在 Oracle® Solaris 11.2 中管理网络数据链路

文件号码 E53798-02
2014 年 9 月

ORACLE

目录

使用本文档

- **概述** – 概述了用来管理网络数据链路以提高网络性能的高级功能。说明了如何通过使用中继或 DLMP 聚合将链路组合为聚合，使用虚拟局域网将网络划分为子网，使用网桥连接独立的网段，使用链路层发现协议交换网络连接信息，以及使用数据中心桥接管理聚合网络。
- **目标读者** – 系统管理员。
- **必备知识** – 基本的和一些高级的网络管理技能。

产品文档库

有关本产品的最新信息和已知问题均包含在文档库中，网址为：http://www.oracle.com/pls/topic/lookup?ctx=E36784。

获得 Oracle 支持

Oracle 客户可通过 My Oracle Support 获得电子支持。有关信息，请访问 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info；如果您听力受损，请访问 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs。

反馈

可以在 http://www.oracle.com/goto/docfeedback 上提供有关本文档的反馈。

♦ ♦ ♦ 第 1 章

1 管理网络数据链路介绍

本章介绍了用来管理网络数据链路的高级功能，本书的其余章节中对网络数据链路进行了介绍。本书中介绍的不同技术的使用取决于具体情况。此外，某些硬件配置可能会要求您使用特定类型的功能。因此，您不需要完成本书中介绍的所有配置过程。而应该选择和部署能满足您的网络需求的技术。

在您执行本书中介绍的任何配置之前，必须首先完成基本网络配置，并了解基本的数据链路配置。有关基本网络配置的信息，请参见《在 Oracle Solaris 11.2 中配置和管理网络组件 》。有关用来管理数据链路配置的元素的信息，请参见《在 Oracle Solaris 11.2 中配置和管理网络组件 》中的第 2 章 “在 Oracle Solaris 中管理数据链路配置”。

有关 Oracle Solaris 中的网络配置功能的摘要，请参见《Oracle Solaris 11.2 中的网络管理策略 》中的第 1 章 “Oracle Solaris 网络管理摘要”。

本章包含以下主题：

Oracle Solaris 11.2 中用来管理网络数据链路的新增功能

本节针对现有的客户重点介绍了此发行版中的以下重要变更：

- **数据链路多路径 (datalink multipathing, DLMP) 聚合中基于探测器的故障检测** – 检测 DLMP 聚合链路与所配置的目标之间的连接是否丢失。该故障检测类型主要用于解决基于链路的故障检测机制的不足，后者只能检测出数据链路和第一中继站交换机间的直接连接丢失所引发的故障。有关更多信息，请参见“DLMP 聚合故障检测” [21]。
- **与对等方之间的增强传输选择 (enhanced transmission selection, ETS) 建议值传输** – 数据中心桥接 (data center bridging, DCB) 中的 ETS 功能已增强，从而使 Oracle Solaris 主机能够向 DCB 网络中的下一中继站交换机提供带宽份额建议。有关更多信息，请参见“向对等方建议 ETS 配置” [104]、“设置与 ETS 相关的数据链路属性” [103]和例 6-8 “显示本地主机同步 ETS 信息的功能”。

- **显示数据链路属性的生效值** – `dladm show-linkprop` 子命令已增强，可以显示某些数据链路属性的 `EFFECTIVE` 字段。`EFFECTIVE` 字段的值是系统根据资源的可用性、底层设备的能力或与对等方的协商确定的。生效值不需要与所配置的值相同。即使没有为数据链路属性配置值，此属性也可以具有一个生效值。有关如何显示 `EFFECTIVE` 字段的示例，请参见“显示数据链路属性” [99]和例 6-7 “显示与 ETS 相关的数据链路属性”。
- **显示网桥统计信息** – `dlstat show-bridge` 子命令显示网桥及与每个网桥相连的链路的统计信息，还会显示网桥统计信息的嵌套视图。有关更多信息，请参见“显示配置的网桥的信息” [69]。

用来管理网络数据链路的功能和组件

管理网络数据链路是指使用相关功能和技术对系统处理网络通信的方式进行微调。配置了这些技术的系统可以更好地管理网络通信，这有助于提高网络的总体性能。虽然这些功能涉及网络运行的不同方面，但都可在网络连接、网络管理和效率等方面带来好处。

链路聚合

通过链路聚合，可以将要管理的多个数据链路资源合并为一个单元。您可以通过将多个物理 NIC 组合在一起来提升带宽并提供应用程序高可用性。网络数据链路的链路聚合可保证系统能够持续访问网络。中继聚合和 DLMP 聚合是链路聚合的两种类型。

中继聚合可为基于聚合配置的客户机提供底层数据链路的整合带宽。DLMP 聚合可以跨越多个交换机为基于聚合配置的客户机提供高可用性。DLMP 聚合还支持基于链路的故障检测和基于探测器的故障检测，以确保网络可持续用来发送和接收通信。有关不同类型的链路聚合以及配置和管理链路聚合的过程的更多信息，请参见第 2 章 使用链路聚合配置高可用性。

虚拟局域网

使用虚拟局域网 (virtual local area network, VLAN)，您可以在不向物理网络环境添加资源的情况下，将网络划分为多个子网。因此，子网是虚拟的，您仍使用相同的物理网络资源。VLAN 为应用程序提供隔离的子网，以便只有同一 VLAN 中的应用程序才能互相通信。您可以将 VLAN 和 Oracle Solaris 区域结合使用，在一个网络单元（例如一个交换机）内配置多个虚拟网络。有关 VLAN 及 VLAN 配置和管理过程的更多信息，请参见第 3 章 使用虚拟局域网配置虚拟网络。

桥接网络

网桥连接不同的网段，而网段是两个节点之间的路径。不同网段通过网桥连接后，连接的网段进行通信时，就如同是一个网段一样。网桥使用某种包转发机制将子网连接在一起，使系统可以通过最短的连接路由将包传送到其目的地。有关桥接网络及管理网桥的过程的更多信息，请参见第 4 章 管理桥接功能。

链路层发现协议

通过链路层发现协议 (Link Layer Discovery Protocol, LLDP)，可以在网络上的系统间交换连接和管理信息，从而执行拓扑发现。该信息可以包括系统能力、管理地址以及与网络操作相关的其他信息。网络诊断服务使用 LLDP 检测可能会导致网络连接受限或降级的问题。有关 LLDP 和 LLDP 的配置过程的更多信息，请参见第 5 章 使用链路层发现协议交换网络连接信息。

数据中心桥接

当共享同一网络链路（例如，在联网协议和存储协议之间共享一条数据链路）时，数据中心桥接 (data center bridging, DCB) 用来管理多种通信类型的带宽、相对优先级和流控制。通过 DCB，可以使用 LLDP 与对等方交换有关支持聚合网络的功能的信息。该信息与影响网络包完整性的配置相关，尤其是在通信量很大的环境中（例如数据中心）。DCB 将存储区域网络 (storage area network, SAN) 和局域网 (local area network, LAN) 整合在一起，因而降低了数据中心的运营和管理成本，从而实现了高效的网络基础结构。

您可以根据服务类 (class of service, CoS) 优先级配置 DCB 功能，例如基于优先级的流控制 (priority-based flow control, PFC)（用于防止包丢失）和增强传输选择 (enhanced transmission selection, ETS)（用于在包间实现带宽共享）。有关更多信息，请参见第 6 章 使用数据中心桥接管理聚合网络。

♦ ♦ ♦ 第 2 章

使用链路聚合配置高可用性

本章概述了链路聚合并介绍了配置和管理链路聚合的过程。

本章包含以下主题：

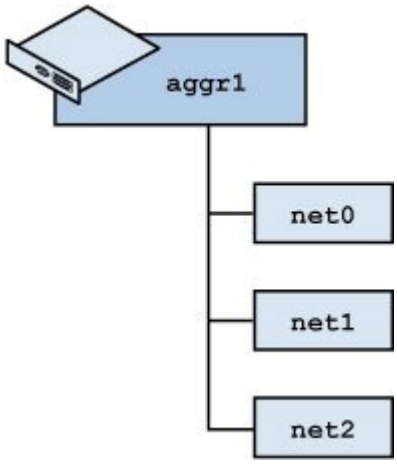
链路聚合概述

链路聚合是一种将系统上的多个物理数据链路结合起来的方法，这种方法可增加应用程序的带宽。这些物理数据链路一起配置为一个逻辑单元，以提高网络通信吞吐量并实现数据链路层的高可用性。

通过使用 Oracle Enterprise Manager Ops Center，您可以包括可用 NIC 中作为单个接口列出的任何链路聚合。使用 Oracle Enterprise Manager Ops Center，还可以显示中继聚合和 DLMP 聚合的详细信息。有关 Oracle Enterprise Manager Ops Center 的更多信息，请参见 http://www.oracle.com/pls/topic/lookup?ctx=oc122&id=OPCCM。

下图显示了在系统上配置的简单链路聚合的示例。

图 2-1 链路聚合配置



图中显示了由三个底层数据链路（net0、net1 和 net2）组成的聚合 aggr1。这些数据链路专用于为通过该聚合穿越系统的通信提供服务。底层链路对外部应用程序是隐藏的。相反，可以访问逻辑数据链路 aggr1。

链路聚合的优势

链路聚合具备以下优势：

- **增加了带宽** – 将多个链路的容量组合到一个逻辑链路中。
- **自动故障转移和故障恢复** – 将来自故障链路的通信自动切换到聚合中的其他工作链路，从而实现高可用性。
- **改进了管理** – 所有底层链路作为一个单元进行管理。
- **减少了网络地址池消耗** – 可以将一个 IP 地址指定给整个聚合。

由于链路聚合可将多个链路组合为单个逻辑数据链路，因此数据链路的链路保护和资源管理等功能可通过链路聚合正常发挥作用。有关链路保护的信息，请参见《在 Oracle Solaris 11.2 中确保网络安全 》中的第 1 章 “在虚拟化环境中使用链路保护”。有关资源管理的信息，请参见《在 Oracle Solaris 11.2 中管理网络虚拟化和网络资源 》中的第 7 章 “管理网络资源”。

链路聚合克服了结合使用高可用性功能与网络虚拟化（例如 IPMP）时遇到的一些问题。使用链路聚合，您可以首先在全局区域中创建聚合，然后在配置非全局区域时将其指定为底层链路。在区域引导时，将在链路聚合上为该区域指定一个虚拟网络接口卡 (virtual network interface card, VNIC)。链路聚合与 IPMP 不同，后者需要在每个非全局

区域进行配置，前者仅在全局区域进行配置，并为非全局区域提供高可用性 VNIC。全局区域还可以配置其他属性，诸如指定给区域的 VNIC 上的带宽。

注 - 链路聚合执行与 IP 多路径 (IP multipathing, IPMP) 类似的功能以提高数据链路层的网络性能和可用性。有关这两种技术的比较，请参见附录 A, 链路聚合和 IPMP：功能比较。

支持以下类型的链路聚合：

- 中继聚合
- 数据链路多路径 (datalink multipathing, DLMP) 聚合

有关这两种链路聚合类型之间的区别，请参见“中继聚合与 DLMP 聚合的比较” [37]。

中继聚合

中继聚合基于 IEEE 802.3ad 标准，并通过使多个通信流分布在一组聚合端口中而发挥作用。IEEE 802.3ad 需要交换机配置以及交换机供应商专有扩展，以便跨多个交换机运行。在中继聚合中，在聚合上配置的客户机可获得底层链路的整合带宽，因为每个网络端口与在聚合上配置的每个数据链路相关联。创建链路聚合时，缺省情况下将在中继模式下创建聚合。在以下情况中您可能会使用中继聚合：

- 对于网络中运行具有分布式大通信流量的应用程序的系统，可以将中继聚合专用于该应用程序的通信以利用增加的带宽。
- 对于具有有限的 IP 地址空间但却需要很大带宽的站点，数据链路的中继聚合仅需要一个 IP 地址。
- 对于需要隐藏任何内部数据链路的站点，中继聚合的 IP 地址对外部应用程序隐藏这些数据链路。
- 对于需要可靠网络连接的应用程序，中继聚合对网络连接进行保护，防止链路故障。

中继聚合支持以下功能：

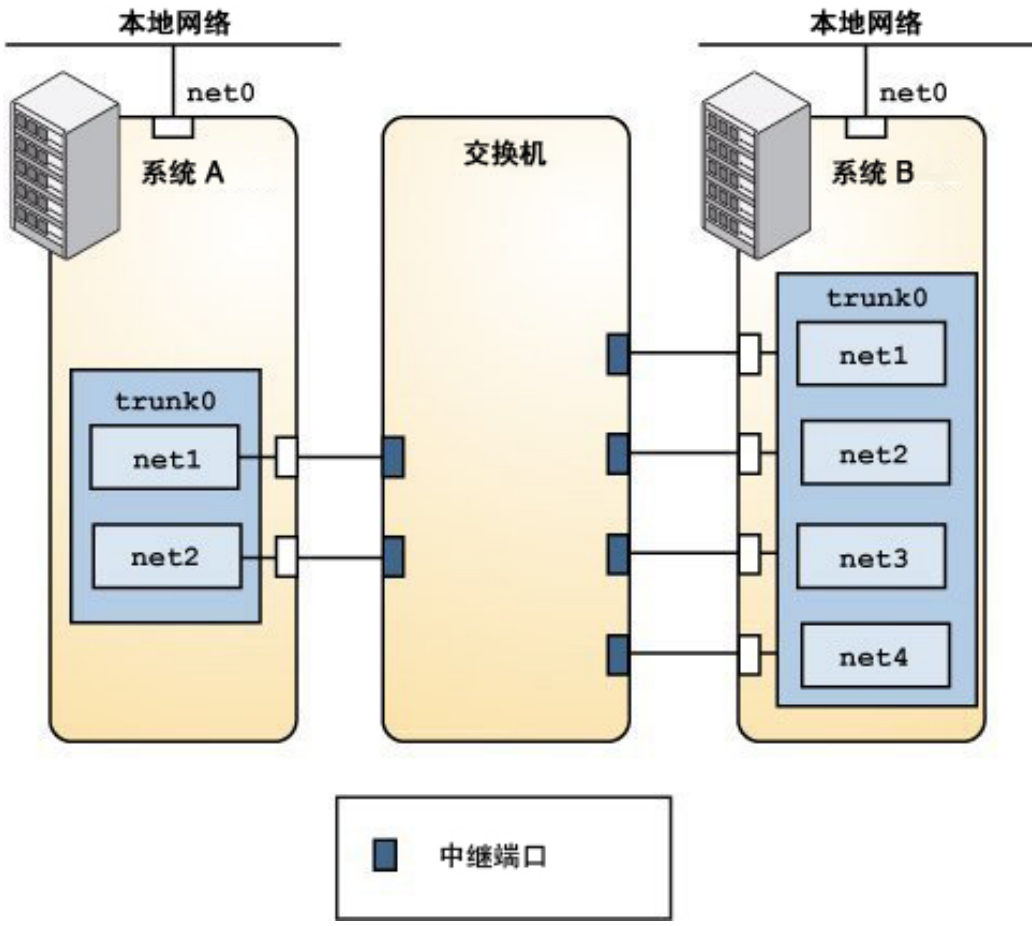
- 使用交换机
- 使用具有链路聚合控制协议 (Link Aggregation Control Protocol, LACP) 的交换机
- 背对背中继聚合配置
- 聚合策略和负载平衡

以下各节介绍了中继聚合的功能。

使用交换机

配置了中继聚合的系统可能会使用外部交换机连接到其他系统。下图描述了具有两个系统的本地网络，其中各个系统均配置了中继聚合。

图 2-2 使用交换机的中继聚合

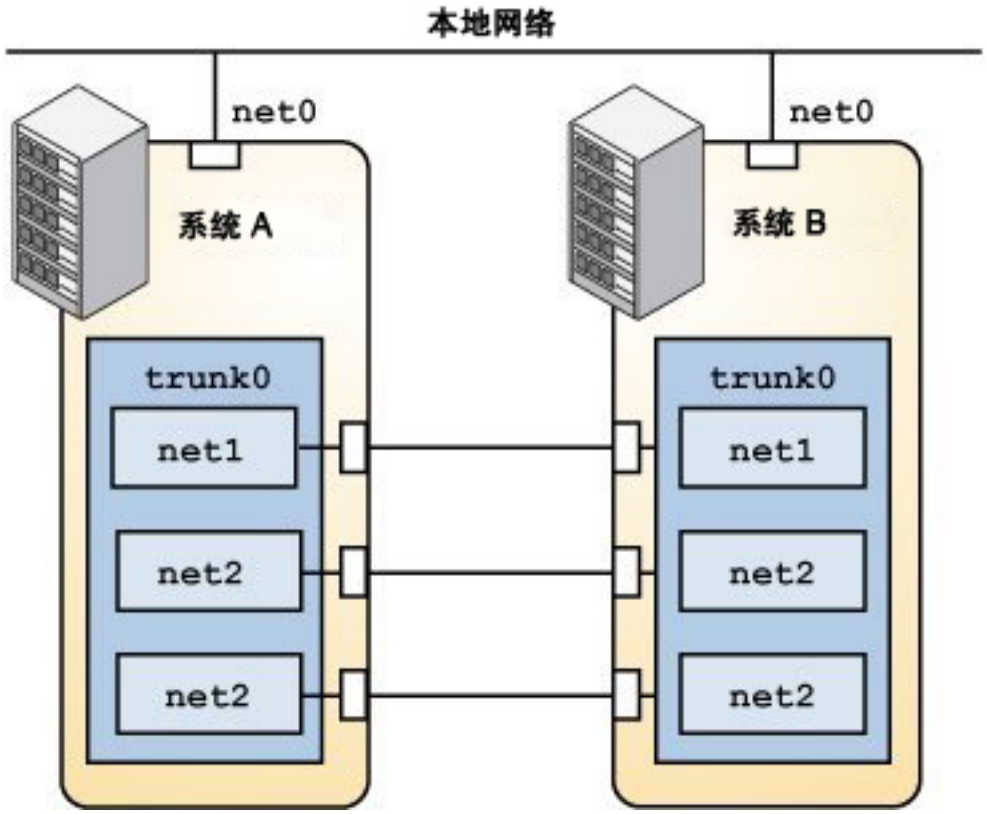


这两个系统由交换机连接在一起。系统 A 的中继聚合由两个数据链路（net1 和 net2）组成。这些数据链路通过聚合端口连接到交换机。系统 B 的中继聚合由四个数据链路（net1 到 net4）组成。这些数据链路也连接到交换机上的聚合端口。在该中继聚合拓扑中，交换机必须支持 IEEE 802.3ad 标准，并且必须为聚合配置交换机端口。请参见交换机制造商文档来配置交换机。

背对背中继聚合配置

中继聚合支持背对背配置。不使用交换机，将两个系统直接连接到一起以运行并行聚合，如下图所示。

图 2-3 背对背中继聚合配置



图中显示，系统 A 上的中继聚合 `trunk0` 直接与系统 B 上的中继聚合 `trunk0` 连接（通过各自底层数据链路之间的相应链路）。通过该设置，系统 A 和 B 可以提供冗余、高可用性以及这两个系统之间的高速通信。每个系统还将 `net0` 配置为用于本地网络内的通信流。

背对背中继聚合最常见的应用是数据中心中镜像数据库服务器的配置。这两个服务器必须一起更新，因此对带宽、高速通信流和可靠性要求很高。

使用具有链路聚合控制协议的交换机

如果中继聚合的设置中包含交换机，且该交换机支持 LACP，则可以为交换机和系统启用 LACP。请参见交换机制造商文档来配置交换机。

LACP 支持更加可靠的数据链路故障检测方法。如果不使用 LACP，链路聚合将仅依赖于设备驱动程序报告的链路状态检测聚合数据链路的故障。如果使用 LACP，将定期交换 LACPDU 以确保聚合数据链路能够发送和接收通信流量。LACP 还可以检测到一些错误配置情况，例如，数据链路的分组在两个对等方之间不匹配。

如果系统上启用了 LACP，LACP 将在聚合和交换机之间交换名为链路聚合控制协议数据单元 (Link Aggregation Control Protocol Data Unit, LACPDU) 的特殊帧。LACP 使用这些 LACPDU 维护聚合数据链路的状态。

使用 `dladm create-aggr` 命令将聚合的 LACP 配置为以下三种模式之一：

- `off` – 聚合的缺省模式。系统不生成 LACPDU。
- `active` – 系统以指定的时间间隔生成 LACPDU。
- `passive` – 系统仅在收到来自交换机的 LACPDU 时才生成 LACPDU。如果聚合和交换机均在 `passive` 模式下进行配置，则它们不会交换 LACPDU。

  有关如何配置 LACP 的信息，请参见如何创建链路聚合 [24]。

为负载平衡定义聚合策略

可以为传出通信定义一个策略，此策略指定如何在聚合的可用链路之间分配负载，从而建立负载平衡。可以使用以下负载说明符强制实施各种负载平衡策略：

- L2 – 通过使用每个包的 MAC (L2) 头来确定传出链路
- L3 – 通过使用每个包的 IP (L3) 头来确定传出链路
- L4 – 通过使用每个包的 TCP、UDP 或其他 ULP (L4) 头来确定传出链路

这些策略的任意组合也是有效的。缺省策略是 L4。

数据链路多路径聚合

数据链路多路径 (datalink multipathing, DLMP) 聚合是一种无需交换机配置即可跨多个交换机提供高可用性的链路聚合类型。DLMP 聚合支持基于链路的故障检测和基于探测器的故障检测，以确保网络可持续用于发送和接收通信。

