネットワーク内のほかの宛先システムに接続されています。

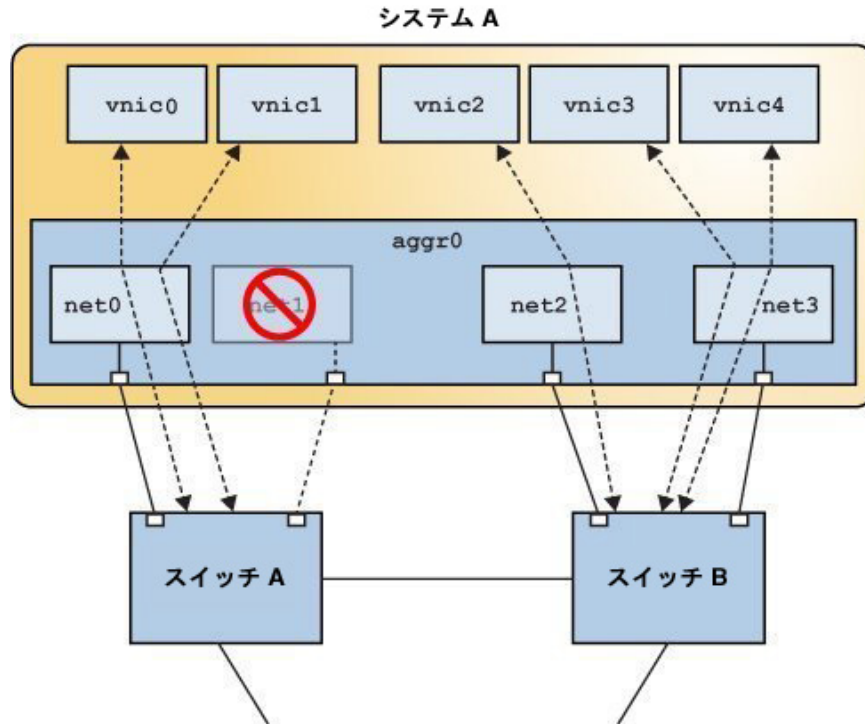
VNIC はベースとなるリンクを介して集約されたポートに関連付けられます。たとえば、この図では vnic0 から vnic3 までが、net0 から net3 までのベースとなるリンクを介して集約されたポートに関連付けられます。つまり、VNIC の数およびベースとなるリンクの数が同じ場合には、各ポートはベースとなるリンクに関連付けられません。

VNIC の数がベースとなるリンクの数を超えている場合、1つのポートに複数のデータリンクが関連付けられます。たとえば、この図では VNIC の合計数がベースとなるリンクの数を超えています。したがって、vnic4 は vnic3 とポートを共有します。

集約されたポートが失敗した場合、そのポートを使用するすべてのデータリンクはほかのポートの間に分散され、それによりフェイルオーバー中にネットワーク高可用性が提供されます。たとえば、net0 が失敗した場合は、DLMP アグリゲーションは残りのポート net1 を VNIC 間で共有します。アグリゲーションポート間の分散は、ユーザーに対しては透過的に、また、アグリゲーションに接続されている外部スイッチとは無関係に行われます。

次の図は、ポートが失敗した場合の DLMP アグリゲーションの動作を示しています。この図では、net1 は失敗し、スイッチと net1 の間のリンクが停止しています。vnic1 は vnic0 から net0 までとポートを共有します。

図 5 ポートが失敗した場合の DLMP アグリゲーション



DLMP アグリゲーションの障害検出

DLMP アグリゲーションの障害検出は、集約されたポートの障害を検出する方法です。ポートはトラフィックを送受信できない場合に失敗したとみなされます。ポートは次の理由により失敗することがあります。

- ケーブルの損傷または切断
- スイッチポートのダウン
- 上位ネットワークパスの障害

DLMP アグリゲーションは、トラフィックを送受信するネットワークの継続的な可用性を保証するために集約されたポートに対して障害検出を実行します。ポートが失敗すると、そのポートに関連付けられたクライアントはアクティブなポートにフェイル

オーバーされます。集約されたポートが失敗すると、そのポートは修復されるまで使用不可能なままになります。残りのアクティブポートが機能し続ける一方で、既存のポートが必要に応じて配備されます。失敗したポートが障害から回復すると、そのほかのアクティブなポートのクライアントをそのポートに関連付けることができます。

DLMP アグリゲーションはリンクベースおよびプローブベースの障害検出をサポートしています。

リンクベースの障害検出

リンクベースの障害検出は、ケーブルの切断時またはスイッチポートのダウン時に障害を検出します。したがって、データリンクと1ホップ目のスイッチの間の直接接続の損失が原因で発生する障害のみを検出できます。リンクベースの障害検出は DLMP アグリゲーションの作成時にデフォルトで有効です。

プローブベースの障害検出

プローブベースの障害検出は、エンドシステムと構成されたターゲット IP アドレスの間の障害を検出します。この機能はリンクベースの障害検出の既知の制限を克服します。プローブベースの障害検出は、デフォルトのルーターがダウンしているか、あるいはネットワークが到達不能になった場合に役立ちます。DLMP アグリゲーションはプローブパケットを送受信することで障害を検出します。

プローブベースの障害検出を DLMP アグリゲーションで有効にするには、`probe-ip` プロパティを構成する必要があります。

注記 - DLMP アグリゲーションに `probe-ip` が構成されていない場合は、プローブベースの障害検出が無効になり、リンクベースの障害検出のみが使用されます。

プローブベースの障害検出は2種類のプローブ (Internet Control Message Protocol (ICMP (L3)) プローブと推移的 (L2) プローブ) の組み合わせを使用して実行されます。これらのプローブは、集約された物理的なデータリンクの状態を判定するために連携して動作します。

■ ICMP プローブ

ICMP プローブは、ICMP パケットをターゲット IP アドレスに送信して動作します。ソースとターゲットの IP アドレスは手動で指定することも、システムで選択させることもできます。ソースアドレスが指定されない場合、DLMP アグリゲーション上または DLMP アグリゲーション上で構成された VNIC 上で構成されている IP アドレスからソースアドレスが選択されます。ターゲットアドレスが指

定されない場合、指定されたソース IP アドレスの 1 つと同じサブネット上にあるいずれかの次のホップルーターからターゲットアドレスが選択されます。詳細は、[37 ページの「DLMP のプローブベースの障害検出の構成方法」](#)を参照してください。

■ 推移的プローブ

推移的プローブはすべてのネットワークポートの健全性の状態が ICMP プローブのみを使用して判断できない場合に実行されます。これは、ICMP プローブ用に選択されたソース IP アドレスが特定のポートで構成されているアドレスのリストに含まれない場合に発生する可能性があります。その結果、このポートは ICMP プローブ応答を受け取ることができません。推移的プローブは、構成済みソース IP アドレスのないポートから構成済みソース IP アドレスを持つ別のポートへ L2 パケットを送信することで動作します。その他のポートが ICMP ターゲットに到達でき、推移的プローブを送信するポートがそのポートから L2 応答を受信する場合、これは推移的プローブを送信するポートも ICMP ターゲットに到達できることを意味します。

DLMP アグリゲーションでは、タグなしトラフィックが制限されている構成をサポートするために ICMP と推移的プローブの両方に使用できる VLAN をオプションで指定できます。probe-vlan-id プロパティを使用して単一の VLAN ID を指定でき、指定された VLAN は、ターゲット IP への ICMP プローブの送信と、推移的プローブに依存するメンバーポートの健全性のチェックの両方に使用されます。L3 プローブターゲットが、このプロパティに基づいて自動的に選択されて検証されます。probe-vlan-id プロパティには 0 - 4094 の値を指定できます。デフォルト値は 0 で、プローブがタグなしであることを示します。

Oracle Solaris には、ネットワークを介して送信される推移的プローブに対する独自のプロトコルパケットが含まれます。詳細は、[付録B 推移的プローブのパケット形式](#)を参照してください。

リンクアグリゲーションの要件

リンクアグリゲーションの構成には次のような要件があります。

- アグリゲーションに構成するデータリンク上には、IP インタフェースを構成しないでください。
- 大域ゾーンからのみリンクアグリゲーションを作成できます。IP インタフェースをデータリンク上に構成しない場合でも、データリンクを使用して非大域ゾーンからリンクアグリゲーションを作成することはできません。リンクアグリゲーションは複数の物理 NIC のみを組み合わせますが、非大域ゾーン内ではすべてのインタフェースが仮想です。したがって、非大域ゾーンからはリンクアグリゲーションを作成できません。
- アグリゲーション内のすべてのデータリンクは、同じ速度かつ全二重モードで実行される必要があります。

- DLMP アグリゲーションの場合、アグリゲーションをほかのシステムのポートに接続するために、少なくとも1つのスイッチが必要です。DLMP アグリゲーションを構成する場合、バックツーバック設定は使用できません。
- SPARC ベースのシステムでは、各データリンクに一意の MAC アドレスが必要です。詳細については、『[Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理](#)』の「[各インタフェースの MAC アドレスが一意であることを確認する方法](#)」を参照してください。
- (トランクアグリゲーションのみ) ポートをトランクアグリゲーションに接続したり、トランクアグリゲーションから切り離したりするには、IEEE 802.3ad Link Aggregation Standard で定義されているリンク状態通知がデバイスでサポートされている必要があります。リンク状態通知をサポートしていないデバイスは、`dladm create-aggr` コマンドの `-f` オプションを使用してのみ集約できます。このようなデバイスの場合、リンク状態は常に UP として報告されます。`-f` オプションの使用については、[30 ページの「リンクアグリゲーションを作成する方法」](#)を参照してください。

リンク集約の作成

リンクアグリゲーションはベースとなるポートを単一の論理グループにグループ化します。アグリゲーションは、これらのベースとなるポートを排他的に使用し、ほかの操作(これらのポート上での VNIC の構成や IP アドレスの割り当てなど)は実行できません。ただし、個々のポート上ではなくアグリゲーション上に VNIC を構成できます。

リンクアグリゲーションを作成する前に、これらのポート上にある既存の IP インタフェースを削除する必要があります。

作成したリンクアグリゲーションの上に VLAN を構成し、VNIC を作成することもできます。リンクアグリゲーション上に VLAN を作成する方法については、[60 ページの「リンクアグリゲーション上に VLAN を構成する方法」](#)を参照してください。

注記 - リンクアグリゲーションは、同一の速度で稼働する全二重のポイントツーポイントリンク上でのみ機能します。アグリゲーション内のデータリンクがこの要件を満たしていることを確認してください。

▼ リンクアグリゲーションを作成する方法

始める前に トランクアグリゲーションを作成して、トランクアグリゲーションのスイッチを使用する場合、アグリゲーションとして使用されるポートをスイッチ上に構成します。ス

スイッチが LACP をサポートしている場合は、LACP を active または passive モードで構成します。

スイッチを構成するには、スイッチの製造元のドキュメントを参照してください。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. アグリゲーションの物理データリンクを識別するデータリンク情報を表示します。

```
# dladm show-phys
```

3. 集約しようとしているデータリンクが、どのアプリケーションでも使用されていないことを確認します。

たとえば、データリンク上に IP インタフェースが作成されている場合は、まずその IP インタフェースを削除します。

a. リンクの状態を判定します。

```
# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE      OVER
lo0         loopback   ok         yes         --
net0        ip         ok         no          --
```

この出力は、データリンク net0 上に IP インタフェースが存在することを示しています。

b. IP インタフェースを削除します。

```
# ipadm delete-ip interface
```

ここで、*interface* はリンク上の IP インタフェースを指定します。詳細は、[ipadm\(1M\)](#) のマニュアルページを参照してください。

4. リンクアグリゲーションを作成します。

```
# dladm create-aggr [-f] [-m mode] [-P policy] [-L LACP-mode] \
[-T time] [-u address] -l link1 -l link2 [...] aggr
```

-f アグリゲーションを強制的に作成します。このオプションは、リンク状態通知をサポートしていないデバイスを集約しようとする場合に使用します。

-m mode モードは次のいずれかの値に設定する必要があります。デフォルトのモードは trunk です。

- trunk – IEEE 802.3ad に準拠するリンクアグリゲーションモード
- d1mp – データリンクマルチパスモード

-P <i>policy</i>	(トランクアグリゲーションのみ) アグリゲーションの負荷分散ポリシーを指定します。サポートされる値は L2、L3、および L4 です。詳細は、 23 ページの「負荷分散のアグリゲーションポリシーの定義」 を参照してください。
-L <i>LACP-mode</i>	(トランクアグリゲーションのみ) LACP が使用される場合にそのモードを指定します。サポートされている値は、 <code>off</code> 、 <code>active</code> 、または <code>passive</code> です。モードについては、 20 ページの「スイッチの使用」 を参照してください。
-T <i>time</i>	(トランクアグリゲーションのみ) LACP タイマー値を指定します。サポートされている値は、 <code>short</code> または <code>long</code> です。
-u <i>address</i>	アグリゲーションの固定ユニキャストアドレスを指定します。
-l <i>linkn</i>	集約するデータリンクを指定します。
<i>aggr</i>	アグリゲーションの名前を指定します。カスタマイズした任意の名前を使用できます。名前を割り当てるルールについては、『 Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理 』の「 有効なリンク名のための規則 」を参照してください。

5. (オプション) 作成したアグリゲーションのステータスを確認します。

- アグリゲーションおよびリンクをステータス情報とともに表示します。
- ステータスおよびポートごとの情報とともにアグリゲーションを表示します。

```
# dladm show-link
```

```
# dladm show-aggr -x
```

アグリゲーションの状態は `up` であるべきです。

例 1 トランクアグリゲーションの作成

この例は、ベースとなる 2 つのデータリンク `net0` および `net1` を含むリンクアグリゲーションを作成するコマンドを示しています。また、アグリゲーションは LACP パケットを送信するように構成されます。この例ではまず、ベースとなるデータリンク上の既存の IP インタフェースを削除します。

```
# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE      OVER
lo0         loopback   ok         yes         --
net0        ip         ok         no          --
# ipadm delete-ip net0
# dladm create-aggr -L active -l net0 -l net1 trunk0
# dladm show-aggr -x
```


LINK	PORT	SPEED	DUPLEX	STATE	ADDRESS	PORTSTATE
trunk0	--	1000Mb	full	up	8:0:27:49:10:b8	--
	net0	1000Mb	full	up	8:0:27:49:10:b8	attached
	net1	1000Mb	full	up	8:0:27:e4:d9:46	attached

例 2 DLMP アグリゲーションの作成と DLMP アグリゲーション上での IP インタフェースの構成

この例は、DLMP アグリゲーションを作成する方法を示しています。アグリゲーションには、ベースとなる 3 つのデータリンク (net0、net1、および net2) があります。IP インタフェースはアグリゲーション aggr0 上に作成され、VNIC vnic1 もそのアグリゲーション上に作成されます。

```
# dladm create-aggr -m dlmp -l net0 -l net1 -l net2 aggr0
# dladm show-link
LINK      CLASS      MTU      STATE    OVER
net0      phys       1500     up       --
net1      phys       1500     up       --
net2      phys       1500     up       --
aggr0     aggr       1500     up       net0 net1 net2
# dladm show-aggr -x
LINK      PORT          SPEED DUPLEX  STATE  ADDRESS          PORTSTATE
aggr0     --            1000Mb full    up     8:0:27:49:10:b8  --
          net0         1000Mb full    up     8:0:27:49:10:b8  attached
          net1         1000Mb full    up     8:0:27:e4:d9:46  attached
          net2         1000Mb full    up     8:0:27:38:7a:97  attached
# ipadm create-ip aggr0
# ipadm create-addr -a local=10.10.10.1 aggr0/v4
# dladm create-vnic -l aggr0 vnic1
```

次の手順 IP インタフェースや VNIC の作成など、アグリゲーションの追加の構成を実行できます。非大域ゾーンおよびカーネルゾーンを構成するために、作成されたアグリゲーションを使用できます。

- IP インタフェースの作成については、『Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理』の第 3 章、「Oracle Solaris での IP インタフェースとアドレスの構成および管理」を参照してください。
- リンクアグリゲーション上の VLAN の構成については、60 ページの「リンクアグリゲーション上での VLAN の構成」を参照してください。
- VNIC の構成については、『Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理』の「VNIC と etherstub を構成する方法」を参照してください。
- ゾーンの構成については、『Oracle Solaris ゾーンの作成と使用』を参照してください。

アグリゲーションへのリンクの追加

既存のアグリゲーションに追加のデータリンクを含めることができます。トランクアグリゲーションにデータリンクを追加する場合、LACP がスイッチに構成されている

場合でも、追加のデータリンクを収容できるようにスイッチを再構成する必要があることがあります。

▼ アグリゲーションにリンクを追加する方法

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. IP インタフェースがリンク上に構成されていないことを確認します。IP インタフェースが構成されている場合は、その IP インタフェースを削除します。

```
# ipadm show-if
# ipadm delete-ip interface
```

ここで、*interface* は、データリンク上に構成されている IP インタフェースです。

3. アグリゲーションにリンクを追加します。

```
# dladm add-aggr -l link [-l link] [...] aggr
```

ここで、*link* はアグリゲーションに追加するデータリンクを表し、*aggr* はアグリゲーションの名前です。

4. (トランクアグリゲーションのみ) 必要に応じてスイッチを再構成します。

LACP がスイッチで有効な場合でも、スイッチの構成方法に基づいて追加のデータリンクを収容できるようにスイッチを再構成する必要があることがあります。

スイッチに対して何らかの再構成タスクを実行するには、スイッチの製造元のドキュメントを参照してください。

例 3 アグリゲーションへのリンクの追加

この例は、アグリゲーション *aggr0* にリンクを追加する方法を示しています。

```
# dladm show-link
LINK CLASS MTU STATE OVER
net0 phys 1500 up --
net1 phys 1500 up --
aggr0 aggr 1500 up net0 net1
net3 phys 1500 up --
# ipadm delete-ip net3
# dladm add-aggr -l net3 aggr0
# dladm show-link
LINK CLASS MTU STATE OVER
net0 phys 1500 up --
net1 phys 1500 up --
aggr0 aggr 1500 up net0 net1 net3
```

```
net3    phys    1500   up     --
```

アグリゲーションからのリンクの削除

`dladm remove-aggr` コマンドを使用してアグリゲーションに関連付けられた個々のデータリンクを削除できます。アグリゲーションからリンクを削除する場合、スイッチを再構成する必要があります。

管理者になり、次のコマンドを使用します。

```
# dladm remove-aggr -l link aggr
```

例 4 アグリゲーションからのリンクの削除

この例は、アグリゲーション `aggr0` からリンクを削除する方法を示しています。

```
# dladm show-link
LINK    CLASS    MTU     STATE   OVER
net0    phys     1500    up      --
net1    phys     1500    up      --
aggr0   aggr     1500    up      net0 net1 net3
net3    phys     1500    up      --
# dladm remove-aggr -l net3 aggr0
# dladm show-link
LINK    CLASS    MTU     STATE   OVER
net0    phys     1500    up      --
net1    phys     1500    up      --
aggr0   aggr     1500    up      net0 net1
net3    phys     1500    unknown --
```

トランクアグリゲーションの変更

選択したトランクアグリゲーション用の属性 (`policy`、`lacpmode`、`time` など) を変更できます。これらの属性は DLMP アグリゲーションではサポートされていません。

- アグリゲーションの負荷分散ポリシーを変更するには、管理者になり、次のコマンドを発行します。

```
# dladm modify-aggr -P policy aggr
```

`policy` 1 つ以上の [23 ページの「負荷分散のアグリゲーションポリシーの定義」](#) で説明されている負荷分散ポリシー (L2、L3、および L4) を表します。

`aggr` ポリシーを変更するアグリゲーションを指定します。

- アグリゲーションの LACP モードを変更するには、管理者になり、次のコマンドを使用します。

```
# dladm modify-aggr -L LACP-mode -T time aggr
```

-L *LACP-mode* アグリゲーションが動作する必要がある LACP モードを示します。指定できる値は、**active**、**passive**、および **off** です。

-T *time* LACP タイマー値を示します。値は、**short** または **long** です。

aggr LACP モードを変更するアグリゲーションを指定します。

注記 - アグリゲーションがアクティブまたは使用中であっても、アグリゲーションのロードバランシングポリシー、LACP モード、および LACP タイマー値を変更できます。また、IP インタフェースと VNIC がアグリゲーション上に構成されている場合でも、これらの属性を変更できます。

例 5 トランクアグリゲーションの変更

この例は、リンクアグリゲーション `aggr0` の負荷分散ポリシーを L2 に変更する方法と、LACP モードを **active** に変更する方法を示しています。

```
# dladm modify-aggr -P L2 aggr0
# dladm modify-aggr -L active -T short aggr0
# dladm show-aggr
LINK   MODE      POLICY  ADDRPOLICY  LACPACTIVITY  LACPTIMER
aggr0  trunk    L2      auto        active         short
```

DLMP アグリゲーションのプロープベースの障害検出の構成

プローブが有効になるように DLMP アグリゲーションの `probe-ip` プロパティを構成する必要があります。それ以外の場合は、プローブはデフォルトで無効になり、リンクベースの障害検出のみが使用されます。詳細については、[28 ページの「プロープベースの障害検出」](#)を参照してください。

`dladm set-linkprop` コマンドで次のデータリンクプロパティを使用して、プロープベースの障害検出を構成できます。

- `probe-ip` – ソース IP アドレス、ターゲット IP アドレス、またはソースとターゲット IP アドレスのコンマ区切りリストを指定します。これらのアドレスは ICMP プローブに使用されます。

注記 - ソースとターゲットの IP アドレスを指定することはオプションです。ソースとターゲットの IP アドレスを指定しない場合は、ソースとプローブの IP アドレスが自動的に選択されます。

ソースとターゲットを区切るために + を使用できます。ターゲットの IP アドレスまたはホスト名としてターゲットを指定できます。ソースアドレスおよびターゲットアドレスを指定する方法については、[37 ページの「DLMP のプローブベースの障害検出の構成方法」](#)を参照してください。

- `probe-vlan-id` - ICMP と推移的プローブの両方に使用される VLAN ID を指定します。有効な値は 0 - 4094 です。値 0 は、推移的プローブがタグなしであることを示します。デフォルト値は 0 です。
- `probe-fdt` - 障害検出時間を指定します。予測される障害検出時間の値 (秒単位) を構成できます。デフォルト値は 10 秒

▼ DLMP のプローブベースの障害検出の構成方法

始める前に DLMP アグリゲーションを作成します。詳細は、[30 ページの「リンクアグリゲーションを作成する方法」](#)を参照してください。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. (オプション) プローブベースの障害検出を構成するアグリゲーションを識別するために、既存のアグリゲーションをすべて表示します。

```
# dladm show-aggr
```

3. 自動選択されたソースとターゲットの IP アドレスを使用するか、手動選択によって、アグリゲーションのプローブを構成します。

■ 自動選択を使用するには、次のコマンドを発行します。

ターゲットアドレスを指定しない場合、DLMP アグリゲーションは、ソース IP アドレスと同じサブネット上にあるいずれかの次のホップルーターからターゲット IP を自動的に選択します。

```
# dladm set-linkprop -p probe-ip=+ aggr
```

ソース IP アドレスが指定されていない場合は、アグリゲーションおよび VNIC 上に構成されたすべての IP アドレスが、ICMP プローブの潜在的なソース IP アドレスです。大域ゾーンで構成された IP インタフェースのみがソースアドレスとして

使用されます。たとえば、DLMP アグリゲーションを使用している非大域ゾーンの IP アドレスは、ソースアドレスとして使用されません。

■ 手動選択を使用するには、次のコマンドを発行します。

```
# dladm set-linkprop -p probe-ip=[source[,...]]+[target[,...]] aggr
```

<i>source</i>	<p>プローブのソース IP アドレスを指定します (オプション)。ソース IP アドレスは、次のいずれかの方法で書式設定できます。</p> <ul style="list-style-type: none"> ■ <code>probe-ip=IP-address[/prefix-length]+</code> IP アドレスとその接頭辞の長さ。たとえば、<code>10.130.10.1/24+</code> とします。 ■ <code>probe-ip=addr-obj-name+</code> アドレスオブジェクト名。たとえば、<code>vnic1/addr1+</code> です。 ■ <code>probe-ip=interface-name+</code> インタフェース名。集約されたインタフェースまたはアグリゲーション上で構成された VNIC の名前。たとえば、<code>[aggr1]</code> です。インタフェースは、構成済み IP アドレスを持つ IP インタフェースである必要があります。 ■ <code>probe-ip=hostname+</code> ホスト名。たとえば、<code>sdg1+</code> です
<i>target</i>	<p>プローブのターゲット IP アドレスを指定します (オプション)。ターゲット IP アドレスを指定するときは、次のいずれかの形式にする必要があります。</p> <ul style="list-style-type: none"> ■ <code>probe-ip+=IP-address</code> たとえば、<code>+10.130.10.1</code> とします。 ■ <code>probe-ip+=hostname</code> たとえば、<code>+sdg1</code> です
<i>aggr</i>	アグリゲーションの名前。

注記 - ターゲットとして使用する特定のアドレスがある場合を除き、ターゲット IP アドレスを明示的に指定しないことをお勧めします。ターゲットアドレスを指定しない場合、DLMP アグリゲーションは、ソース IP アドレスと同じサブネット上にあるいずれかの次のホップルーターからターゲット IP を自動的に選択します。次のホップルーターを決定または構成する方法の詳細については、『[Oracle Solaris 11.3 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理](#)』の第 1 章、「TCP/IP ネットワークの管理」を参照してください。

4. (オプション) プローブの VLAN ID を設定します。

```
# dladm set-linkprop -p probe-vlan-id=probe-VID aggr
```

ここで、*probe-VID* は、ICMP と推移的プローブの両方に使用される VLAN ID です。*probe-VID* の有効な値の範囲は 0 - 4094 です。*probe-VID* 値 0 は、推移的プローブがタグなしであることを示します。デフォルト値は 0 です。

5. (オプション) 障害検出時間を設定します。

```
# dladm set-linkprop -p probe-fdt=fdt aggr
```

ここで、*fdt* は指定された障害検出時間 (秒単位) です。デフォルト値は 10 秒

6. (オプション) プローブの統計情報を表示します。

```
# dlstat show-aggr -n -P [[t],[i],[all]]
```

ここで、*-P* は、表示するプローブのタイプを指定します。プローブのタイプとして次のいずれかをコンマ区切りリストで指定できます。

- *t* – 推移的プローブを表示します。
- *i* – ICMP プローブを表示します。
- *all* – 推移的プローブと ICMP プローブの両方を表示します。

例 6 プローブベースの障害検出の構成

この例は、ソースとプローブの IP アドレスのデフォルトの自動選択を使用して、プローブベースの障害検出を構成する方法を示しています。デフォルトの自動選択を使用するには、DLMP アグリゲーションの *ip-probe* プロパティを + に設定する必要があります。

1. DLMP アグリゲーションを作成します。

```
# dladm create-aggr -m dlmp -l net0 -l net1 -l net2 aggr1
# dladm show-aggr -x
```

LINK	PORT	SPEED	DUPLEX	STATE	ADDRESS	PORTSTATE
aggr0	--	1000Mb	full	up	8:0:27:49:10:b8	--
	net0	1000Mb	full	up	8:0:27:49:10:b8	attached
	net1	1000Mb	full	up	8:0:27:e4:d9:46	attached
	net2	1000Mb	full	up	8:0:27:38:7a:97	attached

2. 集約上に IP インタフェースを作成します。

```
# ipadm create-ip aggr1
```

3. IP アドレスをインタフェースに割り当てます。

```
# ipadm create-addr -a 192.168.85.137/24 aggr1
```

4. プローブベースの障害検出を構成します。

```
# dladm set-linkprop -p probe-ip=+ aggr1
```

ソースとターゲットの IP アドレスが指定されていないため、アグリゲーション aggr1 上に構成された IP アドレス 192.168.85.137 が、ICMP プローブのソース IP アドレスになります。

5. プローブの VLAN ID を設定します。

```
# dladm set-linkprop -p probe-vlan-id=2100 aggr1
```

6. 障害検出時間を設定します。

```
# dladm set-linkprop -p probe-fdt=15 aggr1
```

7. 設定されているプロパティを表示します。

```
# dladm show-linkprop -p probe-ip,probe-fdt aggr1
```

LINK	PROPERTY	PERM	VALUE	EFFECTIVE	DEFAULT	POSSIBLE
aggr1	probe-ip	rw	+	+	--	--
aggr1	probe-fdt	rw	15	15	10	1-600

8. アグリゲーションのプローブの統計情報を表示します。

```
# dlstat show-aggr -n -P t,i aggr1
```

TIME	AGGR	PORT	LOCAL	TARGET	PROBE	NETRTT	RTT
0.45s	aggr1	net0	net0	net1	t16148	--	--
0.45s	aggr1	net0	net0	net1	t16148	0.63ms	0.81ms
1.08s	aggr1	net1	net1	net0	t16148	--	--
1.08s	aggr1	net1	net1	net0	t16148	0.72ms	0.99ms
2.07s	aggr1	net1	192.168.85.137	192.168.85.137	i15535	--	--
2.07s	aggr1	net1	192.168.85.137	192.168.85.137	i15535	0.18ms	0.54ms

出力に表示されるプローブの統計情報の各フィールドについては、[例7「プローブ関連情報の表示」](#)を参照してください。

プローブベースの障害検出のモニタリング

dladm show-aggr、dlstat show-aggr、および ipadm show-addr コマンドを使用して、プローブベースの障害検出をモニターできます。

dlstat show-aggr コマンドを次の構文で使用して、プローブ関連の情報を表示できます。

```
# dlstat show-aggr -n -P [[t],[i],[all]]
```

ここで、-P は、表示するプローブのタイプを指定します。プローブのタイプとして次のいずれかをコンマ区切りリストで指定できます。

- t – 推移的プローブを表示します。

- `i` – ICMP プローブを表示します。
- `all` – 推移的プローブと ICMP プローブの両方を表示します。

例 7 プローブ関連情報の表示

次の例では、DLMP アグリゲーション `aggr1` のプローブの統計情報を表示します。

```
# dlstat show-aggr -n -P t,i aggr1
TIME      AGGR      PORT      LOCAL      TARGET      PROBE  NETRTT      RTT
0.53s     aggr1     net0      net0        net1         t16148 --           --
0.53s     aggr1     net0      net0        net1         t16148 0.62ms      0.87ms
1.17s     aggr1     net1      net1        net0         t16148 --           --
1.17s     aggr1     net1      net1        net0         t16148 0.72ms      0.99ms
2.24s     aggr1     net1      192.168.0.1 192.168.0.2 i15535 --           --
2.24s     aggr1     net1      192.168.0.1 192.168.0.2 i15535 0.11ms      0.55ms
```

TIME プローブが送信される時間 (秒)。この時間は `dlstat` コマンドの発行時を基準とします。`dlstat` コマンドを発行する前にプローブが送信された場合、この時間は負になります。

AGGR プローブが送信されたアグリゲーションの名前。

LOCAL ICMP プローブ: プローブのソース IP アドレス。
推移的プローブ: 推移的プローブの発生元のポート名。

TARGET ICMP プローブ: プローブの宛先 IP アドレス。
推移的プローブ: 対象とされたプローブのポート名。

PROBE プローブを表す ID 番号。接頭辞 `t` は推移的プローブ用で、接頭辞 `i` は ICMP プローブ用です。

NETRTT プローブのネットワーク往復時間。この値は、DLMP アグリゲーションによるプローブ送信から確認応答受信までの時間です。

RTT プローブの合計往復時間。この値は、DLMP アグリゲーションによるプローブ送信から確認応答処理完了までの時間です。

詳細は、[dlstat\(1M\)](#) のマニュアルページを参照してください。

例 8 集約されたポートに関する詳細情報の表示

次の例では、ベースとなる各ポートの詳細なアグリゲーション情報を表示します。

```
# dladm show-aggr -x
LINK      PORT      SPEED  DUPLEX  STATE  ADDRESS          PORTSTATE
aggr1     --        100Mb  full    up     1e:34:db:fa:50:a2 --
```

```
net0 100Mb full up 1e:34:db:fa:50:a2 attached
net1 100Mb full up b2:c0:6a:3e:c5:b5 attached
```

詳細は、[dladm\(1M\)](#) のマニュアルページを参照してください。

例 9 集約されたポートの状態の表示

次の例では、アグリゲーションのポートの状態とポートのターゲット IP アドレスを表示します。

```
# dladm show-aggr -S -n
LINK      PORT      FLAGS   STATE   TARGETS   XTARGETS
aggr1     net1      u--3   active  192.168.0.2  net0
--        net0      u-2-   active  --         net1
```

例 10 probe-ip プロパティ値の表示

次の例では、指定された DLMP アグリゲーションのリンクプロパティ probe-ip の詳細を表示します。

```
# dladm show-linkprop -p probe-ip aggr1
LINK      PROPERTY  PERM   VALUE           EFFECTIVE        DEFAULT        POSSIBLE
aggr1     probe-ip  rw     192.168.0.2    192.168.0.2    --             --
```

例 11 アグリゲーションの IP アドレスおよび状態の表示

次の例では、アグリゲーションの IP アドレスおよび状態を表示します。

```
# ipadm show-addr aggr1
ADDROBJ   TYPE      STATE      ADDR
aggr1/local1  static  ok         192.168.0.1/24
```

リンクアグリゲーションの削除

dladm delete-aggr コマンドを使用してリンクアグリゲーションを削除できます。アグリゲーションを削除する前に、リンクアグリゲーション上に構成されている IP インタフェースおよび VNIC を削除する必要があります。

注記 - dladm delete-phys コマンドを使用してデータリンクを削除するときは、そのデータリンク上で構成されているすべてのアグリゲーション (レイヤー 2 コンポーネント) も削除されます。この dladm delete-phys コマンドは、データリンク上で構成されているフロー、VNIC、VLAN などその他のレイヤー 2 コンポーネントも削除しません。ただし、IP インタフェースなど、データリンク上で構成されているレイヤー 3 コンポーネントは、ipadm コマンドを使用して手動で削除する必要があります。

▼ リンクアグリゲーションを削除する方法

1. 管理者になります。

詳細は、『Oracle Solaris 11.3でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. リンクアグリゲーション上に構成されている IP インタフェースを削除します。

```
# ipadm delete-ip IP-aggr
```

ここで、*IP-aggr* はリンクアグリゲーション上の IP インタフェースです。

3. リンクアグリゲーションを削除します。

```
# dladm delete-aggr aggr
```

例 12 リンクアグリゲーションの削除

この例は、アグリゲーション *aggr0* を削除する方法を示しています。この削除は永続します。

```
# ipadm delete-ip aggr0
# dladm delete-aggr aggr0
```

トランクアグリゲーションと DLMP アグリゲーションの間の切り替え

トランクアグリゲーションと DLMP アグリゲーションの間で切り替えると構成全体が変わるため、ほかのリンクアグリゲーションプロパティを単に変更する場合よりも総合的にアグリゲーションが影響を受けます。アグリゲーションの使用中にアグリゲーションのモードを切り替えるには、アグリゲーション上のすべての既存構成を削除してから、希望のモードでアグリゲーションを再構成する必要があります。

▼ リンクアグリゲーションのタイプを切り替える方法

- 始める前に
- トランクアグリゲーションから DLMP アグリゲーションに切り替える場合は、トランクアグリゲーション用に以前作成したスイッチ構成を削除する必要があります。
 - DLMP アグリゲーションから切り替える場合は、すべてのリンクが同じスイッチ上にあり、スイッチがアグリゲーション用に構成されていることを確認する必要があります。

1. 管理者になります。
詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. リンクアグリゲーションの現在のタイプを判定します。

```
# dladm show-aggr
```

出力の MODE フィールドは、アグリゲーションの現在のタイプを示します。MODE の値は、トランクアグリゲーションの場合は trunk、DLMP アグリゲーションの場合は d1mp です。

3. アグリゲーションを切り替えます。

```
# dladm modify-aggr -m mode aggr
```

ここで、mode は、トランクアグリゲーションに切り替える場合は trunk、DLMP アグリゲーションに切り替える場合は d1mp です。また、aggr はアグリゲーションの名前です。

4. 新しいタイプのリンクアグリゲーションの要件に従って、スイッチを構成します。
スイッチを構成するために、スイッチの製造元のドキュメントを参照してください。

5. (オプション) 現在のリンクアグリゲーション構成を確認します。

```
# dladm show-aggr
```

例 13 トランクアグリゲーションから DLMP アグリゲーションへの切り替え

この例は、アグリゲーションをトランクアグリゲーションから DLMP アグリゲーションに変更する方法を示しています。

```
# dladm show-aggr
LINK  MODE      POLICY  ADDRPOLICY  LACPACTIVITY  LACPTIMER
aggr0 trunk     L2      auto        active        short
```

```
# dladm modify-aggr -m d1mp aggr0
```

```
# dladm show-aggr
LINK  MODE      POLICY  ADDRPOLICY  LACPACTIVITY  LACPTIMER
aggr0 d1mp     --      --          --            --
```

アグリゲーションを切り替えたら、トランクアグリゲーションに適用された以前のスイッチ構成を削除する必要があります。

使用事例: リンクアグリゲーションの構成

次のエンドツーエンドの使用事例は、次のアクションを実行する方法を示しています。

- DLMP アグリゲーションを作成します。

- アグリゲーションにリンクを追加します。
- アグリゲーション上に IP インタフェースを構成します。
- アグリゲーション上に VNIC を構成します。
- アグリゲーションのプロープベースの障害検出を構成します。
- ルーティングテーブル内にターゲット IP アドレスを構成します。
- ICMP プロープと推移的プロープをモニターします。

1. 管理者になります。

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. アグリゲーションのデータリンクを識別するデータリンク情報を表示します。

```
# dladm show-link
LINK      CLASS    MTU     STATE   OVER
net0     phys    1500    up      --
net1     phys    1500    up      --
net2     phys    1500    up      --
```

3. 集約するデータリンク上には IP インタフェースが構成されていないことを確認します。インタフェースがいずれかのリンク上に構成されている場合、インタフェースを削除します。

```
# ipadm show-if
IFNAME    CLASS      STATE    ACTIVE   OVER
lo0       loopback  ok       yes      --
net0      ip         ok       no       --

# ipadm delete-ip net0
```

4. リンク net0 および net1 を含む DLMP アグリゲーションを作成します。

```
# dladm create-aggr -m dlmp -l net0 -l net1 aggr1
```

5. 別のリンク net2 をアグリゲーションに追加します。

```
# dladm add-aggr -l net2 aggr1
```

既存のスイッチ構成に新しいリンクが必要な場合、そのリンクを収容できるようにスイッチを再構成します。スイッチの製造元のドキュメントを参照してください。

6. アグリゲーション aggr1 上に IP インタフェースを構成します。

```
# ipadm create-ip aggr1
# ipadm create-addr -a local=10.10.10.1 aggr1/v4
```

7. アグリゲーション上に VNIC を作成します。

```
# dladm create-vnic -l aggr1 vnic1
```

8. アグリゲーションのプロープベースの障害検出を構成します。

```
# dladm set-linkprop -p probe-ip+= aggr1
```

ソースとターゲットのアドレスは、指定されていないため、自動的に選択されま
す。

9. 集約されたポートの状態とターゲットを表示します。

```
# dladm show-aggr -S
LINK      PORT      FLAGS  STATE  TARGETS  XTARGETS
aggr1     net0      u--3   active 10.10.10.2 net2 net1
--        net1      u-2-   active --        net2 net0
--        net2      u-2-   active --        net0 net1
```

10. ICMP プローブの統計情報をモニターします。

```
# dlstat show-aggr -n -P i
TIME      AGGR  PORT  LOCAL      TARGET      PROBE NETRTT  RTT
1.16s     aggr1 net0   10.10.10.1 10.10.10.2 i33   --      --
1.16s     aggr1 net0   10.10.10.1 10.10.10.2 i33   0.08ms 0.33ms
2.05s     aggr1 net0   10.10.10.1 10.10.10.2 i34   --      --
2.05s     aggr1 net0   10.10.10.1 10.10.10.2 i34   0.01ms 0.64ms
4.05s     aggr1 net0   10.10.10.1 10.10.10.2 i35   --      --
4.05s     aggr1 net0   10.10.10.1 10.10.10.2 i35   0.10ms 0.35ms
5.54s     aggr1 net0   10.10.10.1 10.10.10.2 i36   --      --
5.54s     aggr1 net0   10.10.10.1 10.10.10.2 i36   0.08ms 0.34ms
```

11. ポート間の推移的プローブの統計情報をモニターします。

```
# dlstat show-aggr -n -P t
TIME      AGGR  PORT  LOCAL      TARGET      PROBE NETRTT  RTT
0.30s     aggr1 net2   net2       net0       t38   --      --
0.30s     aggr1 net2   net2       net0       t38   0.46ms 0.59ms
0.46s     aggr1 net0   net0       net1       t39   --      --
0.46s     aggr1 net0   net0       net1       t39   0.46ms 0.50ms
0.48s     aggr1 net1   net1       net0       t39   --      --
0.48s     aggr1 net1   net1       net0       t39   0.34ms 0.38ms
0.72s     aggr1 net2   net2       net1       t38   --      --
0.72s     aggr1 net2   net2       net1       t38   0.38ms 0.42ms
0.76s     aggr1 net0   net0       net2       t39   --      --
0.76s     aggr1 net0   net0       net2       t39   0.33ms 0.38ms
0.87s     aggr1 net1   net1       net2       t39   --      --
0.87s     aggr1 net1   net1       net2       t39   0.32ms 0.38ms
1.95s     aggr1 net2   net2       net0       t39   --      --
1.95s     aggr1 net2   net2       net0       t39   0.36ms 0.42ms
1.97s     aggr1 net2   net2       net1       t39   --      --
1.97s     aggr1 net2   net2       net1       t39   0.32ms 0.38ms
1.99s     aggr1 net0   net0       net1       t40   --      --
1.99s     aggr1 net0   net0       net1       t40   0.31ms 0.36ms
2.12s     aggr1 net1   net1       net0       t40   --      --
2.12s     aggr1 net1   net1       net0       t40   0.34ms 0.40ms
```

```
2.14s  aggr1 net0          net0          net2    t40  --  --
```

IP インタフェースが構成されたアグリゲーション `aggr0` が作成されます。VNIC `vnic1` はアグリゲーション `aggr0` 上に構成されます。プローブベースの障害検出は、プローブのソース IP アドレスおよびターゲット IP アドレスを指定せずに構成されています。プローブを有効にするために、ルーティングテーブル内のターゲットは、指定された IP アドレス `10.10.10.1` と同じサブネット上に IP アドレス `10.10.10.2` を使用して構成されます。ICMP プローブおよび推移的プローブの統計情報がモニターされます。

トランクアグリゲーションと DLMP アグリゲーションの比較

このセクションでは、2 種類のリンクアグリゲーションの全般的な比較を示します。

表 1 トランクアグリゲーションと DLMP アグリゲーションの機能比較

機能	トランクアグリゲーション	DLMP アグリゲーション
リンクベースの障害検出	サポートされています	サポートされています
LACP	サポートされています	サポートされていません
スタンバイインタフェースの使用	サポートされていません	サポートされていません ¹
複数スイッチへのまたがり	ベンダー独自のソリューションを使用している場合を除き、サポートされていません	サポートされています
スイッチ構成	必要です	必要ありません
負荷分散ポリシー	サポートされています	該当しません
アグリゲーションのすべてのポートにわたる負荷分散	サポートされています	制限されています ²
ユーザー定義フローによるリソース管理	サポートされています	サポートされています
リンク保護	サポートされています	サポートされています
バックツーバック構成	サポートされています	サポートされていません ³

¹ 各 DLMP クライアントは DLMP ポート 1 つのみに関連付けられています。残りのポートは DLMP クライアントの使用可能ポートとして動作しますが、これらの使用可能ポートを構成することはできません。

² アグリゲーションはその VNIC をすべてのポートに分散します。ただし、個々の VNIC が複数のポートに負荷を分散することはできません。

³ DLMP アグリゲーションは、ほかの宛先システムにパケットを送信する場合、常に中間のスイッチを使用する必要があります。ただし、DLMP にはスイッチ構成が必要ありません。

◆◆◆ 3 第 3 章

仮想ローカルエリアネットワークを使用した仮想ネットワークの構成

この章では、仮想ローカルエリアネットワーク (VLAN) について説明し、VLAN を構成および変更する手順についても説明します。

この章の内容は、次のとおりです。

- 49 ページの「VLAN のデプロイの概要」
- 54 ページの「仮想化での VLAN の使用」
- 55 ページの「VLAN 構成の計画」
- 55 ページの「VLAN の構成」
- 60 ページの「リンクアグリゲーション上での VLAN の構成」
- 62 ページの「レガシーデバイス上での VLAN の構成」
- 63 ページの「VLAN 情報の表示」
- 63 ページの「VLAN の変更」
- 66 ページの「VLAN の削除」
- 67 ページの「使用事例: リンクアグリゲーションと VLAN 構成を組み合わせる」
- 70 ページの「ユースケース: VLAN のフローの構成」
- 71 ページの「ユースケース: VLAN VNIC の帯域幅の構成」

VLAN のデプロイの概要

仮想ローカルエリアネットワーク (VLAN) は、ローカルエリアネットワークをプロトコルスタックのデータリンク層で分割したものです。スイッチテクノロジーを使用するローカルエリアネットワークの VLAN を作成できます。同じシステム上のインタフェースを異なる VLAN に割り当てることができます。

Oracle Solaris では、エラスティック仮想スイッチ (EVS) が OpenStack ネットワーキングのバックエンドを形成します。EVS は、VLAN または VXLAN 上の VM インスタンス

ス間の通信を容易にします。Oracle Solaris での OpenStack の詳細は、[『Oracle Solaris 11.3 での OpenStack のインストールと構成』](#)を参照してください。

どのような場合に VLAN を使用するか

次を行う必要がある場合は VLAN をデプロイできます。

- 作業グループの論理的な分割を作成する。
たとえば、ビルのフロアにあるすべてのシステムがスイッチベースの 1 つのローカルネットワークに接続している場合は、フロアの作業グループごとに個別の VLAN を作成できます。
- 作業グループに異なるセキュリティーポリシーを適用する。
たとえば、財務部門と情報技術部門のセキュリティー要件はかなり異なります。部門ごとに個別の VLAN を作成し、VLAN ベースで適切なセキュリティーポリシーを強制できます。
- ブロードキャストドメインのサイズが小さくなり、ネットワークの効率が向上します。作業グループを管理可能なブロードキャストドメインに分割できます。
たとえば、25 ユーザーから構成されるブロードキャストドメインでは、ブロードキャストトラフィックが 12 ユーザー専用の場合、それらの 12 ユーザーに個別の VLAN を設定するとトラフィックが減少し、ネットワークの効率を改善できます。

VLAN 名の割り当て

VLAN では、汎用またはカスタマイズされた名前を使用することに明らかな利点があります。以前のリリースでは、VLAN は物理接続点 (PPA) で識別されていたため、ハードウェアベースのデータリンク名と VLAN ID を組み合わせる必要がありました。ただし、Oracle Solaris では、VLAN の識別のためにより意味のある名前を選択できるようになりました。名前は、[『Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理』](#)の「有効なリンク名のための規則」で説明されているデータリンクの命名規則に準拠する必要があります。たとえば、sales0 や marketing1 などのカスタム VLAN 名を割り当てることができます。

VLAN 名は VLAN ID と連携します。ローカルエリアネットワーク内の各 VLAN は、VLAN タグの一部である VLAN ID によって識別されます。VLAN ID は VLAN の構成時に割り当てられます。VLAN をサポートするようにスイッチを構成する場合は、各ポートに VLAN ID を割り当てる必要があります。ポートの VLAN ID は、そのポートに接続するインタフェースに割り当てられた VLAN ID と同じでなければなりません。

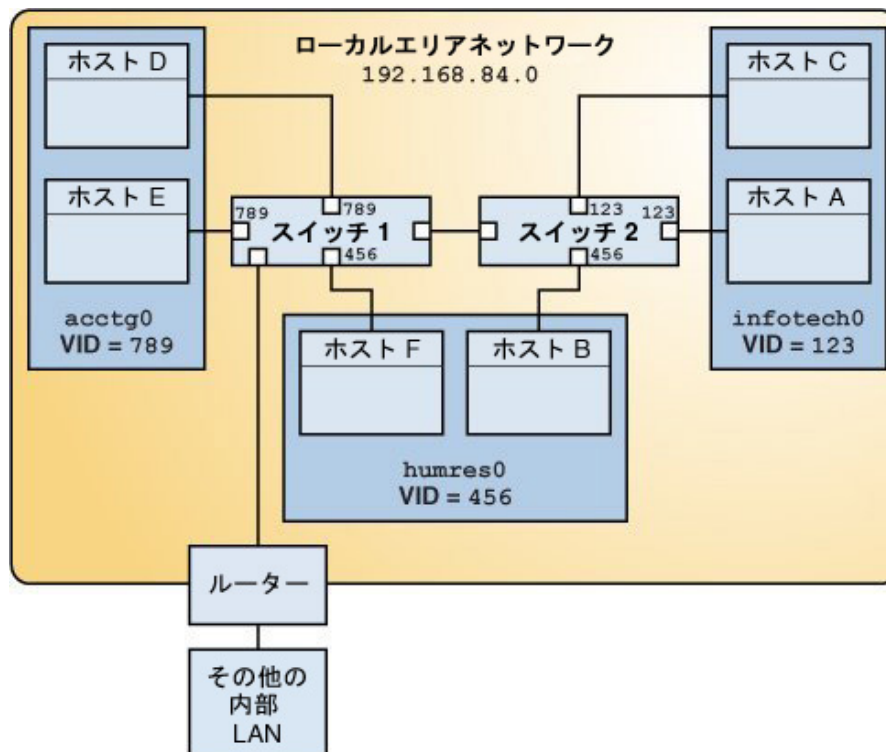
デフォルトでは、各ポートにはポート VLAN ID と呼ばれる VLAN ID があります。この VLAN ID に属するパケットは VLAN タグでタグ付けされません。Oracle Solaris では、データリンクのプロパティ `default_tag` を使用してインタフェース上のポート VLAN ID を表示および変更できます。

VLAN トポロジ

スイッチ LAN テクノロジを使用すると、ローカルネットワーク上のシステムを VLAN に編成できます。ローカルネットワークを VLAN に分割する前に、VLAN テクノロジをサポートするスイッチを入手する必要があります。VLAN トポロジに応じて、スイッチ上のすべてのポートで単一の VLAN を処理するか、複数の VLAN を処理するように構成できます。スイッチのポートを構成する手順はスイッチの製造元によって異なります。Oracle Switch ES1-24 は、VLAN テクノロジをサポートします。Oracle Switch ES1-24 上で VLAN を構成する詳細については、[Sun Ethernet Fabric Operating System, VLAN Administration Guide](#) を参照してください。

次の図は、3 つの VLAN に分割されているローカルエリアネットワークを示しています。

図 6 3つの VLAN を含むローカルエリアネットワーク



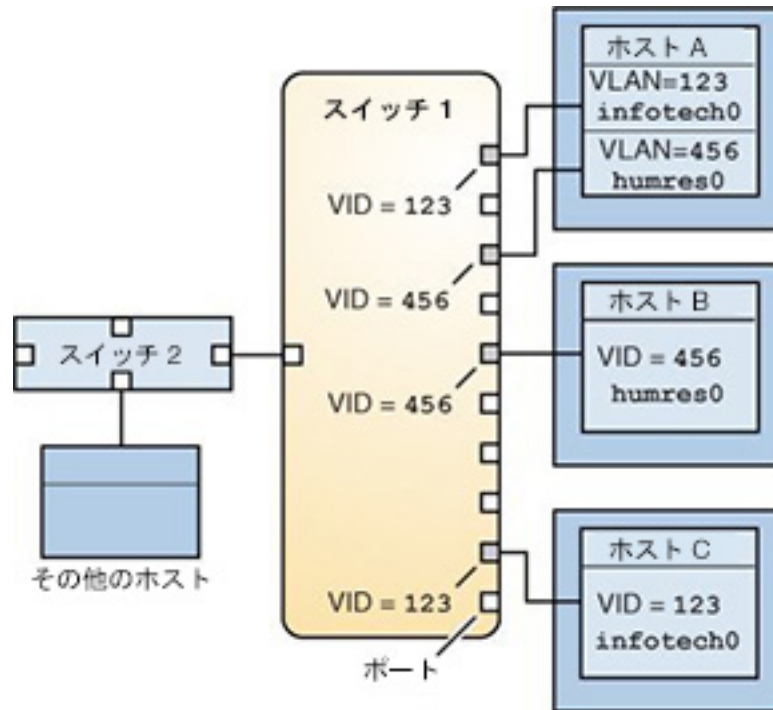
この図の LAN のサブネットアドレスは 192.168.84.0 です。

この LAN は、3つの作業グループに対応する3つの VLAN に分割されています。

- acctg0 (VLAN ID 789) – 経理グループ。このグループはホスト D とホスト E を所有しています。
- humres0 (VLAN ID 456) – 人事グループ。このグループはホスト B とホスト F を所有しています。
- infotech0 (VLAN ID 123) – 情報技術グループ。このグループはホスト A とホスト C を所有しています。

前の図の変化形を次の図に示します。ここではスイッチが1つだけ使用され、さまざまな VLAN に属している複数のホストがその単一のスイッチに接続します。

図 7 異なる VLAN の複数のホストを接続するスイッチ



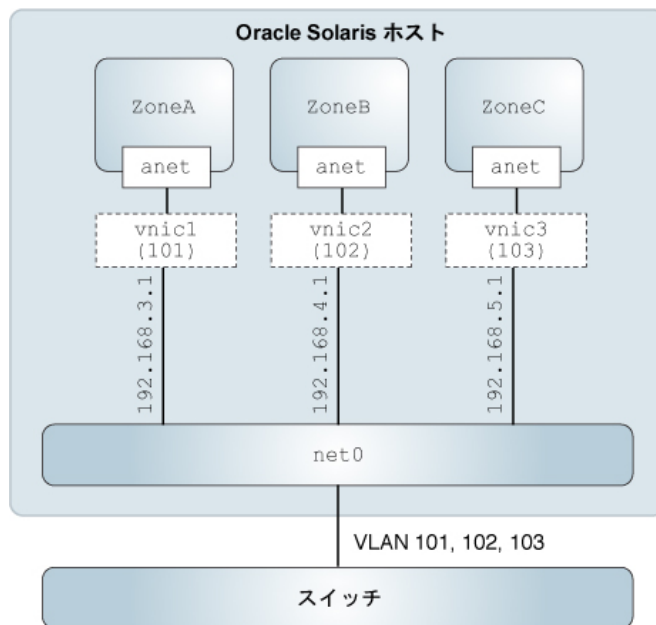
この図で、ホスト A およびホスト C は、VLAN ID 123 を持つ情報技術 VLAN に属しています。ホスト A のインターフェースの 1 つに VLAN ID 123 が構成されています。このインターフェースはスイッチ 1 のポート 1 に接続しており、そのポートにも VLAN ID 123 が構成されています。ホスト B は、VLAN ID 456 を持つ人事 VLAN のメンバーです。ホスト B のインターフェースはスイッチ 1 のポート 5 に接続しており、そのポートには VLAN ID 456 が構成されています。最後に、ホスト C のインターフェースには VLAN ID 123 が構成されています。このインターフェースはスイッチ 1 のポート 9 に接続しています。ポート 9 にも VLAN ID 123 が構成されています。

また、この図は 1 つのホストが複数の VLAN に所属できることも示しています。たとえば、ホスト A にはホストのインターフェースを介して 2 つの VLAN が構成されています。2 番目の VLAN には VLAN ID 456 が構成されており、ポート 3 に接続されています。このポートには VLAN ID 456 が構成されています。したがって、ホスト A は infotech0 VLAN と humres0 VLAN の両方のメンバーです。

仮想化での VLAN の使用

VLAN と Oracle Solaris ゾーンを組み合わせることにより、スイッチなどの 1 つのネットワーク単位内に複数の仮想ネットワークを構成できます。次の図は、物理インターフェースとして net0 を持つシステムを示しています。ZoneA、ZoneB、および ZoneC は、VLAN ID 101、102、および 103 が、それぞれ各ゾーンに割り当てられた状態でシステム内に構成されています。

図 8 ゾーンを持つ VLAN



VNIC データリンクは、`dladm create-vnic` コマンドを使用して作成します。VLAN データリンクは、`dladm create-vlan` コマンドを使用して作成します。

この図に示すように VLAN を構成する方法を確認するには、[例16「ゾーンを持つ VLAN の構成」](#)を参照してください。VLAN VNIC の詳細は、『[Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理](#)』の「[VNIC を VLAN として構成する方法](#)」および『[Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理](#)』の「[VNIC の VLAN ID の変更](#)」を参照してください。

VLAN 構成の計画

VLAN 構成の計画には次の手順が含まれます。

1. LAN のトポロジを調べて、VLAN への分割が適切かどうかを判断します。
このようなトポロジの基本例については、[図6](#)を参照してください。
2. VLAN ID の番号指定スキームを作成し、各 VLAN に VLAN ID を割り当てます。

注記 - VLAN 番号指定スキームがネットワーク上にすでに存在する場合は、既存の VLAN 番号指定スキームに従って VLAN ID を作成する必要があります。

3. 各システム上で、特定の VLAN のコンポーネントにするインタフェースを決定します。
 - a. `dladm show-link` コマンドを使用して、システム上に構成されているリンクを判別します。
 - b. システム上の各データリンクに関連付ける VLAN ID を決定します。
 - c. VLAN を作成します。
4. ネットワークのスイッチへのデータリンクの接続を確認します。
各データリンクの VLAN ID と各インタフェースが接続されているスイッチポートを書き留めます。
5. スイッチの各ポートに、そのポートが接続されているインタフェースと同じ VLAN ID を構成します。
構成手順については、スイッチの製造元のドキュメントを参照してください。

VLAN の構成

次の手順では、`dladm` コマンドを使用してデータリンク上に VLAN を作成する方法について説明します。`ipadm` コマンドを使用して、IP インタフェースを VLAN 上に作成し、そのインタフェースに IP アドレスを構成できます。`dladm` および `ipadm` コマンドの詳細は、[dladm\(1M\)](#) および [ipadm\(1M\)](#) のマニュアルページを参照してください。

VLAN トラフィックをホストするように VLAN ID を持つ VNIC を構成することもできます。詳細については、『[Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理](#)』の「[VNIC を VLAN として構成する方法](#)」を参照してください。

▼ VLAN を構成する方法

始める前に この手順では、ゾーンがすでにシステム上に作成されていることを前提としています。ゾーン構成については、『Oracle Solaris ゾーンの実成と使用』の第1章、「非大域ゾーンの計画および構成方法」を参照してください。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. システムで使用されているリンクのタイプを判定します。

```
# dladm show-link
```

3. データリンク上に VLAN リンクを作成します。

```
# dladm create-vlan -l link -v vid VLAN-link
```

link VLAN インタフェースの作成に使用するリンクを指定します。

vid VLAN ID 番号を示します。

VLAN-link VLAN の名前を指定します。意味のあるカスタム名も使用できます。VLAN 名については、50 ページの「VLAN 名の割り当て」を参照してください。

4. VLAN 構成を確認します。

```
# dladm show-vlan
```

5. VLAN 上に IP インタフェースを作成します。

```
# ipadm create-ip interface
```

ここで、*interface* は VLAN 名を指定します。

6. IP インタフェースに IP アドレスを構成します。

```
# ipadm create-addr -a address interface
```

例 14 VLAN の作成

この例では、図6に示されている VLAN 構成を作成する方法を示します。

1. 使用可能なリンクを確認して特定のリンク上に VLAN を作成します。

```
# dladm show-link
LINK      CLASS    MTU      STATE    OVER
net0      phys     1500     up       --
```



```
net1   phys   1500   up    --
net2   phys   1500   up    --
```

2. ホスト A:

```
# dladm create-vlan -l net0 -v 123 infotech0
```

ホスト C:

```
# dladm create-vlan -l net0 -v 123 infotech0
```

ホスト F:

```
# dladm create-vlan -l net0 -v 456 humres0
```

ホスト B:

```
# dladm create-vlan -l net0 -v 456 humres0
```

ホスト D:

```
# dladm create-vlan -l net0 -v 789 acctg0
```

ホスト E:

```
# dladm create-vlan -l net0 -v 789 acctg0
```

3. 作成された VLAN を表示します。

```
# dladm show-vlan
```

LINK	VID	SVID	PVLAN-TYPE	FLAGS	OVER
infotech0	123	--	--	----	net0
infotech0	123	--	--	----	net0
humres0	456	--	--	----	net0
humres0	456	--	--	----	net0
acctg0	789	--	--	----	net0
acctg0	789	--	--	----	net0