DLMP 聚合的优势

DLMP 聚合具有以下优势：

- 由中继聚合实现的 IEEE 802.3ad 标准不具备跨越多个交换机的条件。要允许在中继模式下在多个交换机之间进行故障转移，需要在供应商之间不兼容的交换机上提供供应商专有扩展。DLMP 聚合允许在多个交换机之间进行故障转移，而无需任何供应商专有扩展。
- 在网络虚拟化环境中使用 IPMP 实现高可用性十分复杂。IPMP 组无法直接指定给区域。如果必须要在多个区域之间共享网络接口卡 (network interface card, NIC)，则必须配置 VNIC 以便每个区域都从各物理 NIC 获得一个 VNIC。各个区域必须将其 VNIC 分组到 IPMP 组，以实现高可用性。扩大配置范围后复杂程度会增加，例如，在包含大量系统、区域、NIC、虚拟 NIC (virtual NIC, VNIC) 和 IPMP 组的方案中。通过 DLMP 聚合，可以创建 VNIC 或在聚合顶部配置区域的 `anet` 资源，该区域将出现高可用性 VNIC。
- DLMP 聚合使您能够使用链路层的功能，例如链路保护、用户定义的流以及定制链路属性（如带宽）的能力。

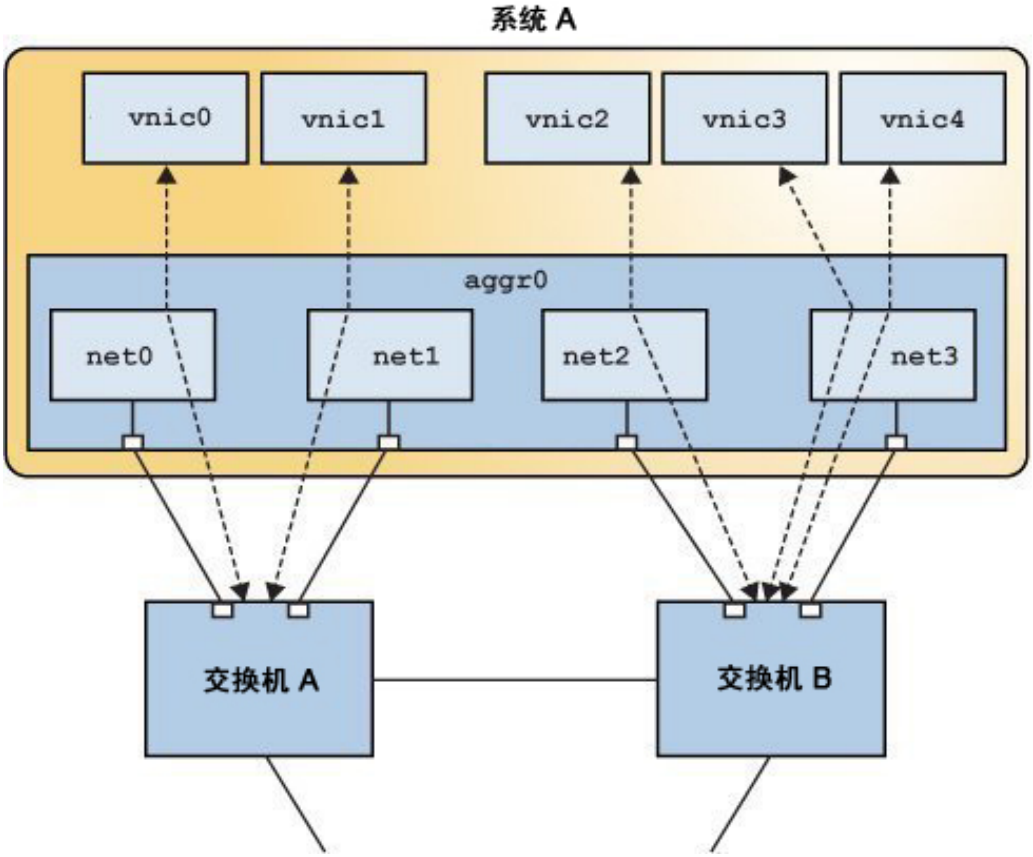
注 - 不能配置基于 DLMP 聚合的 IPMP 组。但是，可以配置基于中继聚合的 IPMP 组。

DLMP 聚合的工作原理

在中继聚合中，每个端口都与聚合上配置的各个数据链路相关联。在 DLMP 聚合中，端口与聚合配置的任何数据链路相关联。

下图显示了 DLMP 聚合的工作原理。

图 2-4 DLMP 聚合



该图显示了具有链路聚合 aggr0 的系统 A。该聚合由 net0 到 net3 四个底层链路组成。还在该聚合上配置了 VNIC vnic0 到 vnic4。该聚合连接到交换机 A 和交换机 B，然后，这两个交换机连接到范围更广的网络中的其他目标系统。

VNIC 通过底层链路与聚合端口相关联。例如，在该表中，vnic0 到 vnic3 通过底层链路 net0 到 net3 与聚合端口相关联。也就是说，如果 VNIC 数与底层链路数相同，则每个端口都与一个底层链路相关联。

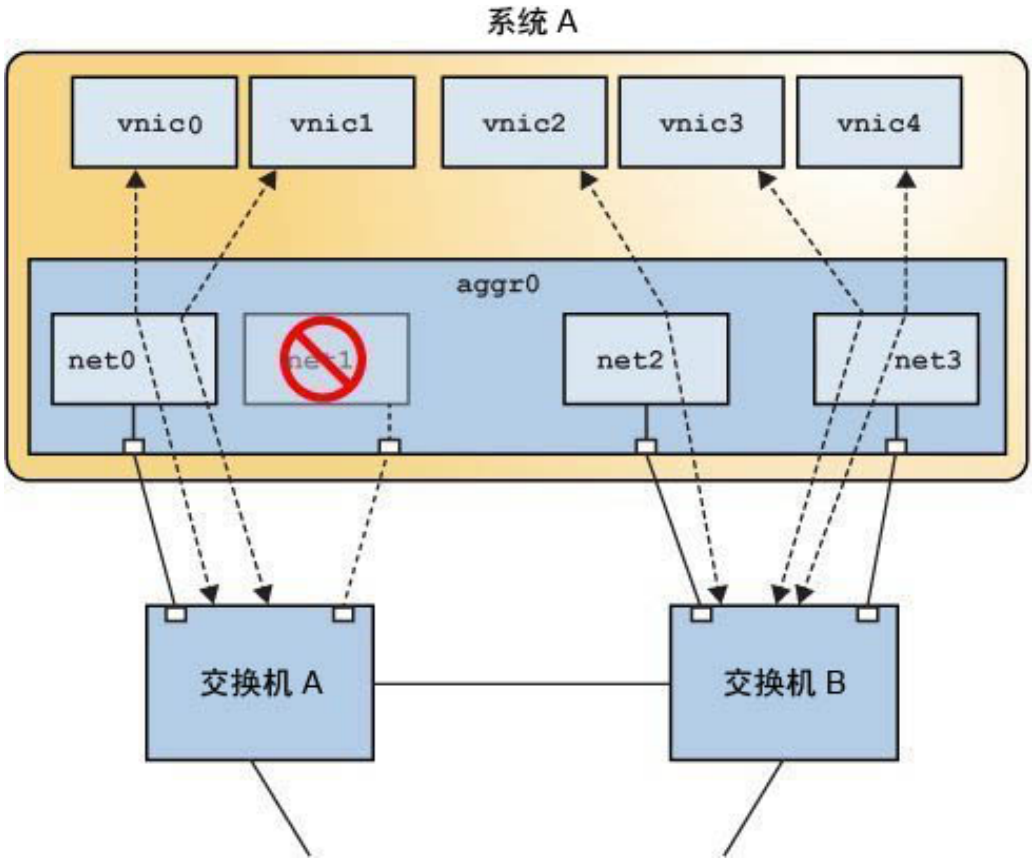
如果 VNIC 数超过底层链路数，则一个端口与多个数据链路相关联。例如，在该图中，VNIC 总数超过了底层链路数。因此，vnic4 与 vnic3 共享一个端口。

如果一个聚合端口出现故障，则使用该端口的所有数据链路将被分布到其他端口，从而在故障转移期间提供网络高可用性。例如，如果 net0 出现故障，则 DLMP 聚合将在

VNIC 之间共享剩余端口 net1。在聚合端口间的分布对用户是透明的，并独立于连接到聚合的外部交换机。

下图显示了一个端口出现故障时 DLMP 聚合的工作原理。在图中，net1 出现故障，交换机和 net1 之间的链路关闭。vnic1 通过 net0 与 vnic0 共享端口。

图 2-5 端口出现故障时的 DLMP 聚合



DLMP 聚合故障检测

DLMP 聚合故障检测是一种检测聚合端口故障的方法。如果端口无法发送或接收通信，则认为该端口出现故障。端口可能会由于以下原因而出现故障：

- 电缆损坏或切断
- 交换机端口关闭
- 上游网络路径故障

DLMP 聚合对聚合端口执行故障检测，以确保网络可持续用于发送或接收通信流量。如果端口出现故障，与该端口关联的客户机将会故障转移到活动端口。出现故障的聚合端口在修复之前不可用。在根据需要部署任何现有端口时，其余活动端口继续工作。出现故障的端口从故障中恢复之后，来自其他活动端口的客户机可以与其相关联。

DLMP 聚合支持基于链路的故障检测和基于探测器的故障检测。

基于链路的故障检测

基于链路的故障检测在电缆切断或交换机端口关闭时检测故障。因此，它只能检测由数据链路和第一中继站交换机之间的直接连接丢失导致的故障。创建 DLMP 聚合时，缺省情况下将启用基于链路的故障检测。

基于探测器的故障检测

基于探测器的故障检测可检测终端主机与所配置的目标之间的故障。此功能克服了基于链路的故障检测的已知限制。当缺省路由器关闭或网络无法访问时，基于探测器的故障检测非常有用。DLMP 聚合通过发送和接收探测包检测故障。

要在 DLMP 聚合中启用基于探测器的故障检测，必须配置 `probe-ip` 属性。

注 - 在 DLMP 聚合中，如果未配置 `probe-ip`，则会禁用基于探测器的故障检测并仅使用基于链路的故障检测。

创建第一个 DLMP 聚合时，服务 `svc:/network/dlmp:default` 将自动启用。此服务可启动 `in.dlmpd` 守护进程，该守护进程在 DLMP 聚合中执行基于探测器的故障检测。如果系统中没有 DLMP 聚合，此服务将处于禁用状态。有关信息，请参见“为 DLMP 聚合配置基于探测器的故障检测” [29]。

基于探测器的故障检测通过结合使用两种类型的探测器来执行：Internet 控制消息协议 (Internet Control Message Protocol, ICMP (L3)) 探测器和传递式 (L2) 探测器，二者配合使用以确定聚合物理数据链路的运行状况。

- ICMP 探测

  可以配置源 IP 地址的逗号分隔列表以及可选目标 IP 地址或主机名。目标 IP 地址必须与指定的源 IP 地址位于相同的子网上。可以通过四种不同的形式指定源 IP 地址。有关更多信息，请参见如何为 DLMP 配置基于探测器的故障检测 [30]。

  仅当 IP 地址与 VNIC 等客户机相关联时，ICMP 探测才会使用为 `probe-ip` 属性配置的源 IP 地址。端口仅在以下情况下与 VNIC 等客户机相关联：该端口接收客户机的

传入通信并传送其传出通信时。在任何特定时间，客户机的传入或传出通信始终会仅通过 DLMP 聚合的一个底层端口。仅当为 `probe-ip` 属性配置的 IP 地址与该端口相关联时，才会使用该 IP 地址来监视端口的运行状况。

对于每个配置的源 IP 地址，`in.dlmpd` 守护程序定期发送指向配置目标的单播 ICMP 包。如果未配置目标 IP 地址，`in.dlmpd` 将使用路由表来查找与指定源 IP 地址相同的子网上的路由，并使用指定的下一中继站作为目标 IP 地址。

仅通过与该 IP 客户机关联的端口发出 ICMP 探测器通信。如果特定端口的所有目标都无法访问，则会将该端口标记为 *ICMP* 出现故障。如果从该端口至少有一个目标可通过 ICMP 探测器访问，则会将该端口标记为 *ICMP* 处于活动状态。

- **传递式探测**

  如果无法通过 ICMP 探测确定所有网络端口的运行状况，则会执行传递式探测。因此，如果所有的端口都与针对 `probe-ip` 属性配置的源 IP 地址不关联，则会执行传递式探测。例如，如果任何端口未与 IP 客户机相关联，或者为 `probe-ip` 属性配置的 IP 地址数少于聚合端口总数，则会执行传递式探测。探测包从未与任何 IP 客户机关联的端口定期发送到对等端口。如果一个端口可以访问 ICMP 活动端口，则该端口将被视为处于 *L2* 活动状态。

  Oracle Solaris 包括通过网络传输的传递式探测器的专有协议包。有关更多信息，请参见附录 B, 传递式探测器的包格式。

在全局区域创建了聚合上的 VNIC 并将其指定给非全局区域时，将在该全局区域执行基于探测器的故障检测。但是，可以在 VLAN 的帮助下将探测器通信从非全局区域中分离。例如，当探测器通信在全局区域的一个 VLAN 上运行时，非全局区域通信可以在其他 VLAN 上运行。

链路聚合的要求

链路聚合配置具有以下要求：

- 要配置到聚合中的数据链路不应具有在这些数据链路上配置的任何 IP 接口。
- 只能从全局区域创建链路聚合。无法使用数据链路从非全局区域创建链路聚合，即使没有在数据链路上配置任何 IP 接口也是如此。链路聚合只是将多个物理 NIC 组合在一起，但在非全局区域中所有接口都是虚拟接口。因此，无法从非全局区域创建链路聚合。
- 聚合中的所有数据链路必须以相同的速度和全双工模式运行。
- 对于 DLMP 聚合，必须至少有一个交换机，以将聚合连接到其他系统的端口。配置 DLMP 聚合时不能使用背对背设置。
- 在基于 SPARC 的系统上，各个数据链路都必须有各自唯一的 MAC 地址。有关信息，请参见《在 Oracle Solaris 11.2 中配置和管理网络组件 》中的“如何确保每个接口的 MAC 地址是唯一的”。
- （仅限中继聚合）如 IEEE 802.3ad 链路聚合标准中所定义，设备必须支持链路状态通知，端口才能连接到中继聚合或者从中继聚合分离。不支持链路状态通知的设

备只能通过使用 dladm create-aggr 命令的 -f 选项进行聚合。对于此类设备，始终将链路状态报告为 UP（活动）。有关使用 -f 选项的信息，请参见如何创建链路聚合 [24]。

创建链路聚合

链路聚合将底层端口组合到单个逻辑组中。聚合以独占方式使用这些底层端口，您无法执行任何其他操作，如在这些端口上配置 VNIC 或指定 IP 地址。但是，可以在聚合之上而非各个端口上配置 VNIC。

在创建链路聚合之前必须删除这些端口上的任何现有 IP 接口。

还可以在您创建的链路聚合上配置 VLAN 并创建 VNIC。有关如何在链路聚合上创建 VLAN 的信息，请参见如何在链路聚合上配置 VLAN [50]。

注 - 链路聚合仅对以相同速度运行的全双工点对点链路起作用。请确保您的聚合中的数据链路符合此要求。

▼ 如何创建链路聚合

开始之前 如果要创建中继聚合并在聚合中使用交换机，请在交换机上配置要用作聚合的端口。如果交换机支持 LACP，请以 active 或 passive 模式配置 LACP。

请参见交换机制造商文档来配置交换机。

1. **成为管理员。**

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. **显示数据链路信息以识别聚合的物理数据链路。**

```
# dladm show-phys
```

3. **确保任何应用程序均未在使用要聚合的数据链路。**

   例如，如果在该数据链路上创建了一个 IP 接口，请先删除该 IP 接口。

   a. **确定链路状态。**

```
# ipadm show-if
IFNAME       CLASS        STATE     ACTIVE     OVER
lo0          loopback     ok        yes        --
```

```
net0        ip          ok        no        --
```

该输出指示数据链路 `net0` 上存在一个 IP 接口。

b. **删除该 IP 接口。**

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

其中，*interface* 指定链路上的 IP 接口。有关更多信息，请参见 `ipadm(1M)` 手册页。

4. **创建链路聚合。**

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

`-f`	强制创建聚合。当试图聚合不支持链路状态通知的设备时，使用此选项。
`-m` *mode*	必须将模式设置为以下值之一：缺省模式为 `trunk`。 ■ `trunk` – 符合 IEEE 802.3ad 标准的链路聚合模式 ■ `dlmp` – 数据链路多路径模式
`-P` *policy*	（仅限中继聚合）指定聚合的负载平衡策略。支持的值包括 `L2`、`L3` 和 `L4`。有关更多信息，请参见“为负载平衡定义聚合策略” [18]。
`-L` *LACP-mode*	（仅限中继聚合）指定 LACP（如果在使用）的模式。支持的值包括 `off`、`active` 或 `passive`。有关模式的信息，请参见“使用交换机” [15]。
`-T` *time*	（仅限中继聚合）指定 LACP 计时器值。支持的值包括 `short` 或 `long`。
`-u` *address*	指定聚合的固定单播地址。
`-l` *linkn*	指定要聚合的数据链路。
aggr	指定聚合的名称，可以是任意定制名称。有关指定名称的规则的信息，请参见《在 Oracle Solaris 11.2 中配置和管理网络组件 》中的“有效链路名称的规则”。

5. **（可选） 检查您创建的聚合的状态。**

- 显示聚合和链路，其中包含状态信息。

```
# dladm show-link
```

- 显示聚合，其中包含状态信息和每个端口的信息。

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

聚合的状态应为 up。

例 2-1 创建中继聚合

此示例显示了用于创建具有两个底层数据链路（net0 和 net1）的链路聚合的命令。该聚合还配置为传送 LACP 包。该示例以删除底层数据链路上的现有 IP 接口开始。

```
# 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 2-2 创建 DLMP 聚合并在聚合之上配置 IP 接口

此示例说明了如何创建 DLMP 聚合。该聚合包含三个底层链路：net0、net1 和 net2。在聚合 aggr0 之上创建了一个 IP 接口并在聚合之上创建了 VNIC vnic1。

```
# 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 -T static -a local=10.10.10.1 aggr0/v4
# dladm create-vnic -l aggr0 vnic1
```

接下来的步骤 可以执行聚合的进一步配置，例如创建 IP 接口和 VNIC。可以使用创建的聚合配置非全局区域和内核区域。

- 有关创建 IP 接口的信息，请参见《在 Oracle Solaris 11.2 中配置和管理网络组件》中的第 3 章 “在 Oracle Solaris 中配置和管理 IP 接口和地址”。
- 有关在链路聚合上配置 VLAN 的信息，请参见“在链路聚合上配置 VLAN” [50]。

- 有关配置 VNIC 的信息，请参见《在 Oracle Solaris 11.2 中管理网络虚拟化和网络资源 》中的“如何配置 VNIC 和 Etherstub”。
- 有关配置区域的信息，请参见《创建和使用 Oracle Solaris 区域 》。

将链路添加到聚合

可以在现有聚合中包括其他数据链路。如果要向中继聚合添加其他数据链路，可能需要重新配置交换机以容纳其他数据链路，即使交换机上配置了 LACP 也是如此。

▼ 如何将链路添加到聚合

1. **成为管理员。**
   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. **确保未在链路上配置任何 IP 接口。如果配置了任何 IP 接口，请将该 IP 接口删除。**

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

   其中，*interface* 是数据链路上配置的 IP 接口。

3. **将链路添加到聚合。**

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

   其中，*link* 代表要添加到聚合中的数据链路，而 *aggr* 是聚合的名称。

4. **（仅限中继聚合）如有必要，重新配置交换机。**
   可能需要重新配置交换机以根据交换机的配置方式容纳其他数据链路，即使交换机上启用了 LACP 也是如此。
   要在交换机上执行任何重新配置任务，请参见交换机制造商文档。

例 2-3 将链路添加到聚合

本示例说明如何将链路添加到聚合 `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    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        --
```

从聚合中删除链路

可以使用 `dladm remove-aggr` 命令删除与聚合关联的各个数据链路。从聚合中删除链路后需要重新配置交换机。

成为管理员并使用以下命令：

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

例 2-4 从聚合中删除链路

此示例说明了如何从聚合 `aggr0` 中删除链路。

```
# 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   --
```

修改中继聚合

可以修改中继聚合的选定属性，例如 `policy`、`lacpmode` 和 `time`。DLMP 聚合不支持这些属性。

- 要修改聚合的负载平衡策略，请成为管理员并发出以下命令：

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

policy 表示一个或多个负载平衡策略（L2、L3 和 L4），如“为负载平衡定义聚合策略” [18]中所述。

aggr 指定要修改其策略的聚合。

- 要修改聚合的 LACP 模式，请成为管理员并使用以下命令：

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

  -L *LACP-mode* 指示聚合运行时必须采用的 LACP 模式。可能的值包括 `active`、`passive` 和 `off`。

  -T *time* 指示 LACP 计时器值（`short` 或 `long`）。

  aggr 指定要修改其 LACP 模式的聚合。

例 2-5 修改中继聚合

此示例说明了如何将链路聚合 `aggr0` 的负载平衡策略修改为 L2 以及如何将 LACP 模式更改为 `active`。

```
# 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
```

为 DLMP 聚合配置基于探测器的故障检测

必须为 DLMP 聚合配置 `probe-ip` 属性才能启用探测。否则，缺省情况下会禁用探测并仅使用基于链路的故障检测。有关更多信息，请参见“DLMP 聚合故障检测” [21]。

以下数据链路属性用于配置基于探测器的故障检测：

- `probe-ip` – 指定源 IP 地址的逗号分隔列表以及目标 IP 地址或主机名。源 IP 地址用于 ICMP 探测。源 IP 地址列表的后面是可选目标地址。目标 IP 地址必须与指定的源 IP 地址位于相同的子网上。

  可以使用 + 分隔源与目标。可以采用目标的 IP 地址或主机名来指定目标。有关如何指定源地址和目标地址的信息，请参见如何为 DLMP 配置基于探测器的故障检测 [30]。
- `probe-fdt` – 指定故障检测时间。可以配置预期故障检测时间值（秒）。缺省值为 10 秒。

▼ 如何为 DLMP 配置基于探测器的故障检测

开始之前　创建 DLMP 聚合。有关更多信息，请参见如何创建链路聚合 [24]。

1. **成为管理员。**

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. **（可选） 显示所有现有聚合以识别用于配置基于探测器的故障检测的聚合。**

   ```
   # dladm show-aggr
   ```

3. **为要配置基于探测器的故障检测的聚合设置探测器目标。**

   ```
   # dladm set-linkprop -p probe-ip=IP-address[/prefix-length]|hostname+[target] aggr
   ```

   可以通过以下方式指定 `probe-ip` 属性的源地址和目标地址：

   - `probe-ip=`*IP-address*[/*prefix-length*]|*hostname*+[*target*]

     IP 地址及其前缀长度。例如，`10.130.10.1/24+`。
   - `probe-ip=`*addr-obj-name*+[*target*]

     地址对象名称。例如，`vnic1/addr1+192.168.0.1`。
   - `probe-ip=`*interface-name*+[*target*]

     接口名称，可以是聚合接口自身的名称或在该聚合上配置的任何 VNIC。例如，`[aggr1]+`。
   - `probe-ip=+`[*target*]

     未指定 IP 地址。如果未指定源 IP 地址，在聚合和 VNIC 上配置的所有 IP 地址将被视为 ICMP 探测器的潜在源 IP 地址。例如，`+10.130.10.1`。

4. **（可选） 设置故障检测时间。**

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

   其中，*fdt* 是指定的故障检测时间（秒）。缺省值为 10 秒。

5. **（可选） 显示聚合以查看与探测器相关的信息。**

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

例 2-6　配置基于探测器的故障检测

1. 显示现有聚合。

   ```
   # dladm show-aggr
   ```

```
LINK    MODE       POLICY  ADDRPOLICY      LACPACTIVITY   LACPTIMER
aggr0   dlmp       --      --              --             --
aggr1   dlmp       --      --              --             --
```

2. 设置 aggr1 的探测器目标。

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

由于未指定源 IP 地址，在聚合 aggr1 和 VNIC 上配置的所有 IP 地址将成为 ICMP 探测器的源 IP 地址。

3. 设置故障检测时间。

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

4. 显示所设置的属性。

```
# dladm show-linkprop -p probe-ip,probe-fdt aggr1
LINK     PROPERTY         PERM   VALUE           EFFECTIVE    DEFAULT      POSSIBLE
aggr1    probe-ip         rw     192.168.85.137   --           --        --
aggr1    probe-fdt        rw     15               15           10          1-600
```

5. 显示聚合的探测器的统计信息。

```
# 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
```

监视基于探测器的故障检测

可以使用 dladm show-aggr、dlstat show-aggr 和 ipadm show-addr 命令监视基于探测器的故障检测。

例 2-7 显示与探测器相关的信息

以下示例显示了 DLMP 聚合的探测器的统计信息。使用探测器类型 -P 选项，可以在 dlstat show-aggr 命令中提供参数的逗号分隔列表（t 表示传递式探测器，i 表示 ICMP 探测器，all 表示 ICMP 和传递式探测器），以分别显示各探测器类型的探测器。

```
# 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
: 发送探测器的时间（秒）。该时间与发出 dlstat 命令的时间有关。如果探测器是在发出 dlstat 命令之前发送的，则时间为负数。

AGGR
: 为其发送探测器的聚合名称。

LOCAL
: ICMP 探测器：探测器的源 IP 地址。
传递式探测器：从其发送传递式探测器的端口名称。

TARGET
: ICMP 探测器：探测器的目标 IP 地址。
传递式探测器：探测器的目标端口名称。

PROBE
: 代表探测器的标识号。前缀 t 表示传递式探测器，前缀 i 表示 ICMP 探测器。

NETRTT
: 探测器的网络往返时间。该值是发送探测器与 DLMP 聚合收到确认之间的时间段。

RTT
: 探测器的总往返时间。该值是发送探测器和 DLMP 聚合完成确认过程之间的时间段。

有关更多信息，请参见 dlstat(1M) 手册页。

例 2-8 显示有关聚合端口的详细信息

以下示例显示了每个底层端口的详细聚合信息。

```
# 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
```

有关更多信息，请参见 dladm(1M) 手册页。

例 2-9 显示聚合端口的状态

以下示例显示了聚合端口的状态以及端口的目标 IP 地址。

```
# 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 2-10 显示 probe-ip 属性值

以下示例显示了有关指定的 DLMP 聚合的链路属性 probe-ip 的详细信息。

```
# 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 2-11 显示聚合的 IP 地址和状态

以下示例显示了聚合的 IP 地址和状态。

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

删除链路聚合

可以使用 dladm delete-aggr 命令删除链路聚合。删除聚合之前，必须删除在链路聚合上配置的 IP 接口和 VNIC。

▼ 如何删除链路聚合

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 删除在链路聚合上配置的 IP 接口。

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

   其中 *IP-aggr* 是链路聚合上的 IP 接口。

3. 删除链路聚合。

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

例 2-12 删除链路聚合

本示例说明如何删除聚合 aggr0。该删除具有永久性。

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

在中继聚合与 DLMP 聚合之间切换

在中继聚合与 DLMP 聚合之间切换会更改整个配置，因此，与仅修改其他链路聚合属性相比，此过程对聚合的影响更广。

▼ 如何切换链路聚合类型

开始之前
- 如果从中继聚合切换为 DLMP 聚合，则必须删除之前为中继聚合创建的交换机配置。
- 如果从 DLMP 聚合切换，则必须确保所有链路均位于同一交换机上，并且为该交换机配置了聚合。

1. **成为管理员。**

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. **确定链路聚合的当前类型。**

   ```
   # dladm show-aggr
   ```

   输出的 MODE 字段指示聚合的当前类型。如果是中继聚合，则 MODE 值为 trunk；如果是 DLMP 聚合，则值为 dlmp。

3. **切换聚合。**

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

   其中，如果要切换为中继聚合，则 *mode* 为 trunk；如果要切换为 DLMP 聚合，则为 dlmp；*aggr* 为聚合名称。

4. **根据新的链路聚合类型的要求配置交换机。**

   请参见交换机制造商文档了解如何配置交换机。

5. **（可选） 验证当前链路聚合配置。**

```
# dladm show-aggr
```

例 2-13 从中继聚合切换为 DLMP 聚合

本示例说明如何将聚合从中继聚合更改为 DLMP 聚合。

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

# dladm modify-aggr -m dlmp aggr0
# dladm show-aggr
LINK    MODE       POLICY  ADDRPOLICY      LACPACTIVITY   LACPTIMER
aggr0   dlmp       --      --              --             --
```

切换聚合之后，必须删除先前应用于中继聚合的交换机配置。

使用案例：配置链路聚合

以下端到端使用案例说明了如何完成以下操作：

- 创建 DLMP 聚合。
- 将链路添加到聚合。
- 在聚合上配置 IP 接口。
- 在聚合上配置 VNIC。
- 为聚合配置基于探测器的故障检测。
- 在路由表中配置 IP 地址。
- 监视 ICMP 和传递式探测器。

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。
2. 显示数据链路信息以识别聚合的数据链路。

   ```
   # dladm show-link
   LINK      CLASS     MTU     STATE   OVER
   net0      phys      1500    up      --
   net1      phys      1500    up      --
   net2      phys      1500    up      --
   ```
3. 确保您希望聚合的数据链路未在链路上配置 IP 接口。如果在任何链路上配置了任何接口，则删除该接口。

   ```
   # ipadm show-if
   IFNAME       CLASS        STATE     ACTIVE     OVER
   ```

```
lo0          loopback     ok        yes        --
net0         ip           ok        no         --
# ipadm delete-ip net0
```

4. 使用链路 net0 和 net1 创建 DLMP 聚合。

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

5. 将另一个链路 (net2) 添加到聚合。

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

根据现有交换机配置的需要，重新配置交换机以容纳新的链路。请参见交换机制造商文档。

6. 在聚合 aggr1 之上配置 IP 接口。

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

7. 在聚合之上创建 VNIC。

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

8. 为聚合配置基于探测器的故障检测。

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

未指定探测器的源 IP 地址和目标 IP 地址。因此，需要在路由表中配置目标以启用探测。

9. 将路由表中的目标配置为与指定的 IP 地址位于相同子网中。

```
# route add -host 10.10.10.2 10.10.10.2 -static
```

10. 显示聚合端口和目标的状态。

```
# 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
```

11. 监视 ICMP 探测器统计信息。

```
# 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
```

12. 监视端口之间的传递式探测器统计信息。

```
# 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   --      --
```

创建了聚合 `aggr0`，其中包含在其上配置的 IP 接口。在聚合 `aggr0` 之上配置了 VNIC `vnic1`。配置了基于探测器的故障检测，而未指定探测器的源 IP 地址或目标 IP 地址。为了启用探测，路由表中的目标配置有 IP 地址 `10.10.10.2`，该地址与指定的 IP 地址 `10.10.10.1` 位于同一子网。同时监视了 ICMP 和传递式探测器统计信息。

中继聚合与 DLMP 聚合的比较

本节提供了两种链路聚合类型的总体比较。

表 2-1 中继聚合与 DLMP 聚合的功能比较

功能	中继聚合	DLMP 聚合
基于链路的故障检测	支持	支持
LACP	支持	不支持
使用备用接口	不支持	不支持[1]
跨越多个交换机	除非使用供应商专有解决方案，否则不支持	支持

功能	中继聚合	DLMP 聚合
交换机配置	必需	非必需
负载平衡策略	支持	不适用
将负载分布在聚合的所有端口中	支持	有限[2]
用于资源管理的用户定义流	支持	支持
链路保护	支持	支持
背对背配置	支持	不支持[3]

[1] 每个 DLMP 客户机都仅与一个 DLMP 端口相关联。其余端口可以作为 DLMP 客户机的可用端口，但是无法配置这些可用端口。

[2] 聚合将其 VNIC 分布在所有端口中。但是，各个 VNIC 无法在多个端口上分配负荷。

[3] DLMP 聚合必须始终使用中间交换机将包发送到其他目标系统。但是，对于 DLMP 而言，不需要配置交换机。

♦ ♦ ♦ 第 3 章

使用虚拟局域网配置虚拟网络

本章论述了虚拟局域网 (virtual local area network, VLAN) 的功能和优势。同时还介绍了配置和修改 VLAN 的过程。

本章包含以下主题：

部署 VLAN 概述

虚拟局域网 (virtual local area network, VLAN) 是在协议栈的数据链路层上对局域网的细分。可以为采用交换机技术的局域网创建 VLAN。可以将同一系统上的接口指定给不同的 VLAN。

何时使用 VLAN

如果需要执行以下操作，则可以部署 VLAN：

- 创建工作组的逻辑分区。

  例如，如果某一建筑一个楼层中的所有主机均连接到一个基于交换机的本地网络，则可以为该楼层的每个工作组创建单独的 VLAN。

- 对各个工作组强制实施不同的安全策略。

  例如，财务部和信息技术部的安全需求大不相同。可以为每个部门创建单独的 VLAN 并在每个 VLAN 上强制执行相应的安全策略。
- 减小广播域的大小并提高网络效率。可以将工作组拆分为易管理的广播域。

  例如，在由 25 个用户组成的广播域中，如果广播通信仅适用于 12 个用户，则为这 12 个用户设置单独的 VLAN 可减少通信流量并提高网络效率。

指定 VLAN 名称

VLAN 演示了使用通用名称或定制名称的优势。在先前的发行版中，VLAN 通过物理连接点 (physical point of attachment, PPA)（需要将数据链路基于硬件的名称与 VLAN ID 组合在一起）进行标识。但是，在 Oracle Solaris 中，现在您可以选择更有意义的名称来标识 VLAN。名称必须符合《在 Oracle Solaris 11.2 中配置和管理网络组件 》中的“有效链路名称的规则”中提供的数据链路命名规则。例如，可以指定 `sales0` 或 `marketing1` 等定制 VLAN 名称。

VLAN 名称与 VLAN ID 结合使用。局域网中的每个 VLAN 均由一个 VLAN ID 标识，该 ID 是 VLAN 标记的一部分。VLAN ID 在 VLAN 配置过程中指定。配置交换机以支持 VLAN 时，需要为每个端口指定一个 VLAN ID。端口的 VLAN ID 必须与为连接到该端口的接口所指定的 VLAN ID 相同。

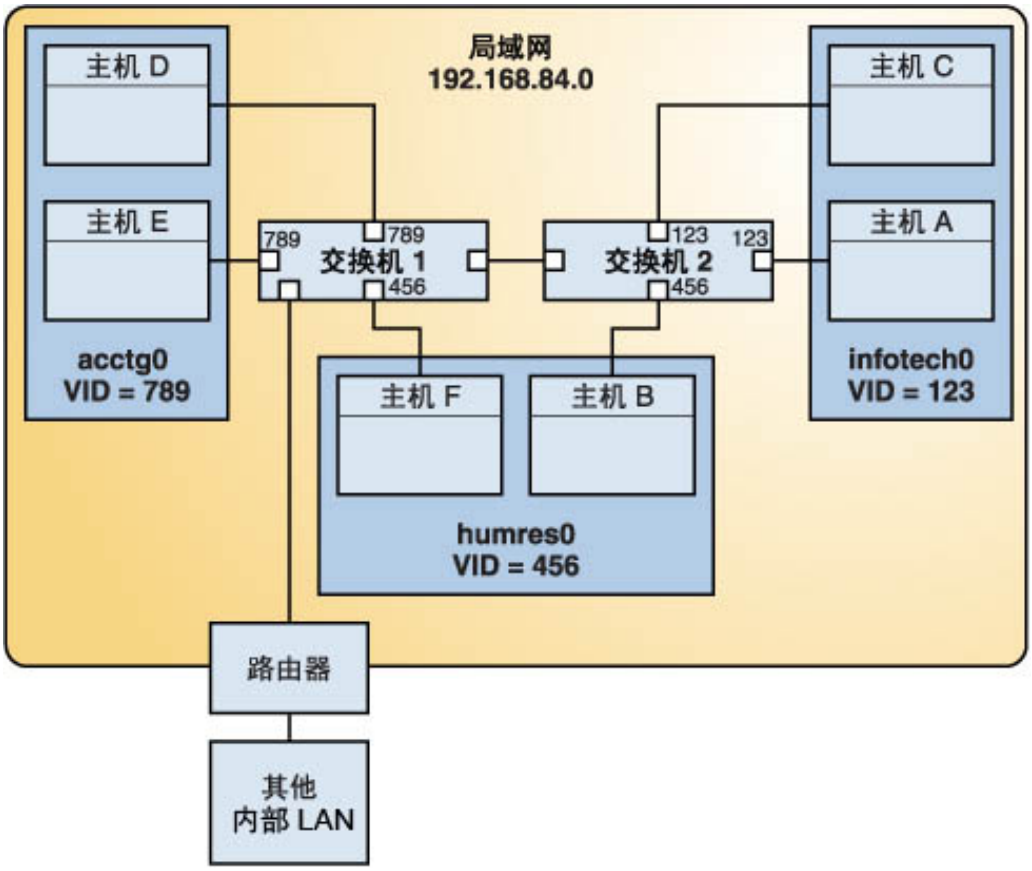
缺省情况下，每个端口均具有一个称为端口 VLAN ID 的 VLAN ID。属于该 VLAN ID 的包不带有 VLAN 标记。在 Oracle Solaris 中，可以使用数据链路属性 `default_tag` 在接口上显示和更改端口 VLAN ID。

VLAN 拓扑

使用交换 LAN 技术，可以将本地网络中的系统组织到 VLAN 中。将本地网络划分为 VLAN 之前，必须先获取支持 VLAN 技术的交换机。可以对交换机上的所有端口进行配置，使其为单个 VLAN 或多个 VLAN 提供服务，具体取决于 VLAN 拓扑。配置交换机端口的过程因交换机制造商而异。

下图显示了被划分为三个 VLAN 的局域网。

图 3-1 具有三个 VLAN 的局域网



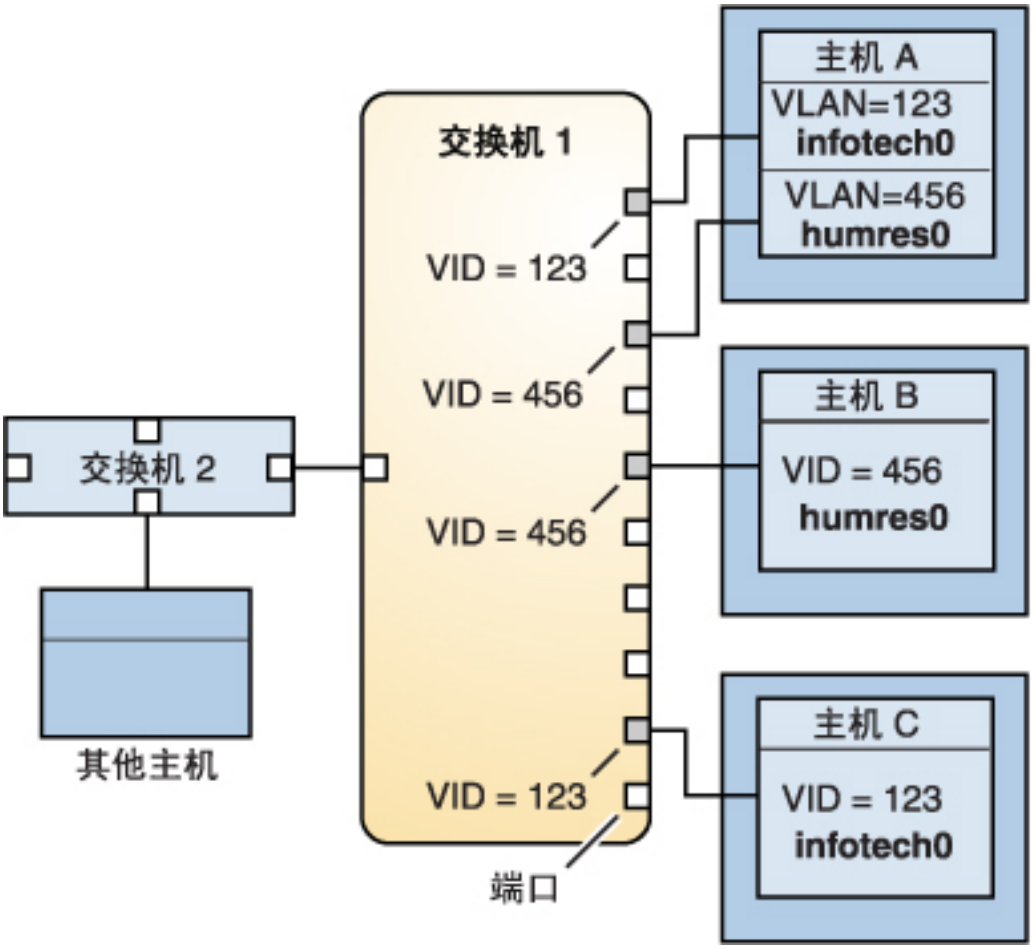
在图中，LAN 的子网地址为 192.168.84.0。

此 LAN 被细分为三个 VLAN，对应于三个工作组：

- VLAN ID 为 789 的 acctg0 – 会计组。此组拥有主机 D 和主机 E。
- VLAN ID 为 456 的 humres0 – 人力资源组。此组拥有主机 B 和主机 F。
- VLAN ID 为 123 的 infotech0 – 信息技术组。此组拥有主机 A 和主机 C。