例 15 ゾーンの anet リソースを使用した VLAN の構成

この例では、ゾーンの anet リソースを使用した VLAN の構成を示します。この例では、システム内に異なるゾーンがすでに構成されていることを前提にしています。ゾーンの構成の詳細は、『[Oracle Solaris ゾーンの作成と使用](#)』を参照してください。

1. VLAN ID 11 および 12 をそれぞれ ZoneA および ZoneB に割り当てます。

```
solaris#zonecfg -z ZoneA
zonecfg:ZoneA> add anet
zonecfg:ZoneA:anet> set linkname=net0
zonecfg:ZoneA:anet> set lower-link=net0
```

```

zonecfg:ZoneA:anet> set vlan-id=11
zonecfg:ZoneA:anet> set allowed-address=192.168.10.10/24
zonecfg:ZoneA:anet> set defrouter=192.168.6.10
zonecfg:ZoneA:net> end

```

```

solaris#zonecfg -z ZoneB
zonecfg:ZoneA> add anet
zonecfg:ZoneA:anet> set linkname=net1
zonecfg:ZoneA:anet> set lower-link=net1
zonecfg:ZoneA:anet> set vlan-id=12
zonecfg:ZoneA:anet> set allowed-address=192.168.11.10/24
zonecfg:ZoneA:anet> set defrouter=192.168.7.11
zonecfg:ZoneA:anet> end

```

2. ゾーンに割り当てられた VLAN ID を表示します。

```

solaris:~# dladm show-vnic
LINK          OVER          SPEED  MACADDRESS    MACADDRTYPE  IDS
ZoneA/anet    net0           1000   2:8:20:47:8c:85  random        VID:11
ZoneB/anet    net1           1000   2:8:20:47:8c:85  random        VID:12

```

注記 - VLAN もスイッチ上で構成されていることを確認してください。

例 16 ゾーンを持つ VLAN の構成

この例では、[図8](#)に示されている VLAN 構成を作成する方法を示します。この例では、システム内に異なるゾーンがすでに構成されていることを前提にしています。ゾーンの構成の詳細は、『[Oracle Solaris ゾーン の作成と使用](#)』を参照してください。

1. まず VLAN の構成に使用できるリンクを確認し、特定のリンク上に VLAN VNIC を作成します。

```

root@solaris# dladm show-link
LINK    CLASS    MTU    STATE    OVER
net0    phys     1500   up       --
net1    phys     1500   up       --
net2    phys     1500   up       --

root@solaris# dladm create-vnic -v 101 -l net0 vnic1
root@solaris# dladm create-vnic -v 102 -l net0 vnic2
root@solaris# dladm create-vnic -v 103 -l net0 vnic3
root@solaris# dladm show-vnic
LINK    OVER    SPEED  MACADDRESS    MACADDRTYPE  IDS
vnic1   net0    1000   2:8:20:35:b:9a  random        VID:101
vnic2   net0    1000   2:8:20:fa:94:57  random        VID:102

```

```
vnic3    net0    1000    2:8:20:51:1c:4a    random    VID:103
```

リンク情報を表示すると、これらの VLAN がリストに含まれています。

```
root@solaris# dladm show-link
LINK      CLASS  MTU    STATE  OVER
net0      phys   1500   up     --
vnic1     vnic   1500   up     net0
vnic2     vnic   1500   up     net0
vnic3     vnic   1500   up     net0
```

2. VLAN ID 101、102、および 103 を、それぞれ ZoneA、ZoneB、および ZoneC に指定します。

```
root@solaris# zonecfg -z ZoneA
zonecfg:ZoneA> add net
zonecfg:ZoneA:net> set physical=vnic1
zonecfg:ZoneA:net> end
zonecfg:ZoneA> verify
zonecfg:ZoneA> commit
zonecfg:ZoneA> exit
root@solaris# zoneadm -z ZoneA reboot
```

```
root@solaris# zonecfg -z ZoneB
zonecfg:ZoneB> add net
zonecfg:ZoneB:net> set physical=vnic2
zonecfg:ZoneB:net> end
zonecfg:ZoneB> verify
zonecfg:ZoneB> commit
zonecfg:ZoneB> exit
root@solaris# zoneadm -z ZoneB reboot
```

```
root@solaris# zonecfg -z ZoneC
zonecfg:ZoneC> add net
zonecfg:ZoneC:net> set physical=vnic3
zonecfg:ZoneC:net> end
zonecfg:ZoneC> verify
zonecfg:ZoneC> commit
zonecfg:ZoneC> exit
root@solaris# zoneadm -z ZoneC reboot
```

3. ゾーンに割り当てられた VLAN VNIC を表示します。

```
root@solaris:~# dladm show-vnic
LINK          OVER          SPEED  MACADDRESS      MACADDRTYPE  IDS
ZoneA/vnic1   net0          1000   2:8:20:47:8c:85  random       VID:101
ZoneB/vnic2   net0          1000   2:8:20:47:8c:85  random       VID:102
```

```
ZoneC/vnic3      net0          1000    2:8:20:47:8c:85    random    VID:103
```

4. 各非大域ゾーンにログインして、VLAN に IP アドレスを構成します。

```
root@solaris:~# zlogin ZoneA
root@ZoneA:~# ipadm create-ip vnic1
root@ZoneA:~# ipadm create-addr -a 192.168.3.1 vnic1
vnic1/v4

root@solaris:~# zlogin ZoneB
root@ZoneB:~# ipadm create-ip vnic2
root@ZoneB:~# ipadm create-addr -a 192.168.4.1 vnic2
vnic2/v4

root@solaris:~# zlogin ZoneC
root@ZoneC:~# ipadm create-ip vnic3
root@ZoneC:~# ipadm create-addr -a 192.168.5.1 vnic3
vnic3/v4
```

注記 - VLAN もスイッチ上で構成されていることを確認してください。

すべての VLAN に IP アドレスが構成されたら、構成は完了です。3 つの VLAN は動作しており、それぞれのゾーンのトラフィックをホストできます。

リンクアグリゲーション上での VLAN の構成

インタフェース上の VLAN 構成と同様の方法でリンクアグリゲーション上に VLAN を作成できます。リンクアグリゲーションについては、[第2章「リンクアグリゲーションを使用した高可用性の構成」](#)で説明されています。このセクションでは、VLAN とリンクアグリゲーションの構成について説明します。

▼ リンクアグリゲーション上に VLAN を構成する方法

始める前に リンクアグリゲーションを作成します。リンクアグリゲーションの作成方法については、[30 ページの「リンクアグリゲーションを作成する方法」](#)を参照してください。

1. 管理者になります。

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. システムで構成されているリンクアグリゲーションを一覧表示します。

```
# dladm show-aggr
```

3. 選択したリンクアグリゲーション上に作成する VLAN ごとに、次のコマンドを使用できます。

```
# dladm create-vlan -l link -v vid VLAN-link
```

link VLAN インタフェースの作成に使用するリンクを指定します。

注記 - この手順では、リンクはリンクアグリゲーションを指します。

vid VLAN ID 番号を示します。

VLAN-link VLAN の名前を指定します。

4. 前の手順で作成した VLAN ごとに、VLAN 上に IP インタフェースを作成します。

```
# ipadm create-ip interface
```

ここで、*interface* は VLAN 名を使用します。

5. VLAN 上の各 IP インタフェースに、有効な IP アドレスを構成します。

```
# ipadm create-addr -a address interface
```

例 17 リンクアグリゲーション上に複数の VLAN を構成する

この例では、リンクアグリゲーション上に 2 つの VLAN を構成します。VLAN には VLAN ID 193 と 194 がそれぞれ割り当てられます。

```
# dladm show-link
LINK    CLASS    MTU     STATE    OVER
net0    phys     1500    up       --
net1    phys     1500    up       --
aggr0   aggr     1500    up       net0 net1

# dladm create-vlan -l aggr0 -v 193 acctg0
# dladm create-vlan -l aggr0 -v 194 humres0

# ipadm create-ip acctg0
# ipadm create-ip humres0

# ipadm create-addr -a 192.168.10.0/24 acctg0
ipadm: acctg0/v4
# ipadm create-addr -a 192.168.20.0/24 humres0
ipadm: humres0/v4
```

レガシーデバイス上での VLAN の構成

特定のレガシーデバイスは、最大転送単位 (MTU) サイズ (フレームサイズとも呼ばれる) が最大 1514 バイトのパケットのみを処理します。フレームサイズがこの上限を超えるパケットは破棄されます。このような場合は、[56 ページの「VLAN を構成する方法」](#)に示されているのと同じ手順に従います。ただし、VLAN を作成するときは、VLAN を強制的に作成するために `-f` オプションを使用します。

▼ レガシーデバイス上で VLAN を構成する方法

1. 管理者になります。

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. `-f` オプションを使用して VLAN を作成します。

```
# dladm create-vlan -f -l link -v vid VLAN-link
```

`-f` VLAN を強制的に作成します。このオプションは、VLAN ヘッダーを含めることができる大きさのフレームサイズが許可されていないデバイスに VLAN を作成する場合に使用します。

`-l link` VLAN インタフェースの作成に使用するリンクを指定します。この手順では、リンクはレガシーデバイスを指します。

`-v vid` VLAN ID 番号を示します。

`VLAN-link` VLAN の名前を指定します。管理用に選択された名前を指定することもできます。

3. 最大転送単位 (MTU) に小さめのサイズを設定します。

次の例では、`mtu` は 1496 に設定されます。

```
# dladm set-linkprop -p mtu=1496 VLAN-link
```

MTU 値を小さくすることによって、リンク層で転送前に VLAN ヘッダーを挿入するための領域が得られます。

4. 手順 2 を繰り返して VLAN 内にあるノードごとに MTU 値を設定します。

リンクプロパティ値の変更の詳細は、『[Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理](#)』の「[データリンクプロパティの管理](#)」を参照してください。

VLAN 情報の表示

VLAN はデータリンクなので、`dladm show-link` コマンドを使用して VLAN に関する情報を表示できます。VLAN に関する特定の情報を表示するには、`dladm show-vlan` コマンドを使用します。

次の例は、`dladm show-link` または `dladm show-vlan` コマンドを使用して取得できるタイプの情報を比較します。最初の例では `dladm show-link` コマンドを使用して、システム上のすべてのデータリンクが、VLAN でないものも含めて表示されます。2 番目の例では `dladm show-vlan` コマンドを使用して、データリンク情報のうち、VLAN に関連するものだけが表示されます。

```
# dladm show-link
LINK      CLASS  MTU    STATE  OVER
net0     phys   1500   up     --
net1     phys   1500   up     --
net2     phys   1500   up     --
web1     vlan   1500   up     net0
auth1    vlan   1500   up     net0
app1     vlan   1500   up     net0
web2     vlan   1500   up     net1
auth2    vlan   1500   up     net1
app2     vlan   1500   up     net1
web3     vlan   1500   up     net2
auth3    vlan   1500   up     net2

# dladm show-vlan
LINK      VID    SVID    PVLAN-TYPE  FLAGS  OVER
web1      111    --      --          ----   net0
auth1     112    --      --          ----   net0
app1      113    --      --          ----   net0
web2      111    --      --          ----   net1
auth2     112    --      --          ----   net1
app2      113    --      --          ----   net1
web3      111    --      --          ----   net2
auth3     113    --      --          ----   net2
```

VLAN の変更

次の方法で `dladm modify-vlan` コマンドを使用して、VLAN を変更できます。

- VLAN の VLAN ID を変更します
- 別のベースとなるリンクに VLAN を移行します

VLAN の VLAN ID の変更

VLAN の VLAN ID を変更するには、次のいずれかのコマンドを使用します。

- `dladm modify-vlan -v vid -L datalink`

ここで、`vid` は、VLAN に割り当てる新しい VLAN ID を指定します。`datalink` は、VLAN が構成されているベースとなるリンクを示します。

注記 - データリンク上に存在する VLAN が 1 つのみである場合のみ、`dladm modify-vlan -v vid -L datalink` コマンド構文を使用できます。複数の VLAN が構成されているデータリンクにこのコマンドを使用すると、データリンク上の各 VLAN には一意の VLAN ID が必要なので、コマンドは失敗します。

リンクの VLAN ID を変更する場合は、新しい VLAN ID のスイッチポートも構成する必要があります。

- `dladm modify-vlan -v vid vlan`

このコマンドは、単一のデータリンク上にある複数の VLAN の一意の VLAN ID を変更する場合に使用します。データリンク上の各 VLAN には一意の VLAN ID があるため、VLAN ID は一度に 1 つずつ変更する必要があります。図8に示されている設定では、`net0` 上に構成されている `web1`、`auth1`、および `app1` の VLAN ID を次のように変更します。

```
# dladm modify-vlan -v 123 web1
# dladm modify-vlan -v 456 app1
# dladm modify-vlan -v 789 auth1
```

別のベースとなるリンクへの VLAN の移行

VLAN の削除や再構成を行わずに、VLAN をベースとなるデータリンクから別のベースとなるデータリンクに移行します。ベースとなるリンクは、物理リンク、リンクアグリゲーション、または `etherstub` です。`etherstub` の詳細は、『Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理』の「ネットワーク仮想化コンポーネント」を参照してください。

VLAN を正常に移行するには、VLAN の移動先であるベースとなるデータリンクが、VLAN のデータリンクプロパティに対応できる必要があります。これらのプロパティがサポートされていない場合、移行は失敗し、ユーザーに通知されます。移行が正常に行われたあとは、VLAN がネットワークに接続されたままであれば、その VLAN を使用しているすべてのアプリケーションが通常の動作を継続します。

ハードウェアに依存する特定のプロパティは、VLAN の移行後に変更される場合があります。たとえば、VLAN は常に、そのベースとなるデータリンクと同じ MAC アドレスを共有します。したがって、VLAN を移行すると、その VLAN の MAC アドレスは、ターゲットデータリンクのプライマリ MAC アドレスに変更されます。ほかにも、データリンク状態、リンク速度、および MTU サイズのプロパティが影響を受ける可能性があります。ただし、アプリケーションは中断されることなく動作を継続します。

注記 - 移行された VLAN では、元のデータリンクのハードウェアレン統計は一切保持されません。ターゲットデータリンク上で VLAN に使用できるハードウェアレンが、統計情報の新しい取得元になります。ただし、`dlstat` コマンドによってデフォルトで表示されるソフトウェア統計は保持されます。

VLAN の移行は、グローバルに実行することも選択的に実行することもできます。

グローバルな移行

グローバルな移行は、あるデータリンク上で構成されているすべての VLAN を別のデータリンクに移行するために使用されます。グローバルな移行を実行するには、ソースデータリンクとターゲットデータリンクを指定するだけで済みます。次の例では、`ether0` 上のすべての VLAN を `net1` に移動する方法を示しています。

```
# dladm modify-vlan -l net1 -L ether0
```

-l VLAN の移行先であるターゲットデータリンクを表します。

-L VLAN が構成されている元のデータリンクを表します。

注記 - 移行元データリンクの前にターゲットデータリンクを指定する必要があります。

選択的移行

選択的移行は、選択された VLAN のみを移行するために使用されます。VLAN の選択的移行を実行する場合は、移動する VLAN を指定します。[図8](#)に基づく次の例では、VLAN は `net0` から `net3` に移動されます。

```
# dladm modify-vlan -l net3 web1,auth1,app1
```

注記 - VLAN を選択的に移行する場合は、`-L` オプションを含めないでください。これはグローバルな移行だけに適用されるオプションです。

移行を実行する間に、VLAN の VLAN ID を変更できます。図8を基にして、次の例では、複数の VLAN を移行すると同時にそれらの VLAN ID を変更する方法を示します。

```
# dladm show-vlan
LINK   VID   SVID   PVLAN-TYPE  FLAGS   OVER
web1   111   --     --           ----- net0
auth1  112   --     --           ----- net0
app1   113   --     --           ----- net0

# dladm modify-vlan -l net3 -v 123 web1
# dladm modify-vlan -l net3 -v 456 auth1
# dladm modify-vlan -l net3 -v 789 app1
# dladm show-vlan
LINK   VID   SVID   PVLAN-TYPE  FLAGS   OVER
web1   123   --     --           ----- net3
auth1  456   --     --           ----- net3
app1   789   --     --           ----- net3
```

注記 - 同等のコマンド `dladm modify-vnic` は、VLAN として構成されている VNIC を移行します。VLAN を移行するのか、VLAN として構成されている VNIC を移行するのかによって、正しいサブコマンドを使用する必要があります。`dladm show-vlan` コマンドで表示される VLAN に対しては、`modify-vlan` サブコマンドを使用します。`modify-vnic` サブコマンドは、`dladm show-vnic` コマンドの出力で表示される VNIC (VLAN ID を持っているものも含む) に対して使用します。VNIC の変更方法については、『[Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理](#)』の「[VNIC の VLAN ID の変更](#)」を参照してください。

VLAN の削除

システム上の VLAN 構成を削除するには、`dladm delete-vlan` コマンドを使用します。

注記 - VLAN を削除する前に、削除しようとしている VLAN 上の既存の IP 構成をすべて削除する必要があります。VLAN 上に IP インタフェースが存在している場合、VLAN の削除は失敗します。

注記 - `dladm delete-phys` コマンドを使用してデータリンクを削除するときは、そのデータリンク上で構成されているすべての VLAN も削除されます。この `dladm delete-phys` コマンドは、データリンク上で構成されているフロー、VNIC、アグリゲーションなどその他のレイヤー 2 コンポーネントも削除します。ただし、IP インタフェースなど、データリンク上で構成されているレイヤー 3 コンポーネントは、`ipadm` コマンドを使用して手動で削除する必要があります。

例 18 VLAN 構成の削除

この例は VLAN 構成を削除する方法を示しています。

```
# dladm show-vlan
LINK      VID   SVID   PVLAN-TYPE  FLAGS   OVER
web1      111   --     --           ----   net0
auth1     112   --     --           ----   net0
app1      113   --     --           ----   net0
web2      111   --     --           ----   net1
auth2     112   --     --           ----   net1
app2      113   --     --           ----   net1
web3      111   --     --           ----   net2
auth3     113   --     --           ----   net2

# ipadm delete-ip web1
# dladm delete-vlan web1
```

SR-IOV に対応するポートでの VLAN タグ付けの使用

SR-IOV をサポートしている一部の Intel ネットワークアダプタの使用時に、仮想機能が悪意のある動作のターゲットになることがあります。ソフトウェアにより生成された予期しないフレームによってホストと仮想スイッチの間のトラフィックが抑制され、パフォーマンスに悪影響を与えることがあります。悪意のある可能性がある予期しないフレームをドロップするには、SR-IOV に対応するすべてのポートで VLAN タグ付けを使用するように構成します。詳細は、『[Oracle VM Server for SPARC 3.3 管理ガイド](#)』を参照してください。

使用事例: リンクアグリゲーションと VLAN 構成を組み合わせる

このセクションでは、リンクアグリゲーションおよび VLAN を使用するネットワーク構成の組み合わせを作成する方法を例で示します。

次の例では、4 枚の NIC を使用するシステムを、8 つの個別のサブネットのルーターとして構成する必要があります。したがって、サブネットごとに 1 つずつ、8 つのリンクが構成されます。最初に、4 枚のすべての NIC 上でトランクアグリゲーションが作成されます。VLAN タグが送信フレームに含まれないこのタグなしリンクが、デフォルトルートが指すネットワークのデフォルトのタグなしサブネットになります。

次に、ほかのサブネットのために、リンクアグリゲーション上で VLAN インタフェースが構成されます。これらのサブネットは、色分けされたスキームに基づいて名前が付けられます。それに応じて、VLAN 名も同様に、それぞれ対応するサブネットに

従って名前が付けられます。最終的な構成は、8つのサブネットに対する8つのリンク(1つのタグなしリンクと7つのタグ付き VLAN リンク)で構成されています。この例は、データリンク上に IP インタフェースがすでに存在しているかどうかの確認から始まります。データリンクをアグリゲーションに結合する前に、これらのインタフェースを削除する必要があります。

1. データリンク上に構成されている IP インタフェースをすべて削除します。

```
# ipadm show-if
IFNAME    CLASS      STATE  ACTIVE  OVER
lo0       loopback  ok     yes     --
net0      ip        ok     yes     --
net1      ip        ok     yes     --
net2      ip        ok     yes     --
net3      ip        ok     yes     --
```

```
# ipadm delete-ip net0
# ipadm delete-ip net1
# ipadm delete-ip net2
# ipadm delete-ip net3
```

2. トランクアグリゲーション default0 を作成します。

```
# dladm create-aggr -P L2,L3 -l net0 -l net1 -l net2 -l net3 default0
```

```
# dladm show-link
LINK      CLASS      MTU   STATE  OVER
net0      phys      1500  up     --
net1      phys      1500  up     --
net2      phys      1500  up     --
net3      phys      1500  up     --
default0  aggr      1500  up     net0 net1 net2 net3
```

3. アグリゲーション上に IP インタフェースを構成します。

```
# ipadm create-ip default0
# ipadm create-addr -a 10.2.3.4/24 default0
```

4. default0 上に VLAN を作成します。

```
# dladm create-vlan -v 2 -l default0 orange0
# dladm create-vlan -v 3 -l default0 green0
# dladm create-vlan -v 4 -l default0 blue0
# dladm create-vlan -v 5 -l default0 white0
# dladm create-vlan -v 6 -l default0 yellow0
# dladm create-vlan -v 7 -l default0 red0
# dladm create-vlan -v 8 -l default0 cyan0

# dladm show-link
```

```
LINK      CLASS  MTU  STATE  OVER
net0      phys   1500 up      --
net1      phys   1500 up      --
net2      phys   1500 up      --
net3      phys   1500 up      --
default0  aggr   1500 up      net0 net1 net2 net3
orange0   vlan   1500 up      default0
green0    vlan   1500 up      default0
blue0     vlan   1500 up      default0
white0    vlan   1500 up      default0
yellow0   vlan   1500 up      default0
red0      vlan   1500 up      default0
cyan0     vlan   1500 up      default0
```

```
# dladm show-vlan
```

```
LINK      VID  SVID  PVLAN-TYPE  FLAGS  OVER
orange0   2    --    --          -----  default0
green0    3    --    --          -----  default0
blue0     4    --    --          -----  default0
white0    5    --    --          -----  default0
yellow0   6    --    --          -----  default0
red0      7    --    --          -----  default0
cyan0     8    --    --          -----  default0
```

5. VLAN リンク上に IP インタフェースを作成し、IP アドレスをインタフェースに割り当てます。

```
# ipadm create-ip orange0
# ipadm create-ip green0
# ipadm create-ip blue0
# ipadm create-ip white0
# ipadm create-ip yellow0
# ipadm create-ip red0
# ipadm create-ip cyan0

# ipadm create-addr -a 10.2.3.5/24 orange0
# ipadm create-addr -a 10.2.3.6/24 green0
# ipadm create-addr -a 10.2.3.7/24 blue0
# ipadm create-addr -a 10.2.3.8/24 white0
# ipadm create-addr -a 10.2.3.9/24 yellow0
# ipadm create-addr -a 10.2.3.10/24 red0
# ipadm create-addr -a 10.2.3.11/24 cyan0
```

ユースケース: VLAN のフローの構成

このセクションでは、ゾーンに割り当てられた VLAN のフローを構成する方法を示す例を提供します。このゾーンでは、生成、レプリケーション、バックアップなどの複数のサービスが同じ VLAN リンク経由で実行されています。この例は、フローを使用したこれらのサービス間でのリンクの帯域幅のプロビジョニングを示しています。

1. 使用可能なリンクを確認し、特定のリンク上に VLAN VNIC を作成します。

```
root@solaris:~# dladm show-link
LINK      CLASS    MTU     STATE   OVER
net0      phys    1500    up      --
net1      phys    1500    up      --
net2      phys    1500    up      --

root@solaris:~# dladm create-vnic -v 101 -l net0 vnic1
```

2. zone1 という名前のゾーンを作成し、それに VLAN VNIC を割り当てます。

```
root@solaris# zonecfg -z zone1
Use 'create' to begin configuring a new zone.
zonecfg:zone1> create -t SYSsolaris
zonecfg:zone1> add net
zonecfg:zone1:net> set physical=vnic1
zonecfg:zone1:net> end
zonecfg:zone1> verify
zonecfg:zone1> commit
zonecfg:zone1> exit
root@solaris# zoneadm -z zone1 boot
```

3. ゾーンにログインし、VLAN VNIC に IP アドレスを割り当てます。

```
root@solaris# zlogin zone1
root@zone1# ipadm create-ip vnic1
root@zone1:~# ipadm create-addr -a 192.168.110.1 vnic1
```

4. ゾーン上のさまざまなサービスのフローを構成します。

```
root@zone1:~# flowadm add-flow -a local_ip=192.168.110.1,local_port=80 \
-p maxbw=500M,priority=high -l vnic1 flow_prod
root@zone1:~# flowadm add-flow -a local_ip=192.168.110.1,local_port=138 \
-p maxbw=250M,priority=medium -l vnic1 flow_repl
root@zone1:~# flowadm add-flow -a local_ip=192.168.110.1,local_port=21 \
-p maxbw=250M,priority=low -l vnic1 flow_bkup
```

複数のサービスが同じ VLAN 上で実行されているため、フローを使用した帯域幅のプロビジョニングがネットワークリソースを効率的に管理するのに役立ちます。

5. 構成されているフローを表示します。

```

root@zone1:~# flowadm
FLOW      LINK      PROTO  LADDR          LPORT  RADDR  RPORT  DSFLD
flow_prod vnic1    --     192.168.110.1  80     --     --     --
flow_repl vnic1    --     192.168.110.1  138    --     --     --
flow_bkup vnic1    --     192.168.110.1  21     --     --     --

```

6. フローに対して設定されているプロパティを表示します。

```

root@zone1:~# flowadm show-flowprop
FLOW      PROPERTY      PERM  VALUE          DEFAULT      POSSIBLE
flow_prod maxbw          rw    500            --           --
flow_prod priority       rw    high           medium       low,medium,high
flow_prod hwflow    r-    off           --           on,off
flow_repl maxbw          rw    250            --           --
flow_repl priority       rw    medium        medium       low,medium,high
flow_repl hwflow    r-    off           --           on,off
flow_bkup maxbw          rw    250            --           --
flow_bkup priority       rw    low           medium       low,medium,high
flow_bkup hwflow    r-    off           --           on,off

```

ユースケース: VLAN VNIC の帯域幅の構成

このセクションでは、VLAN VNIC 間で物理リンクの帯域幅をプロビジョニングするために VLAN VNIC に対する帯域幅制限を構成する方法を示す例を提供します。この例では、同じ物理リンクを共有する、異なるゾーンで実行されている3つのサービスが存在します。

1. 使用可能なリンクを確認し、特定のリンク上に VNIC VLAN を作成します。

```

root@solaris# dladm show-link
LINK      CLASS      MTU      STATE      OVER
net0     phys      1500    up         --
root@solaris# dladm create-vnic -v 10 -l net0 vnic_prod
root@solaris# dladm create-vnic -v 11 -l net0 vnic_repl
root@solaris# dladm create-vnic -v 12 -l net0 vnic_bkup
root@solaris# dladm show-vnic
LINK      OVER      SPEED      MACADDRESS          MACADDRTYPE  IDS
vnic_prod net0     1000      2:8:20:35:b:9a     random       VID:10
vnic_repl net0     1000      2:8:20:fa:94:57     random       VID:11
vnic_bkup net0     1000      2:8:20:51:1c:4a     random       VID:12

```

2. すべての VNIC に帯域幅を割り当てます。

```

root@solaris# dladm set-linkprop -p maxbw=16 vnic_prod
root@solaris# dladm set-linkprop -p maxbw=500M vnic_repl

```

```
root@solaris# dladm set-linkprop -p maxbw=200M vnic_bkup
```

3. zone1 という名前のゾーンを作成し、それに VLAN VNIC を割り当てます。

```
root@solaris# zonecfg -z zone1
Use 'create' to begin configuring a new zone.
zonecfg:zone1> create -t SYSsolaris
zonecfg:zone1> add net
zonecfg:zone1:net> set physical=vnic_prod
zonecfg:zone1:net> end
zonecfg:zone1> verify
zonecfg:zone1> commit
zonecfg:zone1> exit
root@solaris# zoneadm -z zone1 boot
```

4. zone1 にログインし、VNIC に IP アドレスを構成します。

```
root@solaris# zlogin zone1
root@zone1# ipadm create-ip vnic_prod
root@zone1:~# ipadm create-addr -a 192.168.110.1 vnic_prod
vnic_prod/v4
root@solaris# zoneadm -z zone1 reboot
```

5. zone2 という名前のゾーンを作成し、それに VLAN VNIC を割り当てます。

```
root@solaris# zonecfg -z zone2
Use 'create' to begin configuring a new zone.
zonecfg:zone2> create -t SYSsolaris
zonecfg:zone2> add net
zonecfg:zone2:net> set physical=vnic_repl
zonecfg:zone2:net> end
zonecfg:zone2> verify
zonecfg:zone2> commit
zonecfg:zone2> exit
root@solaris# zoneadm -z zone2 boot
```

6. zone2 にログインし、VNIC に IP アドレスを構成します。

```
root@solaris# zlogin zone2
root@zone2# ipadm create-ip vnic_repl
root@zone2:~# ipadm create-addr -a 192.168.111.1 vnic_repl
vnic_repl/v4
```

7. zone3 という名前のゾーンを作成し、それに VLAN VNIC を割り当てます。

```
root@solaris# zonecfg -z zone3
Use 'create' to begin configuring a new zone.
zonecfg:zone3> create -t SYSsolaris
zonecfg:zone3> add net
zonecfg:zone3:net> set physical=vnic_bkup
zonecfg:zone3:net> end
```



```
zonecfg:zone3> verify
zonecfg:zone3> commit
zonecfg:zone3> exit
root@solaris# zoneadm -z zone3 boot
```

8. zone3 にログインし、VNIC に IP アドレスを構成します。

```
root@solaris# zlogin zone3
root@zone3# ipadm create-ip vnic_bkup
root@zone3:~# ipadm create-addr -a 192.168.112.1 vnic_bkup
vnic_bkup/v4
```