下图显示了此图的一种变体，其中仅使用一个交换机，属于不同 VLAN 的多个主机均连接到此单个交换机。

图 3-2 一个交换机将属于不同 VLAN 的多个主机连接起来



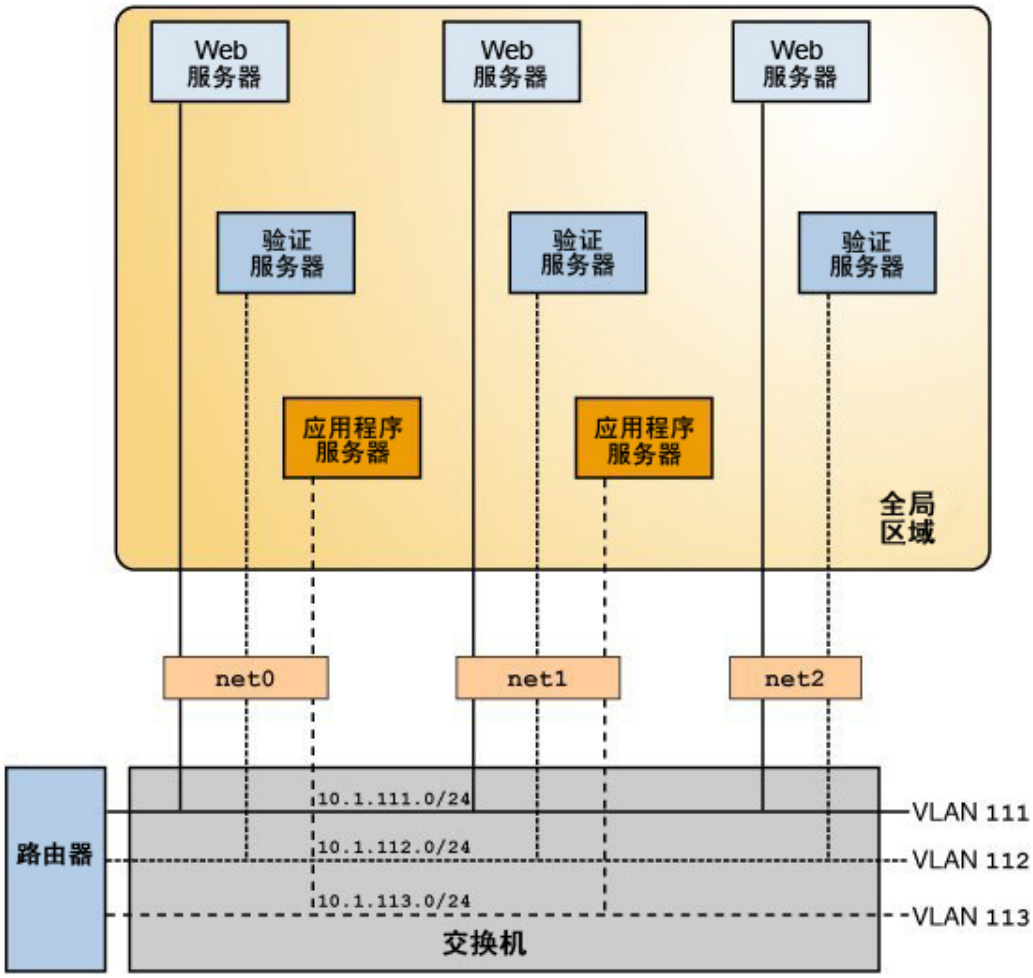
在此图中，主机 A 和主机 C 属于信息技术 VLAN，其 VLAN ID 为 123。主机 A 的其中一个接口配置有 VLAN ID 123。此接口连接到交换机 1 上的端口 1，此端口也使用 VLAN ID 123 进行配置。主机 B 是人力资源 VLAN 的成员，其 VLAN ID 为 456。主机 B 的接口连接到交换机 1 上的端口 5，此端口使用 VLAN ID 456 进行配置。最后，主机 C 的接口使用 VLAN ID 123 进行配置。此接口连接到交换机 1 上的端口 9。端口 9 也使用 VLAN ID 123 进行配置。

此图还显示单个主机可以属于多个 VLAN。例如，主机 A 在其接口上配置了两个 VLAN。第二个 VLAN 配置有 VLAN ID 456 并连接到配置有 VLAN ID 456 的端口 3。因此，主机 A 同时是 `infotech0` 和 `humres0` VLAN 的成员。

将 VLAN 与区域结合使用

通过将 VLAN 与 Oracle Solaris 区域结合使用，可以在单个网络单元（如交换机）内配置多个虚拟网络。以下图为例，其中描述了具有三个物理网卡的系统：`net0`、`net1` 和 `net2`。

图 3-3 具有多个 VLAN 的系统



在没有 VLAN 情况下，您需要配置不同的系统来执行特定功能并将这些系统连接到单独的网络。例如，将 Web 服务器连接到一个 LAN，将验证服务器连接到另一个 LAN，并

将应用服务器连接到第三个 LAN。使用 VLAN 和区域，您可以组合所有八个系统并将它们作为区域配置在单个系统中。然后，可以使用 VLAN ID 将 VLAN 指定给执行相同功能的每组区域。下表列出了该图中提供的信息。

功能	区域名称	VLAN 名称	VLAN ID	IP 地址	NIC
Web 服务器	webzone1	web1	111	10.1.111.0	net0
验证服务器	authzone1	auth1	112	10.1.112.0	net0
应用程序服务器	appzone1	app1	113	10.1.113.0	net0
Web 服务器	webzone2	web2	111	10.1.111.1	net1
验证服务器	authzone2	auth2	112	10.1.112.1	net1
应用程序服务器	appzone2	app2	113	10.1.113.1	net1
Web 服务器	webzone3	web3	111	10.1.111.2	net2
验证服务器	authzone3	auth3	112	10.1.112.2	net2

要创建图中所示的配置，请参阅例 3-2 “配置具有区域的 VLAN”。

规划 VLAN 配置

规划 VLAN 配置包含以下步骤：

1. 检查 LAN 拓扑，并确定在何处划分 VLAN 比较合适。
   有关此类拓扑的基本示例，请参阅图 3-1 “具有三个 VLAN 的局域网”。
2. 创建 VLAN ID 的编号方案，并为每个 VLAN 指定 VLAN ID。

注 - 如果网络上已存在 VLAN 编号方案，则必须在现有的 VLAN 编号方案范围内创建 VLAN ID。

---

3. 在每个系统上，确定哪些接口是特定 VLAN 的组件。
   a. 使用 `dladm show-link` 命令确定系统上配置了哪些链路。
   b. 确定哪个 VLAN ID 将与系统上的每个数据链路相关联。
   c. 创建 VLAN。
4. 检查数据链路与网络交换机的连接。
   记下每个数据链路的 VLAN ID 以及每个接口所连接到的交换机端口。
5. 使用与交换机连接的接口相同的 VLAN ID 配置交换机上的每个端口。
   有关配置说明，请参阅交换机制造商的文档。

配置 VLAN

以下过程说明了如何使用 `dladm` 命令在数据链路上创建 VLAN。可以在 VLAN 上创建 IP 接口，并使用 `ipadm` 命令为接口配置 IP 地址。有关 `dladm` 和 `ipadm` 命令的信息，请参见 `dladm(1M)` 和 `ipadm(1M)` 手册页。

▼ 如何配置 VLAN

开始之前 本过程假定已在系统上创建了区域。有关区域配置的信息，请参阅《创建和使用 Oracle Solaris 区域 》中的第 1 章 “如何规划和配置非全局区域”。

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 确定在系统中使用的链路的类型。

   ```
   # dladm show-link
   ```

3. 在数据链路上创建一个 VLAN 链路。

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

   link 指定正在其上创建 VLAN 接口的链路。

   vid 表示 VLAN ID 号。

   VLAN-link 指定 VLAN 的名称，这也可以是有意义的定制名称。有关 VLAN 名称的信息，请参见“指定 VLAN 名称” [40]。

4. 验证 VLAN 配置。

   ```
   # dladm show-vlan
   ```

5. 在 VLAN 上创建 IP 接口。

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

   其中 *interface* 提供 VLAN 名称。

6. 使用一个 IP 地址配置 IP 接口。

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

例 3-1 创建 VLAN

此示例说明了如何创建图 3-1 “具有三个 VLAN 的局域网” 中所示的 VLAN 配置。

1. 检查可用链路并在特定链路上创建 VLAN。

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

2. 主机 A：

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

主机 C：

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

主机 F：

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

主机 B：

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

主机 D：

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

主机 E：

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

3. 显示所创建的 VLAN。

```
# dladm show-vlan
LINK            VID     OVER        FLAGS
infotech0       123     net0        ----
infotech0       123     net0        ----
humres0         456     net0        ----
humres0         456     net0        ----
acctg0          789     net0        ----
acctg0          789     net0        ----
```

例 3-2 配置具有区域的 VLAN

本示例说明如何创建图 3-3 “具有多个 VLAN 的系统” 所示的 VLAN 配置。此示例假定您已在系统中配置了不同的区域。有关配置区域的更多信息，请参见《创建和使用 Oracle Solaris 区域》。

1. 检查可用于配置 VLAN 的可用链路，然后在特定链路上创建 VLAN。

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

global# dladm create-vlan -l net0 -v 111 web1
global# dladm create-vlan -l net0 -v 112 auth1
global# dladm create-vlan -l net0 -v 113 app1
global# dladm create-vlan -l net1 -v 111 web2
global# dladm create-vlan -l net1 -v 112 auth2
global# dladm create-vlan -l net1 -v 113 app2
global# dladm create-vlan -l net2 -v 111 web3
global# dladm create-vlan -l net2 -v 112 auth3

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

当显示链路信息时，列表中包含 VLAN。

```
global# 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
```

2. 将 VLAN 指定给其各自的区域并显示每个区域的类似于以下内容的信息：

```
global# zonecfg -z webzone1 info net
net:
address not specified
physical: web1
net:
address not specified
physical: web2
net:
address not specified
physical: web3

global# zonecfg -z authzone1 info net
net:
address not specified
physical: auth1
net:
address not specified
physical: auth2
net:
address not specified
physical: auth3

global# zonecfg -z appzone2 info net
net:
address not specified
physical: app1
net:
address not specified
physical: app2
```

属性 `physical` 的值表示为给定区域设置的 VLAN。

3. 显示在区域中指定的 VLAN。

```
global# dladm show-vlan
LINK            VID  OVER  FLAGS
webzone1/web1   111  net0  --
authzone1/auth1 112  net0  --
appzone1/app1   113  net0  --
webzone1/web2   111  net1  --
authzone1/auth2 112  net1  --
appzone1/app2   113  net1  --
webzone1/web3   111  net2  --
```

```
authzone2/auth3 111  net2  --
```

4. 登录到每个非全局区域来为 VLAN 配置 IP 地址。

   在 webzone1 中：

```
webzone1# ipadm create-ip web1
webzone1# ipadm create-addr -a 10.1.111.0/24 web1
ipadm: web1/v4
```

   在 webzone2 中：

```
webzone2# ipadm create-ip web2
webzone2# ipadm create-addr -a 10.1.111.1/24 web2
ipadm: web2/v4
```

   在 webzone3 中：

```
webzone3# ipadm create-ip web3
webzone3# ipadm create-addr -a 10.1.111.2/24 web3
ipadm: web3/v4
```

   在 authzone1 中：

```
authzone1# ipadm create-ip auth1
authzone1# ipadm create-addr -a 10.1.112.0/24 auth1
ipadm: auth1/v4
```

   在 authzone2 中：

```
authzone2# ipadm create-ip auth2
autzone2# ipadm create-addr -a 10.1.112.1/24 auth2
ipadm: auth2/v4
```

   在 authzone3 中：

```
authzone3# ipadm create-ip auth3
authzone3# ipadm create-addr -a 10.1.112.2/24 auth3
ipadm: auth3/v4
```

   在 appzone1 中：

```
appzone1# ipadm create-ip app1
appzone1# ipadm create-addr -a 10.1.113.0/24 app1
ipadm: app1/v4
```

   在 appzone2 中：

```
appzone2# ipadm create-ip app2
appzone2# ipadm create-addr -a 10.1.113.1/24 app2
ipadm: app2/v4
```

配置完所有 VLAN 的 IP 地址后，配置过程即完成。三个 VLAN 可以运行并可以承载其各自区域的通信。

在链路聚合上配置 VLAN

可以在链路聚合上创建 VLAN，方式与在接口上配置 VLAN 类似。第 2 章 使用链路聚合配置高可用性中介绍了链路聚合。本节综合介绍了如何配置 VLAN 和链路聚合。

▼ 如何在链路聚合上配置 VLAN

开始之前 创建链路聚合。有关如何创建链路聚合的信息，请参阅如何创建链路聚合 [24]。

1. **列出系统上配置的链路聚合。**

   ```
   # dladm show-aggr
   ```

2. **对于要在您选择的链路聚合上创建的每个 VLAN，可以使用以下命令：**

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

   link 指定正在其上创建 VLAN 接口的链路。

   注 - 在此过程中，链路指链路聚合。

   vid 表示 VLAN ID 号。

   VLAN-link 指定 VLAN 的名称。

3. **对于上一步中创建的每个 VLAN，创建该 VLAN 上的 IP 接口。**

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

   其中 *interface* 使用 VLAN 名称。

4. **为 VLAN 上的各个 IP 接口配置有效的 IP 地址。**

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

例 3-3 在链路聚合上配置多个 VLAN

在此示例中，基于链路聚合配置了两个 VLAN。为 VLAN 指定的 VLAN ID 分别为 193 和 194。

```
# 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
```

在传统设备上配置 VLAN

某些传统设备只处理最大传输单元 (maximum transmission unit, MTU) 大小（也称为帧大小）为 1514 字节的包。帧大小超出此最大限制的包将被丢弃。对于这种情况，请按照如何配置 VLAN [45]中列出的过程执行操作。但是，当创建 VLAN 时，请使用 `-f` 选项强制创建 VLAN。

▼ 如何在传统设备上配置 VLAN

1. 使用 **-f** 选项创建 VLAN。

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

`-f`	强制创建 VLAN。在不允许足以容纳 VLAN 标头的帧大小的设备上创建 VLAN 时，请使用此选项。
`-l` *link*	指定在其上创建 VLAN 接口的链路。在此过程中，链路指传统设备。
`-v` *vid*	表示 VLAN ID 号。
VLAN-link	指定 VLAN 的名称，也可以是通过管理行为选择的名称。

2. 为最大传输单元 (maximum transmission unit, **MTU**) 设置一个较低的大小。

   在以下示例中，`mtu` 设置为 `1496`。

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

较低的 MTU 值为链路层在传输之前插入 VLAN 标头提供了空间。

3. 重复步骤 2，为 VLAN 中的每个节点设置 MTU 值。
有关更改链路属性值的更多信息，请参阅《在 Oracle Solaris 11.2 中配置和管理网络组件 》中的“管理数据链路属性”。

显示 VLAN 信息

因为 VLAN 是数据链路，所以可以使用 `dladm show-link` 命令显示有关 VLAN 的信息。使用 `dladm show-vlan` 命令显示有关 VLAN 的特定信息。

以下示例比较了可以通过 `dladm show-link` 或 `dladm show-vlan` 命令获取的信息类型。第一个示例使用 `dladm show-link` 命令显示了系统上的所有数据链路，包括非 VLAN 的数据链路。第二个示例使用 `dladm show-vlan` 命令显示了仅与 VLAN 相关的数据链路信息子集。

```
# 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     OVER     FLAGS
web1      111     net0     ----
auth1     112     net0     ----
app1      113     net0     ----
web2      111     net1     ----
auth2     112     net1     ----
app2      113     net1     ----
web3      111     net2     ----
auth3     113     net2     ----
```

修改 VLAN

可以使用 `dladm modify-vlan` 命令通过以下方式修改 VLAN：

- 更改 VLAN 的 VLAN ID
- 将 VLAN 迁移到另一个底层链路

修改 VLAN 的 VLAN ID

要更改 VLAN 的 VLAN ID，请使用以下命令之一：

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

  其中，*vid* 指定要分配给 VLAN 的新 VLAN ID，*datalink* 指的是在其上配置 VLAN 的底层链路。

注 - 仅当数据链路上存在单个 VLAN 时，才能使用 `dladm modify-vlan -v` *vid* `-L` *datalink* 命令语法。如果对配置了多个 VLAN 的数据链路使用此命令语法，则命令将失败，因为数据链路上的每个 VLAN 必须具有唯一的 VLAN ID。

如果修改链路上的 VLAN ID，则还必须配置新 VLAN ID 的交换机端口。

- `dladm modify-vlan -v` *vid vlan*

  使用此命令可更改单个数据链路上多个 VLAN 的唯一 VLAN ID。数据链路上的每个 VLAN 都具有唯一 VLAN ID，因此，一次只能更改一个 VLAN ID。在图 3-3 “具有多个 VLAN 的系统” 所示的设置中，需要更改在 `net0` 上配置的 `web1`、`auth1` 和 `app1` 的 VLAN ID，如下所示：

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

将 VLAN 迁移到另一个底层链路

您可以将 VLAN 从一个底层数据链路迁移到另一个底层数据链路，而无需删除和重新配置 VLAN。底层链路可以是物理链路、链路聚合或 etherstub。有关 etherstub 的更多信息，请参见《在 Oracle Solaris 11.2 中管理网络虚拟化和网络资源》中的“虚拟网络组件”。

要成功迁移 VLAN，VLAN 要移动到的底层数据链路必须能够接纳此 VLAN 的数据链路属性。如果不支持这些属性，则迁移将会失败并通知用户。成功迁移后，如果 VLAN 仍然连接到网络，使用此 VLAN 的所有应用程序将继续正常运行。

迁移 VLAN 后，某些与硬件相关的属性可能会改变。例如，VLAN 始终与其底层数据链路共享相同的 MAC 地址。因此，迁移 VLAN 后，VLAN 的 MAC 地址将更改为目标数

据链路的主 MAC 地址。其他可能会受影响的属性包括数据链路状态、链路速度和 MTU 大小。但是，应用程序不间断地继续运行。

注 - 迁移后的 VLAN 不保留任何原始数据链路的硬件通道统计信息。目标数据链路上 VLAN 的可用硬件通道将成为统计信息的新来源。不过，`dlstat` 命令缺省显示的软件统计信息将会保留。

您可以全局性地或有选择性地执行 VLAN 迁移。

全局迁移

全局迁移用于将一个数据链路上配置的所有 VLAN 迁移到另一个数据链路。要执行全局迁移，只需指定源数据链路和目标数据链路。以下示例说明了如何将 `ether0` 上的所有 VLAN 移动到 `net1`。

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

`-l` 指 VLAN 要迁移到的目标数据链路。

`-L` 指在其上配置 VLAN 的原始数据链路。

注 - 目标数据链路必须在源数据链路之前指定。

选择性迁移

选择性迁移用于仅迁移选定的 VLAN。要执行有选择性的 VLAN 迁移，需要指定要移动的 VLAN。在基于图 3-3 “具有多个 VLAN 的系统” 的以下示例中，VLAN 从 `net0` 移动到 `net3`。

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

注 - 选择性地迁移 VLAN 时，请勿包括 `-L` 选项，此选项仅适用于全局迁移。

执行迁移时，可以更改 VLAN 的 VLAN ID。以下示例基于图 3-3 “具有多个 VLAN 的系统”，说明了如何同时迁移多个 VLAN 并更改其 VLAN ID。

```
# dladm show-vlan
LINK    VID      OVER     FLAGS
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      OVER     FLAGS
web1    123      net3     -----
auth1   456      net3     -----
app1    789      net3     -----
```

注 - 并行命令 `dladm modify-vnic` 用于迁移配置为 VLAN 的 VNIC。必须根据要迁移的是 VLAN 还是配置为 VLAN 的 VNIC，使用正确的子命令。针对由 `dladm show-vlan` 命令显示的 VLAN，使用 `modify-vlan` 子命令。针对在 `dladm show-vnic` 命令的输出中显示的 VNIC（包括具有 VLAN ID 的 VNIC），使用 `modify-vnic` 子命令。有关如何修改 VNIC 的信息，请参见《在 Oracle Solaris 11.2 中管理网络虚拟化和网络资源 》中的“修改 VNIC 的 VLAN ID”。

删除 VLAN

使用 `dladm delete-vlan` 命令删除系统上的 VLAN 配置。

注 - 必须首先删除要删除的 VLAN 上的所有现有 IP 配置，才能删除此 VLAN。如果 VLAN 上存在 IP 接口，删除 VLAN 将会失败。

例 3-4 删除 VLAN 配置

此示例说明了如何删除 VLAN 配置。

```
# dladm show-vlan
LINK      VID     OVER     FLAGS
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
```

使用案例：结合使用链路聚合与 VLAN 配置

本节提供了一个示例，说明了如何创建使用链路聚合和 VLAN 的网络配置组合。

在以下示例中，使用四个 NIC 的系统必须配置为针对八个单独子网的路由器。因此，将配置八个链路，分别用于每个子网。首先，在所有四个 NIC 上创建一个中继聚合。此不带标记的链路在传出帧中不包括 VLAN 标记，将成为缺省路由指向的网络的缺省不带标记的子网。

然后，在链路聚合上为其他子网配置 VLAN 接口。子网的命名基于一种颜色编码方案。因此，VLAN 名称也采用类似命名方式以对应其各自的子网。最终配置包括分别针对八个子网的八个链路：一个不带标记的链路以及七个带标记的 VLAN 链路。该示例首先验证数据链路上是否已经存在 IP 接口。必须先删除这些接口，才能将数据链路组合为聚合。

1. 删除在数据链路上配置的任何 IP 接口。

```
# 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. 创建中继聚合 `default0`。

```
# 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. 在聚合上配置 IP 接口。

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

4. 在 `default0` 上创建 VLAN。

```
# 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   OVER        FLAGS
orange0          2   default0    -----
green0           3   default0    -----
blue0            4   default0    -----
white0           5   default0    -----
yellow0          6   default0    -----
red0             7   default0    -----
cyan0            8   default0    -----
```

5. 在 VLAN 链路上创建 IP 接口并为其指定 IP 地址。

```
# 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
```

♦ ♦ ♦ 第 4 章

管理桥接功能

桥接用于连接不同的网段，使它们可以像单个网段那样通信。本章说明了如何配置和管理桥接网络。

本章包含以下主题：

桥接网络概述

网桥将网络中的各种节点连接成单个网络。这些网段共享一个广播网络，连接后进行通信时，就如同是一个网段一样。因此，每个节点可以通过使用网络协议（如 IP）而不是路由器来跨网段转发通信，从而访问其他节点。如果不使用网桥，就必须配置 IP 路由以允许节点之间的 IP 通信转发。