この例では、ゾーン内の各サービスに 1 つの VLAN VNIC が割り当てられます。物理リンクの帯域幅をプロビジョニングするために VNIC の帯域幅が構成されます。これにより、使用可能なネットワークリソースが効率的に使用されるようになります。

プライベート仮想ローカルエリアネットワークの構成

この章では、プライベート VLAN (PVLAN) について、および PVLAN を構成および変更する手順について説明します。

この章の内容は、次のとおりです。

- 75 ページの「プライベート VLAN の概要」
- 80 ページの「プライベート VLAN の構成」
- 81 ページの「プライベート VLAN の変更」
- 66 ページの「VLAN の削除」
- 82 ページの「ゾーンへの PVLAN の割り当て」

プライベート VLAN の概要

プライベート VLAN (PVLAN) のテクノロジーを使用すると、VLAN をサブ VLAN に分割することによってネットワークトラフィックを隔離できます。通常の VLAN は、単一のブロードキャストドメインです。PVLAN を構成すると、1つのブロードキャストドメインが小さいサブドメインに分割されます。PVLAN を構成する場合、標準 (IEEE 802.1Q) VLAN はプライマリ VLAN と呼ばれ、サブ VLAN はセカンダリ VLAN と呼ばれます。セカンダリ VLAN は、隔離 VLAN またはコミュニティ VLAN のどちらでもかまいません。

■ 隔離 VLAN

隔離 VLAN に関連付けられたポートは、プライマリ VLAN のみと通信でき、その他のセカンダリ VLAN とはできません。隔離 VLAN はプライマリ VLAN ドメイン内に 1 つだけ作成できます。

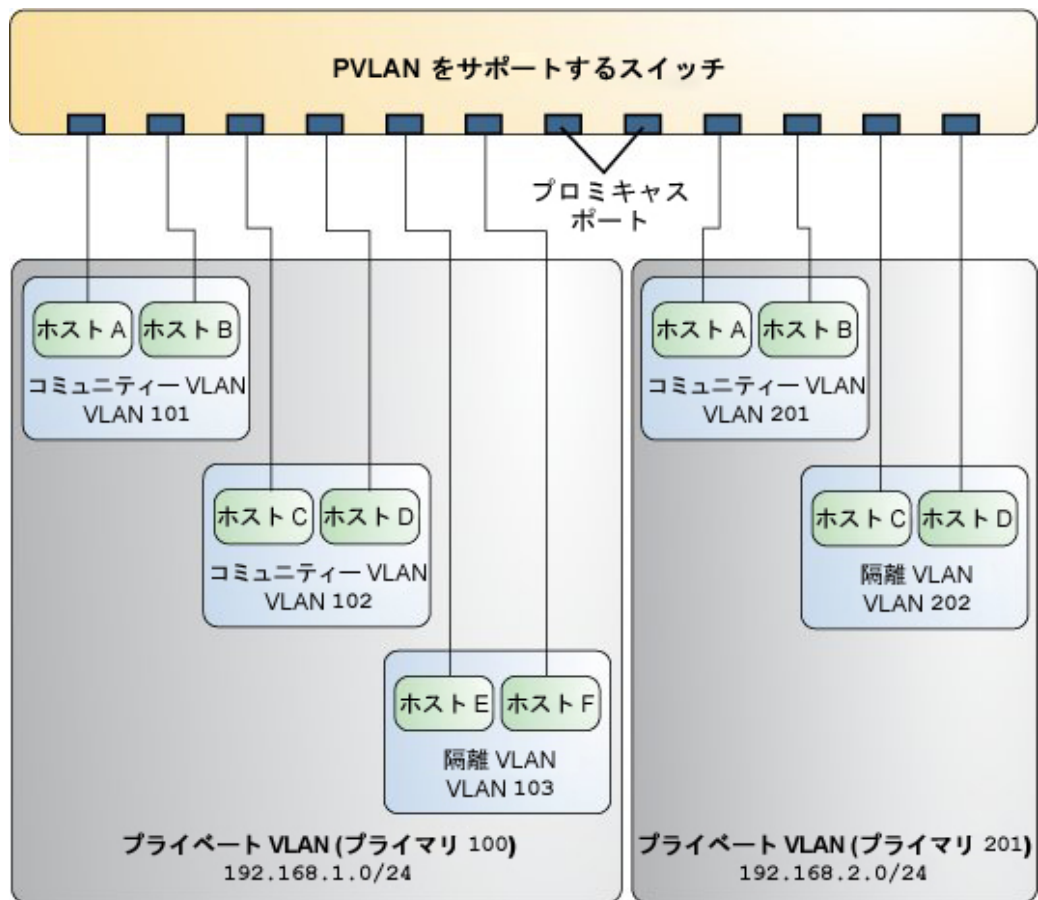
■ コミュニティ VLAN

コミュニティ VLAN に関連付けられたポートは、プライマリ VLAN、および同一のコミュニティ VLAN 内にあるその他のポートと通信できます。コミュニティ VLAN はプライマリ VLAN ドメイン内に複数作成できます。

通常の VLAN については、第3章「仮想ローカルエリアネットワークを使用した仮想ネットワークの構成」を参照してください。

次の図は、PVLAN をサポートするスイッチで構成された2つの PVLAN を持つ単純な PVLAN 構成を示しています。

図 9 プライベート VLAN



この図で、プライマリ VLAN ID が 100 のプライベート VLAN には 3つのセカンダリ VLAN があり、そのセカンダリ VLAN ID は 101、102、および 103 です。セカンダリ VLAN のうち 2つは Host A、Host B、Host C、Host D によるコミュニティー VLAN、もう 1つは Host E および Host F による隔離 VLAN です。

コミュニティ VLAN 101 および 102 に関連付けられたポートは、プライマリ VLAN 100 と通信でき、さらに相互に通信できます。つまり、Host A は Host B と通信でき、Host C は Host D と通信できます。ただし、コミュニティ VLAN 101 内の Host A および Host B は、コミュニティ VLAN 102 内の Host C および Host D と通信できません。隔離 VLAN に関連付けられたポートは、プライマリ VLAN のみと通信でき、相互にはできません。

詳細については、[80 ページの「プライベート VLAN の構成」](#)を参照してください。

PVLAN を使用することの利点

PVLAN の使用には、次の利点があります。

- IP サブネットごとに複数の VLAN を作成できるため、IP アドレスを節約します。
- PVLAN は通常の VLAN よりも多くの L2 分離ネットワークを提供するため、スケーラビリティが向上します。この増加により、VLAN の制限が取り除かれ、最大 4094 の分離されたネットワークを作成できるようになります。
- 通常の VLAN に比べて分離が向上します。

PVLAN ポート

PVLAN は通常の VLAN と同様に複数のスイッチにまたがることができます。トランクポートはプライマリ VLAN またはセカンダリ VLAN からのフレームを搬送します。PVLAN に関連付けられた 2 種類のポートは、プロミスキャストランクポートとセカンダリトランクポートです。

- **プロミスキャストランクポート** – プロミスキャストランクポートから送信されるすべてのフレームは、プライマリ VLAN ID でタグ付けされます。このポートは、最上位のアップリンクポートで構成されます。スイッチは、プライマリ VLAN ID とセカンダリ VLAN ID をマッピングします。
- **PVLAN セカンダリトランクポート** – PVLAN セカンダリトランクポートから送信されるすべてのフレームは、セカンダリ VLAN ID でタグ付けされます。

注記 - PVLAN ポートがレイヤー 2 で分離されている場合でも、外部ポリシーゲートウェイが通信を許可しているかぎり、ポートは引き続きレイヤー 3 で相互に通信できます。

送信トラフィックのタグ付け

Oracle Solaris では、プロミスキャストリンクポートがシステムとスイッチのどちらにあるのかに応じて、PVLAN のタグモードプロパティを設定する必要があります。したがって、PVLAN がスイッチ上で構成されていてセカンダリポートが Oracle Solaris 上で構成されているのか、それとも PVLAN がスイッチ上ではなくシステム上でのみ構成されているのかを識別する必要があります。

プライマリ VLAN ID を持つパケットを送信するには、タグモードを `primary` に設定する必要があります。セカンダリ VLAN ID を持つパケットを送信するには、タグモードを `secondary` に設定する必要があります。スイッチは、セカンダリ VLAN ID をプライマリ VLAN ID に変換します。デフォルトでは、パケットはプライマリ VLAN ID 付きで送信されます。

`tagmode` プロパティを構成して、プライマリ VLAN ID またはセカンダリ VLAN ID で送信トラフィックにタグ付けできます。このプロパティのデフォルト値は、`primary` です。複数の PVLAN スイッチがあり、1 つの PVLAN をこれら複数のスイッチにまたがるようにする場合は、`dladm set-linkprop` コマンドを使用して、プロパティ `tagmode` を `secondary` に設定する必要があります。

例 19 PVLAN 用のタグモードの設定

次の例は、データリンク上のタグモードを変更する方法を示しています。

```
# dladm set-linkprop -p pvlan-tagmode=secondary net0
# dladm show-linkprop -p pvlan-tagmode net0
LINK      PROPERTY      PERM VALUE      EFFECTIVE      DEFAULT      POSSIBLE
net0      pvlan-tagmode rw      secondary      secondary      primary      secondary,
                                         primary
```

注記 - `tagmode` の値は、スイッチが PVLAN をサポートしているかどうかによって異なります。

PVLAN 構成の要件

PVLAN を構成するときは、次の制限事項に注意してください。

- コミュニティー VLAN のプライマリ VLAN ID とセカンダリ VLAN ID は一意である必要があります。

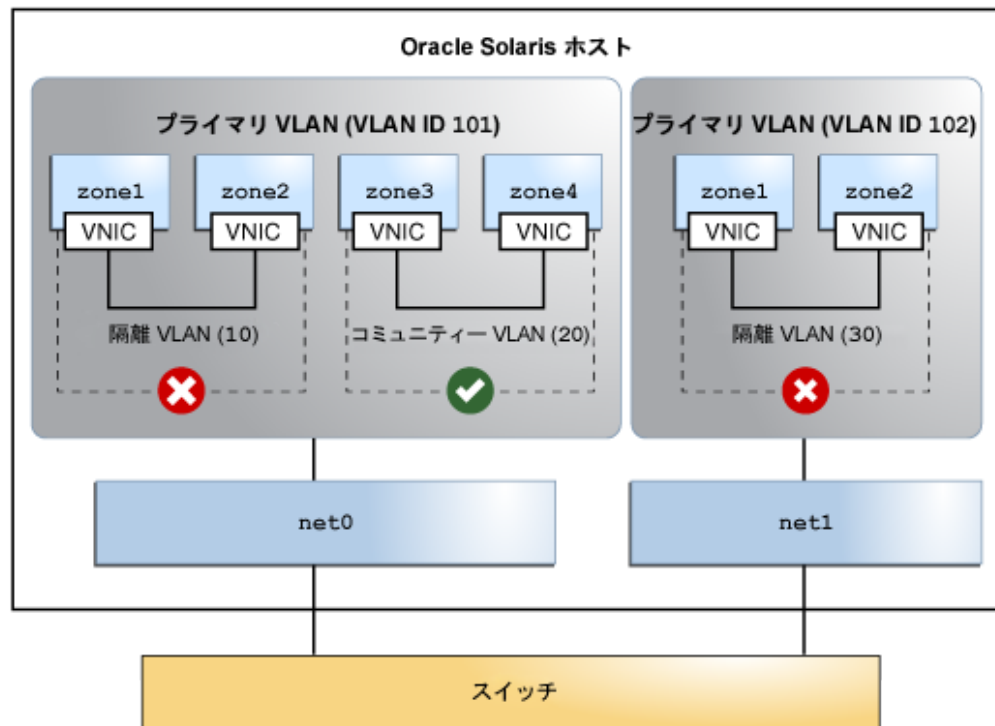
たとえば、3 のプライマリ VLAN ID と 100 のセカンダリ VLAN ID を持つコミュニティ VLAN がある場合は、3 または 100 をセカンダリ VLAN ID として使用する別のコミュニティ VLAN を作成することはできません。つまり、4 と 100 や 4 と 3 といった ID ペアが含まれる組み合わせは無効です。

- 隔離 VLAN のセカンダリ VLAN ID は再利用できます。
 たとえば、3 のプライマリ VLAN ID と 100 のセカンダリ VLAN ID を持つ隔離 VLAN がある場合は、VLAN ID 100 をプライマリ VLAN ID またはコミュニティーセカンダリ VLAN ID として再利用できます。

ゾーンを持つ PVLAN

VLAN と Oracle Solaris ゾーンを組み合わせることにより、スイッチなどの 1 つのネットワーク単位内に複数のプライベート仮想ネットワークを構成できます。PVLAN を使用すると、追加の VLAN を導入しなくても同じ VLAN 内にあるゾーン間のネットワーク分離を提供できます。次の図は、2 つの物理 NIC と 2 つの PVLAN を持つシステムがそれらの上に構成されていることを示しています。

図 10 ゾーンを持つ PVLAN



この図は、101 および 102 の VLAN ID を持つ 2 つの PVLAN を示しています。隔離およびコミュニティ VLAN は、プライマリ VLAN 101 の上に構成されています。プライマリ VLAN 102 上には隔離 VLAN が 1 つだけ構成されています。プライマリ VLAN 101 上の隔離 VLAN およびコミュニティ VLAN 内のゾーンは、相互に通信できません。ただし、コミュニティ VLAN 内のホストは、相互に通信できます。

ゾーンに PVLAN を割り当てる方法については、[82 ページの「ゾーンへの PVLAN の割り当て」](#) を参照してください。

プライベート VLAN の構成

dladm create-vlan を使用して PVLAN を作成したり、dladm create-vnic コマンドを使用して PVLAN トラフィックをホストするように PVLAN VNIC を構成したりできます。PVLAN VNIC の構成については、『Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理』の「VNIC を PVLAN として構成する方法」を参照してください。

PVLAN を作成するには次のコマンドを使用します。

```
dladm create-vlan [ -l link -v VLAN-ID[,PVLAN-SVID[,PVLAN-type]] [VLAN-link]
```

<i>link</i>	VLAN が作成される Ethernet リンクを指定します。
<i>VLAN-ID</i>	VLAN に関連付けられたプライマリ ID。
<i>PVLAN-SVID</i>	VLAN に関連付けられた PVLAN セカンダリ VLAN ID。
<i>PVLAN-type</i>	VLAN に関連付けられた PVLAN タイプ (isolated または community)。デフォルト値は isolated です。
<i>VLAN-link</i>	VLAN の名前。

例 20 dladm create-vlan コマンドを使用した隔離 PVLAN の作成

次の例は、3 のプライマリ VLAN ID、100 のセカンダリ VLAN ID、および isolated に設定された PVLAN タイプを持つ PVLAN を作成する方法を示しています。

```
# dladm create-vlan -v 3,100,isolated -l net0 vlan1
# dladm show-vlan
LINK   VID   SVID   PVLAN-TYPE  FLAGS   OVER
vlan1  3     100    isolated    ----- net0
```


プライベート VLAN の変更

dladm modify-vlan コマンドを使用して PVLAN を変更することで、PVLAN のプライマリ VLAN ID、セカンダリ VLAN ID、および PVLAN タイプを変更できます。dladm create-vnic コマンドを使用せずに dladm create-vlan コマンドを使用して PVLAN を作成した場合のみ、このコマンドを使用できます。PVLAN VNIC の変更については、『Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理』の「PVLAN VNIC の変更」を参照してください。

PVLAN を変更するには次のコマンドを使用します。

```
dladm modify-vlan [-v VID[,PVLAN-SVID[,PVLAN-type]]] VLAN-link
```

プライマリ VLAN ID、セカンダリ VLAN ID、および PVLAN タイプのタプルを同時に変更します。

次の例は、プライマリ VLAN ID を 15 に、セカンダリ VLAN ID を 103 に、および PVLAN タイプを community に設定する方法を示しています。

```
# dladm show-vlan
LINK      VID  SVID  PVLAN-TYPE  FLAGS  OVER
vlan1    10  102   isolated    ----  net0
# dladm modify-vlan -v 15,103,community vlan1
# dladm show-vlan
LINK      VID  SVID  PVLAN-TYPE  FLAGS  OVER
vlan1    15  103   community    ----  net0
```

プライベート VLAN の削除

システム上の PVLAN 構成を削除するには、dladm delete-vlan コマンドを使用します。

注記 - PVLAN を削除する前に、PVLAN 上の既存の IP 構成をすべて削除する必要があります。PVLAN 上に IP インタフェースが存在している場合、PVLAN の削除は失敗します。

例 21 VLAN 構成の削除

この例は VLAN 構成を削除する方法を示しています。

```
# dladm show-vlan
LINK      VID  SVID  PVLAN-TYPE  FLAGS  OVER
vlan1    15  103   community    ----  net0
```

```

vlan2      20    105    isolated    ----    net1
# ipadm delete-ip vlan1
# dladm delete-vlan vlan1
# dladm show-vlan
LINK      VID    SVID    PVLAN-TYPE  FLAGS    OVER
vlan2     20    105    isolated    ----    net1

```

ゾーンへの PVLAN の割り当て

分離を向上するために、ゾーンに PVLAN を割り当てることができます。ゾーンの構成については、『[Oracle Solaris ゾーン作成と使用](#)』を参照してください。

▼ ゾーンに PVLAN を割り当てる方法

1. 管理者になります。

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. VLAN を作成します。

```
# dladm create-vlan [-l link -v VLAN-ID[,PVLAN-SVID[,PVLAN-type]]] [VLAN-link]
```

3. 作成した VLAN を確認します。

```
# dladm show-vlan
```

4. PVLAN を割り当てて、ゾーンを構成します。

```

global# zonecfg -z zone-name
zonecfg:zone2> add anet
zonecfg:zone2:anet> set vlan-id=VID[,PVLAN-SVID[,PVLAN-type]]
zonecfg:zone2:anet> end
zonecfg:zone2> verify
zonecfg:zone2> commit
zonecfg:zone2> exit
global# zoneadm -z zone reboot

```

PVLAN が、ゾーンの anet のベースとなるリンクとして割り当てられます。

例 22 ゾーンへの PVLAN の割り当て

この例は、ゾーンに PVLAN を割り当てる方法を示しています。この例では、ゾーンは作成済みであると仮定されています。

```

# dladm create-vlan -l net0 -v 100 vlan10
# dladm show-vlan
LINK      VID    SVID    PVLAN-TYPE  FLAGS    OVER

```

```
vlan10    100    --      ---      ----      net0
global# zonecfg -z zone1
zonecfg:zone2> add anet
zonecfg:zone2:anet> set vlan-id=100,200,community
zonecfg:zone2:anet> end
zonecfg:zone2> verify
zonecfg:zone2> commit
zonecfg:zone2> exit
global# zoneadm -z zone reboot
```


ブリッジング機能の管理

個別のネットワークセグメントを接続するために使用されるブリッジを使って、それらのネットワークセグメントが単一のネットワークセグメントであるかのように通信できます。この章では、ブリッジネットワークを構成および管理する方法について説明します。

この章の内容は、次のとおりです。

- 85 ページの「ブリッジネットワークの概要」
- 91 ページの「ブリッジの作成」
- 93 ページの「ブリッジの保護タイプの変更」
- 94 ページの「既存のブリッジへのリンクの追加」
- 94 ページの「ブリッジからのリンクの削除」
- 95 ページの「ブリッジ構成情報の表示」
- 98 ページの「システムからのブリッジの削除」
- 98 ページの「ブリッジネットワーク上の VLAN の管理」
- 100 ページの「ブリッジのデバッグ」

ブリッジネットワークの概要

ブリッジはネットワーク内のさまざまなノードを単一のネットワークに接続します。ネットワークセグメントは、単一のブロードキャストネットワークを共有し、接続されると単一のネットワークセグメントであるかのように通信を行います。そのため、ルーターを使用してトラフィックをネットワークセグメント間で転送するのではなく、IP などのネットワークプロトコルを使用することによって、各ノードはほかのノードに到達できます。ブリッジを使用しない場合は、ノード間の IP トラフィックの転送を許可するように IP ルーティングを構成する必要があります。

ブリッジングとルーティングはどちらも、ネットワーク上のリソースの場所に関する情報を配布するために使用できますが、いくつかの点で異なります。ルーティングは IP 層 (L3) で実装され、ルーティングプロトコルを使用します。データリンク層では、ルーティングプロトコルは使用されません。

ブリッジングは、ネットワーク上のリソースの場所に関する情報を配布するために使用されます。ブリッジネットワークでは、パケットの転送先は、ブリッジに接続されているリンクで受信されたネットワークトラフィックを検査することで特定されます。ブリッジネットワークは、STP (Spanning Tree Protocol) や TRILL (Transparent Interconnection of Lots of Links) などのプロトコルを使用します。詳細は、89 ページの「ブリッジングプロトコル」を参照してください。

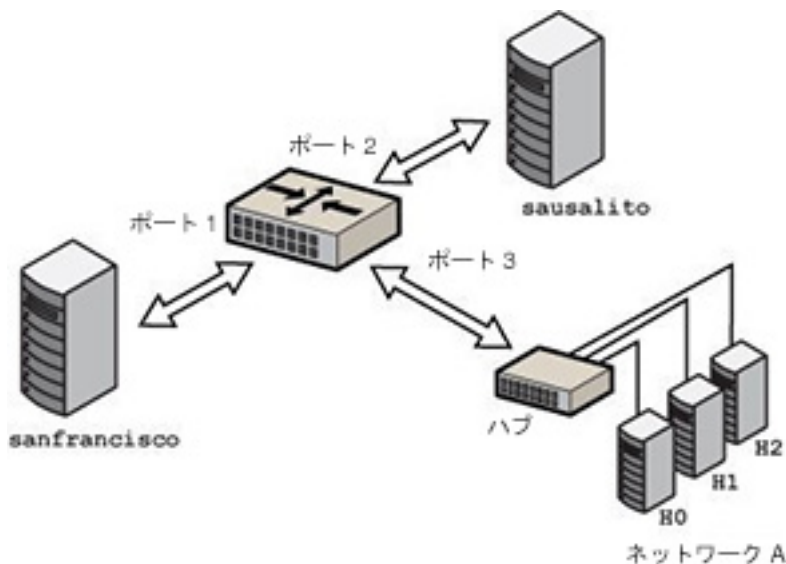


注意 - ブリッジングを使用する SPARC® ベースのシステムで `eeprom` コマンドを使用して、`local-mac-address?` プロパティを `false` に設定しないでください。これを設定すると、これらのシステムは同一の MAC アドレスを複数のポートや同じネットワーク上で誤って使用します。

単純なブリッジネットワーク

次の図は、単純なブリッジネットワーク構成を示しています。

図 11 単純なブリッジネットワーク



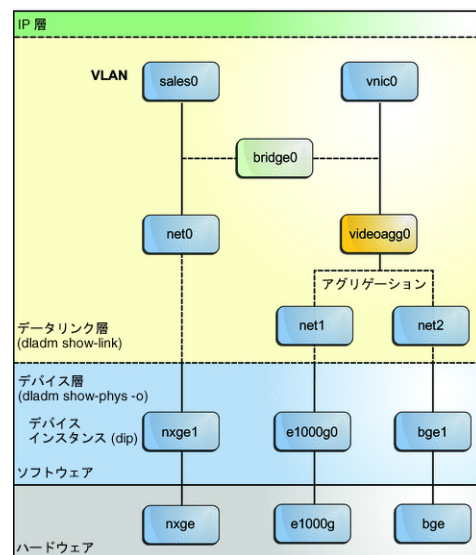
ブリッジ `goldengate` は、ブリッジングが構成された Oracle Solaris システムです。sanfrancisco および sausalito は、物理的にブリッジに接続されたシステムです。ネットワーク A ではハブが使用され、片側では物理的にブリッジに接続され、反

対側では3つのコンピュータシステムに接続されています。ブリッジポートはリンク net0、net1、および net2 です。

ネットワークスタックでの Oracle Solaris ブリッジの実装方法

Oracle Solaris では次の図に示すように、ブリッジを同じネットワークスタック実装のデータリンク層で構成できます。

図 12 Oracle Solaris のネットワークスタックでのブリッジ

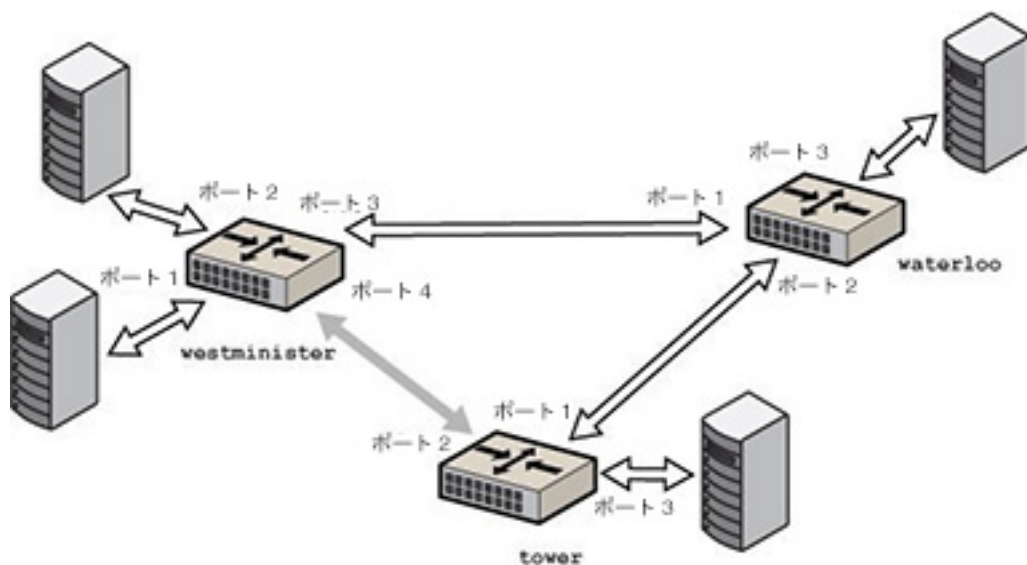


2つのインタフェース net0 と videoagg0 がブリッジ bridge0 として構成されています。1つのインタフェース上で受信されたパケットは、もう一方のインタフェースに転送されます。ブリッジを構成したあとも、引き続き両方のインタフェースを使用して VLAN および IP インタフェースを構成できます。

ブリッジネットワークリング

ブリッジネットワークはリング状に形成して、物理的に複数のブリッジを相互に接続できます。次の図は、ブリッジネットワークリング構成を示しています。

図 13 ブリッジネットワークリング



図は、リング状に構成されたブリッジネットワークを示しています。この構成では、3台のブリッジがあります。2台のシステムが物理的に westminister ブリッジに接続されています。1台のシステムが物理的に waterloo ブリッジに接続されており、もう1台のシステムが物理的に tower ブリッジに接続されています。ブリッジは、物理的にブリッジポートを介して相互に接続されています。

このタイプの構成では、古いパケットがリングの周りをエンドレスにループすることによってネットワークリンクが飽和状態になる問題が発生する可能性があります。このようなループ状況から保護するために、Oracle Solaris のブリッジには STP と TRILL の両方のプロトコルが実装されています。大部分のハードウェアブリッジでは、STP ループ保護も実装されています。

ブリッジネットワークの動作

ブリッジがパケットを受信すると、そのソースアドレスが検査されます。パケットのソースアドレスには、パケットが受信されたリンクにパケットを送信したノードが関連付けられます。その後、受信されたパケットが宛先アドレスと同じアドレスを使用すると、ブリッジはリンク上のパケットをそのアドレスに転送します。

ソースアドレスに関連付けられたリンクは、ブリッジネットワーク内の別のブリッジに接続された中間リンクである場合があります。時間が経過すると、ブリッジネットワーク内のすべてのブリッジが、どのリンクが特定のノードにパケットを送信するかを「学習」します。したがって、ホップバイホップのブリッジング方式で最終宛先にパケットを送信する際には、パケットの宛先アドレスが使用されます。

ローカルの「リンク停止」通知は、特定のリンク上のすべてのノードが到達不能になったことを示します。この状況では、リンクへのパケット転送が停止し、リンク上のすべての転送エントリがフラッシュされます。古い転送エントリも時間が経過するとフラッシュされます。リンクが復元されると、リンク上で受信されたパケットは新規として処理されます。パケットのソースアドレスに基づいた学習プロセスが再開されます。このプロセスによって、アドレスが宛先アドレスとして使用されたときに、ブリッジが正常にパケットをそのリンク上に転送できるようになります。

ブリッジングプロトコル

ブリッジネットワークは次のプロトコルを使用します。

- STP (Spanning Tree Protocol)

STP はブリッジネットワークで使用されるデフォルトのプロトコルです。ブリッジングでは STP メカニズムを使用して、サブネットワークが使用不可になる可能性のあるネットワークループを回避します。パケットを宛先に転送するには、ブリッジに接続されたすべてのリンク上でブリッジがプロミスキャスモードで待機する必要があります。プロミスキャスモードで待機すると、最大回線速度でパケットが無限に循環してしまう転送ループの発生に対して、ブリッジが脆弱になります。



注意 - 可能なかぎり高いレベルのパフォーマンスが必要な場合は、ブリッジにリンクを構成しないでください。ブリッジングでは、ベースとなるインタフェースをプロミスキャスモードにして、システムのハードウェア層 (NIC)、ドライバ層、およびそのほかの層で実装されている数多くの重要な最適化を無効にする必要があります。このようなパフォーマンス拡張機能が無効になることは、ブリッジングメカニズムでは避けられない結果です。

このようなパフォーマンスの問題は、ブリッジの一部になるように構成されたリンクにのみ影響を与えます。システムのリンクの一部がブリッジされていないために、これらの制約の対象ではないシステムでは、ブリッジを使用できます。

- TRILL (Transparent Interconnection of Lots of Links)

Oracle Solaris は、リンクを無効化せずにループを回避する TRILL 保護拡張機能をサポートしています。TRILL は、宛先への複数のパス間のトラフィックを負荷分散するために役立ちます。

STP がループ保護に使用されると、ループ内のいずれかの接続がパケットを転送しないようにすることで、物理的なループが軽減されます。図13は、westminster と tower ブリッジ間の物理リンクがパケット転送に使用されないことを示しています。

STP とは異なり、TRILL ではループを回避するために物理リンクがシャットダウンされません。その代わりに、TRILL はネットワーク内の TRILL ノードごとに最短パス情報を計算し、その情報を使用して個々の宛先にパケットを転送します。

`dladm create-bridge` または `dladm modify-bridge` コマンドで `-P trill` オプションを指定すれば、TRILL を使用できます。詳細は、91 ページの「ブリッジの作成」と93 ページの「ブリッジの保護タイプの変更」を参照してください。

STP の詳細は、IEEE 802.1D-1998 を参照してください。TRILL の詳細は、Internet Engineering Task Force (IETF) TRILL draft documents (<http://tools.ietf.org/wg/trill>) を参照してください。

STP デーモン

`dladm create-bridge` コマンドを使用して作成された各ブリッジは、`svc:/network/bridge` の同じ名前の付いたサービス管理機能 (SMF) インスタンスとして表されます。各インスタンスは、STP を実装する `/usr/lib/bridged` デーモンのコピーを実行します。

たとえば、次のコマンドは `pontevecchio` というブリッジを作成します。

```
# dladm create-bridge pontevecchio
```

`svc:/network/bridge:pontevecchio` という SMF サービスインスタンス、および `/dev/net/pontevecchio0` という可観測性ノードが作成されます。可観測性ノードは、`snoop` コマンドおよび `wireshark` パケットアナライザで使用することを目的としています。`dlstat` コマンドを使用してブリッジのランタイム統計を取得できます。

安全のため、デフォルトで標準 STP がすべてのポートで実行されます。STP などの一部のブリッジプロトコル形式が実行されないブリッジでは、ネットワークで長期にわたる転送ループが発生する可能性があります。Ethernet のパケットにはホップ数や有効期間 (TTL) が存在しないため、このようなループはネットワークにとって致命的です。

特定のポートが別のブリッジに接続されていない (たとえば、ポートがシステムに直接ポイントツーポイント接続されているため) ときは、管理上そのポートの STP を無効にできます。ブリッジ上のすべてのポートで STP が無効になっている場合でも、STP デーモンは次の理由で実行されます。

- 新たに追加されるポートを処理するため
- BPDU ガードを実装するため
- 必要に応じて、ポート上の転送を有効または無効にするため

ポートで STP が無効になっている場合、bridged デーモンは BPDU (BPDU ガード) の待機を続行します。デーモンは syslog を使用してエラーにフラグを付け、ポート上の転送を無効にして、重大なネットワークの構成ミスを示します。リンクが停止してから再起動すると、またはリンクを手動で削除してから再度追加すると、ふたたびリンクが有効になります。

ブリッジの SMF サービスインスタンスを無効にすると、STP デーモンが停止したときに、そのポート上のブリッジも停止します。インスタンスが再起動すると、STP は初期状態から起動します。

TRILL デーモン

dladm create-bridge -P trill コマンドを使用して作成された各ブリッジは、svc:/network/bridge および svc:/network/routing/trill の同じ名前の付いた SMF インスタンスによって表されます。svc:/network/routing/trill の各インスタンスでは、TRILL プロトコルを実装する /usr/lib trilld デーモンのコピーが実行されます。

たとえば、次のコマンドは bridgeofsighs というブリッジを作成します。

```
# dladm create-bridge -P trill bridgeofsighs
```

svc:/network/bridge:bridgeofsighs と svc:/network/routing/trill:bridgeofsighs という 2 つの SMF サービスが作成されます。さらに、/dev/net/bridgeofsighs0 という可観測性ノードも作成されます。

ブリッジの作成

Oracle Solaris では、dladm コマンドおよび SMF 機能を使用してブリッジを管理します。インスタンスの障害管理リソース識別子 (FMRI) svc:/network/bridge を使用してブリッジインスタンスを有効化、無効化、およびモニターするには、SMF コマンドを使用します。ブリッジを作成または破棄したり、ブリッジにリンクを割り当てたり、ブリッジからリンクを削除したりするには、dladm を使用します。ブリッジに割り当てられたリンクは、Ethernet タイプ (802.3 および 802.11 メディアを含む) である必要があります。

リンク間にブリッジを作成するには、少なくとも 1 つのブリッジインスタンスを作成する必要があります。各ブリッジインスタンスは個別のものです。ブリッジにはブリッジ間の転送接続は含まれず、リンクは最大で 1 つのブリッジのメンバーです。

dladm create-bridge コマンドは、ブリッジインスタンスを作成し、任意で新しいブリッジに 1 つ以上のネットワークリンクを割り当てます。デフォルトではブリッジイ

インスタンスがシステムに存在しないため、Oracle Solaris はデフォルトでネットワークリンク間にブリッジを作成しません。

ブリッジを作成するには次のコマンドを使用します。

```
# dladm create-bridge [-P protect] [-p priority] [-d forward-delay] [-l link...] bridge-name
```

- | | |
|------------------|--|
| -P protect | 保護の方式を指定します。これは、次のいずれかの値に設定する必要があります。 |
| | <ul style="list-style-type: none"> ■ stp – STP 保護方式 (デフォルト) ■ trill – TRILL 保護方式 |
| -p priority | IEEE STP 優先順位値を指定して、ネットワーク内のルートブリッジノードを決定します。デフォルト値は 32768 です。有効な値は 0 (もっとも高い優先順位) から 61440 (もっとも低い優先順位) で、4096 ずつ増分します。 |
| -d forward-delay | ブリッジの STP 転送遅延パラメータを指定します。作成されたブリッジがルートノードである場合、ポートが有効になっていると、ネットワーク内のすべてのブリッジがこのタイマーを使用してリンク状態を順序付けます。デフォルト値は 15 秒有効な値は 4 秒から 30 秒までです。 |
| -l link | ブリッジにリンクを追加します。指定されたリンクのいずれかを追加できない場合、コマンドは失敗し、ブリッジは作成されません。 |

bridge-name は任意の文字列であり、正当な SMF サービスインスタンス名である必要があります。この名前は、エスケープシーケンスが含まれない FMRI コンポーネントです。つまり、空白、ASCII 制御文字、および次の文字が存在してはいけません:

```
; / ? : @ & = + $ , % < > # "
```

default という名前と *SUNW* 文字列で始まるすべての名前は、予約されています。末尾に数字が付く名前は、デバッグに使用する可観測性デバイスを作成するために予約されています。可観測性デバイスが使用されるため、正当なブリッジインスタンス名は、正当な *dlpi* 名となるようにさらに制限されます。この名前は、英字または下線文字で開始および終了する必要があります。名前の残りの部分には、英数字と下線文字を含めることができます。

ブリッジ作成のオプションの詳細は、[dladm\(1M\)](#) のマニュアルページで `dladm create-bridge` コマンドの説明を参照してください。

例 23 ブリッジの作成

次の例は、`net0` と `net1` リンクを接続して、`brooklyn` ブリッジを作成する方法を示しています。

```
# dladm create-bridge -P stp -d 12 -l net0 -l net1 brooklyn
# dladm show-bridge
BRIDGE      PROTECT ADDRESS          PRIORITY DESROOT
goldengate  stp      32768/8:0:20:bf:f 32768     8192/0:d0:0:76:14:38
brooklyn    stp      32768/8:0:20:e5:8 32768     8192/0:d0:0:76:14:38
```

次の例は、net0 と net1 リンクを接続して、westminister ブリッジを作成する方法を示しています。

```
# dladm create-bridge -P trill -l net0 -l net1 westminister
# dladm show-bridge
BRIDGE      PROTECT ADDRESS          PRIORITY DESROOT
goldengate  stp      32768/8:0:20:bf:f 32768     8192/0:d0:0:76:14:38
westminister trill    32768/8:0:20:e5:8 32768     8192/0:d0:0:76:14:38
```

ブリッジの保護タイプの変更

STP は、サブネットワークが使用不可になる可能性のあるネットワークループを回避するメカニズムです。ブリッジに STP を使用することに加えて、Oracle Solaris では TRILL 保護拡張機能もサポートされています。デフォルトでは STP が使用されますが、ブリッジングコマンドで `-P trill` オプションを指定すれば、TRILL も使用できます。

保護タイプを STP から TRILL または TRILL から STP に変更するには、次のコマンドを使用します。

```
# dladm modify-bridge -P protection-type bridge-name
```

`-P protection-type` オプションは、使用する保護タイプ `stp` (デフォルト) または `trill` を指定します。

例 24 ブリッジの保護タイプの変更

次の例は、brooklyn ブリッジの保護タイプをデフォルトの STP から TRILL に変更する方法を示しています。

```
# dladm modify-bridge -P trill brooklyn
```

次の例は、brooklyn ブリッジの保護タイプを TRILL から STP に変更する方法を示しています。

```
# dladm modify-bridge -P stp brooklyn
```

既存のブリッジへのリンクの追加

1つのリンクは、最大1つのブリッジのメンバーになることができます。したがって、別のブリッジインスタンスにリンクを移動する場合は、まず現在のブリッジからリンクを削除してから、別のブリッジに追加する必要があります。

同じブリッジに割り当てるリンクはすべて、MTU値が同じである必要があります。既存のリンクのMTU値を変更できますが、ブリッジを再起動する前に、割り当てられたリンクを削除または変更してMTU値が一致するまで、ブリッジインスタンスは保守状態になります。

注記 - ブリッジに割り当てるリンクを VLAN、VNIC、またはトンネルにすることはできません。ブリッジに割り当てることができるのは、アグリゲーションの一部として受け入れ可能なリンク、またはアグリゲーションであるリンクのみです。

既存のブリッジに新しいリンクを追加するには、次のコマンドを使用します。

```
# dladm add-bridge -l new-link bridge-name
```

次の例は、既存のブリッジ `rialto` に `net2` リンクを追加する方法を示しています。

```
# dladm add-bridge -l net2 rialto
```

ブリッジからのリンクの削除

ブリッジを削除する前に、まずそのブリッジのリンクをすべて削除する必要があります。リンクを削除するには次のコマンドを使用します。

```
# dladm remove-bridge [-l link]... bridge-name
```

次の例は、ブリッジ `charles` から `net0`、`net1`、および `net2` リンクを削除する方法を示しています。

```
# dladm remove-bridge -l net0 -l net1 -l net2 charles
```

ブリッジのリンクプロパティの設定

次のリンクプロパティをブリッジに設定できます。

<code>default_tag</code>	リンクとの間で送受信されるタグなしパケットのデフォルト VLAN ID です。有効な値は 0 から 4094 です。デフォルト値は 1 です。
--------------------------	---

forward	ブリッジ経由のトラフィック転送を有効および無効にします。このプロパティは、VNIC リンク以外のすべてのリンクに存在します。有効な値は 1 (true) および 0 (false) です。デフォルト値は 1 です。
stp	STP と RSTP を有効および無効にします。有効な値は 1 (真) と 0 (偽) です。デフォルト値は 1 であり、STP と RSTP が有効になります。
stp_cost	リンクを使用する際の STP と RSTP のコスト値を表します。有効な値は 1 - 65535 です。デフォルト値は 0 であり、自動的にコストがリンクタイプ別に計算されるように指定されます。
stp_edge	ポートがほかのブリッジに接続されているかどうかを指定します。有効な値は 1 (真) と 0 (偽) です。デフォルト値は 1 です。
stp_p2p	接続モードタイプを指定します。有効な値は、true、false、および auto です。デフォルト値は auto です。
stp_priority	STP および RSTP ポートの優先順位値を設定します。有効な値は 0 - 255 です。デフォルト値は 128 です。

詳細は、[dladm\(1M\)](#) のマニュアルページを参照してください。

ブリッジのリンクプロパティを変更するには、次のコマンドを使用します。

```
dladm set-linkprop -p prop=value link
```

例 25 ブリッジのリンクプロパティの設定

次の例は、トラフィック転送を無効にし、接続モードタイプを設定する方法を示しています。ブリッジのプロパティを設定するには、ブリッジを接続するリンクのプロパティを設定する必要があります。

```
# dladm create-bridge -P stp -d 12 -l net0 -l net1 brooklyn
# dladm set-linkprop -p forward=0 net0
# dladm set-linkprop -p stp_p2p=true net1
```

次の例は、ブリッジの複数のプロパティをリセットする方法を示しています。

```
# dladm reset-linkprop -p default_tag,stp_priority brooklyn
```

ブリッジ構成情報の表示

dladm show-bridge コマンドを使用してブリッジ構成情報を表示できます。

構成されているブリッジに関する情報の表示

dladm show-bridge および dlstat show-bridge コマンドを使用して、構成されているブリッジに関する各種情報を表示できます。

次のコマンドオプションを使用します。

- ブリッジのリストを表示するには

```
# dladm show-bridge

# dladm show-bridge
BRIDGE      PROTECT ADDRESS          PRIORITY DESROOT
goldengate  stp      32768/8:0:20:bf:f 32768    8192/0:d0:0:76:14:38
baybridge   stp      32768/8:0:20:e5:8 32768    8192/0:d0:0:76:14:38
```

- ブリッジのリンク関連のステータスを表示するには

```
# dladm show-bridge -l bridge-name
```

- ブリッジのリンク関連の統計情報を表示するには

```
# dlstat show-bridge bridge-name
```

- ブリッジのカーネル転送エントリを表示するには

```
# dladm show-bridge -f bridge-name
```

- ブリッジに関する TRILL 情報を表示するには

```
# dladm show-bridge -t bridge-name
```

- 各ブリッジの統計情報と、各ブリッジに接続されているリンクの統計情報を表示するには

```
# dlstat show-bridge

BRIDGE  LINK      IPKTS  RBYTES  OPKTS  OBYTES  DROPS  FORWARDS
rbblue0  --      1.93K  587.29K  2.47K  3.30M   0      0
         simblue1    72    4.32K   2.12K  2.83M   0      --
         simblue2  1.86K  582.97K  348    474.04K  0      --
stbred0  --      975    976.69K  3.44K  1.13M   0      38
         simred3    347    472.54K  1.86K  583.03K  0      --
         simred4    628    504.15K  1.58K  551.51K  0      --
```

- 各ブリッジのすべての統計情報と、各ブリッジに接続されているリンクの統計情報を表示するには

```
# dlstat show-bridge -o all
```

dladm show-bridge コマンドのオプションの詳細については、[dladm\(1M\)](#) のマニュアルページを参照し、dlstat show-bridge コマンドのオプションについては、[dlstat\(1M\)](#) のマニュアルページを参照してください。

例 26 ブリッジ情報の表示

次の例は、さまざまなオプションを付けた `dladm show-bridge` コマンドの使用方法を示します。

- 次のコマンドは、単一のブリッジインスタンス `tower` に関するリンク関連のステータス情報を表示します。構成されているプロパティを表示するには、`dladm show-linkprop` コマンドを使用します。

```
# dladm show-bridge -l tower
LINK          STATE          UPTIME  DESROOT
net0          forwarding    117     8192/0:d0:0:76:14:38
net1          forwarding    117     8192/0:d0:0:76:14:38
```

- 次のコマンドは、指定されたブリッジ `avignon` のカーネル転送エントリを表示します。

```
# dladm show-bridge -f avignon
DEST          AGE          FLAGS  OUTPUT
8:0:20:bc:a7:dc  10.860    --     net0
8:0:20:bf:f9:69   --         L      net0
8:0:20:c0:20:26  17.420    --     net0
8:0:20:e5:86:11  --         L      net1
```

- 次のコマンドは、指定されたブリッジ `key` に関する TRILL 情報を表示します。

```
# dladm show-bridge -t key
NICK  FLAGS  LINK          NEXTHOP
38628 --     london        56:db:46:be:b9:62
58753 L      --            --
```

ブリッジリンクに関する構成情報の表示

`-o all` オプションを指定して `dladm show-link` コマンドを使用し、出力に `BRIDGE` フィールドを表示します。リンクがブリッジのメンバーである場合、このフィールドによってメンバーであるブリッジの名前が識別されます。リンクがブリッジの一部でない場合、`-p` オプションが使用されていれば、このフィールドは空白になります。それ以外の場合は、フィールドには `--` と表示されます。

`dladm show-link` 出力には、別々のリンクとしてブリッジの可観測性ノードも表示されます。このノードの場合、既存の `OVER` フィールドにブリッジのメンバーであるリンクが一覧表示されます。

ブリッジのメンバーであるリンクに関する構成情報を表示するには、次のコマンドを使用します。

```
# dladm show-link [-p]
```

-p オプションを付けると、解析可能な形式で出力されます。

システムからのブリッジの削除

ブリッジを削除する前に、ブリッジに接続されているすべてのリンクを最初に削除する必要があります。

▼ システムからブリッジを削除する方法

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. ブリッジに接続されているリンクをすべて削除します。

```
# dladm remove-bridge [-l link]... bridge-name
```

3. システムからブリッジを削除します。

```
# dladm delete-bridge bridge-name
```

例 27 システムからのブリッジの削除

次の例は、まず coronado ブリッジから net0、net1、および net2 リンクを削除してから、システムからブリッジ自体を削除する方法を示しています。

```
# dladm remove-bridge -l net0 -l net1 -l net2 coronado  
# dladm delete-bridge coronado
```

ブリッジネットワーク上の VLAN の管理

デフォルトでは、システムに構成された VLAN は、ブリッジインスタンス上のすべてのポート間でパケットを転送します。ベースとなるリンクがブリッジの一部である場合に、`dladm create-vlan` または `dladm create-vnic -v` コマンドを呼び出すと、該当ブリッジリンク上の指定された VLAN のパケット転送も有効になります。VLAN

の詳細は、[第3章「仮想ローカルエリアネットワークを使用した仮想ネットワークの構成」](#)を参照してください。

リンク上に VLAN を構成して、ブリッジ上のそのほかのリンク間のパケット転送を無効にするには、`dladm set-linkprop` コマンドで VLAN の `forward` プロパティを設定して、転送を無効にする必要があります。詳細は、[94 ページの「ブリッジのリンクプロパティの設定」](#)を参照してください。

▼ ブリッジの一部であるデータリンク上に VLAN を構成する方法

始める前に この手順は、ブリッジがすでに存在していることを前提としています。ブリッジを作成する方法については、[91 ページの「ブリッジの作成」](#)を参照してください。

1. 管理者になります。

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. ブリッジのリンク関連の情報をリストして、ブリッジの一部であるリンクを確認します。

```
# dladm show-bridge -l bridge-name
```

3. ブリッジの一部であるリンクの上に VLAN を作成します。

```
# dladm create-vlan -l link -v vid VLAN-link
```

`link` VLAN インタフェースの作成に使用するリンクを指定します。

注記 - この手順では、リンクは作成したブリッジの一部である必要があります。

`vid` VLAN ID 番号を示します。

`VLAN-link` VLAN の名前を指定します。

4. 作成するすべての VLAN に対して上記の手順を繰り返します。作成した VLAN ごとに、VLAN 上に IP インタフェースを作成します。

```
# ipadm create-ip interface
```

ここで、`interface` は VLAN 名です。

5. VLAN 上の各 IP インタフェースに、有効な IP アドレスを構成します。

```
# ipadm create-addr -a IP-address interface
```

VLAN と STP プロトコルおよび TRILL プロトコル

標準準拠の STP では、VLAN は無視されます。ブリッジプロトコルは、タグなし BPDU メッセージを使用して 1 つのループなしのトポロジのみを計算し、このツリートポロジを使用してリンクを有効および無効にします。ネットワークでプロビジョニングされている複製リンクは、STP によって自動的に無効になっても、構成された VLAN が接続解除されないように構成する必要があります。ブリッジされたバックボーンのあらゆる場所ですべての VLAN を実行するか、すべての冗長リンクを慎重に検査する必要があります。

TRILL プロトコルは、複雑な STP の規則には従いません。その代わりに、TRILL は自動的に、VLAN タグが変更されていないパケットをカプセル化して、ネットワーク経由で渡します。

ブリッジのデバッグ

各ブリッジインスタンスには、可観測性ノードが割り当てられます。これは、`/dev/net/` ディレクトリに表示され、ブリッジ名の末尾に `0` を付加した名前が付けられます。たとえば、`/dev/net/bridgeofsigths0` になります。

可観測性ノードは、`snoop` コマンドおよび `wireshark` パケットアナライザで使用することを目的としています。このノードは、パケット転送が暗黙的に破棄される点を除いて、標準の Ethernet インタフェースと同様に動作します。可観測性ノード上の IP は `plumb` することができず、`passive` オプションを使用しないかぎり、バインド要求 (`DL_BIND_REQ`) も実行できません。これにより、パケットの受信のみができ、送信はできません。

可観測性ノードは、ブリッジで処理されるすべてのパケットの未変更コピーを 1 つ作成します。これは、ユーザーがモニタリングおよびデバッグに使用できます。この動作は、従来のブリッジのポートモニタリングと同様であり、通常のデータリンクプロバイダインタフェース (DLPI) のプロミスキャスモード規則の対象になります。`pfmod` コマンド、または `snoop` コマンドと `wireshark` パケットアナライザの機能を使用すると、VLAN ID に基づいてパケットをフィルタリングすることもできます。

配信されたパケットは、可観測性ノードに送信されたパケットであり、ブリッジで受信されたデータを表します。

注記 - ブリッジングプロセスが VLAN タグを追加、削除、または変更する場合、`snoop` コマンドと `wireshark` パケットアナライザによって表示されるデータはプロセスが実行される前の状態を示します。さまざまなリンクで別々の `default_tag` 値が使用されると、このようなまれな状況で混乱する可能性があります。

ブリッジングプロセスの完了後に、特定のリンクで送受信されるパケットを確認するには、ブリッジの可観測性ノード上ではなく、個々のリンクで `snoop` コマンドを実行します。

また、`dlstat` コマンドを使用して、ネットワークパケットがリンク上のネットワークリソースをどのように使用しているかについて統計を取得することもできます。詳細は、『[Oracle Solaris 11.3 での仮想ネットワークとネットワークリソースの管理](#)』の第8章、「[ネットワークトラフィックとリソース使用状況のモニタリング](#)」を参照してください。

リンク層検出プロトコルによるネットワーク接続情報の交換

この章では、システムがリンク層検出プロトコル (LLDP) を使用して、ローカルネットワーク全体にわたってシステムおよびネットワーク接続情報を交換できるようにする方法について説明します。

この章の内容は、次のとおりです。

- 103 ページの「LLDP の概要」
- 106 ページの「LLDP エージェントが通知する情報」
- 109 ページの「システムでの LLDP の有効化」
- 113 ページの「エージェントの LLDP パケットの TLV ユニットと値の指定」
- 117 ページの「LLDP の無効化」
- 118 ページの「LLDP エージェントのモニタリング」

LLDP の概要

LLDP は、ローカルエリアネットワーク (LAN) のシステムが構成情報と管理情報を相互に交換するために使用します。このプロトコルを使用すると、システムは、接続や管理の情報をネットワーク上のほかのシステムに通知できます。これらの情報には、ネットワーク操作に関連するシステムの機能、管理アドレス、およびその他の情報を含めることができます。またこのプロトコルにより、システムは、同じローカルネットワーク上にあるほかのシステムに関する同様の情報を受信することもできます。

どの LAN でも、システムやスイッチなどの個々のコンポーネントが分離された状態で構成されることはありません。ネットワークトラフィックを効率よくホストするには、ネットワーク上のシステムの構成を相互に調整する必要があります。

各システム、スイッチ、およびその他のコンポーネントを手動で構成する場合、それらのコンポーネントの間で互換性を確保することが課題となります。システムの手動構成にはリスクが伴い、特に複数の管理者が異なるシステムを独立して操作する場合は、構成ミスが発生しやすくなります。より優れた方法は LLDP を使用することで

す。LLDP はシステムがそれぞれの構成情報をピアシステムに送信でき、構成ミスの検出に役立ちます。

Oracle Solaris は LLDP の使用をサポートして、ネットワーク上のシステム間でのシステムおよびネットワーク接続情報の交換を拡張し、ネットワークリソースの構成ミスのリスクを軽減します。

このリリースでは、ネットワーク診断サービスが LLDP を使用して、ネットワーク接続の制限または機能低下 (あるいはその両方) につながる可能性のある問題を自動的に検出します。LLDP サービスを有効にすると、使用している Oracle Solaris システムのネットワーク診断を実行する機能が強化されます。ネットワークの診断の詳細は、『[Oracle Solaris 11.3 でのネットワーク管理のトラブルシューティング](#)』の第 5 章、「[network-monitor](#) トランスポートモジュールユーティリティを使用したネットワーク診断の実行」を参照してください。

Oracle Solaris では、LLDP はデータセンターブリッジング交換プロトコル (DCBX) の TLV (Type-Length-Value) ユニットの交換にも使用されます。DCBX は、優先順位ベースのフロー制御 (PFC) や拡張伝送選択 (ETS) などの DCB 機能に関する構成情報を提供します。DCB の詳細は、[第 7 章「データセンターブリッジングを使用した集中ネットワークの管理」](#)を参照してください。

LLDP を使用すると、システム管理者は、特に仮想ローカルエリアネットワーク (VLAN) やリンクアグリゲーションなどの複雑なネットワークで、誤ったシステム構成を容易に検出できます。ネットワークを構成しているサーバー、スイッチ、そのほかのデバイス間の物理的な接続を追跡しなくても、ネットワークトポロジに関する情報を簡単に取得できます。

LLDP 実装のコンポーネント

LLDP は、次のコンポーネントを使用して実装されています。

- **LLDP パッケージ** – このパッケージをインストールして LLDP を有効にします。このパッケージには、LLDP デーモン、コマンド行ユーティリティ、サービスマニフェストとスクリプトのほか、LLDP が動作するために必要なそのほかのコンポーネントが含まれています。
- **LLDP サービス** – `svcadm` コマンドを使用して LLDP サービスを有効にできます。このサービスは、サービス管理機能 (SMF) サービスインスタンス `svc:/network/11dp:default` の障害管理リソース識別子 (FMRI) を使用して、LLDP デーモン `11dps` を管理します。この LLDP サービスは、`11dps` デーモンの起動、停止、再起動、またはリフレッシュを行います。LLDP パッケージをインストールすると、このサービスは自動的に有効になります。
- **11dps コマンド** – このコマンドは、個々のリンク上の LLDP を管理し、LLDP の動作モードを構成したり、送信される TLV ユニットの指定したり、DCBX TLV ユニットの構成したりするために使用できます。TLV ユニットについては、[106 ページの「LLDP エージェントが通知する情報」](#)を参照してください。

エージェントごとの LLDP プロパティとグローバルな LLDP プロパティを設定したり、特定のエージェントまたはそのピアの LLDP 情報を取得するには、このコマンドを使用する必要があります。

lldpadm サブコマンドについては、次のセクションで説明します。lldpadm コマンドについての詳細は、[lldpadm\(1M\)](#) のマニュアルページを参照してください。

- **LLDP デーモン** – LLDP サービスはシステム上の LLDP エージェントを管理します。また、SNMP (Simple Network Management Protocol) を経由してシステム上で受信される LLDP 情報を取得するために、SNMP のためのデーモンである `snmpd` とも相互作用します。
- **LLDP エージェント** – LLDP エージェントは、LLDP が有効になっている物理データリンクに関連付けられた LLDP インスタンスです。LLDP エージェントはピアにデータリンクに関する情報を送信し、ピアからの情報の受信も行います。関連付けられた物理的なデータリンクに関する特定の情報を通知するように、LLDP エージェントを構成できます。物理データリンク上の LLDP のみを有効にできます。

LLDP エージェントの情報源

LLDP エージェントは、LLDP データユニット (LLDPDU) を送受信します。このエージェントは、次のタイプのデータストア内で、これらの LLDPDU に含まれている情報を管理および格納します。

- **ローカル管理情報ベース (MIB)** – このデータストアには、LLDP エージェントが有効になっているシステムの特定のリンクに関連するネットワーク情報が含まれています。ローカル MIB には、一般的な情報と固有の情報の両方が含まれます。たとえば、シャーシ ID は、システム上のすべての LLDP エージェントの間で共有されている一般的な情報です。ただし、システムのデータリンクのポート ID は異なります。そのため、各エージェントは、独自のローカル MIB を管理します。
- **リモート MIB** – このデータストア内の情報は、ピアホストの LLDP エージェントから受信されます。

LLDP エージェントモード

LLDP エージェントは、次のモードで動作します。

- **送信のみ (txonly)** – LLDP エージェントは受信 LLDPDU を処理しません。そのため、リモート MIB は空です。
- **受信のみ (rxonly)** – エージェントは受信 LLDPDU のみを処理し、情報をリモート MIB 内に格納します。ただし、ローカル MIB からの情報は送信されません。
- **送受信 (both)** – エージェントはローカル情報を送信し、受信 LLDPDU を処理し、そのためにローカル MIB とリモート MIB の両方を維持します。

- 無効 (disable) – エージェントは存在しません。

エージェントモードの設定については、[111 ページ](#)の「特定のポートの LLDP を有効にする方法」を参照してください。

LLDP エージェントが通知する情報

LLDP エージェントは、LLDP パケットまたは LLDPDU でシステムおよび接続情報を送信します。このようなパケットには、TLV 形式で個別にフォーマットされた情報ユニットが含まれています。これらの情報ユニットは *TLV* ユニットとも呼ばれます。

必須の TLV ユニット

特定の TLV ユニットは必須であり、LLDP が有効になったときにデフォルトで LLDP パケットに含まれます。l1dadm コマンドを使用して、このようなユニットのいずれかを除外することはできません。

次の TLV ユニットは必須です。

- シャーシ ID – hostid コマンドによって生成される情報
- ポート ID – 物理 NIC の MAC アドレス
- TTL (生存期間)
- プロトコルデータユニット (PDU) の終わり

リンクの数に応じて、1 つのシステム内で複数の LLDP エージェントを有効にすることができます。シャーシ ID とポート ID の組み合わせによってエージェントが一意に識別され、システム上のほかのエージェントから区別されます。

例 28 シャーシ ID とポート ID の表示

次の例では、LLDP エージェントのシャーシ ID およびポート ID を表示します。

```
# hostid
004e434e

# dladm show-phys -m net4
LINK          SLOT      ADDRESS          INUSE CLIENT
net4          primary  0:1b:21:87:8b:b4  yes   net4

# l1dadm show-agent -1 net4
AGENT          CHASSISID          PORTID
net4           004e434e           00:1b:21:87:8b:b4
```

Oracle Solaris LLDP エージェントは、シャーシ ID として hostid を使用し、ポート ID としてポートの MAC アドレスを使用します。