虽然桥接和路由都可用于分发关于网络中资源位置的信息，但它们在多个方面存在区别。路由在 IP 层 (L3) 实现并使用路由协议。数据链路层不使用路由协议，

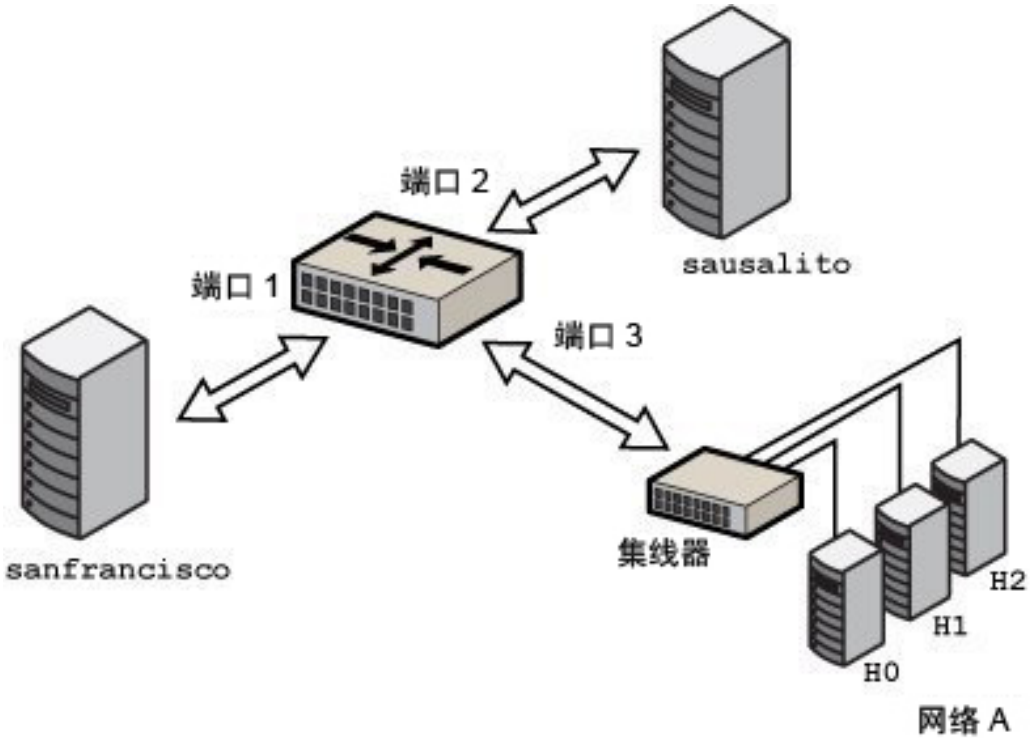
桥接用于分发关于网络中资源的位置的信息。在桥接网络中，通过检查连接到网桥的链路中接收的网络通信流量确定转发包的目的地。桥接网络使用生成树协议 (Spanning Tree Protocol, STP) 和多链路透明互连 (Transparent Interconnection of Lots of Links, TRILL) 等协议。有关更多信息，请参见“桥接协议” [64]。

注意 - 不要通过在基于 SPARC® 的使用桥接的系统上使用 `eeprom` 命令将 `local-mac-address?` 属性设置为 `false`。如果这样做，这些系统会错误地为同一个网络中的多个端口使用同一 MAC 地址。

简单的桥接网络

下图显示了一个简单的桥接网络配置。

图 4-1 简单的桥接网络

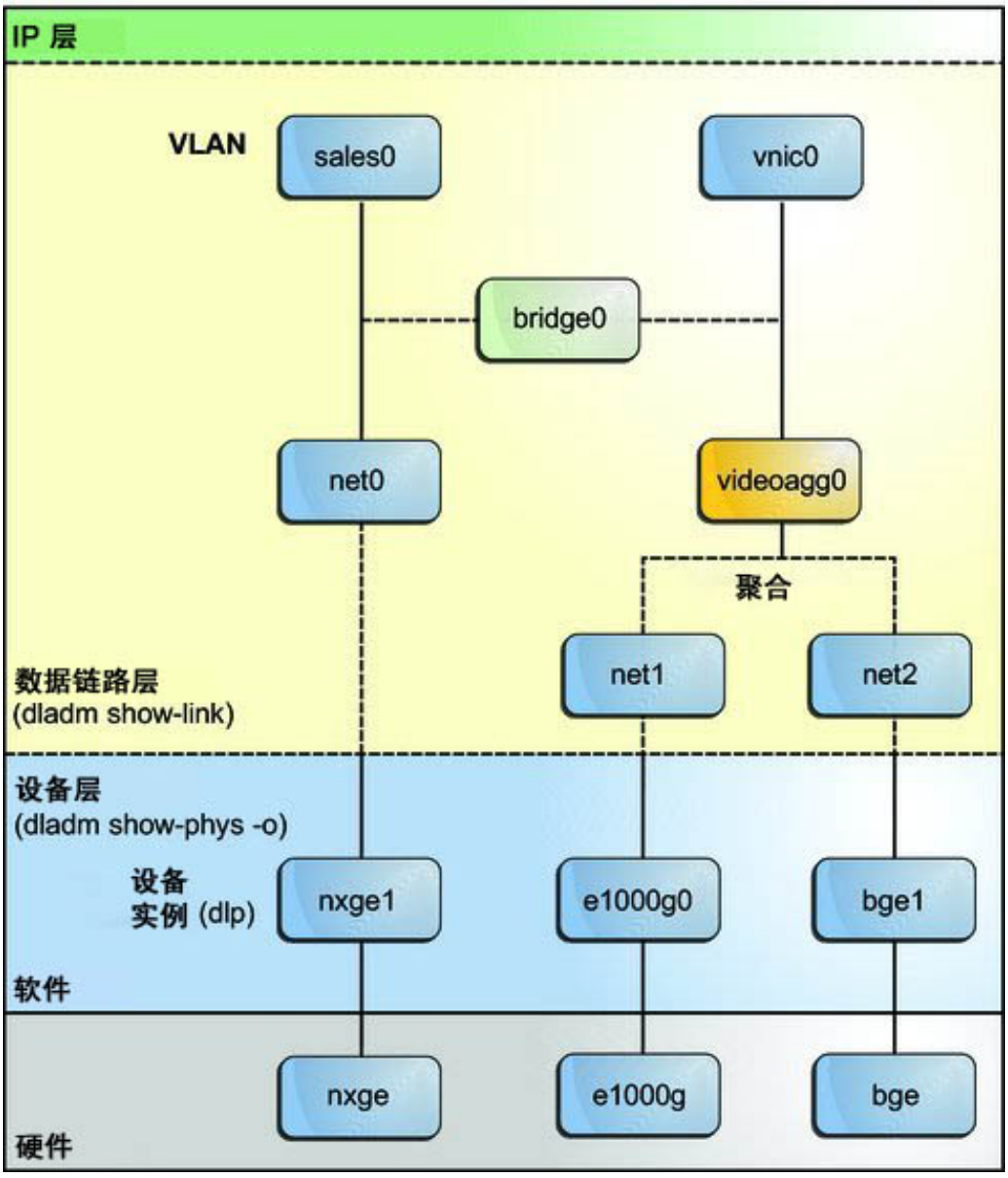


网桥 `goldengate` 是一个配置了桥接的 Oracle Solaris 系统。系统 `sanfrancisco` 和 `sausalito` 以物理方式连接到该网桥。网络 A 使用一个集线器，该集线器一侧物理连接到网桥，另一侧连接到三个计算机系统。网桥端口是链路 `net0`、`net1` 和 `net2`。

如何在网络栈中实现 Oracle Solaris 网桥

在 Oracle Solaris 中，您可以在同一网络栈实现的数据链路层上配置网桥，如下图所示。

图 4-2 Oracle Solaris 网络栈中的网桥

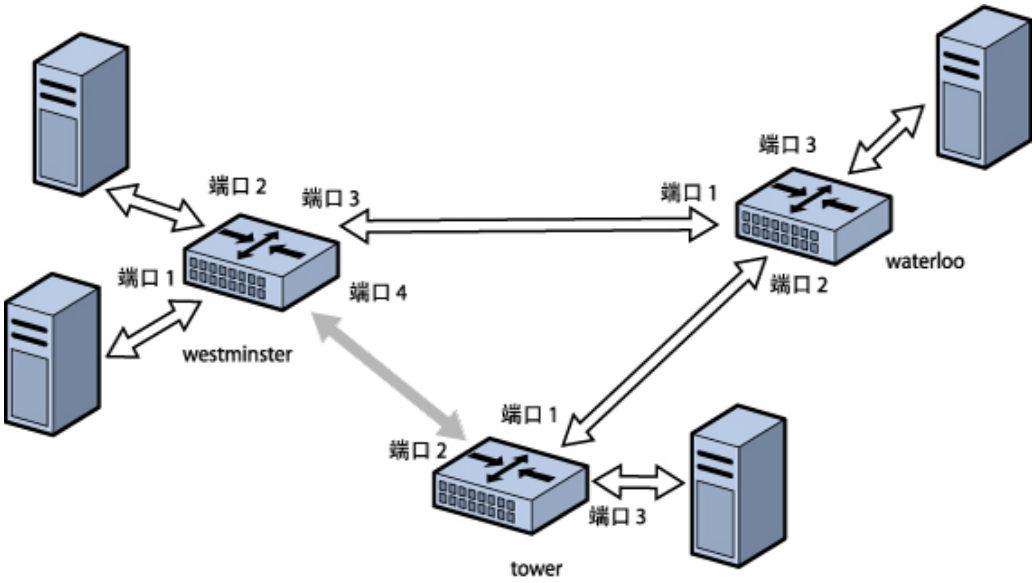


两个接口 `net0` 和 `videoagg0` 被配置为网桥 `bridge0`。在一个接口上接收到的数据包会转发到另一个接口。采用网桥配置后，这两个接口仍可用于配置 VLAN 和 IP 接口。

桥接网络环

桥接网络可以形成环，这些环将多个网桥物理连接在一起。下图显示了一个桥接网络环配置。

图 4-3 桥接网络环



此图显示了配置成一个环的桥接网络。此配置显示三个网桥。两个系统物理连接到 `westminister` 网桥。一个系统物理连接到 `waterloo` 网桥，一个系统物理连接到 `tower` 网桥。网桥通过网桥端口彼此物理连接。

这种类型的配置可能导致因旧包无休止地在环中循环而使网络链路饱和的问题。为了防止出现这种循环情况，Oracle Solaris 网桥实施 STP 和 TRILL 协议。请注意，大多数硬件网桥还实现 STP 循环保护。

桥接网络的工作方式

如果网桥接收到一个包，将检查其源地址。包的源地址将发送该包的节点与接收该包的链路相关联。此后，当接收的包使用同一地址作为目标地址时，网桥将包通过链路转发到该地址。

与源地址关联的链路可能是连接到桥接网络中另一个网桥的中间链路。随着时间的推移，桥接网络中所有网桥都将“学会”通过哪个链路向给定的节点发送包。因此，包的目标地址用于通过逐跳桥接的方式将包导向其最终目标。

本地“链路断开”通知指示给定的链路上的所有节点都不再可访问。在此情况下，到该链路的数据包转发将停止，而通过该链路的所有转发条目将被清除。随时间的推移，还将清除早期转发条目。当恢复一个链路时，此链路上接收的包被视为新包。学习过程将基于包的源地址再次开始。通过此过程，当该地址用作目标地址时，网桥能够通过该链路正确转发包。

桥接协议

桥接网络使用以下协议：

- 跨越树协议 (Spanning Tree Protocol, STP)

  STP 是桥接网络使用的缺省协议。桥接使用 STP 机制防止网络循环可能使子网不可用。要将包转发到目的地，网桥必须以混杂模式侦听连接到该网桥的每个链路。以混杂模式侦听导致网桥容易发生转发循环，发生转发循环时，包以全线速无限循环。

注意 - 如果要求最高级别的性能，请不要将链路配置到网桥中。桥接要求底层接口处于混杂模式下，而该模式会禁用对硬件 (NIC)、驱动程序和系统其他层的多种重要优化。禁用这些性能增强功能是桥接机制不可避免的结果。

这些性能问题只影响那些被配置为网桥的一部分的链路。如果系统中的某些链路不是桥接的，因而不会受到这些限制，您就可以在该系统中使用网桥。

- 多链路透明互连 (Transparent Interconnection of Lots of Links, TRILL)

  Oracle Solaris 支持 TRILL 保护增强功能，这可以在不禁用链路的情况下避免循环。TRILL 有助于实现通向目标的多个路径之间的通信流量负载平衡。

当使用 STP 进行循环保护时，通过防止循环中的连接之一转发包，减少了物理循环。图 4-3 “桥接网络环” 显示 `westminster` 和 `tower` 网桥之间的物理链路不用于转发包。

与 STP 不同，TRILL 不通过关闭物理链路来防止出现循环。相反，TRILL 计算网络中的每个 TRILL 节点的最短路径信息，并使用该信息将包转发到各个目的地。

通过在 `dladm create-bridge` 或 `dladm modify-bridge` 命令中指定 `-P trill` 选项可以使用 TRILL。有关更多信息，请参见“创建网桥” [66]和“修改网桥的保护类型” [67]。

有关 STP 的信息，请参见 IEEE 802.1 D-1998。有关 TRILL 的信息，请参见 Internet 工程任务组 (Engineering Task Force, IETF) TRILL 草稿 (http://tools.ietf.org/wg/trill)。

STP 守护进程

使用 dladm create-bridge 命令创建的各网桥被表示为 svc:/network/bridge 的同名服务管理工具 (Service Management Facility, SMF) 实例。每个实例运行实现 STP 的 /usr/lib/bridged 守护进程的一个副本。

例如，以下命令创建名为 pontevecchio 的网桥：

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

系统将创建名为 svc:/network/bridge:pontevecchio 的 SMF 服务实例和名为 /dev/net/pontevecchio0 的可观测节点。可观测节点可以与 snoop 命令和 wireshark 包分析器结合使用。您可以使用 dlstat 命令获得网桥的运行时统计信息。

出于安全目的，缺省情况下所有端口都运行标准 STP。不运行某种形式的桥接协议（如 STP）的网桥，可能会在网络中形成不终结的转发循环。因为以太网数据包上没有中继站计数或生存时间 (time-to-live, TTL)，所以这样的任何循环对网络都是致命的。

如果特定端口未连接到另一网桥（例如，由于该端口有到主机系统的直接点对点连接），您可以从管理角度禁用该端口的 STP。即使网桥上的所有端口都禁用了 STP，STP 守护进程仍然会在以下情况下运行：

- 处理添加的任何新端口
- 实现 BPDU 保护
- 根据需要在端口上启用或禁用转发

当端口已禁用 STP 时，bridged 守护进程会继续侦听 BPDU（BPDU 保护）。此守护进程使用 syslog 标记所有错误并在端口上禁用转发来指示严重错误的网络配置。当链路从不可用重新变为可用，或者将链路手动删除再重新添加时，链路将重新启用。

如果禁用网桥的 SMF 服务实例，随着 STP 守护进程的停止，这些端口上的网桥也将停止。如果重新启动该实例，STP 将从初始状态启动。

TRILL 守护进程

使用 dladm create-bridge -P trill 命令创建的各网桥以 svc:/network/bridge 和 svc:/network/routing/trill 的同名 SMF 实例表示。svc:/network/routing/trill 的每个实例运行实现 TRILL 协议的 /usr/lib/trilld 守护进程的一个副本。

例如，以下命令创建名为 bridgeofsighs 的网桥：

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

系统将创建两个分别名为 svc:/network/bridge:bridgeofsighs 和 svc:/network/routing/trill:bridgeofsighs 的 SMF 服务。另外，系统将创建名为 /dev/net/bridgeofsighs0 的可观测节点。

创建网桥

在 Oracle Solaris 中，使用 dladm 命令和 SMF 功能管理网桥。可以使用 SMF 命令通过实例 svc:/network/bridge 的故障管理资源标识符 (fault-managed resource identifier, FMRI) 来启用、禁用和监视网桥实例。可以使用 dladm 命令创建或销毁网桥，以及将链路指定给网桥或从网桥中删除链路。指定给网桥的链路必须是以太网类型，其中包括 802.3 和 802.11 介质。

要在链路之间创建网桥，必须创建至少一个网桥实例。每个网桥实例是独立的。网桥之间并不包括转发连接，一个链路至多是一个网桥的成员。

dladm create-bridge 命令创建一个网桥实例，并可以选择将一个或多个网络链路指定给该新网桥。因为缺省情况下系统中不存在网桥实例，所以，缺省情况下 Oracle Solaris 不在网络链路之间创建网桥。

要创建网桥，请使用以下命令：

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

-P *protect*
: 指定保护方法。可设置为以下值之一。
  - stp – STP 保护方法（缺省）
  - trill – TRILL 保护方法

-p *priority*
: 指定网桥的 IEEE STP 优先级值，以确定网络中的根网桥节点。缺省值为 32768。生效值介于 0（最高优先级）和 61440（最低优先级）之间，并以 4096 为增量。

-d *forward-delay*
: 指定网桥的 STP 转发延迟参数。如果创建的网桥为根节点，当启用端口时，网络中的所有网桥都会使用此计时器确定链路状态的顺序。缺省值为 15 秒。生效值介于 4 至 30 秒之间。

-l *link*
: 向网桥添加链路。如果无法添加任何指定的链路，该命令将失败，且不会创建网桥。

bridge-name 可以是任意字符串，但必须是合法的 SMF 服务实例名称。此名称是没有转义序列的 FMRI 组成部分，这意味着不能包含空格、ASCII 控制字符和以下字符：

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

名称 default 以及所有以 SUNW 字符串开头的名称都是保留名称。保留具有数字后缀的名称以创建用于调试的可观测设备。由于可观测设备的使用，进一步将合法的网桥实例名

称约束为合法的 dlpi 名称。该名称必须以字母字符或下划线字符开始和结束。其余的名称可以包含字母数字和下划线字符。

有关网桥创建选项的更多信息，请参见 dladm(1M) 手册页中的 dladm create-bridge 命令说明。

例 4-1 创建网桥

以下示例说明了如何通过连接 net0 和 net1 链路创建 brooklyn 网桥：

```
# 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
```

以下示例说明了如何通过连接 net0 和 net1 链路创建 westminister 网桥：

```
# 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
```

修改网桥的保护类型

STP 是防止网络循环可能使子网不可用的一种机制。除了将 STP 用于网桥之外，Oracle Solaris 还支持 TRILL 保护增强功能。缺省情况下使用 STP，但您可以通过为桥接命令指定 -P trill 选项来使用 TRILL。

要将保护类型从 STP 修改为 TRILL 或反之，请使用以下命令：

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

-P *protection-type* 选项指定要使用的保护类型：stp（缺省值）或 trill。

例 4-2 修改网桥的保护类型

以下示例说明了如何将 brooklyn 网桥的保护类型从缺省的 STP 改为 TRILL。

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

以下示例说明了如何将 brooklyn 网桥的保护类型从 TRILL 改为 STP。

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

为现有网桥添加链路

一个链路只能是一个网桥的成员。因此，如果要将链路从一个网桥实例移动到另一网桥实例，必须先从当前网桥删除该链路，然后再将其添加到另一网桥。

指定给同一网桥的链路必须具有相同的 MTU 值。尽管您可以更改现有链路的 MTU 值，但网桥实例会进入维护状态，直到您删除或更改了已指定的链路，以便使 MTU 值在网桥重新启动之前匹配。

注 - 指定给网桥的链路不能是 VLAN、VNIC 或隧道。仅可将被视为聚合一部分的链路或作为聚合的链路指定给网桥。

要向现有网桥添加新链路，请使用以下命令：

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

以下示例说明了如何将 `net2` 链路添加到现有网桥 `rialto`。

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

从网桥删除链路

只有先删除网桥的所有链路后，才能将网桥删除。要删除链路，请使用以下命令：

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

以下示例说明了如何从网桥 `charles` 删除 `net0`、`net1` 和 `net2` 链路：

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

设置网桥的链路属性

您可以设置网桥的以下链路属性：

`default_tag` 发送至某链路或从某链路接收的不带标记的包的缺省 VLAN ID。生效值为 `0` 到 `4094`。缺省值为 `1`。

`forward` 启用和禁用通过网桥的通信转发。除了 VNIC 链路外的所有链路都具有此属性。生效值是 `1` (true) 和 `0` (false)。缺省值为 `1`。

`stp`	启用和禁用 STP 和 RSTP。生效值是 `1` (true) 和 `0` (false)。缺省值是 `1`，这将启用 STP 和 RSTP。
`stp_cost`	表示与使用此链路对应的 STP 和 RSTP 成本值。生效值介于 `1` 至 `65535` 之间。缺省值为 `0`，用于指示成本由链路类型自动计算。
`stp_edge`	指定该端口是否连接到其他网桥。生效值是 `1` (true) 和 `0` (false)。缺省值为 `1`。
`stp_p2p`	指定连接模式类型。生效值为 `true`、`false` 和 `auto`。缺省值为 `auto`。
`stp_priority`	设置 STP 和 RSTP 端口优先级值。生效值介于 `0` 至 `255` 之间。缺省值为 `128`。

有关更多信息，请参见 `dladm(1M)` 手册页。

要修改网桥的链路属性，请使用以下命令：

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

例 4-3 设置网桥的链路属性

以下示例说明了如何禁用通信流量转发及设置连接模式类型。要设置网桥的属性，您必须设置连接该网桥的链路的属性。

```
# 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
```

以下示例说明了如何重置网桥的多个属性。

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

显示网桥配置信息

您可以使用 `dladm show-bridge` 命令显示网桥配置信息。

显示配置的网桥的信息

您可以使用 `dladm show-bridge` 和 `dlstat show-bridge` 命令显示所配置的网桥的各种信息。

使用以下命令选项：

- 查看网桥列表：

```
# 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
```

- 显示网桥的链路相关状态：

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

- 显示网桥的与链路相关的统计信息：

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

- 显示网桥的内核转发项：

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

- 显示网桥的 TRILL 信息：

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

- 显示每个网桥的统计信息和与每个网桥连接的链路的统计信息：

```
# 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        --
```