オプションの TLV ユニット

オプションの TLV ユニットの LLDP パケットに追加できます。これらのオプションの TLV ユニットにより、ベンダーは通知するベンダー固有の TLV ユニットの挿入できます。LLDP により、組織の一意識別子 (OUI) を使用して追加の TLV ユニットの定義できます。OUI は、OUI が準拠するのが IEEE 802.1 標準か IEEE 802.3 標準かによって、TLV ユニットのカテゴリを識別します。LLDP エージェントのプロパティを構成して、このようなオプションの TLV ユニットの送信を有効または無効にできます。

次の表は、各 TLV グループ、それに対応する名前、プロパティごとの TLV ユニット、およびそれらの説明を示しています。LLDP が有効になったときにパケットに含まれる TLV ユニットの指定するには、これらのプロパティのいずれかを構成します。

表 2 LLDP エージェントのオプションの TLV ユニット

TLV グループ	TLV 名	TLV ユニット	説明
基本的な管理	basic-tlv	sysname、portdesc、syscapab、sysdesc、mgmtaddr	通知されるシステム名、ポートの説明、システムの機能、システムの説明、および管理アドレスを指定します。
802.1 OUI	dot1-tlv	vlanname、pvid、linkaggr、pfc、appln、evb、etscfg、etsreco	通知される次の項目を指定します: VLAN 名、ポートの VLAN ID、リンクアグリゲーション、優先順位ベースのフロー制御の TLV ユニット、アプリケーション、拡張伝送選択、およびエッジ仮想ブリッジング。
802.3 OUI	dot3-tlv	max-framesize	通知される最大フレームサイズを指定します。
Oracle 固有の OUI (0x0003BA として定義される)	virt-tlv	vnic	仮想ネットワークが構成されている場合は、通知される VNIC を指定します。

TLV ユニットプロパティ

各 TLV ユニットにはプロパティがあり、これらのプロパティは特定の値を使用してさらに構成できます。TLV ユニットが LLDP エージェントのプロパティとして有効になった場合、その TLV ユニットは、指定された値のみを使用してネットワーク内で通知されます。たとえば、システムの機能を通知する TLV ユニット `syscapab` を考えてみます。これらの機能には、ルーター、ブリッジ、リピータ、電話などのデバイスに対するサポートが含まれる可能性があります。ただし、ルーターやブリッジなど、特定のシステム上で実際にサポートされている機能のみが通知されるように `syscapab` を設定できます。

TLV ユニットの構成するための手順は、グローバルな TLV ユニットの構成またはエージェントごとの TLV ユニットのどちらを構成するかによって異なります。TLV ユニットの構成方法については、113 ページの「エージェントの LLDP パケットの TLV ユニットの値の指定」を参照してください。

グローバルな TLV ユニットの構成は、システム上のすべての LLDP エージェントに適用されます。次の表は、グローバルな TLV ユニットの構成とそれに対応する、取り得る構成を示しています。

表 3 グローバルな TLV ユニットの構成とそのプロパティ

TLV ユニットの	プロパティ名	取り得るプロパティ値	値の説明
syscapabsupported		other、repeater、bridge、wlan-ap、router、telephone、docsis-cd、station、cvlan、sylvan、tpmr	システムのサポートされている主要な機能を表します。デフォルト値は、router、station、および bridge です。
	enabled	supported に対して示されている値のサブセット	システムの有効になっている機能を表します。
mgmtaddrpaddr		ipv4 または ipv6	ローカルの LLDP エージェントに関連付けられる IP アドレスのタイプを指定します。これらのアドレスは、上位階層エンティティに到達するために使用され、ネットワーク管理による検出に役立ちます。指定できるのは 1 つのタイプだけです。

LLDP エージェントに固有の TLV ユニットの構成はエージェント単位で管理されます。エージェントごとの TLV ユニットの構成では、指定した値は、特定の LLDP エージェントがその TLV ユニットの転送を有効にしたときに使用されます。

次の表は、LLDP エージェントの TLV 値とそれに対応する、取り得る構成を示しています。

表 4 エージェントごとの TLV ユニットの構成とそのプロパティ

TLV ユニットの	プロパティ名	取り得るプロパティ値	値の説明
pfc	willing	on、off	優先順位ベースのフロー制御に関連するリモートシステムからの構成情報を受け入れるか、または拒否するように LLDP エージェントを設定します。
appln	apt	値は、アプリケーション優先順位表で定義されている情報から取得されます。	アプリケーション優先順位表を構成します。この表には、アプリケーション TLV ユニットの構成とそれに対応する優先順位の一覧が含まれています。アプリケーションは、id/selector のペアで識別されます。この表の内容では、次の形式が使用されます。

TLV ユニット	プロパティ名	取り得るプロパティ値	値の説明
			id/selector/priority
etscfg	willing	on、off	<p>詳細は、135 ページの「アプリケーション優先順位の構成」を参照してください。</p> <p>拡張伝送選択に関連するリモートシステムからの構成情報を受け入れるか、または拒否するように LLDP エージェントを設定します。</p>

エージェントごとの TLV ユニットについては、[第7章「データセンターブリッジングを使用した集中ネットワークの管理」](#)を参照してください。

システムでの LLDP の有効化

システム情報をネットワーク上のほかのシステムまたはピアと交換するように LLDP を構成できます。

SMF プロパティ `auto-enable-agents` は、システム上で LLDP エージェントを有効にできる方法を制御します。このプロパティでは、LLDP をすべての物理リンクにわたってグローバルに有効にするか、一度に 1 つの物理リンクでのみ有効にするかを選択できます。

SMF プロパティ `auto-enable-agents` は、次の 3 つの値のいずれかを使用できます。

- ポートに以前の LLDP 構成が存在していない場合は、`yes` により、すべてのポートの LLDP が送受信 (both) モードで有効になります。ポートに構成が存在している場合は、そのポートの構成が保持されます。たとえば、ポートに `rxonly` モードの LLDP がすでに構成されている場合、LLDP サービスはエージェントを送受信 (both) モードでの実行に切り替えません。そのポートの LLDP は引き続き `rxonly` モードになります。これは SMF プロパティ `auto-enable-agents` のデフォルト値です。
- `force` は、すべてのポートの LLDP を送受信 (both) モードで有効にし、ポートの既存の LLDP 構成をすべてオーバーライドします。たとえば、ポートの以前の LLDP 構成が `rxonly` モードで動作している場合、LLDP エージェントは、送受信 (both) モードで実行されるデフォルトの LLDP モードに切り替えられます。
- `no` は、既存の LLDP 構成があるポートを除くすべてのポートで、LLDP の自動有効化を無効にします。これらのポートでは、既存の LLDP 構成が保持されます。

注記 - `auto-enable-agents` プロパティをカスタマイズするたびに、新しい値を有効にするために LLDP サービスを再起動する必要があります。

▼ LLDP パッケージのインストール方法

デフォルトでは、LLDP パッケージのインストールが完了すると、LLDP が有効で使用できる状態になります。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. パッケージをインストールします。

```
# pkg install lldp
```

3. LLDP サービスが開始されたかどうかを確認します。

```
# svcs lldp
STATE          STIME      FMRI
online         Jul_10     svc:/network/lldp:default
```

LLDP サービスが無効になっている場合は、次のコマンドでサービスを起動します。

```
# svcadm enable svc:/network/lldp:default
```

▼ LLDP をグローバルに有効にする方法

始める前に LLDP を有効にするには、最初に LLDP パッケージをインストールする必要があります。詳細は、110 ページの「LLDP パッケージのインストール方法」を参照してください。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. **auto-enable-agents SMF** プロパティーが **no** に設定されている場合、**yes** に変更します。

```
# svccfg -s svc:/network/lldp:default setprop lldp/auto-enable-agents = "yes"
```

デフォルトでは、このプロパティーは **yes** に設定されます。

3. LLDP サービスを再起動します。

```
# svcadm restart svc:/network/lldp:default
```

4. (オプション) グローバルな TLV ユニットのカスタマイズします。

```
# lldpadm set-tlvprop -p property=value global-TLV
```

ここで、*property* はグローバルな TLV ユニットのプロパティを示します。

次の手順 グローバルな TLV ユニットの説明については、[107 ページ](#)の「[TLV ユニットプロパティ](#)」を参照してください。

グローバルな TLV のリストを表示するには、`lldpadm show-tlvprop` を入力するか、[表3](#)を参照してください。

TLV 値の定義方法については、[115 ページ](#)の「[TLV ユニットの定義方法](#)」を参照してください。

`lldpadm` コマンドについては、[lldpadm\(1M\)](#) のマニュアルページを参照してください。

▼ 特定のポートの LLDP を有効にする方法

始める前に LLDP を有効にするには、最初に LLDP パッケージをインストールする必要があります。詳細は、[110 ページ](#)の「[LLDP パッケージのインストール方法](#)」を参照してください。

1. 管理者になります。

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. `auto-enable-agents` SMF プロパティが `yes` に設定されている場合、`no` に変更します。

```
# svccfg -s svc:/network/lldp:default setprop lldp/auto-enable-agents = "no"
```

デフォルトでは、このプロパティは `yes` に設定されます。

3. 手順 2 で SMF プロパティ `auto-enable-agents` を変更した場合は LLDP サービスを再起動します。

```
# svcadm restart svc:/network/lldp:default
```

4. 選択したポートまたはリンク上で LLDP エージェントを有効にします。

```
# lldpadm set-agentprop -p mode=value agent
```

ここで、*agent* は LLDP エージェントであり、エージェントが有効になっている物理リンクによって識別されます。たとえば、LLDP を `net0` で有効にした場合、エージェントは `net0` になります。

プロパティ `mode` は、LLDP エージェントの動作モードを表す 4 つの取り得る値 `txonly`、`rxonly`、`both`、`disable` のいずれかに設定できます。これらの値の説明については、[105 ページの「LLDP エージェントモード」](#)を参照してください。

5. LLDP エージェントで通知できる TLV ユニットの指定します。

```
# lldpadm set-agentprop -p property=value agent
```

LLDP エージェントのプロパティの説明については、[106 ページの「LLDP エージェントが通知する情報」](#)を参照してください。

LLDP エージェントのほかのプロパティのリストを表示するには、`lldpadm show-agentprop` を入力するか、[表2](#)を参照してください。

エージェントの LLDP パケットの TLV ユニットの指定する方法については、[114 ページの「エージェントの LLDP パケットの TLV ユニットの指定する方法」](#)を参照してください。

6. (オプション) エージェントごとの TLV ユニットをカスタマイズします。

```
# lldpadm set-agenttlvprop -p property=value -a agent per-agent-TLV
```

ここで、`property` はエージェントごとの TLV ユニットのプロパティを示します。

エージェントごとの TLV ユニットの説明については、[107 ページの「TLV ユニットのプロパティ」](#)を参照してください。

エージェントごとの TLV のリストを表示するには、`lldpadm show-agenttlvprop` を入力するか、[表4](#)を参照してください。

TLV 値の定義方法については、[115 ページの「TLV ユニットの定義する方法」](#)を参照してください。

`lldpadm` コマンドについては、[lldpadm\(1M\)](#) のマニュアルページを参照してください。

例 29 auto-enable-agents SMF プロパティのカスタマイズ

次の例では、SMF プロパティ `auto-enable-agents` の値を変更すると LLDP が異なる方法で有効になることを示します。たとえば、4 つのポートを備えたシステムで、LLDP が 2 つのポート上で次のように構成されているとします。

- net0: both モード
- net1: rxonly モード
- net2 および net3: なし

SMF プロパティ `auto-enable-agents` のデフォルト値が `yes` である場合、`net2` および `net3` で LLDP が自動的に有効になります。次のように LLDP 構成を表示できます。


```
# lldpadm show-agentprop -p mode
AGENT  PROPERTY  PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
net0   mode       rw    both   --         disable  txonly,rxonly,both,disable
net1   mode       rw    rxonly --         disable  txonly,rxonly,both,disable
net2   mode       rw    both   --         disable  txonly,rxonly,both,disable
net3   mode       rw    both   --         disable  txonly,rxonly,both,disable
```

この SMF プロパティを no に切り替えると、サービスの再起動時に構成が変更されます。

```
# svccfg -s svc:/network/lldp:default setprop lldp/auto-enable-agents = "no"
# svcadm restart svc:/network/lldp:default
# lldpadm show-agentprop -p mode
AGENT  PROPERTY  PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
net0   mode       rw    both   --         disable  txonly,rxonly,both,disable
net1   mode       rw    rxonly --         disable  txonly,rxonly,both,disable
net2   mode       rw    disable --         disable  txonly,rxonly,both,disable
net3   mode       rw    disable --         disable  txonly,rxonly,both,disable
```

出力例では、それまで LLDP モードが自動的に有効になっていた net2 と net3 に、現在は無効としてとしてフラグが付けられています。ただし、LLDP エージェントがあらかじめ構成されていた net0 と net1 には、変更は発生しません。

例 30 複数のデータリンク上の LLDP を有効にする

この例では、LLDP を選択的に有効にする方法を示します。システムに 2 つのデータリンク net0 と net1 が存在します。net0 に LLDP パケットを送受信するエージェントを設定し、net1 に LLDP パケットの送信のみを行うエージェントを設定するには、次のコマンドを入力します。

```
# svccfg -s svc:/network/lldp:default setprop lldp/auto-enable-agents = "no"
# svcadm restart svc:/network/lldp:default
# lldpadm set-agentprop -p mode=both net0
# lldpadm set-agentprop -p mode=txonly net1
# lldpadm show-agentprop -p mode
AGENT  PROPERTY  PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
net0   mode       rw    both   --         disable  txonly,rxonly,both,disable
net1   mode       rw    txonly --         disable  txonly,rxonly,both,disable
```

エージェントの LLDP パケットの TLV ユニットと値の指定

dot1-tlv や basic-tlv などの TLV ユニットの LLDP エージェントのプロパティ値として指定できます。これらのプロパティ値をさらに構成できます。TLV ユニットの指定するには、lldpadm set-agentprop コマンドを使用します。詳細は、[lldpadm\(1M\)](#) のマニュアルページを参照してください。TLV ユニットが LLDP エージェントのプロパティとして指定されると、その TLV ユニットは、TLV ユニットに対して指定された値のみを使用してネットワーク内で通知されます。TLV ユニットとプロパティについては、[107 ページの「TLV ユニットプロパティ」](#)を参照してください。

▼ エージェントの LLDP パケットの TLV ユニットの指定する方法

この手順では、エージェントが送信する LLDP パケット内で通知する、TLV ユニットの指定する方法について説明します。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. (オプション) TLV ユニットを表示して、追加する TLV ユニットを含めることのできる LLDP エージェントプロパティを確認します。

```
# lldpadm show-agentprop agent
```

このコマンドでは、プロパティごとにすでに設定されている TLV ユニットを確認できます。プロパティを指定しない場合、このコマンドはすべての LLDP エージェントプロパティと、その TLV 値を表示します。エージェントプロパティのリストについては、表2を参照してください。

3. プロパティに TLV ユニットを追加、またはプロパティから TLV ユニットを削除します。

```
# lldpadm set-agentprop -p property[+|-]=value[,...] agent
```

修飾子を使用して、複数の値を受け入れるプロパティ値のリストに値を追加 (+) したり、そのリストから値を削除 (-) したりできます。

追加修飾子 (+) や削除修飾子 (-) を使用しない場合は、設定する値により、以前にそのプロパティに対して定義されていたすべての値が置き換えられます。

4. (オプション) プロパティの新しい値を表示します。

```
# lldpadm show-agentprop -p property agent
```

例 31 LLDP パケットへのオプションの TLV ユニットの追加

次の例では、LLDP エージェントには、LLDP パケットで VLAN 情報を通知するように net0 が構成されています。LLDP パケットは、LLDP が通知できる項目として、システム機能、リンクアグリゲーション、および仮想 NIC 情報を含むようにさらに構成されます。VLAN の記述はあとでパケットから削除されます。

1. 既存のエージェントプロパティを表示します。

```
# lldpadm show-agentprop net0
```

AGENT	PROPERTY	PERM	VALUE	EFFECTIVE	DEFAULT	POSSIBLE
net0	mode	rw	both	--	disable	txonly, rxonly, both, disable
net0	basic-tlv	rw	sysname, sysdesc	--	none	none, portdesc, sysname, sysdesc, syscapab, mgmtaddr, all
net0	dot1-tlv	rw	vlanname, pvid, pfc	--	none	none, vlanname, pvid, linkaggr, pfc, appln, evb, etscfg, etsreco, all
net0	dot3-tlv	rw	max-framesize	--	none	none, max-framesize, all
net0	virt-tlv	rw	none	--	none	none, vnic, all

出力は、LLDP エージェントのプロパティごとに既存の値、デフォルトの値、および指定可能な値を表示します。

- ネットワーク経由で通知する項目として、システムの機能、リンクアグリゲーション、およびネットワーク仮想化の情報を設定します。

```
# lldpadm set-agentprop -p basic-tlv+=syscapab,dot1-tlv+=linkaggr,virt-tlv=vnic net0
```

- パケットから VLAN の説明を削除します。

```
# lldpadm set-agentprop -p dot1-tlv-=vlanname net0
```

- エージェントプロパティを表示します。

```
# lldpadm show-agentprop -p net0
```

AGENT	PROPERTY	PERM	VALUE	EFFECTIVE	DEFAULT	POSSIBLE
net0	mode	rw	both	--	disable	txonly, rxonly, both, disable
net0	basic-tlv	rw	sysname, sysdesc, syscapab	--	none	none, portdesc, sysname, sysdesc, syscapab, mgmtaddr, all
net0	dot1-tlv	rw	pvid, pfc, linkaggr	--	none	none, vlanname, pvid, linkaggr, pfc, appln, evb, etscfg, etsreco, all
net0	dot3-tlv	rw	max-framesize	--	none	none, max-framesize, all
net0	virt-tlv	rw	vnic	--	none	none, vnic, all

▼ TLV ユニットの定義する方法

この手順では、特定の TLV ユニットの値を指定する方法について説明します。

ヒント - TLV プロパティをデフォルト値にリセットするには、グローバルな TLV ユニットの場合は `lldpdm reset-tlvprop` コマンドを使用し、エージェントごとの TLV ユニットの場合は `lldpdm reset-agenttlvprop` コマンドを使用します。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. グローバルな TLV ユニットまたはエージェントごとの TLV ユニットの構成します。

- グローバルな TLV ユニットの構成する場合は、該当する TLV プロパティを、通知しようとする値を含むように設定します。

```
# lldpdm set-tlvprop -p TLV-property=value[,value,value,...] TLV-name
```

ここで、*TLV-name* はグローバルな TLV ユニットの名前、*TLV-property* はその TLV ユニットのプロパティです。プロパティに複数の値を割り当てることができます。グローバルな TLV ユニットおよびそのプロパティのリストについては、[表3](#)を参照してください。

- エージェントごとの TLV ユニットの構成する場合は、LLDP エージェントの該当する TLV プロパティを、エージェントで通知しようとする値を含むように構成します。

```
# lldpdm set-agenttlvprop -p TLV-property[+|-]=value[,value,value,...] -a agent TLV-name
```

ここで、*TLV-name* はエージェントの TLV ユニットの名前、*TLV-property* はその TLV ユニットのプロパティです。プロパティに複数の値を割り当てることができます。エージェントごとの TLV ユニットおよびそのプロパティのリストについては、[表4](#)を参照してください。

修飾子を使用して、複数の値を受け入れるプロパティ値のリストに値を追加 (+) したり、そのリストから値を削除 (-) したりできます。

3. (オプション) 構成した TLV プロパティの値を表示します。

- グローバルな TLV プロパティ値を表示するには

```
# lldpdm show-tlvprop
```

- エージェントの TLV プロパティの値を表示するには

```
# lldpdm show-agenttlvprop
```

例 32 syscapab および mgmtaddr TLV ユニットの TLV 値の定義

次の例では、LLDP パケットおよび管理 IP アドレスで通知されるシステムの機能に関する特定の情報が構成されます。

1. syscapab TLV ユニットの supported および enabled プロパティを構成します。

```
# lldpadm set-tlvprop -p supported=bridge,router,repeater syscapab
# lldpadm set-tlvprop -p enabled=router syscapab
```

2. mgmtaddr TLV ユニットの管理 IP アドレスを指定します。

```
# lldpadm set-tlvprop -p ipaddr=192.168.1.2 mgmtaddr
```

3. エージェントプロパティの TLV 値を表示します。

```
# lldpadm show-tlvprop
TLVNAME    PROPERTY  PERM  VALUE          DEFAULT          POSSIBLE
syscapab   supported  rw    bridge,        bridge,router,   other,router,
          router,        repeater,bridge,
          repeater      wlan-ap,telephone,
          docis-cd,station,
          cvlan,svlan,tpmr

syscapab   enabled   rw    router         none              bridge,router,
          repeater

mgmtaddr   ipaddr    rw    192.162.1.2   none              --
```

出力には TLV ユニットのデフォルト値や、プロパティに設定できる指定可能な値も含まれます。

エージェントごとの TLV プロパティの構成については、[第7章「データセンターブリッジングを使用した集中ネットワークの管理」](#)を参照してください。

LLDP の無効化

このセクションでは、個々のポートで LLDP を選択的に無効にする方法について説明します。

▼ LLDP を無効にする方法

LLDP をシステムのすべてのインタフェースで無効にするには、次の手順を実行します。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. SMF LLDP プロパティを `no` に変更して、既存の LLDP 構成があるポートを除くすべてのポートで、LLDP の自動有効化を無効にします。

```
# svccfg -s svc:/network/lldp:default setprop lldp/auto-enable-agents = "no"
```

3. LLDP サービスを再起動します。

```
# svcadm restart svc:/network/lldp:default
```

4. 以前の LLDP 構成が保持されている各ポートで、LLDP を無効にします。

- エージェントのモードを変更することによって LLDP を無効にするには

```
# lldpadm set-agentprop -p mode=disable agent
```

ここで、*agent* は LLDP エージェントであり、エージェントが有効になっている物理リンクによって識別されます。たとえば、LLDP を `net0` で有効にした場合、エージェントは `net0` になります。

- ポートから LLDP 構成を削除することによって LLDP を無効にするには

```
# lldpadm reset-agentprop -p mode agent
```

このコマンドでは、`mode` プロパティの値を設定しません。



注意 - `no` に設定されている `auto-enable-agents` が `yes` に切り替わると、LLDP は、そのポートのエージェントモードが単に無効にされた場合と異なる動作をします。

LLDP エージェントのモニタリング

`lldpadm show-agent` コマンドでは、LLDP エージェントによって通知される完全な情報が表示されます。特定のシステムに応じて、通知は、ネットワークのほかの部分に送信される、そのローカルシステムに関する情報であるか、そのシステムによって受信される、同じネットワーク上のほかのシステムからの情報である場合があります。

通知される情報の表示

情報はローカルまたはリモートのいずれかにできます。ローカル情報は、ローカル LLDP エージェントから得られます。リモート情報は、ネットワーク上のほかの LLDP エージェントから得られ、ローカル LLDP エージェントによって受信されます。

通知される情報を表示するには、`lldpadm show-agent` コマンドを使用します。

```
# lldpadm show-agent -[l|r][v] agent
```

- `-l` は、ローカル LLDP エージェントによって通知されるローカル情報を表示します。
- `-r` は、LLDP エージェントによって受信されるリモート情報を表示します。
- `-v` は、詳細なローカル情報またはリモート情報を表示します。

例 33 通知される LLDP エージェント情報の表示

次の例は、LLDP エージェントによってローカルに、またはリモートから通知されている情報を表示する方法を示しています。デフォルトでは、情報は省略形式で表示されます。`-v` オプションを使用すると、詳細情報を取得できます。

LLDP エージェントによって通知されるローカル情報を表示するには

```
# lldpadm show-agent -l net0
AGENT CHASSISID PORTID
net0 004bb87f 00:14:4f:01:77:5d
```

LLDP エージェントによって通知されるリモート情報を表示するには

```
# lldpadm show-agent -r net0
AGENT SYSNAME CHASSISID PORTID
net0 hostb 0083b390 00:14:4f:01:59:ab
```

冗長モードでローカル情報を表示するには、`-v` オプションを使用します。

```
# lldpadm show-agent -l -v net4
Agent: net4
Chassis ID Subtype: Local(7)
Chassis ID: 00843300
Port ID Subtype: MacAddress(3)
Port ID: 00:1b:21:89:03:d0
Port Description: --
Time to Live: 21 (seconds)
System Name: --
System Description: --
Supported Capabilities: --
Enabled Capabilities: --
Management Address: --
Maximum Frame Size: --
Port VLAN ID: --
VLAN Name/ID: vlan1/22
VNIC PortID/VLAN ID: 02:08:20:63:2d:9d,02:08:20:e5:6c:af/21
```

```

Aggregation Information: --
    PFC Willing: On
    PFC Cap: 8
    PFC MBC: False
    PFC Enable: 4
    PFC Pending: True
Application(s)(ID/Sel/Pri): --
    ETS Willing: On
    ETS Configured CBS: 0
    ETS Configured TCS: 8
    ETS Configured PAT: 0,1,2,3,4,5,6,7
    ETS Configured BAT: 40,20,0,40,0,0,0,0
    ETS Configured TSA: 2,2,2,2,2,2,2,2
    ETS Recommended PAT: 0,1,2,3,4,5,6,7
    ETS Recommended BAT: 40,20,0,40,0,0,0,0
    ETS Recommended TSA: 2,2,2,2,2,2,2,2
    EVB Mode: Station
    EVB GID (Station): Not Supported
    EVB ReflectiveRelay REQ: Not Requested
    EVB ReflectiveRelay Status: RR Not Enabled
    EVB GID (Bridge): Not Supported
    EVB ReflectiveRelay Capable (RRCAP): Not Supported
    EVB ReflectiveRelay Control (RRCTR): Not Enabled
    EVB max Retries (R): 0
    EVB Retransmission Exponent (RTE): 0
    EVB Remote or Local(ROL) and
    Resource Wait Delay (RWD): Local
    EVB Resource Wait Delay (RWD): 0
    EVB Remote or Local (ROL) and
    Reinit Keep Alive (RKA): Local
    EVB Reinit Keep Alive (RKA): 0
    Next Packet Transmission: 4 (seconds)

```

冗長モードでリモート情報を表示するには、`-v` オプションを使用します。

```

# lldpadm show-agent -r -v net4
Agent: net4
Chassis ID Subtype: Local(7)
Chassis ID: 00843300
Port ID Subtype: MacAddress(3)
Port ID: 00:1b:21:89:03:d0
Port Description: --
Time to Live: 21 (seconds)
System Name: --
System Description: --
Supported Capabilities: --
Enabled Capabilities: --
Management Address: --
Maximum Frame Size: --
Port VLAN ID: --
VLAN Name/ID: vlan1/22
VNIC PortID/VLAN ID: 02:08:20:63:2d:9d,02:08:20:e5:6c:af/21
Aggregation Information: --
    PFC Willing: On
    PFC Cap: 8
    PFC MBC: False
    PFC Enable: 4
Application(s)(ID/Sel/Pri): --
    ETS Willing: On
    ETS Configured CBS: 0
    ETS Configured TCS: 8
    ETS Configured PAT: 0,1,2,3,4,5,6,7
    ETS Configured BAT: 40,20,0,40,0,0,0,0
    ETS Configured TSA: 2,2,2,2,2,2,2,2

```



```

ETS Recommended PAT: 0,1,2,3,4,5,6,7
ETS Recommended BAT: 40,20,0,40,0,0,0,0
ETS Recommended TSA: 2,2,2,2,2,2,2,2
  EVB Mode: Station
    EVB GID (Station): Not Supported
  EVB ReflectiveRelay REQ: Not Requested
  EVB ReflectiveRelay Status: RR Not Enabled
    EVB GID (Bridge): Not Supported
  EVB ReflectiveRelay Capable (RRCAP): Not Supported
  EVB ReflectiveRelay Control (RRCTR): Not Enabled
    EVB max Retries (R): 0
  EVB Retransmission Exponent (RTE): 0
    EVB Remote or Local(ROL) and
      Resource Wait Delay (RWD): Local
    EVB Resource Wait Delay (RWD): 0
  EVB Remote or Local (ROL) and
    Reinit Keep Alive (RKA): Local
  EVB Reinit Keep Alive (RKA): 0
  Information Valid Until: 19 (seconds)

```

LLDP 統計情報の表示

ローカルシステムまたはリモートシステムによって通知されている LLDP パケットに関する情報を取得するために、LLDP 統計情報を表示できます。この統計情報は、LLDP パケットの送信および受信を含む重大なイベントを示します。

- LLDP パケットの送信および受信に関するすべての統計情報を表示するには

```
# lldpadm show-agent -s agent
```

- 選択された統計情報を表示するには、`-o` オプションを使用します。

```
# lldpadm show-agent -s -o field[,field,...]agent
```

ここで、*field* は `show-agent -s` コマンドの出力にあるいずれかのフィールド名を示します。

例 34 LLDP パケット統計情報の表示

この例は、LLDP パケット通知に関する情報を表示する方法を示しています。

```
# lldpadm show-agent -s net0
AGENT IFRAMES IERR IDISCARD OFRAMES OLENERR TLVDISCARD TLVUNRECOG AGEOUT
net0      9      0          0      14          0          4          5          0
```

この出力では、次の情報が提供されます。

- AGENT は、LLDP エージェントの名前を指定します。これは、この LLDP エージェントが有効になっているデータリンクの名前と同じです。
- IFRAMES、IERR、および IDISCARD には、受信されているパケット、エラーを含む受信パケット、および破棄された受信パケットに関する情報が表示されます。

- OFRAMES と OLENERR は、送信パケットと、長さのエラーを含むパケットを示します。
- TLVDISCARD と TLVUNRECOG には、破棄された TLV ユニットと、認識されない TLV ユニットに関する情報が表示されます。
- AGEOUT は、タイムアウトしたパケットを示します。

この例は、システムで受信した 9 フレームのうち、おそらく標準に準拠していないために 5 つの TLV ユニットが認識されないことを示します。この例はまた、ローカルシステムによって 14 フレームがネットワークに送信されたことも示しています。

例 35 選択された LLDP パケット統計情報の表示

この例では、選択された統計情報を表示する方法を示します。

```
# # lldpadm show-agent -s -o iframes,oframes net4
IFRAMES  OFRAMES
0          10
```

データセンターブリッジングを使用した集中ネットワークの管理

従来より、アプリケーション要件に基づいてトラフィックを管理する場合や、使用可能な帯域幅に基づいてネットワークトラフィックを負荷分散する場合に異なるネットワークが使用されます。たとえば、ローカルエリアネットワーク (LAN) は Ethernet を使用し、ストレージエリアネットワーク (SAN) はファイバチャネルを使用します。ただし、データセンターブリッジングは Ethernet を拡張して、異なるタイプのトラフィック (集中トラフィック) の実行をより適切に行えるようにし、無損失などの機能をサポートできるようにします。DCB は、SAN や LAN を統合して大規模な設備の運用および管理コストを削減することにより、効率的なネットワークインフラストラクチャーを可能にします。

この章の内容は、次のとおりです。

- [123 ページの「データセンターブリッジングの概要」](#)
- [125 ページの「優先順位ベースのフロー制御」](#)
- [126 ページの「拡張伝送選択」](#)
- [127 ページの「DCBX の有効化」](#)
- [128 ページの「DCB の操作モードの設定」](#)
- [130 ページの「DCB の優先順位ベースのフロー制御のカスタマイズ」](#)
- [132 ページの「PFC 構成情報の表示」](#)
- [135 ページの「アプリケーション優先順位の構成」](#)
- [135 ページの「DCB の拡張伝送選択のカスタマイズ」](#)
- [138 ページの「ピアに対する ETS 構成の推奨」](#)
- [140 ページの「ETS 構成情報の表示」](#)

データセンターブリッジングの概要

データセンターブリッジングは、ネットワークとストレージのプロトコルの間でデータリンクを共有する場合など、同じネットワークリンクを共有する際に複数のトラフィックタイプの帯域幅、関連する優先順位、およびフロー制御を管理するため

に使用されます。このようなトラフィックのホスティングにはファイバチャネルを専用に割り当てることができます。ただし、専用リンクをファイバチャネルトラフィックだけに使用すると、コストが高くなる可能性があります。そのため、FCoE (Fibre channel over Ethernet) がより一般的に使用されます。DCB は、Ethernet ネットワークのトラバース中に、パケットロスに対するファイバチャネルの感度を処理します。

DCB は、サービスクラス (CoS) 優先順位とも呼ばれる優先順位に基づいてトラフィックを区別します。ホストと次のホップは DCB 交換プロトコル (DCBX) を使用して、優先順位に基づいてネットワーク構成をネゴシエーションします (トラフィックロス削除や最小帯域幅共有など)。このプロセスを使用すると、ホスト上およびネットワーク内の異なるアプリケーションのパケットを優先順位に従って処理でき、対応する構成を DCBX を使用してネゴシエートできます。

Oracle Solaris は、IEEE 802.1qaz DCBX 仕様および標準化前の Converged Enhanced Ethernet (CEE) DCBX 仕様 v1.01 をサポートしており、DCB を使用するときには多様なスイッチとの相互運用を可能にします。ネゴシエーションの異なるバージョンを選択する方法の詳細については、[128 ページの「DCB の操作モードの設定」](#)を参照してください。

DCB ネットワーク内の各パケットは DCB 3 ビット優先順位の値 (DCB 優先順位) を含む VLAN ヘッダーを備えています。この IEEE 802.1p 優先順位の値により、ネットワーク内の各 Ethernet パケットとほかのパケットが区別されます。パケットの優先順位の値に応じて、特定の帯域幅をパケットに割り当てるように DCB を構成できます。たとえば、優先順位が 1 のすべてのパケットは PFC を有効にする必要があり、優先順位が 2 のすべてのパケットは PFC を無効にし、帯域幅を 10% 共有する必要があります。

優先順位ベースのフロー制御 (PFC) や拡張伝送選択 (ETS) などの DCB 機能を優先順位に基づいて構成できます。PFC および ETS の詳細は、[125 ページの「優先順位ベースのフロー制御」](#)と[126 ページの「拡張伝送選択」](#)を参照してください。

DCB cos データリンクプロパティを使用すると、データリンクの CoS または優先順位を指定できます。プライマリデータリンク上に設定された cos 値は、この物理リンク上に作成された VNIC に適用されません。cos プロパティに基づく PFC のカスタマイズについては、[130 ページの「DCB の優先順位ベースのフロー制御のカスタマイズ」](#)を参照してください。cos プロパティに基づく ETS のカスタマイズについては、[135 ページの「DCB の拡張伝送選択のカスタマイズ」](#)を参照してください。

Oracle Solaris では、DCBX TLV (Type-Length-Value) ユニットの交換に LLDP が使用されます。LLDP の詳細は、[第6章「リンク層検出プロトコルによるネットワーク接続情報の交換」](#)を参照してください。ベースとなるネットワークインタフェースカード (NIC) が DCB 機能 (優先順位ベースのフロー制御や拡張伝送選択など) をサポートしている場合、これらの機能の構成情報を次のようにネットワーク上のピアと共有できます。

- PFC は、定義されたサービスクラス (CoS) を持つパケットに対して、トラフィックフローを一時停止するメカニズムを実装することにより、パケットロスを防止します。

す。CoSの詳細については、[dladm\(1M\)](#)のマニュアルページで `cos` リンクプロパティの説明を参照してください。

- ETSは定義されているCoSに基づくパケット間での帯域幅共有を可能にします。[126 ページの「拡張伝送選択」](#)を参照してください。

DCBを使用する場合の考慮事項

DCBの使い方に関しては、次の考慮事項に注意してください。

- DCBはIntel Niantic 物理NICでのみサポートされています。
NICがDCBをサポートしているかどうかを確認するには、次のコマンドを発行します。

```
# dladm show-linkprop -p ntcs agent
```

ゼロ (0) より大きいプロパティ値は、NICがDCBをサポートしていることを示します。

- DCBモードで構成されているDCBポートは集約できません(トランクまたはDLMPモード)。
- DCBは、CINバージョンではなくDCBXのIEEEおよびCEEバージョンのみをサポートするため、外部ブリッジはOracle Solaris DCBと対話するIEEEまたはCEEバージョンをサポートする必要があります。
- DCBはETS構成および推奨TLVをサポートしています。
- DCBは8つのトラフィッククラス構成でのみサポートされています。
- DCBは輻輳通知(CN)をサポートしていません。

優先順位ベースのフロー制御

優先順位ベースのフロー制御(PFC)はIEEE 802.1p CoS値を含むように標準のIEEE 802.3x PAUSEフレームを拡張します。PFCでは、PAUSEフレームの送信時にリンク上のすべてのトラフィックを停止する代わりに、PFCフレーム内で有効になっているCoS値のトラフィックだけを停止します。PFCフレームは、トラフィックを一時停止する必要のある有効な`cos`プロパティ値に対して送信されます。送信側ホストはその`cos`プロパティ値のトラフィックを停止しますが、ほかの無効な`cos`プロパティ値のトラフィックへの影響はありません。PFCフレームで指定された時間間隔のあと、一時停止されたパケットに対して伝送が再開されます。

CoS値に基づく一時停止により、その`cos`プロパティ値のパケットが破棄されることはなくなります。定義されたCoS値がないパケット、またはPFCが有効になっていないCoS値を持つパケットに対しては、PAUSEフレームは送信されません。した

がって、トラフィックフローは継続し、トラフィックの輻輳時にパケットが破棄される可能性があります。パケットロスの処理はプロトコルスタック (TCP など) によって異なります。

ホスト上には2種類の DCB 情報 (ローカル DCB 情報とリモート DCB 情報) が存在します。PFC 機能が有効になるには、ホスト上の PFC に関するローカルタイプおよびリモートタイプの DCB 情報が対称でなければなりません。ローカルホストはピアから受信する DCB 情報を照合する必要があります。DCB をシステム上で有効にする場合、DCB はピアと DCB 情報を同期します。

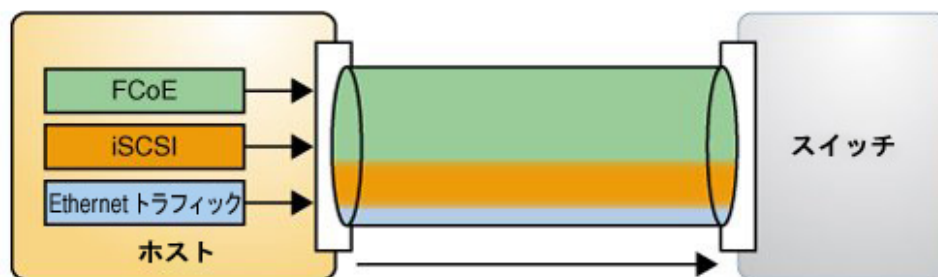
ほとんどの場合、PFC のデフォルトの構成で十分です。この構成は、LLDP を有効にすると自動的に設定されます。ただし、PFC の構成時に異なるオプションを調整できます。詳細は、[130 ページの「DCB の優先順位ベースのフロー制御のカスタマイズ」](#)と[132 ページの「PFC 構成情報の表示」](#)を参照してください。

拡張伝送選択

ETS は、アプリケーションの DCB 優先順位に基づいて NIC 上の帯域幅をアプリケーションに割り当てることができる DCB 機能です。

次の図はネットワーク内の DCB の ETS 機能を示します。

図 14 DCB の拡張伝送選択



この図のホストには、リンク帯域幅を共有する異なるタイプのトラフィック (FCoE や iSCSI など) が含まれています。図では、異なるタイプのトラフィックに対して次の表に示すような優先順位および帯域幅が割り当てられます。

トラフィック	優先順位	帯域幅
FCoE	3	60%
iSCSI	4	30%
Ethernet トラフィック (iSCSI 以外)	0	10%

対応する ETS 帯域幅の割り当てがある DCBX ETS TLV は次のとおりです。

優先順位	0	1	2	3	4	5	6	7
帯域幅の割り当て割合	10	0	0	60	30	0	0	0

ETS 機能を使用するには、NIC が機能をサポートしており、DCB モードで実行する必要があります。LLDP を有効にすると、ベースとなるリンクが DCB をサポートする場合に ETS 機能のデフォルト構成が自動的に設定されます。ただし、デフォルト構成を変更できません。詳細は、[135 ページの「DCB の拡張伝送選択のカスタマイズ」](#)、[138 ページの「ピアに対する ETS 構成の推奨」](#)、および [140 ページの「ETS 構成情報の表示」](#) を参照してください。

DCBX の有効化

LLDP を有効にすると、DCBX のサポートは自動的に有効になります。この手順では、何らかの自動プロセスが失敗した場合に手動で行う代替手順を示します。

▼ データセンターブリッジングの交換機能を手動で有効にする方法

始める前に LLDP がインストールされていることを確認します。LLDP の有効化の詳細は、[109 ページの「システムでの LLDP の有効化」](#) を参照してください。

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. LLDP サービスが実行されていることを確認します。

```
# svcs llpd
```

LLDP サービスが無効になっている場合は、次のコマンドでサービスを起動します。

```
# svcadm enable svc:/network/lldp:default
```

3. LLDP エージェントが Rx および Tx モードで実行されていることを確認します。

```
# llpdadm show-agentprop -p mode agent
```

LLDP エージェントが両方のモードで有効でない場合は次のコマンドを入力します。

```
# llpdadm set-agentprop -p mode=both agent
```

ほかの取り得る LLDP エージェント構成については、[109 ページの「システムでの LLDP の有効化」](#)を参照してください。

4. ベースとなる NIC が DCB をサポートしていることを確認してください。

```
# dladm show-linkprop -p ntcs agent
```

ゼロ (0) より大きいプロパティ値は、NIC が DCB をサポートしていることを示します。

DCB の操作モードの設定

Oracle Solaris ホストは、IEEE DCBX または CEE DCBX プロトコルを使用して、直接接続されたピア (最初のホップスイッチなど) と DCB 機能に関する情報を交換できます。この交換により、ホストとピアの両方で DCB 機能のネゴシエーションおよび構成が可能になります。

次の表は、IEEE DCBX および CEE DCBX バージョンでサポートされている DCBX アプリケーションを示しています。

表 5 IEEE DCBX および CEE DCBX バージョンでサポートされるアプリケーション

アプリケーション	IEEE DCBX	CEE DCBX
PFC	はい	はい
アプリケーション TLV	はい	はい
ETS 構成	はい	いいえ
ETS 推奨	はい	いいえ

スイッチがサポートする標準に応じて、`ieee`、`cee`、または `auto` を選択できます。デフォルトのモードは `auto` で、デフォルトで IEEE モードで動作し、ホストがピアから CEE パケットを受信すると CEE モードに切り替わります。モードを選択するには、`llpdadm` コマンドを使用してプロパティ `dcbx-version` を設定します。

明示的にモードを設定した場合は、CEE または IEEE パケットを受信しても、モードが CEE または IEEE に遷移しません。

IEEE から CEE への遷移は 1 回のみ発生します。その後は、ピアが DCBX モードに変化したときに Oracle Solaris はモードを自動的に切り替えません。

一部のスイッチは、ピアに関係なく情報の交換を開始します。ただし、一部のスイッチはピアを待機して、ピアが DCBX バージョンをサポートする場合のみ応答することがあります。たとえば、Oracle Solaris ホストの DCBX モードを `auto` として構成した場合、デフォルトでは IEEE DCBX パケットを送信します。このホストが IEEE DCBX をサポートしないスイッチに接続されると、CEE DCBX バージョンがサポートされている場合であっても、スイッチが応答しない可能性があります。そのような場合は、DCBX モードを明示的に `cee` として構成する必要があります。

▼ DCB の操作モードを設定する方法

1. 管理者になります。
2. (オプション) 現在の DCBX モードを表示します。

```
# llpdadm show-agentprop -p dcbx-version net0
AGENT      PROPERTY      PERM VALUE  EFFECTIVE    DEFAULT      POSSIBLE
net0       dcbx-version  rw  auto      ieee         auto         auto,ieee,cee
```

3. 操作モードを設定します。

```
# llpdadm set-agentprop -p dcbx-version=DCBX-mode net0
```

ここで `DCBX-mode` は、次のいずれかの値に設定できます。

- `auto` – デフォルトの DCBX モード。モードが `auto` の場合、DCBX は IEEE モードで動作し、ホストがピアから CEE パケットを受信すると CEE モードに切り替わります。
- `ieee` – 情報を交換するときに IEEE プロトコルのみを使用します。
- `cee` – 情報を交換するときに CEE プロトコルのみを使用します。

4. (オプション) 現在のモードを表示します。

```
# llpdadm show-agentprop -p dcbx-version net0
AGENT      PROPERTY      PERM VALUE  EFFECTIVE    DEFAULT      POSSIBLE
net0       dcbx-version  rw  auto      cee         auto         auto,ieee,cee
```

ヒント - DCBX モードをデフォルト設定にリセットするには、次のコマンドを使用します。

```
# llpdadm reset-agentprop -p dcbx-version net0
```

DCB の優先順位ベースのフロー制御のカスタマイズ

PFC および ETS はベースとなる NIC が DCB モードの場合のみデフォルトで有効になります。PFC だけを使用する場合は、次のコマンドを使用して LLDP エージェントの dot1 -tlv プロパティから etscfg を削除する必要があります。

```
# lladm set-agentprop -p dot1-tlv==etscfg net0
```

dot1 -tlv の取り得る値のリストについては、[表2](#)を参照してください。

PFC 関連データリンクプロパティの設定

DCB の PFC 機能は次のデータリンクプロパティを提供します。

- pfcmap – 優先順位の定義およびマッピングの情報を提供します。pfcmap プロパティは優先順位を表す 8 ビットマスク (0-7) を指します。最下位ビットは優先順位 0 を表し、最上位ビットは優先順位 7 を表します。このマスクの各ビットは、それに対応する優先順位に対して PFC が有効かどうかを示します。デフォルトでは、pfcmap は 1111111 に設定され、したがって、すべての優先順位に対して PFC が有効になります。
- pfcmap-rmt – リモートピアで有効になっている PFC マッピングを指定します。このプロパティは読み取り専用です。

DCB ネットワークでは、受信側がトラフィックの受信速度に対応できない場合は、送信側に PFC フレームを送信して、PFC が有効になっている優先順位のトラフィックを一時停止するように送信者に要求します。受信側ホストでトラフィックの輻輳が蓄積した場合、リンク上で伝送される任意のパケットについて、DCB により送信側ホストに PFC フレームが送信されます。PFC フレームを正しく送信するには、通信を行うホストの DCB 構成情報が対称でなければなりません。システムは、リモートピアの PFC 構成と一致するように自身の PFC 構成を自動的に調整できます。pfcmap プロパティの EFFECTIVE 値を表示する dladm show-linkprop コマンドを使用して、ローカルホスト上で動作する PFC のマッピングを決定できます。詳細は、[132 ページの「データリンクプロパティの表示」](#)を参照してください。

▼ DCB の優先順位ベースのフロー制御をカスタマイズする方法

1. 管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. データリンクの `flowctrl` プロパティが `pfc` に設定されていることを確認します。

```
# dladm show-linkprop -p flowctrl datalink
```

プロパティが `pfc` や `auto` に設定されていない場合は次のコマンドを使用します。

```
# dladm set-linkprop -p flowctrl=pfc datalink
```

3. `pfcmap` プロパティをデフォルト値の `11111111` 以外の値に設定します。

```
# dladm set-linkprop -p pfcmap=value datalink
```

たとえば、CoS 優先順位 6 だけを有効にするには、次のコマンドを入力します。

```
# dladm set-linkprop -p pfcmap=01000000 net0
```

PFC TLV ユニットの設定

DCB は PFC TLV ユニットを使用してホスト間で PFC 情報を交換します。いずれかのホストの `pfcmap` プロパティが同じ値であるか、少なくとも 1 つのホストがピアの構成を受け入れる用意がある必要があります。 `dladm set-linkprop` コマンドを使用して `pfcmap` プロパティを設定できます。

TLV プロパティ `willing` は、ホスト構成がピアの構成と異なる場合にホストがピアの構成を受け入れる準備ができていないかどうかを示します。デフォルトでは、`willing` のプロパティ値は、ホストがピアの構成を受け入れることを示す `on` に設定されています。

ホストが自身の PFC 情報とリモートピアの PFC 情報を同期できることを確認するには、次のコマンドを使用して、`willing` が `on` に設定されているかどうかを判断する必要があります。

```
# lldpdm show-agenttlvprop -p willing -a agent pfc
```

PFC TLV プロパティ `willing` が `off` に設定されている場合は、次のコマンドを入力してプロパティ `willing` を `on` に設定し、同期を有効にします。

```
# lldpdm set-agenttlvprop -p willing=on -a agent pfc
```

この `agent` は、エージェントが有効になっているデータリンクです。

例 36 ホストとピアの間での同期の有効化

`net0` データリンクの同期を有効にするには、次のコマンドを入力します。

```
# lldpdm set-agenttlvprop -p willing=on -a net0 pfc
```

```
# dladm show-linkprop -p pfcmap,pfcmap-rmt net0
LINK PROPERTY PERM VALUE EFFECTIVE DEFAULT POSSIBLE
net0 pfcmap rw 11111111 00010000 11111111 00000000-11111111
net0 pfcmap-rmt r- -- -- 00010000 11111111 --
```

たとえば、`pfcmap` および `pfcmap-rmt` プロパティの値は `00010000` です。これはローカルホストがピアと同期していることを示します。したがって、PFC はホストとピアの両方で優先順位 4 で有効になります。

詳細は、[lldpdm\(1M\)](#) のマニュアルページを参照してください。

PFC 構成情報の表示

このセクションでは、LLDP と DCB の構成後に PFC に関連する情報を表示するコマンドについて説明し、これらのコマンドの使用を示す例を提供します。

データリンクプロパティの表示

次のコマンドは、優先順位の定義とデータリンク上で有効になっている PFC マッピングを表示します。

```
# dladm show-linkprop -p pfcmap,pfcmap-rmt datalink
```

ローカルピアとリモートピアの間で PFC 情報が一致しているデータリンクの場合、`pfcmap` プロパティに設定されている値にかかわらず、`pfcmap` プロパティと `pfcmap-rmt` プロパティの `EFFECTIVE` 列の値は同じになります。ローカルホストで同期機能が無効になっている場合、`pfcmap` プロパティの `EFFECTIVE` フィールドにはローカルホストの `pfcmap` プロパティの値が反映されます。

例 37 PFC 関連のデータリンクプロパティの表示

この例では、優先順位ベースのフロー制御に関連する物理データリンクプロパティのステータスを表示する方法を示します。

```
# dladm show-linkprop -p pfcmap,pfcmap-rmt net0
LINK PROPERTY PERM VALUE EFFECTIVE DEFAULT POSSIBLE
net0 pfcmap rw 11111111 11111111 11111111 00000000-11111111
net0 pfcmap-rmt r- -- -- -- --
```

たとえば、`pfcmap` プロパティの `value` フィールドの値は `11111111` です。この値は、ローカルホストの PFC マッピングがデフォルト値で、8 つの優先順位がすべて

有効になっていることを示しています。pfcmap および pfcmap-rmt プロパティの EFFECTIVE 値は 11111111 および -- です。EFFECTIVE フィールドのこれらの値が一致していないことは、ローカルホストが PFC 情報をリモートピアと同期していないことを示しています。

lldpadm show-agenttlvprop コマンドを使用して willing プロパティの値を確認でき、lldpadm show-agent -r コマンドを使用してピアの PFC TLV 情報を確認できます。

ローカルホストで PFC 情報を同期する機能の表示

次のコマンドは、ホストが PFC マッピングをピアと同期する機能を制御する PFC TLV プロパティを表示します。

```
# lldpadm show-agenttlvprop -a agent pfc
```

この *agent* は、LLDP が有効になっているデータリンクによって識別されます。

例 38 ローカルホストで PFC 情報を同期する機能の表示

この例では、ホストがピアの PFC 構成に合わせて調整する機能の、現在のステータスを表示する方法を示します。

```
# lldpadm show-agenttlvprop -a net0 pfc -p willing
AGENT  TLVNAME  PROPERTY  PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
net0    pfc         willing   rw    off    --         on       on,off
```

詳細は、[131 ページの「PFC TLV ユニットの設定」](#)を参照してください。

ホストとピアの間の PFC マッピング情報の表示

PFC Pending 値は、ホストとピアの間で PFC 情報が収束しない場合に True ステータスを返します。不一致が解決されたあとは、PFC Pending のステータスは False に戻ります。

次のコマンドは、ローカルホストとピアの間で PFC マッピング情報が一致しないことをユーザーに警告します。

```
# lldpadm show-agent -lv -o "PFC Pending" agent
# lldpadm show-agent -lv -o "PFC Pending" agent
```

例 39 ホストとピアの間で PFC 情報が対称であることの確認

次の例では、実際の実行時にホストとピアの間で PFC 情報が同期されているか、あるいは不一致が発生しているかを確認する方法を示します。

```
# lldpdm show-agent -lv -o "PFC Pending" net0
PFC Pending: True
```

エージェントによって通知されるすべての情報を表示するには、`lldpdm show-agent` コマンドの `-v` (verbose) オプションを使用します。

```
# lldpdm show-agent -v net0
```

優先順位定義の表示

次のコマンドは、NIC で有効になっている優先順位に関連する、物理リンクの PFC 情報を表示します。

```
# dladm show-phys -D pfc datalink
```

例 40 CoS 優先順位定義の表示

この例では、`pfcmmap` プロパティの値に基づいて特定の物理リンクの現在の優先順位定義を表示する方法を示します。たとえば、`pfcmmap` が `01000000` に構成されているとします。対応する物理リンクの優先順位マッピングを表示するには、次のように操作します。

```
# dladm show-phys -D pfc net0
LINK COS PFC PFC_EFFECT CLIENTS
net0 0 YES NO net0,vnic1
      1 YES YES vnic2
      2 YES NO vnic3
      3 YES NO vnic4
      4 YES NO vnic5
      5 YES NO vnic6
      6 YES NO vnic7
      7 YES NO vnic8
```

物理リンク `net0` では、データリンク上に構成されているすべての VNIC クライアントに対して優先順位が有効になっています。ただし、ローカルホストは自身の PFC マッピングをピアの PFC マッピングに合わせて調整するため、`PFC_EFFECT` フィールドの値で示されているように、`COS 0` および `2-7` では優先順位が無効になっています。そのため、リソースが使用可能かどうかにかかわらず、`vnic2` を除くすべての VNIC のトラフィックに対して PFC フレームの交換は行われません。この構成では、`vnic2` 以外のすべての VNIC を通過するトラフィックで、パケットの破棄が許可されます。`vnic2` のトラフィックに対しては、トラフィックの輻輳が発生すると、このクライアントでのパケットロスを防ぐために `PFC PAUSE` フレームが送信されます。

アプリケーション優先順位の構成

DCBX により、アプリケーションに関連付けられている優先順位情報が交換されます。DCBX を介して交換されるアプリケーション TLV ユニットには、ホスト上のアプリケーションに使用される優先順位の情報が含まれます。優先順位はアプリケーション優先順位表で定義されます。表の各エントリには、アプリケーションの名前およびアプリケーションに割り当てられた優先順位が含まれます。アプリケーションの優先順位を設定すると、その優先順位の PFC および ETS の DCB 設定はすべてアプリケーションに適用できます。アプリケーション TLV は、アプリケーション優先順位情報をほかのホストと交換するためにこの表データを使用します。

デフォルトでは、アプリケーション機能はピアから優先順位マッピングを受け入れます。アプリケーション TLV `appln` の `willing` プロパティは PFC に似ており、ピア間の情報交換を可能にします。

この表のエントリでは、次の形式が使用されます。

protocol-ID/selector/priority

protocol-ID と *selector* のペアによってアプリケーションが識別されます。対応するアプリケーションの優先順位は、0 から 7 までの値を含む優先順位によって識別されます。

アプリケーションの優先順位に関するこの情報をほかのホストと交換するには、アプリケーション TLV を次のように設定します。

```
#lldpdm set-agenttlvprop -p property=value -a agent appln
```

たとえば、FCoE トラフィックの場合、プロトコル ID は 0x8906、セレクタ ID は 1 です。このアプリケーションに優先順位 4 が割り当てられているとします。アプリケーション TLV の設定用パラメータを示す表 4 に基づいて、次のコマンドを入力します。

```
#lldpdm set-agenttlvprop -p apt=8906/1/4 -a net0 appln
#lldpdm show-agenttlvprop -a net0 appln -p apt
```

AGENT	TLVNAME	PROPERTY	PERM	VALUE	EFFECTIVE	DEFAULT	POSSIBLE
net0	appln	apt	rw	8906/1/4	8906/1/4	--	--

DCB の拡張伝送選択のカスタマイズ

LLDP が有効になっており、ベースとなるリンクが DCB をサポートしている場合、デフォルト構成は自動的に設定されます。このデフォルト構成では、`cos` 値 0 がすべての帯域幅に割り当てられます。ただし、`dladm set-linkprop` コマンドを使用してデータリンクに `cos` 値を構成し、一部の帯域幅をそのデータリンクに割り当てることができます。

ETS 構成および推奨の TLV は NIC に対してデフォルトで有効になります。dot1-tlv の取り得る値のリストについては、表2を参照してください。

pfic TLV を削除する場合は次のコマンドを入力します。

```
# llpadm set-agenttlvprop -p dot1-tlv=-pfic agent
```

ETS 関連データリンクプロパティの設定

PFC 情報を参照するデータリンクプロパティは、パケットに定義されている優先順位に基づくパケットロス防止に適用されます。ETS プロパティは、ベースとなるリンクの帯域幅共有の優先順位に基づく割り当てに関連しています。

DCB には次の ETS 関連のプロパティが用意されています。

- **cos** – データリンクのサービスクラスまたは優先順位を指定します。このプロパティの値の範囲は 0 から 7 です。デフォルト値は 0 です。cos 値は、このリンク上で転送されるパケットの VLAN タグに設定されます。

- **etsbw-lcl** – データリンクの送信 (Tx) 側に割り当てられた ETS 帯域幅を示します。ベースとなる物理 NIC が DCB 機能を持ち、ETS をサポートしており、リンクの cos プロパティが 0 に設定されている場合のみ、このプロパティは構成可能です。このプロパティの値をデータリンクに設定するには、ベースとなる物理リンクの合計帯域幅の割合を指定します。同じ物理 NIC 上にあるすべてのデータリンクの etsbw-lcl プロパティ値の合計は 100% を超えてはいけません。

etsbw-lcl に定義される帯域幅の割合は、そのデータリンクのみのために予約されてはいません。割り当てられた帯域幅が使用されていない場合は、その物理 NIC 上にあるほかのデータリンクがそれを使用できます。また、帯域幅割り当ては、ホストのトラフィックの送信側だけに適用されます。

- **etsbw-rmt-advice** – ピアに送信される推奨の ETS 帯域幅の値を指定します。デフォルトでは、etsbw-lcl プロパティのローカル構成値がピアに対して推奨されます。ただし、etsbw-rmt-advice データリンクプロパティを明示的に構成して、etsbw-lcl プロパティとは異なる値にすることを勧めます。

データリンクの帯域幅割り当てが非対称である場合、つまり、受信 (Rx) 帯域幅と送信 (Tx) 帯域幅が異なる場合に、etsbw-rmt-advice プロパティの構成が役立ちます。明示的に etsbw-rmt-advice プロパティを設定すると、ETS 推奨 DCBX TLV の伝送が自動的に開始されます。

- **etsbw-lcl-advice** – ピアがローカルホストに送信する、データリンクの推奨の ETS 帯域幅共有を指定します。これは読み取り専用プロパティです。
- **etsbw-rmt** – データリンクのピア上に構成された帯域幅共有を指定します。これは読み取り専用プロパティです。

優先順位を設定する場合と VNIC に帯域幅を割り当てる場合は、次のコマンドを使用します。

- VNIC に優先順位を設定するには、次のようにします。


```
# dladm set-linkprop -p cos=value VNIC
```

- ベースとなる物理リンクの帯域幅の割合を VNIC に割り当てるには、次のようにします。

```
# dladm set-linkprop -p etsbw-lcl=value VNIC
```

`etsbw-lcl` プロパティに割り当てる値は、ベースとなるリンクの合計帯域幅容量に対する割合を表します。クライアントに割り当てるすべての割り当て帯域幅値の合計は、100 パーセントを超えてはいけません。

- ピアに送信される帯域幅を明示的に推奨するには、次のようにします。

```
# dladm set-linkprop -p etsbw-rmt-advice=value VNIC
```