- 显示每个网桥的统计信息和与每个网桥连接的链路的全部统计信息：

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

有关 `dladm show-bridge` 命令选项的更多信息，请参见 dladm(1M) 手册页，有关 `dlstat show-bridge` 命令选项的信息，请参见 dlstat(1M) 手册页。

例 4-4 显示网桥信息

以下示例显示如何使用带有多个选项的 `dladm show-bridge` 命令。

- 以下命令显示单个网桥实例 `tower` 的与链路相关的状态信息。要查看已配置的属性，请使用 `dladm show-linkprop` 命令。

```
# 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
```

- 以下命令显示指定网桥 avignon 的内核转发条目：

```
# 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
```

- 以下命令显示指定网桥 key 的 TRILL 信息：

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

显示有关网桥链路的配置信息

使用带有 -o all 选项的 dladm show-link 命令在输出中显示 BRIDGE 字段。如果链路是网桥成员，此字段标识该链路所属的网桥的名称。对于不属于网桥的链路，如果使用 -p 选项，此字段为空。否则，该字段显示 --。

网桥可观测节点也作为一个单独的链路显示在 dladm show-link 输出中。对于此节点，现有 OVER 字段会列出那些是网桥成员的链路。

使用以下命令查看作为网桥成员的任何链路的配置信息。

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

-p 选项生成可解析的格式的输出。

从系统中删除网桥

在删除网桥之前，必须首先删除连接到网桥的任何链路。

▼ 如何从系统中删除网桥

1. 成为管理员。

有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 删除连接到网桥的任何链路。

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

3. 从系统中删除网桥。

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

例 4-5 从系统中删除网桥

以下示例说明了如何先从 coronado 网桥中删除 net0、net1 和 net2 链路，再从系统中删除该网桥本身。

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

管理桥接网络上的 VLAN

缺省情况下，系统中配置的 VLAN 在网桥实例的所有端口之间转发包。如果底层链路是网桥的一部分，则调用 dladm create-vlan 或 dladm create-vnic -v 命令时该命令还将启用此网桥链路上指定 VLAN 的包转发。有关 VLAN 的更多信息，请参见第 3 章 使用虚拟局域网配置虚拟网络。

要在一个链路上配置 VLAN 并禁用通过网桥上其他链路发送或接收的包转发，就必须通过使用 dladm set-linkprop 命令设置 VLAN 的 forward 属性来禁用转发。有关更多信息，请参见“设置网桥的链路属性” [68]。

▼ 如何在属于网桥的一部分的数据链路上配置 VLAN

开始之前 此过程假定该网桥已经存在。有关如何创建网桥的信息，请参见“创建网桥” [66]。

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 列出该网桥的与链路相关的信息，以确定属于网桥的一部分的链路。

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

3. 在属于网桥的一部分的链路上创建 VLAN。

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

   link 指定正在其上创建 VLAN 接口的链路。

   注 - 在此过程中，链路应该属于您已创建的网桥的一部分。

   vid 表示 VLAN ID 号。

   VLAN-link 指定 VLAN 的名称。

4. 对希望创建的每个 VLAN 重复此命令。对于您创建的每个 VLAN，创建该 VLAN 上的 IP 接口。

   # **ipadm create-ip** *interface*

   其中 *interface* 是 VLAN 名称。

5. 为 VLAN 上的各个 IP 接口配置有效的 IP 地址。

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

VLAN 与 STP 及 TRILL 协议

在符合标准的 STP 中，VLAN 将被忽略。桥接协议通过使用无标记 BPDU 消息只计算一个无环拓扑，并使用此树拓扑启用和禁用链路。必须配置网络中预分配的所有重复链路，以使配置的 VLAN 在 STP 自动禁用这些重复链路时不会断开连接。您必须在桥接主干中的所有位置运行所有 VLAN 或仔细检查所有冗余链路。

TRILL 协议不遵循复杂的 STP 规则。相反，TRILL 自动封装有完整 VLAN 标记的包并通过网络传递它们。

调试网桥

为每个网桥实例指定一个可观测节点，此节点显示在 `/dev/net/` 目录下，用网桥名称加后缀 `0` 的形式命名，例如 `/dev/net/bridgeofsighs0`。

可观测节点可以与 `snoop` 命令和 `wireshark` 包分析器结合使用。此节点行为类似于标准的以太网接口，不同的是，此节点传递包时会无提示丢弃。除非您使用 `passive` 选项（该选项只允许您接收包而不发送包），否则不能在可观测节点的顶部激活 IP，而且不能执行绑定请求 (`DL_BIND_REQ`)。

可观测节点会为网桥处理的每个包制作一个未经修改的副本。该副本可供用户监视和调试。此行为与在传统网桥上监视端口类似，它遵循普通的数据链路提供者接口 (datalink provider interface, DLPI) 混杂模式规则。您还可以使用 `snoop` 命令或 `pfmod` 命令中的功能以及 `wireshark` 包分析器根据 VLAN ID 过滤包。

发送的包（发送到可观测节点的包）表示网桥接收的数据。

注 - 如果桥接过程添加、删除或修改 VLAN 标记，`snoop` 命令以及 `wireshark` 包分析器显示的数据描述该进程发生之前的状态。如果不同的链路中使用不同的 `default_tag` 值（这种情况极为少见），可能会造成混乱。

要查看特定链路（桥接过程完成后）传送和接收的包，请在单个链路上（而不是在网桥的可观测节点上）运行 `snoop` 命令。

您也可以使用 `dlstat` 命令获取有关网络包如何使用链路上的网络资源的统计信息。有关信息，请参见《在 Oracle Solaris 11.2 中管理网络虚拟化和网络资源 》中的第 8 章 “监视网络通信流量和资源使用情况”。

♦ ♦ ♦ 第 5 章

使用链路层发现协议交换网络连接信息

本章介绍了如何使各个系统能够使用链路层发现协议 (Link Layer Discovery Protocol, LLDP) 在整个本地网络内交换系统和网络连接信息。

本章包含以下主题：

LLDP 概述

局域网 (local area network, LAN) 中的系统使用 LLDP 相互交换配置和管理信息。使用此协议，系统可以向网络上的其他系统通告连接和管理信息。此信息可以包括系统能力、管理地址以及其他与网络操作相关的信息。此协议还使系统能够接收有关同一本地网络上其他系统的类似信息。

在任一 LAN 上，都不会孤立地配置诸如系统和交换机之类的个体组件。要有效地承载网络通信，网络中的系统配置必须相互协调。

在手动配置每个系统、交换机以及其他组件时，确保这些组件之间的兼容性是一个难题。手动配置系统存在风险，很容易导致配置错误，尤其是当不同的管理员在不同的系统上独立工作时。更好的选择是使用 LLDP，这使系统可以将各自的配置信息传送到对等方系统，有助于检测任何配置错误。

Oracle Solaris 支持使用 LLDP，这可以改进网络中各系统之间的系统和网络连接信息交换，从而减少网络资源错误配置的风险。

在本发行版中，网络诊断服务使用 LLDP 自动检测可能导致网络连接有限或/和降级的问题。启用 LLDP 服务可以提高在您的 Oracle Solaris 系统上执行网络诊断的能力。有关网络诊断的更多信息，请参见《在 Oracle Solaris 11.2 中排除网络管理问题 》中的第 4 章 “使用 network-monitor 传输模块实用程序执行网络诊断”。

在 Oracle Solaris 中，LLDP 还用于交换数据中心桥接交换 (data center bridging exchange, DCBX) 协议类型-长度-值 (Type-Length-Value, TLV) 单元。DCBX 提供了有关基于优先级的流控制 (priority-based flow control, PFC) 和增强传输选择 (enhanced transmission selection, ETS) 之类的 DCB 功能的配置信息。有关 DCB 的更多信息，请参见第 6 章 使用数据中心桥接管理聚合网络。

使用 LLDP，系统管理员可以轻松检测有故障的系统配置，尤其是在虚拟局域网 (virtual local area network, VLAN)、链路聚合等复杂网络中。可以轻松得到有关网络拓扑的信息，而无需跟踪构成网络的服务器、交换机以及其他设备之间的物理连接。

LLDP 实现的组件

LLDP 是使用以下组件实现的：

- **LLDP 软件包** – 安装此软件包以启用 LLDP。此软件包中包含 LLDP 守护进程、命令行实用程序、服务清单和脚本以及 LLDP 运行所需的其他组件。
- **LLDP 服务** – 您可以使用 `svcadm` 命令启用 LLDP 服务。此服务使用服务管理工具 (service management facility, SMF) 服务实例的故障管理资源标识符 (fault management resource identifier, FMRI) `svc:/network/lldp:default` 管理 LLDP 守护进程 `lldpd`。此 LLDP 服务负责启动、停止、重新启动或刷新 `lldpd` 守护进程。安装 LLDP 软件包后，此服务将自动启用。
- **`lldpadm` 命令** – 您可以使用此命令管理各个链路上的 LLDP，配置 LLDP 的运行模式，指定传送的 TLV 单元以及配置 DCBX TLV 单元。有关 TLV 单元的信息，请参见“LLDP 代理通告的信息” [77]。

  您必须使用此命令设置每代理 LLDP 属性和全局 LLDP 属性，并获取特定代理或其对等方的 LLDP 信息。

  下面几节将对 `lldpadm` 子命令进行介绍。有关 `lldpadm` 命令的更多信息，请参见 `lldpadm(1M)` 手册页。
- **LLDP 守护进程** – LLDP 服务管理系统上的 LLDP 代理。它们还与 `snmpd`（简单网络管理协议 (Simple Network Management Protocol, SNMP) 的守护进程）交互，以通过 SNMP 检索系统上接收的 LLDP 信息。
- **LLDP 代理** – LLDP 代理是与启用了 LLDP 的物理数据链路相关联的 LLDP 实例。LLDP 代理将有关数据链路的信息传递给对等代理，并接收对等代理的信息。您可以配置 LLDP 代理，以通告关联物理数据链路的特定信息。您只能在物理数据链路上启用 LLDP。

LLDP 代理的信息源

LLDP 代理传送并接收 LLDP 数据单元 (LLDP data unit, LLDPDU)。代理在以下类型的数据存储中管理和存储这些 LLDPDU 中所包含的信息：

- 本地管理信息库 (management information base, MIB) – 此数据存储包含与系统中启用 LLDP 代理的特定链路相关的网络信息。本地 MIB 既包含公用信息，也包含独特信息。例如，机箱 ID 是在系统上的所有 LLDP 代理之间共享的公用信息。不过，系统的各个数据链路的端口 ID 是不同的。因此，每个代理管理它自己的本地 MIB。
- 远程 MIB – 此数据存储中的信息是从对等主机的 LLDP 代理接收的。

LLDP 代理模式

LLDP 代理在以下模式下运行：

- 仅传送 (`txonly`) – LLDP 代理不处理传入 LLDPDU。因此，远程 MIB 为空。
- 仅接收 (`rxonly`) – 代理仅处理传入 LLDPDU 并将信息存储在远程 MIB 中。不过，不从本地 MIB 传送信息。
- 传送和接收 (`both`) – 代理传送本地信息并处理传入 LLDPDU，因此同时维护本地和远程 MIB。
- 禁用 (`disable`) – 代理不存在。

有关设置代理模式的信息，请参见如何为指定端口启用 LLDP [82]。

LLDP 代理通告的信息

LLDP 代理在 LLDP 包或 LLDPDU 中传送系统和连接信息。这些包包含以 TLV 格式单独格式化的信息单元。信息单元也称为 *TLV* 单元。

强制性 TLV 单元

某些 TLV 单元是强制性的，在缺省情况下，在启用 LLDP 时它们就包含在 LLDP 包中。无法使用 `lldpadm` 命令排除这些单元中的任何单元。

以下 TLV 单元是强制性的：

- 机箱 ID – `hostid` 命令生成的信息
- 端口 ID – 物理 NIC 的 MAC 地址
- TTL（生存时间）
- 协议数据单元 (protocol data unit, PDU) 的末尾

根据链路的数量，可以在单个系统中启用多个 LLDP 代理。机箱 ID 和端口 ID 的组合唯一地标识代理，将其与系统上的其他代理区分开来。

例 5-1 显示机箱 ID 和端口 ID

以下示例显示了 LLDP 代理的机箱 ID 和端口 ID。

```
# hostid
004e434e


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

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

Oracle Solaris LLDP 代理使用 `hostid` 作为机箱 ID，使用端口的 MAC 地址作为端口 ID。

可选 TLV 单元

可以将可选的 TLV 单元添加到 LLDP 数据包。供应商可通过这些可选 TLV 单元插入要通告的特定于供应商的 TLV 单元。LLDP 可以通过使用组织唯一标识符 (organization unique identifier, OUI) 定义其他 TLV 单元。根据 OUI 是否符合 IEEE 802.1 或 IEEE 802.3 标准，OUI 标识 TLV 单元的类别。可以配置 LLDP 代理属性以启用或禁用这些可选 TLV 单元的传输。

下表列出了各个 TLV 组、其对应的名称以及每个属性的 TLV 单元及其说明。您配置上述任一属性以指定在启用 LLDP 时要包含在包中的 TLV 单元。

表 5-1 LLDP 代理的可选 TLV 单元

TLV 组	TLV 名称	TLV 单元	说明
基本管理	`basic-tlv`	`sysname`、`portdesc`、`syscapab`、`sysdesc`、`mgmtaddr`	指定要通告的系统名称、端口说明、系统能力、系统说明和管理地址。
802.1 OUI	`dot1-tlv`	`vlanname`、`pvid`、`linkaggr`、`pfc`、`appln`、`evb`、 `etscfg`、`etsreco`	指定以下要通告的内容：VLAN 名称、端口 VLAN ID、链路聚合、针对基于优先级的流控制的 TLV 单元、应用程序、增强传输选择以及边缘虚拟桥接。
802.3 OUI	`dot3-tlv`	`max-framesize`	指定要通告的最大帧大小。
特定于 Oracle 的 OUI（定义为 `0x0003BA`）	`virt-tlv`	`vnic`	指定要通告的 VNIC（如果配置了虚拟网络）。

TLV 单元属性

每个 TLV 单元都有您可以使用特定值进一步配置的属性。如果将 TLV 单元启用为 LLDP 代理的属性，则仅使用指定的值在网络中通告该 TLV 单元。例如，请考虑对系统能力进行通告的 TLV 单元 `syscapab` 这一示例。这些功能可能包括对路由器、网桥、中继器、电话和其他设备的支持。但是，您可以将 `syscapab` 设置为仅通告在您的特定系统中实际支持的功能，例如路由器和网桥。

配置 TLV 单元的过程取决于要配置全局 *TLV* 单元还是每代理 *TLV* 单元。有关如何配置 TLV 单元的信息，请参见“为代理的 LLDP 包指定 TLV 单元和值” [84]。

全局 TLV 单元应用于系统上的所有 LLDP 代理。下表列出了全局 TLV 单元及其对应的可能配置。

表 5-2 全局 TLV 单元及其属性

TLV 单元	属性名称	可能的属性值	值说明
`syscapab`	`supported`	`other`、`repeater`、`bridge`、`wlan-ap`、`router`、`telephone`、`docsis-cd`、`station`、`cvlan`、`sylvan`、`tpmr`	代表系统主要支持的功能。缺省值是 `router`、`station` 和 `bridge`。
	`enabled`	为 `supported` 列出的值的子集	代表系统的已启用的功能。
`mgmtaddr`	`ipaddr`	`ipv4` 或 `ipv6`	指定与本地 LLDP 代理相关联的 IP 地址的类型。地址用于到达更高的层实体，并有助于网络管理的发现。只允许指定一种类型。

以代理为单位管理特定于 LLDP 代理的 TLV 单元。使用每代理 TLV 单元，通过特定 LLDP 代理启用 TLV 单元传输时将使用您提供的值。

下表列出了一个 LLDP 代理的 TLV 值及其对应的可能配置。

表 5-3 每代理 TLV 单元及其属性

TLV 单元	属性名称	可能的属性值	值说明
`pfc`	`willing`	`on`、`off`	设置 LLDP 代理以接受或拒绝来自远程计算机的与基于优先级的流控制相关的配置信息。
`appln`	`apt`	值是从应用程序优先级表中定义的信息中获取的。	配置应用程序优先级表。此表包含应用程序 TLV 单元及其相应的优先级的列表。应用程序由 `id/selector` 对标识。表的内容使用以下格式： `id/selector/priority`

TLV 单元	属性名称	可能的属性值	值说明
			有关更多信息，请参见“应用程序优先级配置” [102]。
etscfg	willing	on、off	设置 LLDP 代理以接受或拒绝来自远程计算机的与增强传输选择相关的配置信息。

有关每代理 TLV 单元的信息，请参见第 6 章 使用数据中心桥接管理聚合网络。

在系统上启用 LLDP

您可以配置 LLDP 以与网络上的其他主机或对等方交换系统信息。

SMF 属性 auto-enable-agents 控制在系统上启用 LLDP 代理的方式。通过该属性，可以选择在所有物理链路上全局性地启用 LLDP，或者一次仅在一个物理链路上启用 LLDP。

SMF 属性 auto-enable-agents 可以具有以下三个可能值之一：

- yes 会在所有端口上以传送和接收模式 (both) 启用 LLDP（如果端口上不存在之前的 LLDP 配置）。如果端口上已存在一个配置，则将保留该端口的配置。例如，如果端口上先前配置了 rxonly 模式的 LLDP，则 LLDP 服务不会将代理切换为在传送和接收模式 (both) 下运行。该端口上的 LLDP 继续处于 rxonly 模式。这是 SMF 属性 auto-enable-agents 的缺省值。
- force 在所有端口上以传送和接收模式 (both) 启用 LLDP，并覆盖任何端口上的任何现有 LLDP 配置。例如，如果端口上先前的 LLDP 配置在 rxonly 模式下运行，则 LLDP 代理会切换为在传送和接收 (both) 模式（缺省的 LLDP 模式）下运行。
- no 将禁止在所有端口上自动启用 LLDP，但那些已存在现有 LLDP 配置的端口除外。在这些端口上，现有 LLDP 配置会保留。

注 - 每次定制 auto-enable-agents 属性后，必须重新启动 LLDP 服务才能使新值生效。

▼ 如何安装 LLDP 软件包

缺省情况下，安装完 LLDP 软件包后，LLDP 即已启用并可以使用。

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 安装软件包。

```
# pkg install lldp
```

3. 确定是否已启用 LLDP 服务。

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

如果已禁用 LLDP 服务，请使用以下命令启动该服务：

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

▼ 如何全局启用 LLDP

开始之前 要启用 LLDP，您必须首先安装 LLDP 软件包。有关更多信息，请参见如何安装 LLDP 软件包 [80]。

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 如果 SMF **auto-enable-agents** 属性设置为 **no**，更改为 **yes**。

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

   缺省情况下，该属性设置为 yes。

3. 重新启动 LLDP 服务。

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

4. （可选） 定制全局 TLV 单元。

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

   其中 *property* 指全局 TLV 单元的属性。

接下来的步骤 有关全局 TLV 单元的说明，请参见“TLV 单元属性” [79]。

要显示全局 TLV 的列表，请键入 `lldpadm show-tlvprop` 或参考 表 5-2 “全局 TLV 单元及其属性”。

有关如何定义 TLV 值的说明，请参见如何定义 TLV 单元 [86]。

有关 `lldpadm` 命令的信息，请参见 `lldpadm(1M)` 手册页。

▼ 如何为指定端口启用 LLDP

开始之前 要启用 LLDP，您必须首先安装 LLDP 软件包。有关更多信息，请参见如何安装 LLDP 软件包 [80]。

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 如果 SMF **auto-enable-agents** 属性设置为 **yes**，请将其更改为 **no**。

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

   缺省情况下，该属性设置为 `yes`。

3. 如果您在步骤 2 中更改了 SMF 属性 **auto-enable-agents**，请重新启动 LLDP 服务。

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

4. 在选定的端口或链路上启用 LLDP 代理。

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

   其中，*agent* 是 LLDP 代理，通过启用该代理的物理链路进行标识。例如，如果您在 `net0` 上启用了 LLDP，则代理为 `net0`。

   可将属性 `mode` 设置为以下四个可能值（代表 LLDP 代理的操作模式）之一：`txonly`、`rxonly`、`both` 和 `disable`。有关这些值的说明，请参见“LLDP 代理模式” [77]。

5. 指定 LLDP 代理可以通告的 TLV 单元。

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

   有关 LLDP 代理的属性的说明，请参见“LLDP 代理通告的信息” [77]。

   要显示 LLDP 代理的其他属性的列表，请键入 `lldpadm show-agentprop` 或参考 表 5-1 “LLDP 代理的可选 TLV 单元”。

   有关如何为代理的 LLDP 包指定 TLV 单元的说明，请参见如何为代理的 LLDP 包指定 TLV 单元 [84]。

6. （可选） 定制每代理 TLV 单元。

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

   其中 *property* 指每代理 TLV 单元的属性。

   有关每代理 TLV 单元的说明，请参见“TLV 单元属性” [79]。

要显示每代理 TLV 的列表，请键入 `lldpadm show-agenttlvprop` 或参考表 5-3 “每代理 TLV 单元及其属性”。

有关如何定义 TLV 值的说明，请参见如何定义 TLV 单元 [86]。

有关 `lldpadm` 命令的信息，请参见 `lldpadm(1M)` 手册页。

例 5-2 定制 `auto-enable-agents` SMF 属性

以下示例显示了更改 SMF 属性 `auto-enable-agents` 的值后启用 LLDP 的不同方式。例如，假定一个系统上有四个端口，在其中两个端口上配置了 LLDP，如下所述：

- `net0`: `both` mode
- `net1`: `rxonly` mode
- `net2` 和 `net3`：无

如果 SMF 属性 `auto-enable-agents` 具有缺省值 `yes`，将在 `net2` 和 `net3` 上自动启用 LLDP。您可以如下所示显示 LLDP 配置：