`dladm show-linkprop` コマンドを使用して、ローカルホストのデータリンクに実装されている実際の帯域幅共有と、ピアのデータリンク上に構成されている帯域幅共有を決定できます。`etsbw-lcl` および `etsbw-rmt` プロパティの出力の `EFFECTIVE` フィールド値は、実装されている実際の帯域幅共有を示します。詳細は、[140 ページの「ETS 構成情報の表示」](#)を参照してください。

特定の優先順位を持つパケットに適切な帯域幅が使用されるようにするには、通信を行うホスト間で ETS 情報が対称になっているか同期されている必要があります。特に、`etsbw-lcl-advice` の値に合わせて自身の帯域幅共有を調整できるローカルシステムである必要があります。Oracle Solaris システムはピアの ETS 推奨と一致するように自身の ETS 構成を自動的に調整できます。

ETS TLV ユニットの設定

ETS TLV (`etscfg`) 構成は、ホストがピアからの ETS 推奨に応答する方法を決定します。この TLV ユニットには構成可能なプロパティが 1 つだけあり、それは `willing` です。デフォルトでは、このプロパティは `on` に設定され、ローカルホストは自身の ETS 構成をリモートピアの ETS 推奨と同期できます。

ホストが自身の ETS 情報とリモートピアの ETS 情報を同期できることを確認するには、次のコマンドを使用します。

```
# llpdadm show-agenttlvprop -p willing -a agent etscfg
```

`willing` プロパティが `off` に設定されている場合は、次のコマンドを入力して同期を確立します。

```
# llpdadm set-agenttlvprop -p willing=on -a agent etscfg
```

特定のエージェントの情報の同期を防止するには、次のように `willing` プロパティを `off` に設定します。

```
# llpdadm set-agenttlvprop -p willing=off -a agent etscfg
```

この agent は、エージェントが有効になっているデータリンクです。

ピアに対する ETS 構成の推奨

ピアが同じ値を構成できるように、優先順位ごとに構成された ETS 帯域幅の値 (etsbw-lc1) をピアに対して推奨できます。ETS 帯域幅の値を推奨するために、LLDP エージェントの dot1-tlv タイプの etsreco プロパティを NIC 上で有効にする必要があります。推奨値はローカルに構成された ETS 値と同じ値を設定できます。または、dladm set-linkprop コマンドを使用して新しいデータリンクプロパティ etsbw-rmt-advice を設定し、明示的に推奨値を構成することもできます。データリンクに割り当てられた帯域幅が非対称である場合、つまり、受信 (Rx) 帯域幅と送信 (Tx) 帯域幅が異なる場合に、etsbw-rmt-advice プロパティの構成が役立ちます。

デフォルトでは、etsbw-lc1 プロパティの構成値はピアに値を推奨するために使用されます。ただし、etsbw-rmt-advice プロパティに異なる値を設定して別の ETS 値を推奨できます。たとえば、Tx でのネットワークトラフィックが多い場合、etsbw-lc1 プロパティ (ホスト上の Tx) に対して高い ETS 値を構成でき、etsbw-rmt-advice プロパティ (ホストへの Rx) に対して低い ETS 値を構成できます。

例 41 ピアに対する ETS 構成の推奨

1. net5 に対する LLDP エージェントの dot1-tlv タイププロパティを表示して、etsreco プロパティが有効であることを確認します。

```
# lldpadm show-agentprop -p dot1-tlv net5
AGENT  PROPERTY  PERM VALUE          EFFECTIVE  DEFAULT  POSSIBLE
net5   dot1-tlv  rw  etsreco,etscfg    etsreco,  none    none,vlanname,pvid,
                                     etscfg    linkaggr,pfc,appln,
                                               evb,etscfg,etsreco,
                                               all
```

2. ベースとなるリンクの帯域幅の 20% の共有を vnic1 に割り当てます。

```
# dladm set-linkprop -p etsbw-lc1=20 vnic1
# dladm show-linkprop -p etsbw-lc1 vnic1
LINK    PROPERTY    PERM VALUE          EFFECTIVE  DEFAULT  POSSIBLE
vnic1   etsbw-lc1   rw  20                20        0        --
```

デフォルトでは同じ値がピアに対して推奨されます。

```
# dladm show-linkprop -p etsbw-rmt-advice vnic1
LINK    PROPERTY    PERM VALUE  EFFECTIVE  DEFAULT  POSSIBLE
```

```
vnic1  etsbw-rmt-advice rw  --  20      0      --
```

- LLDP によって交換された情報を表示します。

```
# lldpadm show-agent -l -v net5
```

- ピアに対して推奨される帯域幅を表示します。

```
# dladm show-phys -D ets -r net5
LINK      COS  ETSBW_RMT_EFFECT  ETSBW_RMT_ADVICE  CLIENTS
--        0   0                  80                 net5
          1   0                  0                  --
          2   0                  0                  --
          3   0                  20                 vnic1
          4   0                  0                  --
          5   0                  0                  --
          6   0                  0                  --
          7   0                  0                  --
```

デフォルトでは、net5 の優先順位ごとに etsbw-lc1 プロパティに構成された値が、ピアの推奨値として送信されます。ETSBW_RMT_ADVICE はピアに対して推奨される値を示します。この出力は、ピアがその端に ETS 帯域幅を構成していないことも示します。lldpadm show-agent コマンドを使用して、ピアに対して推奨される帯域幅を表示することもできます。

- ピアに対して異なる値を推奨します。

```
# dladm set-linkprop -p etsbw-rmt-advice=10 vnic1
# dladm show-linkprop -p etsbw-rmt-advice vnic1
LINK      PROPERTY          PERM VALUE    EFFECTIVE    DEFAULT    POSSIBLE
vnic1     etsbw-rmt-advice rw  10          10          0          --
```

- ピアに対して推奨される帯域幅を表示します。

```
# dladm show-phys -D ets -r net5
LINK      COS  ETSBW_RMT_EFFECT  ETSBW_RMT_ADVICE  CLIENTS
--        0   0                  90                 net5
          1   0                  0                  --
          2   0                  0                  --
          3   0                  10                 vnic2
          4   0                  0                  --
          5   0                  0                  --
          6   0                  0                  --
          7   0                  0                  --
```

ETSBW_RMT_EFFECT フィールドは vnic2 の値 0 を示します。この値は、帯域幅の値が推奨されている場合でも、ピアがその端に帯域幅を設定していないことを示しています。この状況はピアが LLDP を有効にしていない可能性や、ETS をサポートしていないことを意味します。

ETS 構成情報の表示

次のコマンドを使用して ETS 構成に関する情報を表示できます。

```
■ # dladm show-linkprop -p etsbw-lcl,etsbw-rmt,etsbw-lcl-advice,etsbw-rmt-advice datalink
```

このコマンドは物理リンクの ETS に関連する情報を表示します。

```
■ # dladm show-phys -D ets phys-link
```

このコマンドは、リンク上の帯域幅の割り当てと分配に関連する、物理リンク全体のローカルおよびリモートの ETS 構成を表示します。

```
■ # lldpadm show-agenttlvprop -a agent etscfg
```

この *agent* は、LLDP が有効になっているデータリンクです。このコマンドは、ホストが ETS 情報をピアと同期する機能を制御する ETS TLV プロパティを表示します。

例 42 ETS 関連のデータリンクプロパティの表示

この例では、同期が有効になる前に ETS に関連するデータリンクプロパティのステータスを表示する方法を示します。

```
# dladm show-linkprop -p cos,etsbw-lcl,etsbw-rmt,etsbw-lcl-advice, \
etsbw-rmt-advice vnic1
LINK    PROPERTY          PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
vnic1   cos                rw    2      2          0        0-7
vnic1   etsbw-lcl          rw    10     10         0        --
vnic1   etsbw-rmt          r-    20     20         --       --
vnic1   etsbw-lcl-advice  r-    20     20         --       --
vnic1   etsbw-rmt-advice  rw    10     10         0        --
```

この出力は、2 の *cos* 値を使用して *vnic1* に対して、ホストが 10% の ETS 値を設定および推奨していることを示します。ただし、2 の *cos* 値を使用して *vnic1* に対して、ピアが 20% の ETS 値を設定および推奨しています。同期が有効ではない (*willing* が有効ではない) ため、ホストは *etsbw-lcl* プロパティの *EFFECTIVE* 値 (ローカル構成値) に反映されているピアの推奨を受け入れていません。

例 43 ローカルホストで ETS 情報を同期する機能の表示

この例では、ローカルホストがピアの ETS 構成に合わせて調整する機能の、現在のステータスを表示する方法を示します。

```
# lldpadm show-agenttlvprop -a net0 etscfg
AGENT  TLVNAME  PROPERTY  PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
net0   etscfg   willing  rw    off    --         on       on,off
```

同期を有効にするには、次のコマンドを入力します。

```
# lldpdm set-agentttlprop -p willing=on -a net0 etscfg

# dladm show-linkprop -p cos,etsbw-lcl,etsbw-rmt, \
etsbw-lcl-advice,etsbw-rmt-advice vnic1
LINK    PROPERTY      PERM  VALUE  EFFECTIVE  DEFAULT  POSSIBLE
vnic1   cos             rw    2      2          0        0-7
vnic1   etsbw-lcl      rw    10     20        0        --
vnic1   etsbw-rmt     r-    20     20        --       --
vnic1   etsbw-lcl-advice r-    20     20        --       --
vnic1   etsbw-rmt-advice rw    10     10        0        --
```

同期が有効 (`willing` プロパティが有効) であるため、ホストは `etsbw-lcl` の `EFFECTIVE` 値に反映されているピアの推奨を受け入れます。

次の例は、物理リンクの優先順位値ごとにホストおよびピアで有効な ETS 値を表示します。

```
# dladm show-phys -D ets net4
LINK    COS  ETSBW_LCL_EFFECT  ETSBW_RMT_EFFECT  ETSBW_LCL_SOURCE  CLIENTS
net4    0   0                 30                 local              net4
        1   0                 0                 local              --
        2   0                 0                 local              --
        3   0                 0                 local              --
        4   0                 70                local              --
        5   0                 0                 local              --
        6   0                 0                 local              --
        7   0                 0                 local              --
```

`ETSBW_LCL_EFFECT` 有効な ETS 帯域幅を優先順位に対する割合として表示します。

`ETSBW_RMT_EFFECT` 有効な ETS 帯域幅をピアの優先順位値に対する割合として表示します。

`ETSBW_LCL_SOURCE` `ETSBW_LCL_EFFECT` 値のソースを示します。この値は `local` (構成値) または `remote` (推奨値) のいずれかになります。

次の例は、ローカル構成値、ローカルの有効値、およびピアによる推奨値を含むローカルの ETS 情報を表示します。

```
# dladm show-phys -D ets -l net5
LINK    COS  ETSBW_LCL  ETSBW_LCL_EFFECT  ETSBW_LCL_ADVICE  CLIENTS
--      --      --          --                --                --
        0   80         80                0                 net5
        1   0          0                0                 --
        2   0          0                0                 --
        3   20         20                0                 vnic2
        4   0          0                0                 --
        5   0          0                0                 --
        6   0          0                0                 --
        7   0          0                0                 --
```

ピアはいずれの値も推奨していないため、ローカルの有効値 (`ETSBW_LCL_EFFECT`) はローカル構成値 (`ETSBW_LCL`) を使用して設定されます。

次の例はピアに関する情報を示します。

```
# dladm show-phys -D ets -r net5
LINK    COS  ETSBW_RMT_EFFECT  ETSBW_RMT_ADVICE  CLIENTS
```

ETS 構成情報の表示

```
--          0  0          20          net5
            1  0          0          --
            2  0          0          --
            3  0          80         vnic2
            4  0          0          --
            5  0          0          --
            6  0          0          --
            7  0          0          --
```

この出力は、優先順位 3 に対して 80% を設定することをホストがピアに推奨している場合でも、リモートピアが ETSBW_RMT_EFFECT ฟิลด์に値を設定していないことを示します。

リンクアグリゲーションと IPMP: 機能比較

リンクアグリゲーションとIPMPはネットワークパフォーマンスの向上とネットワークの可用性の維持を行うさまざまなテクノロジーです。

次の表は、リンクアグリゲーションと IPMP の全般的な比較を示します。

機能	リンクアグリゲーション	IPMP
ネットワークテクノロジーのタイプ	レイヤー 2 (リンク層)。	レイヤー 3 (IP 層)。
構成ツール	dladm	ipadm
リンクベースの障害検出	サポートされています。	サポートされています。
プローブベースの障害検出	トランク: LACP ベースであり、直接のピアホストまたはスイッチをターゲットとします。 DLMP: サポートされます。ICMP ベースであり、介在するレイヤー 2 スwitchの複数のレベルにわたって DLMP アドレスと同じサブネット内で定義された任意のシステムをターゲットとします。	ICMP ベースであり、介在するレイヤー 2 スwitchの複数のレベルにわたって検査用アドレスと同じ IP サブネット内で定義された任意のシステムをターゲットとします。
スタンバイインターフェースの使用	トランク: サポートされていません。 DLMP: サポートされていません。	サポートされています。スタンバイインターフェースを構成できます。
複数スイッチへのまたがり	トランク: サポートされています。ただし、スイッチのベンダー拡張機能が必要です。 DLMP: サポートされます。	サポートされています。
スイッチ構成	トランク: 必須。 DLMP: 必要ありません。	必要ありません。
バックツーバック構成	トランク: サポートされています。 DLMP: サポートされていません。	サポートされていません。
サポートされるメディアタイプ	Ethernet 固有。	ブロードキャスト可能。

機能	リンクアグリゲーション	IPMP
負荷分散のサポート	<p>トランク: サポートされており、管理者が <code>dladm</code> コマンドを使用して制御します。インバウンド負荷分散はサポートされます。</p> <p>DLMP: アグリゲーションのクライアントと VNIC の間でサポートされます。ただし、個々のクライアントおよび VNIC によるアグリゲーションへの負荷分散はサポートされていません。</p>	サポートされています。カーネルによって制御されます。インバウンド負荷分散は、発信元アドレス選択によって間接的に影響されます。
VNIC と統合する場合のサポートレベル	卓越したサポート。アグリゲーションは制御ドメインまたは大域ゾーンでのみ構成されており、ゾーンに対して透過的です。	<p>サポートされています。ただし、VNIC プロパティ (帯域幅制限、専用の Rx リングまたは Tx リング、リンク保護など) は IPMP グループに適用できません。</p> <p>複数の VNIC をゾーンに割り当てて、各ゾーンに構成する必要があります。</p>
ユーザー定義フローによるリソース管理	サポートされています。	サポートされていません。
リンク保護	サポートされています。	サポートされていません。
プロトコル要件	ありません。	ありません。

リンクアグリゲーションでは、受信トラフィックは、トランクモードでアグリゲーションを構成する複数のリンクにわたって分散されます。したがって、より多くの NIC を取り付けてアグリゲーションにリンクを追加すると、ネットワークのパフォーマンスが向上します。

DLMP アグリゲーションは複数のスイッチにまたがります。レイヤー 2 テクノロジとして、アグリゲーションはほかの Oracle Solaris 仮想化テクノロジーと最適に統合されません。

IPMP のトラフィックは IPMP インタフェースのデータアドレスを使用しますが、これは、それらのデータアドレスが、使用可能なアクティブインタフェースにバインドされるからです。たとえば、すべてのデータトラフィックは 2 つの IP アドレスの間のみを流れるが、それらのトラフィックが同じ接続上を流れるとはかぎらない場合には、より多くの NIC を追加しても IPMP のパフォーマンスは改善しません。なぜなら、使用可能な IP アドレスは依然として 2 つだけだからです。

推移的プローブのパケット形式

推移的なプローブパケットは、Ethernet タイプ ETHERTYPE_ORCL (0x881b) を使用した独自のプロトコルパケットです。推移的プローブについては、28 ページの「プローブベースの障害検出」を参照してください。プローブの統計情報を表示する例については、例7「プローブ関連情報の表示」を参照してください。次の図は推移的プローブパケットの形式を示しています。

図 15 推移的プローブパケット



フィールド	説明
Dest MAC	宛先 MAC アドレス。
Src MAC	ソース MAC アドレス。
Proto#	プロトコル番号。ETHERTYPE_ORCL パケットのレイヤー 2 ペイロードは 16 ビットのプロトコル番号 (推移的プローブパケットの場合は 1) で始まる必要があります。
Typ	プローブパケットのタイプ。プローブパケットのタイプは要求の場合は 0 であり、応答の場合は 1 です。
Pad	パディング(すべて 0 (ゼロ))。

フィールド	説明
Seq#	プローブのシーケンス番号。
Timestamp	プローブのタイムスタンプ。
TL	ターゲット情報の長さ。Ethernet の場合は、ターゲットの長さは 6 ビット (Ethernet MAC アドレスの長さ) です。
Target	ターゲットポートの MAC アドレス。
SL	ソース情報の長さ。Ethernet の場合は、ソースの長さは 6 ビットです。
Source	ソースポートの MAC アドレス。

索引

あ

アグリゲーション 参照 リンクアグリゲーション
アプリケーション優先順位構成, 135
アプリケーション TLV ユニット, 135, 135
 参照 PFC
移行
 VLAN, 64
インストール
 LLDP パッケージ, 110
エージェントごとの TLV ユニット, 108
オプションの TLV ユニット, 107

か

拡張伝送選択 参照 ETS
隔離 VLAN, 75
カスタマイズ
 DCB の PFC, 130
仮想ローカルエリアネットワーク 参照 VLAN
管理情報ベース (MIB), 105
グローバルな TLV ユニット, 108
高可用性
 DLMP アグリゲーション, 24
構成
 PVLAN, 80
コミュニティ VLAN, 75

さ

サービスクラス 参照 CoS
削除
 PVLAN, 81
 VLAN, 66
 アグリゲーションのリンク, 35
 ブリッジ, 98

ブリッジのリンク, 94

作成

PVLAN, 80
VLAN, 56
ブリッジ, 91
ブリッジの一部であるリンク上の VLAN, 99
リンクアグリゲーション, 30
リンクアグリゲーション上の VLAN, 60
レガシーデバイス上の VLAN, 62

指定

エージェントの LLDP パケットの TLV ユニ
ット, 114

情報

集約されたポートの情報, 41

新機能

CEE DCBX, 14
PVLAN, 13
プライベート仮想ローカルエリアネットワ
ーク, 13

推移的プローブ, 29

推奨

ピアに対する ETS 構成, 138

設定

DCB の操作モード, 128
ETS TLV ユニット, 137
ETS 関連のデータリンクプロパティ, 136
PFC TLV ユニット, 131
tagmode, 78

た

タグモード, 78
単純なブリッジネットワーク, 86
追加
 外部ブリッジへのリンク, 94

定義

- LLDP TLV, 115
- データセンターブリッジング 参照 DCB
- データリンクマルチパスアグリゲーション 参照
- DLMP アグリゲーション
- トポロジの検出
 - LLDP, 104
- トランクアグリゲーション, 19
 - DLMP アグリゲーションへの切り替え, 43
 - 固有の機能, 20
 - 使用する場合, 20
 - スイッチを使用, 21
 - 前提条件, 30
 - トランクアグリゲーションを作成する例, 32
 - トランクアグリゲーションを変更する例, 36
 - バックアップ, 22
 - 負荷分散ポリシー, 23
 - 変更, 35
- トランクリンクアグリゲーション
 - Link Aggregation Control Protocol (LACP), 23

な

- ネットワークスタック
 - ブリッジの実装, 87
- ネットワークデータリンクの管理
 - DCB, 16
 - LLDP, 16
 - VLAN, 15
 - 概要, 13
 - 機能とコンポーネント, 14
 - 新機能, 13
 - ブリッジネットワーク, 15
 - リンクアグリゲーション, 14

は

- 必須の TLV ユニット, 106
- 表示
 - ETS 構成情報, 140
 - LLDP 統計情報, 121
 - LLDP の通知される情報, 119
 - PFC 構成情報, 132
 - PFC 同期ステータス, 133
 - PFC マッピング情報, 133

- probe-ip プロパティ値, 42
- VLAN 情報, 63
- willing プロパティ値, 133, 140
- アグリゲーションの IP アドレスの状態, 42
- 集約されたポートの状態, 42
- データリンクプロパティ, 132, 140
- 同期ステータス, 140
- ブリッジ構成情報, 95
- プローブ関連情報, 41
- 優先順位定義, 134
- 負荷分散
 - トランクアグリゲーション, 23
- プライベート仮想ローカルエリアネットワーク, 75
- プライベート VLAN, 75
 - 構成, 80
 - 削除, 81
 - 変更, 81
- ブリッジ
 - 既存のブリッジへのリンクの追加, 94
 - 削除, 98
 - 作成, 91
 - ブリッジの一部であるリンクへの VLAN の構成, 99
 - ブリッジのネーミング, 92
 - リンクの削除, 94
- ブリッジネットワーク, 15, 85
 - STP デーモン, 90
 - TRILL デーモン, 91
 - VLAN と STP プロトコルおよび TRILL プロトコル, 100
 - 概要, 85
 - 既存のブリッジへのリンクの追加, 94
 - 構成情報の表示, 95
 - システムからブリッジを削除する例, 98
 - 単純なブリッジネットワーク, 86
 - ネットワークスタック, 87
 - ブリッジ上の VLAN の管理, 98
 - ブリッジ情報を表示する例, 97
 - ブリッジネットワークの動作, 88
 - ブリッジネットワークリング, 87
 - ブリッジの削除, 98
 - ブリッジの作成, 91
 - ブリッジのデバッグ, 100
 - ブリッジの保護タイプを変更する例, 93

ブリッジリンクに関する構成情報の表示, 97
 ブリッジを作成する例, 92
 プロトコル, 89
 保護タイプの変更, 93
 リンクの削除, 94
 リンクプロパティの設定, 94
 ブリッジネットワーク上の VLAN の管理, 98
 ブリッジネットワークリング, 87
 ブリッジングプロトコル, 89
 プローブベースの障害検出, 28
 ICMP プローブ, 28
 probe-ip プロパティ値の表示, 42
 アグリゲーションの IP アドレスの状態表示, 42
 構成, 36
 集約されたポートの状態表示, 42
 集約されたポートの情報表示, 41
 推移的プローブ, 29
 プローブ関連情報の表示, 41
 プローブベースの障害検出の構成例, 39
 モニタリング, 40
 プロトコル
 DCBX, 123
 LLDP, 103
 STP, 87, 89
 TRILL, 87, 89
 プロトコルデータユニット (PDU), 105
 プロミスキャストランクポート, 77
 変更
 PVLAN, 81
 VLAN の VLAN ID, 64
 トランクアグリゲーション, 35
 ブリッジの保護タイプ, 93

ま

無効化
 LLDP, 117
 モニタリング
 LLDP エージェント, 118

や

有効化
 DCBX, 127

LLDP、グローバル, 110
 PFC 情報の同期, 131
 特定のポートの LLDP, 111
 優先順位ベースのフロー制御 参照 PFC

ら

リモート MIB, 105
 リンクアグリゲーション, 14
 DLMP アグリゲーション, 24
 DLMP アグリゲーションとトランクアグリゲーションの間の切り替え, 43
 IPMP、比較, 143
 VLAN と組み合わせた使用, 67
 アグリゲーションからリンクを削除する例, 35
 概要, 17
 機能, 18
 削除, 35, 43
 作成, 30
 データリンクの追加, 34
 トランクアグリゲーション, 19
 トランクアグリゲーションと DLMP アグリゲーションの機能比較, 47
 メリット, 18
 要件, 29
 リンクアグリゲーションを削除する例, 43
 リンクをアグリゲーションに追加する例, 34
 リンク状態通知, 30
 リンク層検出プロトコル 参照 LLDP
 リンクプロパティ
 ブリッジの設定, 94
 リンクベースの障害検出, 28
 例
 VLAN 構成の削除, 67, 81
 VLAN の作成, 56
 ゾーンを使用した VLAN の作成, 57, 58
 複数の VLAN の移行, 66
 リンクアグリゲーション上に複数の VLAN を作成, 61
 ローカル MIB, 105

A

auto-enable-agents, 109

B

basic-tlv, 107

C

CoS

優先順位の定義, 124

D

DCB, 16

cos プロパティ, 124

DCBX の有効化, 127

ETS の構成, 135

PFC のカスタマイズ, 130

概要, 123

拡張伝送選択 (ETS), 124

考慮事項, 125

優先順位ベースのフロー制御 (PFC), 124, 125

DCB の操作モード

設定, 128

DCBX プロトコル, 103, 123

dladm コマンド

add-aggr, 34

add-bridge, 94

create-aggr, 30

create-bridge, 91

create-vlan, 56, 80

delete-aggr, 43

delete-bridge, 98

delete-vlan, 66, 81

modify-aggr, 35

modify-bridge, 93

modify-vlan, 63, 81

remove-bridge, 94

show-aggr, 30

show-bridge, 96

show-linkprop, 132

show-vlan, 63

DLMP アグリゲーション, 24

DLMP アグリゲーションの動作, 25

DLMP アグリゲーションを作成する例, 33

障害検出, 27

トポロジ, 25

トランクアグリゲーションへの切り替え, 43

プローブベースの障害検出, 28

プローブベースの障害検出の構成, 37

プローブベースの障害検出のモニタリング, 40

プローブベースの障害検出を構成する例, 39

ポート障害, 27

メリット, 24

リンクベースの障害検出, 28

DLMP アグリゲーションとトランクアグリゲーションの間の切り替え, 43

DLMP アグリゲーションの障害検出, 27

プローブベースの障害検出, 28

リンクベースの障害検出, 28

dot1-tlv, 107

dot3-tlv, 107

E

ETS, 124, 126

ETS TLV ユニット, 137

ETS 関連のデータリンクプロパティの設定, 136

ETS 関連のデータリンクプロパティの表示例, 140

ETS 情報を同期する機能の表示例, 140

構成, 135

帯域幅共有, 136

ピアに対する ETS 構成の推奨, 138

ピアに対する ETS 構成の推奨例, 138

表示情報, 140

プロパティ, 136

ローカルおよびリモートの情報, 136

I

ICMP プローブ, 28

IPMP

リンクアグリゲーション、比較, 143

L

LACP

LACPDU, 23

スイッチの使用, 23

定義, 23

モード, 23

Link Aggregation Control Protocol (LACP) 参照

LACP

LLDP, 16, 103

- auto-enable-agents, 109
- auto-enable-agents SMF プロパティをカスタマイズする例, 112
- lldpd デーモン, 105
- Oracle Solaris のコンポーネント, 104
- SMF プロパティ, 109
- TLV 値の定義, 115
- TLV 値を定義する例, 117
- TLV ユニット, 106, 107
- インストール, 110
- エージェント, 105
- エージェント TLV ユニットの指定, 114
- エージェントごとの TLV ユニット, 104, 107
- エージェントのモニタリング, 118
- エージェントモード, 105
- オプションの TLV ユニット, 107
- オプションの TLV ユニットの追加する例, 114
- 管理情報ベース (MIB), 105
- グローバルな TLV ユニット, 104, 107
- グローバルな有効化, 110
- シャーシ ID とポート ID を表示する例, 106
- 選択された統計情報を表示する例, 122
- 通知される情報の表示, 119
- 通知される情報を表示する例, 119
- 統計情報の表示, 121
- 統計情報を表示する例, 121
- 特定のポートの有効化, 111
- パッケージ, 104
- 複数のデータリンクで LLDP を有効にする例, 113
- 無効化, 117
- 有効化, 109

LLDP SMF サービス, 104

lldpadm コマンド, 104

- reset-agentprop, 117
- set-agentprop, 111, 117
- set-agenttlvprop, 112, 115, 130, 131, 135
- set-tlvprop, 111, 115
- show-agent, 118, 121
- show-agenttlvprop, 115, 132
- show-tlvprop, 115

LLDPDU, 105

P

PAUSE フレーム, 125

PFC, 124, 125

- CoS 優先順位定義の表示例, 134
- CoS 優先順位マッピング, 130
- DCB の PFC のカスタマイズ, 130
- PAUSE フレーム, 125
- pfcmap-rmt, 130
- pfcmap, 125, 130
- PFC 関連のデータリンクプロパティの表示例, 132
- PFC 情報が対称であることの確認例, 134
- PFC 情報を同期する機能の表示例, 133
- VNIC クライアント, 134
- カスタマイズ, 130
- 情報の表示, 132
- データリンクプロパティ、関連, 130
- データリンクプロパティの表示, 132
- 同期された情報, 130
- ホストとピアの間で同期を有効にする例, 131
- ローカルおよびリモートの情報, 130

PFC TLV ユニット, 131

PFC マッピング, 125

PVLAN, 75

- 概要, 75
- 隔離 VLAN, 75
- 構成, 80
- 構成の要件, 78
- コミュニティ VLAN, 75
- 削除, 81
- セカンダリトランクポート, 77
- ゾーンへの割り当て, 82
- ゾーンを持つ, 79
- プロミスキャストランクポート, 77
- 変更, 81
- ポート, 77
- 利点, 77
- PVLAN セカンダリトランクポート, 77
- PVLAN のゾーンへの割り当て, 82
- PVLAN ポート, 77

S

setting

- PFC 関連データリンクプロパティ, 130

STP

- ブリッジの保護タイプとして設定, 93
- STP デーモン, 90
- STP プロトコル, 89
- TRILL との対比, 90

T

- TLV (Type-Length-Value) ユニット, 106
- TLV プロパティ
- willing, 131
- TRILL, 89
- ブリッジの保護タイプとして設定, 93
- TRILL デーモン, 91
- TRILL プロトコル
- STP との対比, 90

V

- virt-tlv, 107
- VLAN, 15, 49
- MAC アドレス, 65
 - STP および TRILL プロトコル, 100
 - VLAN ID の変更, 63
 - VLAN 構成の計画, 55
 - VLAN 構成を削除する例, 67, 81
 - VLAN 名, 50
 - VLAN を作成する例, 56
 - 移行, 63
 - 概要, 49
 - 構成, 56
 - 削除, 66
 - 情報の表示, 63
 - ゾーンでの使用, 54
 - ゾーンを使用して VLAN を作成する例, 57, 58
 - どのような場合に VLAN を使用するか, 50
 - トポロジ, 51
 - リンクアグリゲーション上に作成, 60
 - リンクアグリゲーション上に複数の VLAN を作成する例, 61
 - リンクアグリゲーションと組み合わせた使用, 67
 - レガシーデバイス, 上, 62
 - ワークグループ, 50