```
# lldpadm show-agentprop -p mode
AGENT   PROPERTY   PERM   VALUE   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
```

如果将 SMF 属性切换为 `no`，则当重新启动该服务时，配置会发生变化。

```
# 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    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
```

在此输出样例中，`net2` 和 `net3` 的 LLDP 模式（先前已经自动启用）现在标记为禁用。但是，先前已配置了 LLDP 代理的 `net0` 和 `net1` 没有变化。

例 5-3 在多个数据链路上启用 LLDP

此示例说明了如何选择性地启用 LLDP。一个系统具有两个数据链路（`net0` 和 `net1`）。在 `net0` 上设置传送和接收 LLDP 包的代理，在 `net1` 上设置只传送 LLDP 包的代理，键入以下命令：

```
# 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    DEFAULT   POSSIBLE
net0    mode       rw     both     disable   txonly,rxonly,both,disable
net1    mode       rw     txonly   disable   txonly,rxonly,both,disable
```

为代理的 LLDP 包指定 TLV 单元和值

您可以指定 TLV 单元（例如 `dot1-tlv` 和 `basic-tlv`）作为 LLDP 代理的属性值。您可以进一步配置这些属性值。要指定 TLV 单元，请使用 `lldpadm set-agentprop` 命令。有关更多信息，请参见 `lldpadm`(1M) 手册页。如果将 TLV 单元指定为 LLDP 代理的属性，则仅通告网络中具有指定值的 TLV 单元。有关 TLV 单元及其属性的信息，请参见“TLV 单元属性” [79]。

▼ 如何为代理的 LLDP 包指定 TLV 单元

此过程说明如何在代理传送的 LLDP 包中指定要通告的 TLV 单元。

1. **成为管理员。**

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. **（可选） 通过显示 TLV 单元来识别可以包含您要添加的 TLV 单元的 LLDP 代理属性。**

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

   此命令可帮助您查看已为每个属性设置的 TLV 单元。如果未指定属性，此命令将显示所有 LLDP 代理属性及其 TLV 值。有关代理属性的列表，请参见表 5-1 “LLDP 代理的可选 TLV 单元”。

3. **在该属性中添加或删除 TLV 单元。**

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

   对于接受多个值的属性，您可以在值列表中使用限定符添加 (+) 或删除 (-) 值。

   如果您不使用添加 (+) 或删除 (-) 限定符，则您设置的值将取代以前为该属性定义的所有值。

4. **（可选） 显示属性的新值。**

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

例 5-4 将可选 TLV 单元添加到 LLDP 包

在以下示例中，LLDP 代理配置了 net0，以在其 LLDP 包中通告 VLAN 信息。进一步配置 LLDP 包，以将系统能力、链路聚合和虚拟 NIC 信息包括为 LLDP 可以通告的项。然后，从包中删除 VLAN 说明。

1. 显示现有的代理属性。

```
# lldpadm show-agentprop net0
AGENT   PROPERTY   PERM   VALUE          DEFAULT   POSSIBLE
net0    mode       rw     both           disable   txonly,rxonly,both,disable
net0    basic-tlv  rw     sysname,        none     none,portdesc,
                          sysdesc                  sysname,sysdesc,
                                                   syscapab,mgmtaddr,
                                                   all
net0    dot1-tlv   rw     vlanname,       none     none,vlanname,pvid,
                          pvid,pfc                 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
```

   输出显示了 LLDP 代理的每个属性的现有值、缺省值和可能值。

2. 将系统能力、链路聚合和网络虚拟化信息设置为通过网络通告的项。

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

3. 从包中删除 VLAN 说明。

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

4. 显示代理属性。

```
# lldpadm show-agentprop -p net0
AGENT   PROPERTY   PERM   VALUE          DEFAULT   POSSIBLE
net0    mode       rw     both           disable   txonly,rxonly,both,
                                                   disable
net0    basic-tlv  rw     sysname,        none     none,portdesc,
                          sysdesc,                 sysname,sysdesc,
                          syscapab                 syscapab,mgmtaddr,
                                                   all
net0    dot1-tlv   rw     pvid,pfc        none     none,vlanname,pvid,
                          linkaggr                 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
```

▼ 如何定义 TLV 单元

此过程说明如何为特定 TLV 单元提供值。

提示 - 通过针对全局 TLV 单元使用 `lldpadm reset-tlvprop` 命令，针对每代理 TLV 单元使用 `lldpadm reset-agenttlvprop` 命令，您可以将 TLV 属性重置为其缺省值。

---

1. 成为管理员。

   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 配置全局或每代理 TLV 单元。

   - 要配置全局 TLV 单元，请设置适当的 TLV 属性以包含您要通告的值。

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

     其中，*TLV-name* 是全局 TLV 单元的名称，而 *TLV-property* 是该 TLV 单元的一个属性。可为该属性分配多个值。有关全局 TLV 单元及其属性的列表，请参见表 5-2 “全局 TLV 单元及其属性”。

   - 要配置每代理 TLV 单元，请配置 LLDP 代理的相应 TLV 属性以包含希望代理通告的值。

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

     其中，*TLV-name* 是代理 TLV 单元的名称，而 *TLV-property* 是该 TLV 单元的一个属性。可为该属性分配多个值。有关每代理 TLV 单元及其属性的列表，请参见表 5-3 “每代理 TLV 单元及其属性”。

     对于接受多个值的属性，您可以在值列表中使用限定符添加 (+) 或删除 (-) 值。

3. （可选） 显示您配置的 TLV 属性的值。

   - 显示全局 TLV 属性值：

     ```
     # lldpadm show-tlvprop
     ```

   - 显示代理的 TLV 属性值：

     ```
     # lldpadm show-agenttlvprop
     ```

例 5-5 为 syscapab 及 mgmtaddr TLV 单元定义 TLV 值

在以下示例中，配置了要在 LLDP 包中通告的关于系统能力的特定信息，且配置了管理 IP 地址。

1. 配置 syscapab TLV 单元的 supported 和 enabled 属性。

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

2. 指定 mgmtaddr TLV 单元的管理 IP 地址。

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

3. 显示代理属性的 TLV 值。

```
# lldpadm show-tlvprop
TLVNAME    PROPERTY    PERM  VALUE        DEFAULT         POSSIBLE
syscapab   supported   rw    bridge,      bridge,router,  other,router,
                             router,      station         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            --
```

输出包括 TLV 单元的缺省值，以及可以为该属性设置的可能值。

有关配置每代理 TLV 属性的信息，请参见第 6 章 使用数据中心桥接管理聚合网络。

禁用 LLDP

本节介绍了如何选择性地在各个端口上禁用 LLDP。

▼ 如何禁用 LLDP

要在所有系统接口中禁用 LLDP，请执行以下步骤。

1. **成为管理员。**
   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. 将 SMF LLDP 属性更改为 **no**，这将禁止在所有端口上自动启用 LLDP，但那些已存在现有 LLDP 配置的端口除外。

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

3. 重新启动 LLDP 服务。

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

4. 在保留了先前 LLDP 配置的各个端口上禁用 LLDP。

   - 通过更改代理的模式禁用 LLDP：

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

     其中，*agent* 是 LLDP 代理，通过启用该代理的物理链路进行标识。例如，如果您在 `net0` 上启用了 LLDP，则代理为 `net0`。

   - 通过从端口中删除 LLDP 配置来禁用 LLDP：

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

     在此命令中，不设置 `mode` 属性的值。

注意 - 如果设置为 `no` 的 `auto-enable-agents` 切换回 `yes`，则 LLDP 的行为方式与只是在该端口上禁用了代理模式的情况不同。

监视 LLDP 代理

`lldpadm show-agent` 命令显示由 LLDP 代理通告的完整信息。相对于给定系统，通告可以是有关传送到网络其余部分的本地系统的信息，也可以是系统从同一网络上的其他系统接收的信息。

显示通告的信息

信息可以是本地信息，也可以是远程信息。本地信息来自本地 LLDP 代理。远程信息来自网络上的其他 LLDP 代理，由本地 LLDP 代理接收。

使用 `lldpadm show-agent` 命令可以显示通告的信息。

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

- `-l` 显示本地 LLDP 代理通告的本地信息。

- -r 显示 LLDP 代理接收的远程信息。
- -v 显示详细的本地或远程信息。

例 5-6 显示通告的 LLDP 代理信息

以下示例说明了如何显示 LLDP 代理以本地或远程方式通告的信息。缺省情况下，信息以短格式显示。通过使用 -v 选项，您可以获取详细信息。

显示 LLDP 代理通告的本地信息：

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

显示 LLDP 代理通告的远程信息：

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

要以详细模式显示本地信息，请使用 -v 选项：

```
# 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)
```

要以详细模式显示远程信息，请使用 -v 选项：

```
# 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)
```

显示 LLDP 统计信息

您可以显示 LLDP 统计信息，以获取有关由本地系统或远程系统通告的 LLDP 包的信息。统计信息引用涉及 LLDP 包传送和接收的重大事件。

- 显示有关 LLDP 包传送和接收的所有统计信息：

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

- 要显示所选的统计信息，请使用 -o 选项：

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

  其中 *field* 指 `show-agent -s` 命令的输出中的任何字段名称。

例 5-7 显示 LLDP 包统计信息

本示例说明如何显示有关 LLDP 包通告的信息。

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

此输出提供了以下信息：

- `AGENT` 指定 LLDP 代理的名称，它与在其上启用 LLDP 代理的数据链路相同。
- `IFRAMES`、`IERR` 和 `IDISCARD` 显示有关正在接收的包、有错误的传入包和丢弃的传入包的信息。
- `OFRAMES` 和 `OLENERR` 指传出包以及有长度错误的包。
- `TLVDISCARD` 和 `TLVUNRECOG` 显示有关被丢弃的和未识别的 TLV 单元的信息。

- AGEOUT 指已超时的包。

该示例表明在系统收到的 9 个帧中，有 5 个 TLV 单元无法识别，原因可能是不符合标准。该示例还显示本地系统向网络传送了 14 个帧。

例 5-8 显示所选 LLDP 包统计信息

此示例说明了如何显示所选统计信息。

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

♦ ♦ ♦ 第 6 章

使用数据中心桥接管理聚合网络

过去，人们使用不同的网络来根据应用程序要求进行通信管理，并根据可用带宽进行网络通信的负载分配。例如，局域网 (local area network, LAN) 使用的是以太网，存储区域网 (storage area network, SAN) 使用的是光纤通道。但是，数据中心桥接可增强以太网，使其更加适用于运行不同类型的通信（即聚合通信），还可以支持无损性等功能。DCB 可将 SAN 和 LAN 整合在一起并进而降低数据中心的运营和管理成本，从而实现高效的网络基础结构。

本章包含以下主题：

数据中心桥接概述

当共享同一网络链路（例如，在联网协议和存储协议之间共享一条数据链路）时，数据中心桥接用来管理多种通信类型的带宽、相对优先级和流控制。光纤通道可专用于承载此类型的通信。但是，如果使用专用链路来仅提供光纤通道通信，成本可能会很高。因此，更多情况下使用以太网光纤通道 (fiber channel over Ethernet, FCoE)。DCB 可解决光纤通道在穿过以太网时对丢包的敏感性问题。

DCB 基于优先级区分通信，优先级也称为服务类 (class of service, CoS) 优先级。主机和下一个中继站使用 DCB 交换 (DCB exchange, DCBX) 协议基于优先级来协商网络配

置，例如，无通信丢失以及最低带宽份额。此过程允许根据包优先级对来自主机上和网络中不同应用程序的包进行处理，并使用 DCBX 协商相应的配置。

DCB 网络中的每个包均有一个 VLAN 头，其中包含一个 DCB 3 位优先级值，即 DCB 优先级。此 IEEE 802.1p 优先级值可将网络中的每个以太网包与其他包区分开来。您可以根据包的优先级值对 DCB 进行配置以向包分配特定带宽。例如，优先级为 1 的所有包必须启用 PFC，优先级为 2 的所有包必须禁用 PFC 并具有 10% 的带宽份额。

您可以配置 DCB 功能，例如，基于优先级的流控制 (priority-based flow control, PFC) 和基于优先级的增强传输选择 (enhanced transmission selection, ETS)。有关 PFC 和 ETS 的更多信息，请参见“基于优先级的流控制” [95]和“增强传输选择” [95]。

通过 DCB `cos` 数据链路属性，可以指定数据链路的 CoS 或优先级。在主数据链路上设置的 `cos` 值不应用于在此物理链路上创建的 VNIC。有关基于 `cos` 属性定制 PFC 的信息，请参见“为 DCB 定制基于优先级的流控制” [97]。有关基于 `cos` 属性定制 ETS 的信息，请参见“为 DCB 定制增强传输选择” [102]。

在 Oracle Solaris 中，使用 LLDP 来交换 DCBX 类型-长度-值 (type-length-value, TLV) 单元。有关 LLDP 的更多信息，请参见第 5 章 使用链路层发现协议交换网络连接信息。如果底层网络接口卡 (network interface card, NIC) 支持基于优先级的流控制和增强传输选择等 DCB 功能，则可以与网络上的对等方主机共享这些功能的配置信息，如下所述：

- PFC 通过实施一种机制来防止包丢失，该机制可暂停具有所定义的服务类 (class of service, CoS) 的包的通信流。有关 CoS 的更多信息，请参见 `dladm(1M)` 手册页中 `cos` 链路属性的描述。
- ETS 支持基于所定义的 CoS 在包之间共享带宽。请参见“增强传输选择” [95]。

使用 DCB 时的注意事项

请注意有关使用 DCB 的以下事项：

- DCB 仅在 Intel Niantic 物理 NIC 上受支持。

  要验证 NIC 是否支持 DCB，请发出以下命令：

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

  大于零 (`0`) 的属性值表示 NIC 支持 DCB。
- 在 DCB 模式下配置的 DCB 端口无法聚合（中继或 DLMP 模式）。
- 由于 DCB 仅支持 DCBX 的 IEEE 版本（而非 CEE 或 CIN 版本），因此外部网桥必须支持 IEEE 版本才能与 Oracle Solaris DCB 进行互操作。
- DCB 支持 ETS 配置和建议 TLV。
- DCB 仅在八通信类配置中受支持。
- DCB 不支持拥塞通知 (congestion notification, CN)。

基于优先级的流控制

基于优先级的流控制 (priority-based flow control, PFC) 扩展了标准 IEEE 802.3x PAUSE 帧以包含 IEEE 802.1p CoS 值。使用 PFC，当发送 PAUSE 帧时不会停止链路上的所有通信，而是仅暂停 PFC 帧中启用的 CoS 值所对应的通信。对于其通信需要暂停的已启用 `cos` 属性值，会发送一个 PFC 帧。发送主机会停止该 `cos` 属性值的通信，而其他禁用的 `cos` 属性值的通信不受影响。在经过 PFC 帧中指定的时间间隔之后，将恢复暂停的包的传输。

基于 CoS 值暂停可确保不会丢失该 `cos` 属性值的包。对于未定义任何 CoS 值或 CoS 值未启用 PFC 的包，将不会发送 PAUSE 帧。因此，通信将继续进行，但在通信拥塞时可能会丢弃包。对包丢失的处理取决于协议栈（例如 TCP）。

主机上存在两种类型的 DCB 信息：本地 DCB 信息和远程 DCB 信息。要使 PFC 功能生效，主机上用于 PFC 的本地和远程 DCB 信息必须对称。本地主机必须能够匹配它从对等方接收到的 DCB 信息。如果在系统上启用 DCB，DCB 可将 DCB 信息与对等方同步。

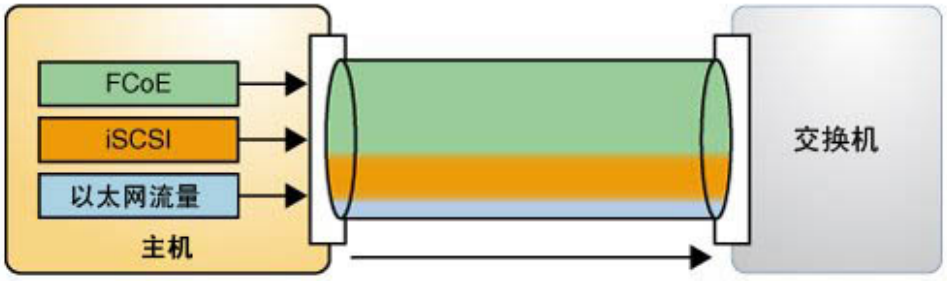
大多数情况下，PFC 的缺省配置足以满足要求。启用 LLDP 时会自动设置此配置。但是，可以在配置 PFC 时调整不同的选项。有关更多信息，请参见“为 DCB 定制基于优先级的流控制” [97]和“显示 PFC 配置信息” [99]。

增强传输选择

ETS 是一项 DCB 功能，通过该功能可根据应用程序的 DCB 优先级为其分配 NIC 上的带宽。

下图显示了网络中 DCB 的 ETS 功能。

图 6-1 DCB 中的增强传输选择



图中的主机具有不同类型的通信，如 FCoE 和 iSCSI，这些通信共享链路带宽。在图中，为不同类型的通信分配了下表中所示的优先级和带宽。

通信	优先级	带宽
FCoE	3	60%
iSCSI	4	30%
以太网（非 iSCSI）通信	0	10%

具有相应 ETS 带宽分配的 DCBX ETS TLV 如下所示：

优先级	0	1	2	3	4	5	6	7
带宽分配百分比	10	0	0	60	30	0	0	0

要使用 ETS 功能，NIC 必须支持该功能并在 DCB 模式下运行。启用 LLDP 时，如果底层链路支持 DCB，则会自动设置 ETS 功能的缺省配置。但是，您可以修改缺省配置。有关更多信息，请参见“为 DCB 定制增强传输选择” [102]、“向对等方建议 ETS 配置” [104]和“显示 ETS 配置信息” [106]。

启用 DCBX

启用 LLDP 时，将自动启用对 DCBX 的支持。此过程提供了备用的手动步骤，以防某些自动过程失败。

▼ 如何手动启用数据中心桥接交换功能

开始之前 确保您已安装 LLDP。有关启用 LLDP 的更多信息，请参见“在系统上启用 LLDP” [80]。

1. 成为管理员。
   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。
2. 验证 LLDP 服务是否正在运行。

   ```
   # svcs lldp
   ```

   如果已禁用 LLDP 服务，请使用以下命令启动该服务：

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

3. 确保 LLDP 代理正在以 Rx 和 Tx 模式运行。

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

   如果未在这两种模式下启用 LLDP 代理，请键入以下命令：

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

   有关 LLDP 代理的其他可能配置，请参见“在系统上启用 LLDP” [80]。
4. 确认底层 NIC 支持 DCB。

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

   大于零 (0) 的属性值表示 NIC 支持 DCB。

为 DCB 定制基于优先级的流控制

仅当底层 NIC 处于 DCB 模式时，才会在缺省情况下启用 PFC 和 ETS。如果希望仅使用 PFC，则必须使用以下命令将 `etscfg` 从 LLDP 代理的 `dot1 -tlv` 属性中删除：

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

有关 `dot1 -tlv` 的可能值的列表，请参阅表 5-1 “LLDP 代理的可选 TLV 单元”。

设置与 PFC 相关的数据链路属性

DCB 的 PFC 功能提供了以下数据链路属性：

- pfcmap – 提供有关优先级定义和映射的信息：pfcmap 属性指表示优先级的 8 位掩码 (0–7)。最低位表示优先级 0，而最高位表示优先级 7。此掩码中每位均可表示是否已为对应的优先级启用 PFC。缺省情况下，pfcmap 设置为 1111111，表示已为所有优先级启用 PFC。
- pfcmap-rmt – 指定远程对等方上的有效 PFC 映射。此属性为只读。

在 DCB 网络中，当接收方无法跟上通信的传入速率时，将会向发送方发送 PFC 帧，请求发送方为启用了 PFC 的优先级暂停通信。对于通过链路传输的任何包，如果在接收主机上产生通信拥塞，则 DCB 将向发送主机发送 PFC 帧。要正常发送 PFC 帧，通信主机必须具有对称的 DCB 配置信息。系统可以自动调整其 PFC 配置，以匹配远程对等方的 PFC 配置。可以使用 dladm show-linkprop 命令（该命令显示 pfcmap 属性的 EFFECTIVE 值）确定本地主机上实施的 PFC 映射。有关更多信息，请参见“显示数据链路属性” [99]。

▼ 如何为 DCB 定制基于优先级的流控制

1. **成为管理员。**
   有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

2. **确保数据链路的 `flowctrl` 属性设置为 `pfc`。**

   ```
   # dladm show-linkprop -p flowctrl datalink
   ```

   如果属性未设置为 pfc 或 auto，请使用以下命令：

   ```
   # dladm set-linkprop -p flowctrl=pfc datalink
   ```

3. **将 `pfcmap` 属性设置为除缺省值 `11111111` 以外的其他值。**

   ```
   # dladm set-linkprop -p pfcmap=value datalink
   ```

   例如，要仅启用 CoS 优先级 6，请键入以下命令：

   ```
   # dladm set-linkprop -p pfcmap=01000000 net0
   ```

设置 PFC TLV 单元

DCB 使用 PFC TLV 单元在主机之间交换 PFC 信息。两台主机的 pfcmap 属性必须具有相同的值或者必须至少有一台主机愿意接受其对等方的配置。可以使用 dladm set-linkprop 命令设置 pfcmap 属性。

TLV 属性 willing 指示如果主机配置与对等方的配置不同，主机是否已准备好接受对等方的配置。缺省情况下，willing 的属性值设置为 on，表示主机将接受对等方的配置。

要验证主机能否将其 PFC 信息与远程对等方上的 PFC 信息进行同步，必须使用以下命令确定 `willing` 是否已设置为 `on`：

```
# lldpadm show-agenttlvprop -p willing -a agent pfc
```

如果 PFC TLV 属性 `willing` 设置为 `off`，请键入以下命令将属性 `willing` 设置为 `on` 并启用同步。

```
# lldpadm set-agenttlvprop -p willing=on -a agent pfc
```

其中，*agent* 是启用了代理的数据链路。

例 6-1 在主机和对等方之间启用同步

要启用 `net0` 数据链路的同步，请键入以下命令：

```
# lldpadm 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   --
```

在示例中，`pfcmap` 和 `pfcmap-rmt` 属性具有值 `00010000`。这表示本地主机已经与对等方同步。因此，PFC 同时在主机和对等方上启用，优先级为 `4`。

有关更多信息，请参见 `lldpadm(1M)` 手册页。

显示 PFC 配置信息

本节介绍了用于显示在配置 LLDP 和 DCB 后与 PFC 相关的信息的命令，并提供了示例来说明这些命令的用法。

显示数据链路属性

以下命令显示了优先级定义以及数据链路上生效的 PFC 映射：

```
# dladm show-linkprop -p pfcmap,pfcmap-rmt datalink
```

对于本地和远程对等方之间 PFC 信息匹配的数据链路，不管将 `pfcmap` 属性设置为何值，`pfcmap` 和 `pfcmap-rmt` 属性的 `EFFECTIVE` 列的值均相同。如果在本地主机上禁用了同步功能，则 `pfcmap` 属性的 `EFFECTIVE` 字段将反映本地主机的 `pfcmap` 属性的值。

例 6-2 显示与 PFC 相关的数据链路属性

此示例说明了如何显示与基于优先级的流控制相关的物理数据链路属性的状态。

```
# 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-     --         --         --         --
```

在示例中，`pfcmap` 属性的 `value` 字段具有值 `11111111`。该值指示本地主机上的 PFC 映射使用缺省值，其中所有八个优先级均已启用。`pfcmap` 和 `pfcmap-rmt` 属性的 `EFFECTIVE` 值为 `11111111` 和 `--`。`EFFECTIVE` 字段的值不匹配表示本地主机未将其 PFC 信息与远程对等方同步。

可以使用 `lldpadm show-agenttlvprop` 命令验证 `willing` 属性的值，并使用 `lldpadm show-agent -r` 命令检查来自对等方的 PFC TLV 信息。

显示本地主机同步 PFC 信息的功能

以下命令显示 PFC TLV 属性，该属性控制主机将其 PFC 映射与对等方进行同步的功能。

```
# lldpadm show-agenttlvprop -a agent pfc
```

其中，*agent* 通过启用 LLDP 的数据链路进行标识。

例 6-3 显示本地主机同步 PFC 信息的功能

此示例说明了如何显示主机用于适应对等方 PFC 配置的功能的当前状态。

```
# lldpadm show-agenttlvprop -a net0 pfc
AGENT   TLVNAME   PROPERTY   PERM   VALUE   DEFAULT   POSSIBLE
net0    pfc       willing    rw     off     on        on,off
```

有关更多信息，请参见“设置 PFC TLV 单元” [98]。

显示主机和对等方之间的 PFC 映射信息

如果主机和对等方之间的 PFC 信息未聚合，则 `PFC Pending` 值会返回 `True` 状态。解决不匹配问题之后，`PFC Pending` 的状态将恢复为 `False`。

以下命令提醒您本地主机和对等方之间的 PFC 映射信息不匹配。

```
# lldpadm show-agent -lv -o "PFC Pending" agent

# lldpadm show-agent -lv -o "PFC Pending" agent
```

例 6-4 验证主机和对等方之间的 PFC 信息的对称性

以下示例说明了如何在实际运行时验证主机和对等方之间的 PFC 信息是否同步，或者是否发生了不匹配。

```
# lldpadm show-agent -lv -o "PFC Pending" net0
PFC Pending: True
```

要显示代理所通告的所有信息，请使用 `lldpadm show-agent` 命令的 `-v`（详细）选项：

```
# lldpadm show-agent -v net0
```

显示优先级定义

以下命令显示物理链路上与 NIC 上启用的优先级有关的 PFC 信息。

```
# dladm show-phys -D pfc datalink
```

例 6-5 显示 CoS 优先级定义

此示例说明了如何显示特定物理链路上基于 `pfcmap` 属性值的当前优先级定义。例如，假定 `pfcmap` 配置为 `01000000`。要显示物理链路上相应的优先级映射，请执行如下命令：

```
# 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
```

对于物理链路 `net0`，为数据链路上配置的所有 VNIC 客户机启用了优先级。但是，本地主机将其 PFC 映射调整为对等方的 PFC 映射，如 `PFC_EFFECT` 字段的值所示，其中，已对 `COS 0` 和 `2-7` 禁用优先级。因此，不会为除 `vnic2` 之外的任何 VNIC 上的通信交换 PFC 帧，无论资源是否可用。此配置允许对流经 `vnic2` 之外的任何 VNIC 的通信丢弃

包。对于 vnic2 上的通信，在出现通信拥塞时将发送 PFC PAUSE 帧，以防止此客户机上发生包丢失。

应用程序优先级配置

DCBX 交换与应用程序关联的优先级信息。通过 DCBX 交换的应用程序 TLV 单元包含有关要用于主机上某个应用程序的优先级的信息。该优先级在应用程序优先级表中进行定义。表中的每一项都包含应用程序的名称以及指定给该应用程序的优先级。为应用程序设置优先级时，该优先级的 PFC 和 ETS 的所有 DCB 设置均适用于该应用程序。应用程序 TLV 使用该表数据与其他主机交换应用程序优先级信息。

缺省情况下，应用程序功能接受来自对等方的优先级映射。应用程序 TLV appln 的 willing 属性与 PFC 类似，可实现对等方之间的信息交换。

表中的条目使用以下格式：

protocol-ID/*selector*/*priority*

protocol-ID/*selector* 对用于标识应用程序。相应应用程序的优先级由包含 0 到 7 之间的值的优先级予以标识。

要与其他主机交换此有关某个应用程序优先级的信息，可如下所示设置应用程序 TLV：

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

例如，对于 FCoE 通信，协议 ID 为 0x8906，选择器 ID 为 1。假定为此应用程序指定优先级 4。根据表 5-3 “每代理 TLV 单元及其属性”（列出了用于设置应用程序 TLV 的参数），键入以下命令：

```
# lldpadm set-agenttlvprop -p apt=8906/1/4 -a net0 appln
# lldpadm show-agenttlvprop -a net0 appln
AGENT   TLVNAME   PROPERTY   PERM   VALUE      DEFAULT   POSSIBLE
net0    appln     apt        rw     8906/1/4   --        --
```

为 DCB 定制增强传输选择

如果启用了 LLDP，并且底层链路支持 DCB，则将自动设置缺省配置。在此缺省配置中，为 cos 值 0 分配了所有带宽。但是，可以使用 dladm set-linkprop 命令配置数据链路上的 cos 值，以将部分带宽分配给该数据链路。

缺省情况下会为 NIC 启用 ETS 配置和建议 TLV。有关 dot1-tlv 的可能值的列表，请参阅表 5-1 “LLDP 代理的可选 TLV 单元”。

如果要删除 pfc TLV，请键入以下命令：

```
# lldpadm set-agenttlvprop -p dot1-tlv-=pfc agent
```

设置与 ETS 相关的数据链路属性

引用 PFC 信息的数据链路属性适用于基于为包定义的优先级来防止包丢失。ETS 属性与基于优先级分配底层链路的带宽份额有关。

DCB 提供了以下与 ETS 相关的属性：

- cos – 指定数据链路的服务类或优先级。该属性的值的范围为从 0 到 7。缺省值为 "0"。cos 值是在通过此链路传输的包的 VLAN 标记中设置的。
- etsbw-lcl – 指示为数据链路的传送 (Tx) 端分配的 ETS 带宽。仅当底层物理 NIC 具有 DCB 功能并支持 ETS，且链路的 cos 属性未设置为 0 时，此属性才可配置。可通过指定底层物理链路总带宽的百分比设置此数据链路属性的值。同一物理 NIC 上所有数据链路的 etsbw-lcl 属性的值之和不得超过 100%。

  在 etsbw-lcl 中定义的带宽百分比不会仅保留给该数据链路。如果未使用分配的带宽，则该带宽可由该物理 NIC 上的其他数据链路使用。此外，仅在主机通信的传输端强制进行带宽分配。
- etsbw-rmt-advice – 指定发送给对等方的建议 ETS 带宽值。缺省情况下，建议向对等方发送本地配置的 etsbw-lcl 属性值。但是，可以通过显式配置 etsbw-rmt-advice 数据链路属性建议一个不同于 etsbw-lcl 属性的值。

  如果数据链路的带宽分配不对称（这意味着接收 (Rx) 和传送 (Tx) 带宽不同），则配置 etsbw-rmt-advice 属性非常有用。在显式设置 etsbw-rmt-advice 属性时，ETS 建议 DCBX TLV 的传输将自动启动。
- etsbw-lcl-advice – 指定数据链路的建议带宽份额，该带宽份额由对等方发送至本地主机。该属性是只读属性。
- etsbw-rmt – 指定在对等方上为数据链路配置的带宽份额。该属性是只读属性。

要对 VNIC 设置优先级并向其分配带宽，请使用以下命令：

- 为 VNIC 设置优先级：

  ```
  # dladm set-linkprop -p cos=value VNIC
  ```
- 向 VNIC 分配底层物理链路的带宽百分比：

  ```
  # dladm set-linkprop -p etsbw-lcl=value VNIC
  ```

  指定给 etsbw-lcl 属性的值表示底层链路的带宽总量的百分比。为客户机指定的所有已分配带宽值的总和不得超过 100%。
- 显式建议发送到对等方的带宽：

  ```
  # dladm set-linkprop -p etsbw-rmt-advice=value VNIC
  ```

可以使用 dladm show-linkprop 命令确定在本地主机的数据链路上实施的实际带宽份额，以及在对等方的数据链路上配置的带宽份额。etsbw-lcl 和 etsbw-rmt 属性输出的 EFFECTIVE 字段中的值显示了实施的实际带宽份额。有关更多信息，请参见“显示 ETS 配置信息” [106]。

对于具有特定优先级的包要使用的相应带宽，最好使通信主机之间的 ETS 信息对称或进行同步。确切地说，本地系统应该能够将其带宽份额调整为 etsbw-lcl-advice 的值。Oracle Solaris 系统可以自动调整其 ETS 配置，以匹配对等方的 ETS 建议配置。

设置 ETS TLV 单元

ETS TLV (etscfg) 配置确定主机如何响应对等方的 ETS 建议配置。此 TLV 单元只有一个可配置属性 willing。缺省情况下，此属性设置为 on，从而使本地主机可以将其 ETS 配置与远程对等方的 ETS 建议配置同步。

要验证主机能否将其 ETS 信息与远程对等方的 ETS 信息进行同步，请使用以下命令：

```
# lldpadm show-agenttlvprop -p willing -a agent etscfg
```

如果 willing 属性设置为 off，请键入以下命令建立同步：

```
# lldpadm set-agenttlvprop -p willing=on -a agent etscfg
```

要防止同步某特定代理的信息，请将 willing 属性设置为 off，如下所示：

```
# lldpadm set-agenttlvprop -p willing=off -a agent etscfg
```

其中，*agent* 是启用了代理的数据链路。

向对等方建议 ETS 配置

可以向对等方建议为每个优先级配置的 ETS 带宽值 (etsbw-lcl)，以便对等方能够配置相同的值。必须启用 NIC 上 LLDP 代理的 dot1-tlv 类型中的 etsreco 属性才能建议 ETS 带宽值。建议的值可与本地配置的 ETS 值相同，或者您也可以使用 dladm set-linkprop 命令，通过设置新的数据链路属性 etsbw-rmt-advice 显式配置建议的值。如果为数据链路分配的带宽不对称（这意味着接收 (Rx) 和传送 (Tx) 带宽不同），则配置 etsbw-rmt-advice 属性非常有用。

缺省情况下，使用为 etsbw-lcl 属性配置的值向对等方建议值。但是，可以通过为 etsbw-rmt-advice 属性设置其他值来建议其他 ETS 值。例如，如果 Tx 上的网络通信流量较多，则可以为 etsbw-lcl 属性（主机上的 Tx）配置较高的 ETS 值，为 etsbw-rmt-advice 属性（Rx 到主机）配置较低的值。

例 6-6 向对等方建议 ETS 配置

1. 通过显示 net5 的 LLDP 代理的 dot1-tlv 类型属性确保 etsreco 属性已启用。

```
# lldpadm show-agentprop -p dot1-tlv net5
AGENT        PROPERTY   PERM VALUE          DEFAULT        POSSIBLE
net5         dot1-tlv   rw   etsreco,etscfg none           none,vlanname,pvid,
                                                           linkaggr,pfc,appln,
                                                           evb,etscfg,etsreco,
                                                           all
```

2. 为 vnic1 分配 20% 的底层链路带宽份额。

```
# dladm set-linkprop -p etsbw-lcl=20 vnic1
# dladm show-linkprop -p etsbw-lcl vnic1
LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
vnic1    etsbw-lcl       rw   20           20           0         --
```

缺省情况下，将为对等方建议相同的值。

```
# dladm show-linkprop -p etsbw-rmt-advice vnic1
LINK     PROPERTY         PERM VALUE  EFFECTIVE    DEFAULT   POSSIBLE
vnic1    etsbw-rmt-advice rw   --      20           0         --
```

3. 显示通过 LLDP 交换的信息。

```
# lldpadm show-agent -l -v net5
```

4. 显示向对等方建议的带宽。

```
# 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                --
```

缺省情况下，为 net5 上每个优先级的 etsbw-lcl 属性配置的值将作为建议值发送至对等方。ETSBW_RMT_ADVICE 显示了向对等方建议的值。该输出还显示了对等方未在其上配置任何 ETS 带宽。可以使用 lldpadm show-agent 命令显示向对等方建议的带宽。

5. 向对等方建议不同的值。

```
# 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         --
```

6. 显示向对等方建议的带宽。

```
# 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 字段显示 vnic2 的值为 0，表示对等方未在其上设置任何带宽，尽管已建议带宽值。此情况表明，对等方可能未启用 LLDP 或不支持 ETS。

显示 ETS 配置信息

可以使用以下命令显示有关 ETS 配置的信息：

- `# dladm show-linkprop -p etsbw-lcl,etsbw-rmt,etsbw-lcl-advice,etsbw-rmt-advice` *datalink*

  此命令显示与物理链路上的 ETS 相关的信息。

- `# dladm show-phys -D ets` *phys-link*

  此命令显示物理链路上与链路间的带宽分配和分发有关的本地和远程 ETS 配置。

- `# lldpadm show-agenttlvprop -a` *agent* `etscfg`

  其中，*agent* 是启用 LLDP 的数据链路。此命令显示 ETS TLV 属性，该属性控制主机将 ETS 信息与对等方同步的功能。

例 6-7 显示与 ETS 相关的数据链路属性

此示例说明了如何显示在启用同步之前与 ETS 相关的数据链路属性的状态。

```
# 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          --
```

输出显示主机为 cos 值为 2 的 vnic1 设置并建议了 10% 的 ETS 值。但是，对等方为 cos 值为 2 的 vnic1 设置并建议了 20% 的 ETS 值。由于未启用同步（未启用 willing），因此主机未接受对等方的建议，这一点反映在 etsbw-lcl 属性的 EFFECTIVE 值中（本地配置的值）。

例 6-8 显示本地主机同步 ETS 信息的功能

此示例说明了如何显示本地主机用于适应对等方的 ETS 配置的功能的当前状态。

```
# lldpadm show-agenttlvprop -a net0 etscfg
AGENT   TLVNAME   PROPERTY   PERM   VALUE   DEFAULT   POSSIBLE
net0    etscfg    willing    rw     off     on        on,off
```

要启用同步，请键入以下命令：

```
# lldpadm set-agenttlvprop -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          --
```

由于已启用同步（已启用属性 willing），因此主机接受了对等方的建议，这一点反映在 etsbw-lcl 的 EFFECTIVE 值中。

以下示例显示了主机和对等方上针对物理链路上各优先级值的生效 ETS 值。

```
# 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　将生效 ETS 带宽显示为优先级的百分比。

ETSBW_RMT_EFFECT　将生效 ETS 带宽显示为对等方上优先级值的百分比。

ETSBW_LCL_SOURCE　指示 ETSBW_LCL_EFFECT 值的源。该值可以是 local（配置值）或 remote（建议值）。

以下示例显示了本地 ETS 信息，包括本地配置的值、本地生效值和对等方建议的值。

```
# 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                --
```

由于对等方未建议任何值，因此将使用本地配置的值 (ETSBW_LCL) 设置本地生效值 (ETSBW_LCL_EFFECT)。

以下示例显示了有关对等方的信息。

```
# dladm show-phys -D ets -r net5
LINK                COS ETSBW_RMT_EFFECT ETSBW_RMT_ADVICE CLIENTS
--                  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                --
```

输出显示远程对等方未在 ETSBW_RMT_EFFECT 字段中设置任何值，尽管主机已经建议对等方为优先级 3 设置 80%。

♦ ♦ ♦ 附 录 A

链路聚合和 IPMP：功能比较

链路聚合和IPMP是用来提高网络性能和维护网络可用性的不同技术。

下表列出了链路聚合与 IPMP 的一般比较。

功能	链路聚合	IPMP
网络技术类型	第 2 层（链路层）。	第 3 层（IP 层）。
配置工具	`dladm`	`ipadm`
基于链路的故障检测	支持。	支持。
基于探测器的故障检测	中继：基于 LACP，面向直接对等主机或交换机。 DLMP：支持。基于 ICMP，面向与 DLMP 地址位于同一子网中的任何已定义系统，跨多个级别的干预第 2 层交换机。	基于 ICMP，面向与测试地址位于同一 IP 子网中的任何已定义系统，跨多个级别的干预第 2 层交换机。
使用备用接口	中继：不支持。 DLMP：不支持。	支持。可以配置备用接口。
跨越多个交换机	中继：支持。不过，需要交换机供应商扩展。 DLMP：支持。	支持。
交换机配置	中继：必需。 DLMP：非必需。	非必需。
背对背配置	支持。	不支持。
支持的介质类型	特定于以太网。	支持广播。
负荷分配支持	中继：支持，由管理员通过 `dladm` 命令控制。支持传入负荷分配。 DLMP：在聚合的客户机和 VNIC 中受支持。不过，不支持按聚合上的个体客户机和 VNIC 进行负荷分配。	支持。由内核控制。传入负荷分配受源地址选择间接影响。
与 VNIC 集成时的支持级别	出色支持。聚合只能在控制域或全局区域中进行配置，且对这些区域是透明的。	支持。但是，无法在 IPMP 组上强制使用带宽限制、专用 Rx 或 Tx 环以及链路保护等 VNIC 属性。

功能	链路聚合	IPMP
		需要将多个 VNIC 指定给这些区域，并需要在每个区域中进行配置。
用于资源管理的用户定义流	支持。	不支持。
链路保护	支持。	不支持。
协议要求	无。	无。

在链路聚合中，传入通信流量以中继模式在构成聚合的多个链路上分配。因此，由于安装了更多 NIC 以将链路添加到聚合，提高了网络性能。

DLMP 聚合跨越多个交换机。作为第 2 层技术，聚合与其他 Oracle Solaris 虚拟化技术进行了很好的集成。

IPMP 的通信使用 IPMP 接口的数据地址，因为它们绑定到了可用的活动接口。举例来说，如果所有数据通信只在两个 IP 地址之间流动，但不一定在同一连接上，则添加多个 NIC 不会借助 IPMP 提高性能，因为只有两个 IP 地址保持可用。

♦ ♦ ♦ 附 录 B

传递式探测器的包格式

传递式探测器包是具有以太网类型 ETHERTYPE_ORCL (0x881b) 的专有协议包。有关这些传递式探测器的更多信息，请参见“基于探测器的故障检测” [22]。有关显示探测器统计信息的示例，请参见例 2-7 “显示与探测器相关的信息”。下图显示了传递式探测器包格式。

图 B-1 传递式探测器包

字段	说明
Dest MAC（目标 MAC）	目标 MAC 地址。
Src MAC（源 MAC）	源 MAC 地址。
Proto#（协议编号）	协议编号。ETHERTYPE_ORCL 包的第 2 层有效负载必须以 16 位协议编号（对于传递式探测器包来说，为 1）开头。
Typ（类型）	探测器包类型。对于请求，探测器包类型为 0，对于响应则为 1。
Pad（填充）	填充（均为零）。
Seq#（序列号）	探测器序列号。
Timestamp（时间戳）	探测器时间戳。
TL	目标信息长度。对于以太网，目标长度为 6 位（以太网 MAC 地址长度）。

字段	说明
Target（目标）	目标端口 MAC 地址。
SL	源信息长度。对于以太网，源长度为 6 位。
Source（源）	源端口 MAC 地址。

索引

M

P

Q

W

X

Y

Z